

DFSORT



# Messages, Codes and Diagnosis Guide

*Release 14*



DFSORT



# Messages, Codes and Diagnosis Guide

*Release 14*

**Note!**

Before using this information and the product it supports, be sure to read the general information under “Appendix C. Notices” on page 117.

**Fifth Edition (July 2000)**

This edition replaces and makes obsolete the previous edition, SC26-7050–03. Technical changes for Release 14 are summarized under “Summary of Changes” and are indicated by a vertical bar to the left of a change. A vertical bar to the left of a figure caption indicates that the figure has changed. Editorial changes that have no technical significance are not noted.

This edition applies to Release 14 of DFSORT, (5740-SM1) and to any subsequent releases until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

**Note:** This document is not classified because all the information has previously been released to customers in other formats.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

© **Copyright International Business Machines Corporation 1995, 2000. All rights reserved.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>Figures</b> . . . . .	vii
<b>Preface</b> . . . . .	ix
How to Use the Diagnosis Guide . . . . .	ix
DFSORT Publications . . . . .	ix
DFSORT Library Softcopy Information . . . . .	x
Referenced Publications . . . . .	x
How to Send Your Comments . . . . .	xi
<b>Summary of Changes</b> . . . . .	xiii
Fifth Edition, July 2000 . . . . .	xiii
New Programming Support for Release 14 (PTFs — July, 2000) . . . . .	xiii
Fourth Edition, December 1999 . . . . .	xv
Third Edition, March 1999 . . . . .	xv
New Programming Support for Release 14 (PTFs - March 1999) . . . . .	xv
Second Edition, September 1998 . . . . .	xv
New Programming Support for Release 14. . . . .	xv

---

## Part 1. Eliminating Common Sources of Error . . . . . 1

<b>Chapter 1. Eliminating Common Sources of Error</b> . . . . .	3
How to Get Messages . . . . .	3
How to Get a Dump . . . . .	3
Eliminating DFSORT Installation as a Source of Error . . . . .	4
Eliminating Your Invoking Program as a Source of Error . . . . .	4
Eliminating Your User Exit Routines as a Source of Error . . . . .	4
Errors in Using Registers . . . . .	4
Errors in Using Main Storage . . . . .	5
Errors in Record Contents . . . . .	5
Correcting Common Errors . . . . .	5
Common System Abends Related to Storage . . . . .	5
Miscellaneous System Abends . . . . .	6
User Abends at Run Time . . . . .	9
User Abends from DFSORT Panels . . . . .	10
Common Problems Reported by DFSORT Users . . . . .	10

---

## Part 2. Messages and Codes . . . . . 13

<b>Chapter 2. Message Considerations</b> . . . . .	15
Printing Messages and Control Statements . . . . .	15
Printing EFS Messages and Control Statements . . . . .	16
Writing Messages to the Master Console . . . . .	16
Finding DFSORT Control Statement Coding Errors . . . . .	17
<b>Chapter 3. Informational and Error Messages</b> . . . . .	19
<b>Chapter 4. Diagnostic Messages</b> . . . . .	69
<b>Chapter 5. Informational Messages for Debugging</b> . . . . .	71
<b>Chapter 6. Return Codes and User Abends</b> . . . . .	73
DFSORT Return Codes . . . . .	73

ICETOOL Return Codes . . . . .	73
ICEGENER Return Codes. . . . .	74

---

**Part 3. Diagnosis Guide . . . . . 77**

<b>Chapter 7. Resolving Failures in the DFSORT Program . . . . .</b>	<b>79</b>
Developing Your Keyword String . . . . .	80
Constructing Keyword Strings . . . . .	82
Component Identification Keyword Procedure . . . . .	83
FMID Keyword Procedure . . . . .	84
Type-of-Technique Keyword Procedure . . . . .	84
Type-of-Application Keyword Procedure . . . . .	85
Type-of-Failure Keyword Procedures . . . . .	85
Wait and Loop Keywords Procedure . . . . .	86
System Abend Keyword Procedure . . . . .	87
User Abend Keyword Procedure . . . . .	87
Message Keyword Procedure . . . . .	88
Incorrect-Output Keyword Procedure . . . . .	88
Performance Keyword Procedure . . . . .	89
Document Keyword Procedure . . . . .	89
Module Keyword Procedure . . . . .	89
Offset Keyword Procedure. . . . .	91
Message Variable Keywords . . . . .	92
ICE061A Variable Keywords . . . . .	92
ICE117A Variable Keywords . . . . .	93
<b>Chapter 8. Resolving Problems When Using DFSORT Panels . . . . .</b>	<b>95</b>
Obtaining Diagnostic Information . . . . .	95
Determining Where the Failure Occurred . . . . .	95
Resolving Failures in DFSORT Panels . . . . .	96
DFSORT Panels Abend Panel . . . . .	96
TSO Abend Message . . . . .	97
Resolving Failures in ISPF or ISMF . . . . .	98
Resolving Failures in the DFSORT Program . . . . .	99
Foreground Execution Failures . . . . .	100
Background Execution Failures . . . . .	100
Recording Information . . . . .	100
Inter-Module Traces . . . . .	101
ISMF Logging with SORTREC. . . . .	101
Using the Profile Application . . . . .	102
Directing Abend Processing. . . . .	102
Obtaining Diagnostic Messages . . . . .	103
<b>Chapter 9. Searching the IBM Software Support Database . . . . .</b>	<b>105</b>
<b>Chapter 10. Fixing or Bypassing the Problem . . . . .</b>	<b>107</b>
Fixing the Problem . . . . .	107
Bypassing the Problem . . . . .	107
<b>Chapter 11. Reporting a Problem . . . . .</b>	<b>109</b>

---

**Part 4. Appendixes . . . . . 111**

<b>Appendix A. Valid Input/Output Options at User Exits—ICE044I . . . . .</b>	<b>113</b>
---	------------

<b>Appendix B. User Exit Return Codes—ICE071A</b> . . . . .	115
<b>Appendix C. Notices</b> . . . . .	117
Programming Interface Information . . . . .	117
Trademarks. . . . .	117
<b>Summary of Changes</b> . . . . .	119
New Programming Support for Release 13 (PTFs after April, 1996) . . . . .	119
Additional Year 2000 Features . . . . .	119
OS/390 Registration . . . . .	119
New Programming Support for Release 13 (PTFs - April, 1996) . . . . .	119
Year 2000 Features. . . . .	119
Performance Improvements for FLR and VLR Blockset Sorts . . . . .	119
Floating Point for SUM . . . . .	120
Security Improvements . . . . .	120
EXCPVR Processing Removed . . . . .	120
New Device Support for Release 13 (PTFs). . . . .	120
New Programming Support for Release 13 . . . . .	121
DFSORT's Performance Booster for The SAS** System . . . . .	121
Dynamic Hipersorting . . . . .	121
Performance . . . . .	121
OUTFIL Processing. . . . .	121
National Language Support. . . . .	122
ICETOOL Enhancements . . . . .	122
INCLUDE/OMIT Substring Search . . . . .	123
SMF Type-16 Record Enhancements . . . . .	123
Other Enhancements . . . . .	123
New Programming Support for Release 12 (PTFs) . . . . .	124
New Device Support for Release 12 (PTFs). . . . .	125





---

## Figures

1. Keyword String Flowchart . . . . .	81
2. Keyword String Syntax. . . . .	83
3. Finding the System Abend Code and the PSW in the SYSUDUMP . . . . .	86
4. Finding the Wait State Bit in the PSW . . . . .	87
5. Finding the Abend Address . . . . .	90
6. Finding the Module Name . . . . .	91
7. Finding the Offset . . . . .	92
8. DFSORT Panels Abend Panels Indicate the Problem Is in DFSORT Panels . . . . .	97
9. TSO Messages Also Indicate That the Problem Is in DFSORT Panels . . . . .	97
10. Two Symptom Dumps from TSO Appear on Your Screen . . . . .	98
11. ISPF Abend Panels Indicate the Problem Is in ISPF . . . . .	99
12. ISMF Abend Panels Indicate the Problem Is in ISMF . . . . .	99



---

# Preface

This book is intended to help you:

- Eliminate common sources of errors. “Part 1. Eliminating Common Sources of Error” on page 1 explains how to get messages and dumps and how to identify and eliminate various problems.
- Interpret the informational and error messages issued by DFSORT. “Part 2. Messages and Codes” on page 13 contains the messages returned from the program, explains their meanings, and suggests appropriate responses.
- Diagnose a program failure that you suspect is caused by DFSORT. “Part 3. Diagnosis Guide” on page 77 is a step-by-step guide that will assist you in searching for a solution and, if necessary, reporting the problem to the IBM Support Center.

---

## How to Use the Diagnosis Guide

The Diagnosis Guide is designed to help you solve problems you encounter when running DFSORT. To use the information in this part most efficiently, first determine whether the problem is caused by:

### A user error

Read “Chapter 1. Eliminating Common Sources of Error” on page 3, for solutions to some of the most common errors users encounter when running DFSORT.

### A program error

Use the numbered steps listed in “Chapter 7. Resolving Failures in the DFSORT Program” on page 79, to describe the problem and to find out if corrections for it are already available.

### A DFSORT Panels error

A failure while working in DFSORT Panels might be due to an abend in the DFSORT Panels program, ISPF, or ISMF. Begin with “Chapter 8. Resolving Problems When Using DFSORT Panels” on page 95. If the problem actually is a DFSORT program failure, the steps in that section will direct you back to this page.

---

## DFSORT Publications

*DFSORT Messages, Codes and Diagnosis Guide Release 14* is a part of a more extensive DFSORT library. These books can help you work with DFSORT more effectively.

Task	Publication	Order Number
Planning For and Customizing DFSORT	<i>DFSORT Installation and Customization Release 14</i>	SC33-4034
Learning to Use DFSORT	<i>Getting Started with DFSORT Release 14</i>	SC26-4109
Learning to Use DFSORT Panels	<i>DFSORT Panels Guide</i>	GC26-7037
Application Programming	<i>DFSORT Application Programming Guide Release 14</i>	SC33-4035
Quick Reference	<i>DFSORT Reference Summary Release 14</i>	SX33-8001
Tuning DFSORT	<i>DFSORT Tuning Guide Release 14</i>	SC26-3111

You can order a complete set of DFSORT books with the order number SBOF-1243, except for *DFSORT Licensed Program Specifications Release 14* which must be ordered separately.

See *DFSORT Application Programming Guide* for a complete list of related publications that might be of use to you.

## DFSORT Library Softcopy Information

The DFSORT library, with the exception of the *DFSORT Reference Summary Release 14*, is available on CD-ROM.

Order Number	Title
SK2T-6700	<i>IBM Online Library Omnibus Edition OS/390 Collection</i>

## Referenced Publications

For up-to-date descriptions of all of the books that support OS/390, refer to the *OS/390 Information Roadmap*, GC28-1727.

Within the text, references are made to the following publications:

Short Title	Publication	Order Number
Authorized Assembler Services Guide	<i>MVS/ESA Programming: Authorized Assembler Services Guide (for MVS/ESA SP Version 5)</i>	GC28-1467
Application Development Guide for Authorized Programs	<i>MVS/ESA Application Development Reference: Services for Assembler Language Programs (for MVS/ESA SP Version 4)</i>	GC28-1642
Application Development Reference for Authorized Programs	<i>MVS/ESA Programming: Assembler Services Reference (for MVS/ESA SP Version 5)</i>	GC28-1474
	<i>MVS/ESA Application Development Reference: Services for Assembler Language Programs (for MVS/ESA SP Version 4)</i>	GC28-1648
ISPF Dialog Management Guide	<i>Interactive System Productivity Facility Dialog Management Guide and Reference (for ISPF Version 3 Release 2 or later)</i>	SC34-4266
	<i>Interactive System Productivity Facility Dialog Management Guide (for ISPF Version 3 Release 1)</i>	SC34-4213
	<i>Interactive System Productivity Facility Dialog Management Guide (for ISPF Version 2 Release 3)</i>	SC34-4112
Macro Instructions for Data Sets	<i>DFSMS/MVS Version 1 Release 2: Macro Instructions for Data Sets</i>	SC26-4913
	<i>OS/390 DFSMS Macro Instructions for Data Sets</i>	SC26-7337
	<i>MVS/ESA Data Administration: Macro Instruction Reference</i>	SC26-4506
	<i>MVS/ESA VSAM Administration: Macro Instruction Reference</i>	SC26-4517

Short Title	Publication	Order Number
SPL: System Management Facilities	<i>MVS/ESA System Management Facilities (for MVS/ESA SP Version 5)</i>	GC28-1457
	<i>MVS/ESA System Management Facilities (for MVS/ESA SP Version 4)</i>	GC28-1628
System Messages	<i>MVS/ESA System Messages, Volume 1 (for MVS/ESA SP Version 4)</i>	GC28-1656
	<i>MVS/ESA System Messages, Volume 2 (for MVS/ESA SP Version 4)</i>	GC28-1657
	<i>MVS/ESA System Messages, Volume 3 (for MVS/ESA SP Version 4)</i>	GC28-1658
	<i>MVS/ESA System Messages, Volume 4 (for MVS/ESA SP Version 4)</i>	GC28-1659
	<i>MVS/ESA System Messages, Volume 5 (for MVS/ESA SP Version 4)</i>	GC28-1660
	<i>OS/390 System Messages, Volume 1 (ABA-ASA)</i>	GC28-1784
	<i>OS/390 System Messages, Volume 2 (ASB-EZM)</i>	GC28-1785
	<i>OS/390 System Messages, Volume 3 (GDE-IEB)</i>	GC28-1786
	<i>OS/390 System Messages, Volume 4 (IEC-IFD)</i>	GC28-1787
	<i>OS/390 System Messages, Volume 5 (IGD-IZP)</i>	GC28-1788
Using Data Sets	<i>DFSMS/MVS Version 1 Release 2: Using Data Sets</i>	SC26-4922
	<i>MVS/ESA Data Administration Guide</i>	SC26-4505
	<i>MVS/ESA VSAM Administration Guide</i>	SC26-4518
Locale Processing	<i>IBM Language Environment for MVS &amp; VM Debugging Guide and Run-Time Messages</i>	SC26-4829

## How to Send Your Comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book or any other DFSORT documentation:

- Send your comments by e-mail to:
  - IBMLink from US: [starpubs@us.ibm.com](mailto:starpubs@us.ibm.com)
  - IBMLink from Canada: STARPUBS at TORIBM
  - IBM Mail Exchange: USIB3VVD at IBMMAIL
  - Internet: [starpubs@us.ibm.com](mailto:starpubs@us.ibm.com)

Be sure to include the name of the book, the part number of the book, version and product name, and if applicable, the specific location of the text you are commenting on (for example, a page number or a table number).

- Fill out one of the forms at the back of this book and return it by mail or by giving it to an IBM representative. If the form has been removed, address your comments to IBM Corporation, RCF Processing Department G26/050, 5600 Cottle Road, San Jose, California 95193-0001, U.S.A.



---

# Summary of Changes

---

## Fifth Edition, July 2000

### New Programming Support for Release 14 (PTFs — July, 2000)

#### **Larger Tape Block Sizes with OS/390 R10**

DFSORT can now use tape data sets with block sizes greater than 32760 bytes for input and output, providing improved performance and tape utilization.

DFSORT can now select system-determined optimum block sizes greater than 32760 bytes for tape output data sets, if allowed by the BLKSZLIM value in effect. Installation and run-time options SDB=INPUT (the new IBM-supplied default), SDB=LARGE (new), SDB=YES (or its alias SDB=SMALL) and SDB=NO allow you to control DFSORT's use of system-determined block sizes, including block sizes greater than 32760 bytes for tape output data sets.

DFSORT's ICEGENER, like IEBGENER, will use SDB=value parameters you specify.

#### **New Device Support**

DFSORT now supports FICON channels.

DFSORT now recognizes the IBM 2105 control unit and provides optimizations for it appropriate for cached control units, such as cache fast write for work data sets.

DFSORT now treats work data sets on emulated 3390 model 9 devices as desirable choices with respect to performance. DFSORT continues to treat work data sets on real 3990 model 9 devices as undesirable choices with respect to performance.

#### **DFSMSrmm-Managed Tapes**

DFSORT can now automatically obtain accurate input file size information for DFSMSrmm-managed tapes. This can result in improved sort performance and more accurate dynamic workspace allocation.

DFSORT can now automatically obtain input and output attributes such as RECFM, LRECL and BLKSIZE for DFSMSrmm-managed tapes.

#### **VSAM Processing**

DFSORT now supports the VSAM extended addressability function for extended format VSAM data sets, which provides the capability of VSAM data sets larger than four gigabytes.

DFSORT now allows an empty VSAM input data set to be accepted and processed as having zero records. New installation option VSAMEMT=YES/NO and run-time options VSAMEMT and NVSAMEMT allow you to control the processing of empty VSAM input data sets.

DFSORT now allows a VSAM output data set defined with REUSE to be processed as a NEW or MOD data set. New installation option RESET=YES/NO and run-time options RESET and NORESET allow you to control the processing of VSAM output data sets.

DFSORT now allows a VSAM data set defined with REUSE to be sorted in-place. New installation option VSAMIO=YES/NO and run-time options VSAMIO and NOVAMIO allow you to control the use of the same VSAM data set for input and output.

### **HFS Files**

DFSORT now supports Hierarchical File System (HFS) files for input and output.

### **Productivity and Easier Migration from other Sort Products**

By default, DFSORT now uses the L3, OUTREC or INREC record length as the SORTOUT LRECL when the SORTOUT LRECL is unavailable. New installation option SOLRF=YES/NO and run-time options SOLRF and NOSOLRF allow you to control the use of the reformatted record length for the SORTOUT LRECL.

DFSORT now provides enhanced capabilities for OUTFIL, including:

- Padding short records with blanks, by default, when converting variable-length records to fixed-length records.
- Creation of edited counts and subcounts in trailers.

DFSORT now provides enhanced capabilities for INREC and OUTREC that were previously only available with OUTFIL OUTREC, including:

- Hexadecimal display.
- Transformation of dates.
- Editing of numeric fields.
- Lookup and change.

DFSORT now provides enhanced capabilities for INREC, OUTREC and OUTFIL OUTREC, including:

- Conversion of numeric fields to other numeric formats.
- Editing of transformed dates.
- Conversion of transformed dates to numeric formats.
- Creation of sequence numbers.

DFSORT now allows INCLUDE/OMIT fields that extend beyond variable-length records (that is, short fields) to be compared as if they were padded with binary zeros. This allows both the short and non-short fields to be compared. New installation option VLSCMP=YES/NO and run-time options VLSCMP and NOVLSMCP allow you to control the use of binary zero padding for short control fields.

DFSORT now allows -0 and +0 to be treated as unsigned (that is, the same) for processing by DFSORT statements INCLUDE, INREC, MERGE, OMIT, OUTFIL, OUTREC and SORT. New installation option SZERO=YES/NO and run-time options SZERO and NOSZERO allow you to select signed or unsigned zero processing for these statements.

DFSORT now allows -0 and +0 to be treated as unsigned (that is, the same) for processing by ICETOOL operators DISPLAY, OCCUR, SELECT and UNIQUE. New option UZERO allows you to select unsigned zero processing for these operators.

DFSORT now always determines or assigns the record type (F or V) when the input is VSAM, or when an E15 or E32 exit supplies all of the input records.



DFSORT now allows an LRECL greater than 80 for SYSIN, SORTCNTL and DFSPARM; only the first 80 bytes of each control statement are used.

DFSORT now allows variable-length output records that are longer than the output data set LRECL to be truncated to the LRECL. New installation option VLLONG=YES/NO and run-time options VLLONG and NOVLLONG allow you to control the truncation of long variable-length output records.

DFSORT now accepts and processes the following as EXEC/DFSPARM PARM options: COBEXIT=COB1/COB2, NOVERIFY, NOVLSHRT, NZDPRINT, OVFL0=RC0/RC4/RC16, PAD=RC0/RC4/RC16, SPANINC=RC0/RC4/RC16, TRUNC=RC0/RC4/RC16, VERIFY, VLSHRT and ZDPRINT.

DFSORT now accepts and processes the following as OPTION statement options: EXITCK=STRONG/WEAK, WRKREL and WRKSEC.

DFSORT now accepts and processes Y2PAST=s/f as a SORT and MERGE statement option.

DFSORT now accepts, and processes or ignores as appropriate, many more aliases for DFSORT options.

DFSORT now accepts and processes the following as 24-Bit Parmlist control statement entry codes: X'10', X'16', X'18' and X'20' through X'29'.

---

## **Fourth Edition, December 1999**

This edition did not include a change to the product. It improved the usability of the softcopy version of this book by adding links to the messages.

---

## **Third Edition, March 1999**

### **New Programming Support for Release 14 (PTFs - March 1999)**

#### **New Generation of Year 2000 Features**

New Y2T, Y2U, Y2V, Y2W, Y2X, and Y2Y formats handle CH, ZD, and PD full dates like yymmdd, yyddd, yymm, yyq, mmdyy, dddy, mmyy, and qyy, and their special indicators like zeros and nines, according to the specified fixed or sliding century window.

You can use these new formats in the SORT, MERGE, INCLUDE, OMIT, and OUTFIL statements to sort, merge, compare, and transform full dates.

---

## **Second Edition, September 1998**

### **New Programming Support for Release 14**

#### **Symbols for Fields and Constants**

DFSORT now provides a simple and flexible method for using symbols in DFSORT and ICETOOL statements. You can define and use a symbol for any field or constant that is recognized in a DFSORT control statement or ICETOOL operator. This makes it easy to create and reuse collections of symbols (that is, mappings) for your frequently used data.

In addition, you can obtain and use collections of DFSORT symbols created specifically for data associated with other products (for example, RACF, DFSMSrmm and DCOLLECT) or by your site.

DFSORT symbols can increase your productivity by automatically providing the positions, lengths and formats of the fields and the literals, numbers, and bit flags of the constants, associated with the particular records you are processing with DFSORT or ICETOOL.

### **Improvements in Performance, Capacity and Storage Usage**

Blockset copy and merge applications can now use storage above 16MB virtual, providing improved performance and virtual storage constraint relief.

Blockset copy and merge modules will now reside above 16MB virtual, providing virtual storage constraint relief.

DFSORT can now handle a significantly larger number of INCLUDE and OMIT conditions.

DFSORT can now handle a significantly larger number of SUM fields.

The upper limit for the number of JCL and dynamically allocated work data sets that can be specified and used by DFSORT's Blockset technique has been raised from 100 to 255. The use of more work data sets increases the maximum amount of data DFSORT can process in a single sort application. Any valid ddname of the form SORTWKdd or SORTWKd can now be used for DASD work data sets (for example, SORTWK01, SORTWK3, SORTWK2, SORTWK#5, SORTWKA, SORTWKXY and so on).

The upper limit for the number of input data sets that can be specified and used for a Blockset merge application has been raised from 16 to 100. The use of more merge input data sets increases the maximum amount of data DFSORT can process in a single merge application.

### **Time-of-Day Option Controls**

New time-of-day installation modules (ICETD1-4) allow different sets of installation defaults to be used, based on the day and time DFSORT applications run. Each environment installation module (ICEAM1-4) can enable one or more time-of-day installation modules. This capability allows new levels of control for installation defaults. For example, larger storage, hiperspace and data space limits could be used only for batch program-invoked DFSORT applications that run off-shift during the week, and all weekend.

### **Repackaging**

The product has been repackaged to simplify installation and customization:

- IBM's DFSORT, DFSMSdftp, and MVS/DFP teams have simplified the process of replacing IEBGENER with ICEGENER. You now only need to apply a DFSMS or DFP PTF that supplies an alias of "IEBGENR" for IEBGENER and place ICEGENER with an alias of "IEBGENER" ahead of IEBGENER in the system's search order for programs.
- The number of FMIDs has been reduced from 10 to 3.
- The number of libraries required to install DFSORT has been reduced from 40 to 26.

- DFSORT R14 now supports a single installation of the product for both resident and nonresident features. This allows you to decide how to use DFSORT independent of the installation method, thus reducing the number of decisions you have to make at installation time.
- All FMIDs in DFSORT R14 can be installed together, including the FMIDs for both English and Japanese messages and panels.

### **OUTFIL Processing Enhancements**

OUTFIL now supports creation of multiple output records using the fields of the input record. This allows you to split each record into pieces, include a field in more than one record, include different fields in different records, and more.

OUTFIL now supports processing of variable-length input records which are too short to contain all specified OUTFIL OUTREC fields. OUTFIL's new VLFILL=byte operand can be used to replace missing bytes in OUTFIL OUTREC fields with the specified fill byte so the filled fields can be processed.

### **ICETOOL Enhancement**

A new DISCARD(savedd) operand of ICETOOL's SELECT operator allows you to save the records that are not selected, in the savedd data set. Thus, in one pass, you can create an outdd data set with the records that meet your specified criteria, and a savedd data set with the records that do not meet your specified criteria. DISCARD(savedd) can be used to save the records discarded by ALLDUPS, NODUPS, HIGHER(x), LOWER(y), EQUAL(v), FIRST or LAST.

### **Installation and Run-Time Option Enhancements**

A new p% value for the EXPRES, EXPOLD, and EXPMAX installation options and the HIPRMAX installation and run-time options is now available. p% can be used to vary the limit DFSORT calculates for the corresponding option as a percentage of the configured expanded storage on the system at run time. If the configured expanded storage on a system changes, p% will cause a corresponding change in the run-time limit calculated for the corresponding option. When sharing DFSORT installation options between systems, such as in a sysplex, p% can be used to tailor the limit DFSORT calculates for the corresponding option to the system on which the application runs.

A new SPANINC installation and run-time option allows you to specify what you want DFSORT to do if it detects incomplete spanned records. This gives you control over the action (continue by eliminating incomplete spanned records and recovering valid records, or terminate), type of message (informational or error) and return code (0, 4 or 16) for incomplete spanned records.

A new OVFLO installation and run-time option allows you to specify what you want DFSORT to do when BI, FI, PD or ZD summary fields overflow. This gives you control over the action (continue or terminate), type of message (informational or error) and return code (0, 4 or 16) for summary overflow.

A new PAD installation and run-time option allows you to specify what you want DFSORT to do when the SORTOUT LRECL is larger than the SORTIN/SORTINnn LRECL. This gives you control over the action (continue or terminate), type of message (informational or error) and return code (0, 4 or 16) for LRECL padding.

A new TRUNC installation and run-time option allows you to specify what you want DFSORT to do when the SORTOUT LRECL is smaller than the SORTIN/SORTINnn LRECL. This gives you control over the action (continue or terminate), type of message (informational or error) and return code (0, 4 or 16) for LRECL truncation.

The IBM-supplied default for ICEMAC option DSA has been changed from 16MB to 32MB.

The IBM-supplied default for ICEMAC option GENER has been changed from IEBGENER to IEBGENR.

The maximum value for ICEMAC option OVERRGN has been changed from 64KB to 16128KB.

### **Installation and Run-Time Option Enhancements**

A new p% value for the EXPRES, EXPOLD, and EXPMAX installation options and the HIPRMAX installation and run-time options is now available. p% can be used to vary the limit DFSORT calculates for the corresponding option as a percentage of the configured expanded storage on the system at run time. If the configured expanded storage on a system changes, p% will cause a corresponding change in the run-time limit calculated for the corresponding option. When sharing DFSORT installation options between systems, such as in a sysplex, p% can be used to tailor the limit DFSORT calculates for the corresponding option to the system on which the application runs.

A new SPANINC installation and run-time option allows you to specify what you want DFSORT to do if it detects incomplete spanned records. This gives you control over the action (continue by eliminating incomplete spanned records and recovering valid records, or terminate), type of message (informational or error) and return code (0, 4 or 16) for incomplete spanned records.

A new OVFL0 installation and run-time option allows you to specify what you want DFSORT to do when BI, FI, PD or ZD summary fields overflow. This gives you control over the action (continue or terminate), type of message (informational or error) and return code (0, 4 or 16) for summary overflow.

A new PAD installation and run-time option allows you to specify what you want DFSORT to do when the SORTOUT LRECL is larger than the SORTIN/SORTINnn LRECL. This gives you control over the action (continue or terminate), type of message (informational or error) and return code (0, 4 or 16) for LRECL padding.

A new TRUNC installation and run-time option allows you to specify what you want DFSORT to do when the SORTOUT LRECL is smaller than the SORTIN/SORTINnn LRECL. This gives you control over the action (continue or terminate), type of message (informational or error) and return code (0, 4 or 16) for LRECL truncation.

The IBM-supplied default for ICEMAC option DSA has been changed from 16MB to 32MB.

The IBM-supplied default for ICEMAC option GENER has been changed from IEBGENER to IEBGENR.

The maximum value for ICEMAC option OVERRGN has been changed from 64KB to 16128KB.

### **Other Enhancements**

New messages ICE178I and ICE179A provide information about reallocation of VIO work data sets.

The option-in-effect messages (ICE127I-ICE133I) are now printed for Blockset copy and merge applications.

The user exit address constant can now be passed to E32 user exits for Blockset merge applications.

Null segments in variable spanned input records are now processed by DFSORT and no longer result in termination. A null segment means that there are no more segments in the block.

**OS/390 and MVS/ESA Only**

DFSORT Release 14 only supports the OS/390 and MVS/ESA environments. MVS/XA and VIRTDSP processing for MVS/XA are no longer supported.



---

## Part 1. Eliminating Common Sources of Error

<b>Chapter 1. Eliminating Common Sources of Error</b> . . . . .	3
How to Get Messages. . . . .	3
How to Get a Dump . . . . .	3
Eliminating DFSORT Installation as a Source of Error . . . . .	4
Eliminating Your Invoking Program as a Source of Error . . . . .	4
Eliminating Your User Exit Routines as a Source of Error. . . . .	4
Errors in Using Registers . . . . .	4
Errors in Using Main Storage . . . . .	5
Errors in Record Contents . . . . .	5
Correcting Common Errors . . . . .	5
Common System Abends Related to Storage . . . . .	5
Miscellaneous System Abends . . . . .	6
User Abends at Run Time . . . . .	9
User Abends from DFSORT Panels . . . . .	10
Common Problems Reported by DFSORT Users . . . . .	10





---

## Chapter 1. Eliminating Common Sources of Error

Some problems in running DFSORT are not caused by DFSORT failures, and using the information in this section can save you time. This section explains how to get messages and dumps as well as how to identify and eliminate problems caused by errors in your installation of DFSORT, your calling programs, or your user exit routines. The section also discusses other common errors.

---

### How to Get Messages

When a message data set is required, but neither a SYSOUT DD statement nor an alternative message data set ddname is provided, DFSORT can either terminate or continue. Using the defaults with which it was delivered, DFSORT issues message ICE158A to the console, and terminates with either a return code of 20 or a user abend code of 158 (depending on the ERET, ABEND, or NOABEND option in effect at run-time). You can change the default value during installation with the ICEMAC option NOMSGDD to allow DFSORT to continue even when a required message data set is not supplied.

If you do not receive any DFSORT messages, make sure that you supply a message data set, and resubmit your job using a SORTDIAG DD statement with a DUMMY parameter. The SORTDIAG DD statement specifies that all messages, including diagnostic messages (ICE800I through ICE999I), and control statements be printed.

You can provide either a SYSOUT DD statement or a DDNAME DD statement (where DDNAME is the alternate message data set ddname specified at installation or execution time with MSGDDN=ddname).

For example:

```
//SORTDIAG DD DUMMY  
//SYSOUT DD SYSOUT=A
```

If you use a temporary or permanent message data set, it is best to specify a disposition of MOD to ensure you see all messages and control statements in the message data set. For example:

```
//SYSOUT DD DSN=&MSG,DISP=(MOD,PASS),SPACE=(CYL,(2,2),UNIT=SYSDA
```

---

### How to Get a Dump

To get a dump at an abend, use the JCL and program control statements shown in Table 1. You get additional abend information when DFSORT's ESTAE recovery routine is in effect. At the beginning of each run, DFSORT establishes an ESTAE recovery routine to trap system or user abends for Blockset and Peerage/Vale applications. The ESTAE recovery routine remains in effect unless you use ICEMAC option ESTAE=NO or the run-time option NOESTAE to delete it.

*Table 1. Getting a Dump*

<b>If you received</b>	<b>Then use</b>
A user or system abend code	//SYSUDUMP DD SYSOUT=A
A condition code 16	//SYSUDUMP DD SYSOUT=A //DFSPARM DD * DEBUG ABEND

## Eliminating Common Sources of Error

Table 1. Getting a Dump (continued)

If you received	Then use
<b>Note:</b> If your program invokes DFSORT and does not pass the DFSORT return code to the system, the listing's condition code might not be the actual DFSORT return code. To determine if the code in the listing is from DFSORT, find out how your program handles return codes.	

If you invoke DFSORT from a COBOL program compiled with VS COBOL, you might receive a COBOL dump and a DFSORT dump. For the procedures in this manual, use only the DFSORT dump.

If you are using PL/I, you must specify the NOSPIE and NOSTAE macro at run time to ensure that you receive a DFSORT dump. If you do not, you might receive only a PL/I dump.

---

## Eliminating DFSORT Installation as a Source of Error

If you recently installed a new release or Program Temporary Fix (PTF) level of DFSORT, make sure any necessary additional alias names were added to module ICEMAN. If not, then DFSORT might be executing mixed levels of program modules, which can cause an abnormal end (abend).

---

## Eliminating Your Invoking Program as a Source of Error

Main storage space can be a problem when you invoke DFSORT from another program, especially if you use MAINSIZE/SIZE=MAX and invoke user exit E15 or E35 (or, from COBOL, use an input or output procedure). In this case, you usually receive a system ABEND 80A message.

If you invoked DFSORT from another program, and particularly if you opened a file in your user exit routine, check that you specified a sufficiently large amount of reserved storage with the RESALL, ARESALL, RESINV, or ARESINV run-time option.

For more detailed information on main storage, see *DFSORT Application Programming Guide*.

---

## Eliminating Your User Exit Routines as a Source of Error

To eliminate your own routines or programs as possible sources of error, check the use of registers, the main storage space allocated to the routines or programs, and the record contents.

## Errors in Using Registers

Make sure your routines or programs save and restore the appropriate registers. If the routines or programs return to DFSORT with, for example, a different value in register 12 the outcome is probably an abend.

Ensure that registers you use for loading or storing are not accidentally overlaying DFSORT code or work areas. In this case, DFSORT might work without errors with one technique but fail with another.

For more detailed information on register usage, see *DFSORT Application Programming Guide*.

### Errors in Using Main Storage

Check whether your routines need more main storage space than you have allocated. Have you increased the size of your routines without increasing the corresponding exit size on your MODS statement? A change in the size of your user exit routines can lead to an abend in your own routine, or it can lead to insufficient space for DFSORT.

You can see if too little space was left for sorting by studying the information in messages ICE092I, ICE093I, ICE156I and ICE231I. If you did not receive any messages, refer to “How to Get Messages” on page 3.

Similar situations can occur if the MAINSIZE/SIZE=MAX option is specified or taken by default and a fairly large reserved value is passed to DFSORT through the RESALL, ARESALL, RESINV, or ARESINV option or taken by default. Another problem can arise if your E15 routine issues a GETMAIN without a corresponding FREEMAIN at the end. This can be done indirectly, for example, by leaving a data set open so that a buffer pool remains reserved.

For more detailed information, see *DFSORT Application Programming Guide*.

### Errors in Record Contents

If the output records do not appear to contain the same data as the input records, and either E15 or E35 has been used, check that your routine is handling register 1 correctly; especially, check that it is correct on return to DFSORT. If, for example, you first load register 1 and then restore all registers (including register 1), register 1 probably has the wrong contents.

Also, if you first restore all registers and then try to load register 1 from a changed base register, you might pass the wrong information to DFSORT.

---

### Correcting Common Errors

You can encounter abends or performance problems not caused by failures in the DFSORT program or the DFSORT Panels program. For example, your job might end abnormally because of system errors beyond your control. Abends can also result from specifying inadequate storage or incorrect storage parameters. You can sometimes improve degraded performance by allocating resources differently.

This section lists abends you might encounter when running DFSORT, and suggests ways of correcting the problems. Not every possible problem can be listed, but the list includes those most commonly reported to IBM personnel. Use the information here in conjunction with “Part 2. Messages and Codes” on page 13 and *System Codes*.

### Common System Abends Related to Storage

Table 2 on page 6 covers common system abend codes that might appear when abends result from errors in storage parameters. The items are listed by system abend code. For details on DFSORT’s use of storage and storage parameters, see *DFSORT Installation and Customization* and *DFSORT Application Programming Guide*.

## Common Errors

Table 2. Common System Abends Related to Storage

Abend Code	Problem
106	There was not sufficient storage to load a module. Ensure that the values for REGION, MAINSIZE, SIZE, RESALL or RESINV are sufficient.
804	The amount of storage requested exceeded that available. Ensure that the values for REGION, MAINSIZE, SIZE, RESALL or RESINV are sufficient.
878	User exits required more storage than was available. Ensure that the values for each user exit size in the MODS statement and for REGION, MAINSIZE, SIZE, RESALL or RESINV are sufficient.
80A	This abend can result from the following: <ul style="list-style-type: none"><li>• A DFSORT job required more storage than was available. Ensure that the values for REGION, MAINSIZE, SIZE, RESALL or RESINV are sufficient.</li><li>• The '-1' flag required to mark the end of an extended parameter list was missing. Include an F'-1' to mark the last item in the parameter list.</li><li>• The invoking program used AL3 to define addresses for DFSORT control statements. Be sure to use full 4-byte addresses to prevent DFSORT from "running off the end" while allocating storage.</li></ul>
D37	A SORTWK, SORTOUT, or OUTFIL data set was specified with insufficient primary space allocation and no secondary allocation. Increase the primary allocation, or include a value for secondary allocation.
E37	A SORTOUT or OUTFIL data set was allocated on a primary volume that had either insufficient storage or excessive fragmentation, and no secondary volume was specified. Specify more than one volume in the VOLSER parameter for the SORTOUT or OUTFIL data set or compress the volume to allow it to contain the full 16 extents.

## Miscellaneous System Abends

Table 3 covers other common system abend codes. The items are listed by system abend code.

Table 3. Miscellaneous System Abends

Abend Code	Problem
001	This abend can result from the following: <ul style="list-style-type: none"><li>• The data control block (DCB) for SORTIN contained incorrect LRECL or BLKSIZE data set attributes. Check the SORTIN DD statement to ensure that the LRECL and BLKSIZE parameters are correct.</li><li>• When a COBOL program invoked DFSORT:<ul style="list-style-type: none"><li>– The SELECT or ASSIGN statement specified an incorrect file name or DDNAME parameter.</li><li>– The SORTIN or SORTOUT files specified DCB attributes different from those defined for the data set.</li><li>– The FD file description for SORTIN or SORTOUT files contained inaccurate specifications (such as RECORD CONTAINS, BLOCK CONTAINS, RECORDING MODE, and so on).</li></ul></li></ul>

Table 3. Miscellaneous System Abends (continued)

Abend Code	Problem
002	<p data-bbox="699 258 1133 281">This abend can result from the following:</p> <ul data-bbox="699 296 1458 474" style="list-style-type: none"> <li data-bbox="699 296 1458 407">• For variable-length record SORTOUT or OUTFIL data sets, the record descriptor word was invalid (less than 4 bytes or greater than 32KB) or, for nonspanned records, was greater than the specified block size.</li> <li data-bbox="699 422 1458 474">• For fixed-length record SORTOUT or OUTFIL data sets, the LRECL parameter was greater than the BLKSIZE parameter.</li> </ul>
013	<p data-bbox="699 491 1325 514">The SORTIN, SORTOUT, or OUTFIL BLKSIZE parameter:</p> <ul data-bbox="699 529 1458 707" style="list-style-type: none"> <li data-bbox="699 529 1458 611">• Was greater than 32760 for a DASD data set, or was greater than the maximum block size supported by the access method for a tape data set.</li> <li data-bbox="699 625 1458 707">• Was not an integer multiple of the LRECL parameter for fixed-length records, or was not at least four bytes longer than the LRECL for variable-length records.</li> </ul>

## Common Errors

Table 3. Miscellaneous System Abends (continued)

Abend Code	Problem
0Cx	<p>These abends can result from the following:</p> <ul style="list-style-type: none"> <li>• Locale processing with a run-time library that contains down-level dynamically loadable routines.</li> <li>• An invalid PD or ZD sign (0-9) or digit (A-F) can result in a data exception (0C7 ABEND) for SUM processing. An invalid PD or ZD digit can result in a data exception for OUTFIL edit or statistics processing. ICETOOL's DISPLAY or VERIFY operators can be used to identify decimal values with invalid digits. ICETOOL's VERIFY operator can be used to identify decimal values with invalid signs.</li> <li>• A user exit passed back a variable-length record that was shorter than the record descriptor word indicated.</li> <li>• A user exit passed back a fixed-length record that was shorter than the length specified in the RECORD statement (L2 for E15 and E32, or L3 for E35).</li> <li>• The contents of register 1 (and possibly other registers) used by DFSORT were altered before returning to DFSORT. Ensure the registers are restored before returning to DFSORT.</li> <li>• The SORTLIB DD statement specified an incorrect library.</li> <li>• The DEBUG CTRx option was in effect and the criteria were met.</li> <li>• For COBOL: <ul style="list-style-type: none"> <li>– DISPLAY, EXHIBIT, or TRACE was used for COBOL messages, but the message data set was missing. Supply a message data set DD statement. (See the discussion of MSGDDN in <i>DFSORT Application Programming Guide</i> for an alternative solution.)</li> <li>– An attempt was made to reopen an open data set, or a data set was already open prior to executing the SORT USING statement.</li> <li>– A STOP RUN or GOBACK statement was required but not supplied; therefore, the COBOL program did not exit as intended but processed the next instruction.</li> <li>– Invalid logic was used in a sort I/O procedure (possibly using the STOP RUN, GOBACK, CALL, or GO TO statements), or PERFORM referred to a paragraph outside the section.</li> <li>– The file name or DDNAME parameter of the SELECT/ASSIGN statement was incorrect.</li> <li>– The LINKAGE SECTION was specified incorrectly or misplaced.</li> <li>– The PROCEDURE DIVISION USING statement gave an invalid parameter list.</li> <li>– An attempt was made to move data from the COBOL SD area prior to execution of the SORT verb.</li> <li>– The FD entry RECORD CONTAINS clause (LRECL) for the USING data set is different than the SD entry RECORD CONTAINS clause for the SORTWK data set.</li> </ul> </li> </ul>
213	<p>This abend can result from the following:</p> <ul style="list-style-type: none"> <li>• A specified data set was not found.</li> <li>• Space for the SORTOUT or OUTFIL data set was not allocated.</li> <li>• For COBOL: <ul style="list-style-type: none"> <li>– The file name or DDNAME parameter of the SELECT or ASSIGN statement was incorrect.</li> <li>– The SORT verb required the SORTLIB DD statement.</li> </ul> </li> </ul>

Table 3. Miscellaneous System Abends (continued)

Abend Code	Problem
30A	The SORTIN LRECL parameter was greater than the SORTOUT or OUTFIL LRECL parameter (for VSAM data sets).
513	For COBOL, the SORT USING statement was used with an input tape data set that was already opened prior to invoking DFSORT.
806	DFSORT required a program, such as a user exit or invoking program, during execution and was unable to find it. Correct the job step or supply the necessary program.  ABEND 806, along with message “CSV003I REQUESTED MODULE CEEBINIT NOT FOUND”, may occur if during locale processing DFSORT does not have access to the library containing the dynamically loadable routines. For example, the data set containing these routines might be called SYS1.SCEERUN. If you are unsure of the name of the data set, contact your system administrator.
813	The SORTIN, SORTOUT, or OUTFIL data set was tape, and the data set name was incorrect or missing or DISP=MOD was specified inappropriately for a tape data set.
913	For COBOL, the file name or DDNAME parameter of the SELECT or ASSIGN statement was incorrect.
C03	A user exit opened data sets, but did not close them upon completion.
C13	For COBOL, the file name or DDNAME parameter of the SELECT or ASSIGN statement was incorrect.

## User Abends at Run Time

DFSORT issues two types of user abends at run time:

- If ABEND or ABSTP is in effect and DFSORT terminates with an error message, you receive a user abend. The abend code is equal to either the error message number or to a number between 1 and 99 (defined during DFSORT installation with the ICEMAC ABCODE option). For user abend codes that are error message numbers, refer to the explanation for the corresponding message.  
For a tape work data set sort or Conventional merge, user abend 0 can be issued if DFSORT terminates with an error message and ABEND or ABSTP is in effect.
- If you receive a user abend with a code from 1000 through 1599, DFSORT detected an error in its internal logic. The abend is issued in this case to prevent infinite loops and to aid diagnosis of the problem. Please report such problems to your IBM representative.

In some cases, you may be able to bypass a DFSORT user 1nnn abend by making more storage available to DFSORT; try adding 32KB by changing the REGION parameter or the MAINSIZE or SIZE value. See *DFSORT Application Programming Guide* for ways to specify MAINSIZE and SIZE.

If DFSORT determines that a SmartBatch pipe data set is being used for input or output, it:

- Automatically forces the ABEND option on, to ensure that an ABEND will be generated if an error is detected, and
- If an E15, E32, or E35 user exit requests termination, terminates with user ABEND zero instead of return code 16.



## Common Errors

If ICETOOL determines that a SmartBatch pipe data set is being used for input or output, it automatically terminates with user ABEND 2222 instead of return code 12.

Generation of an ABEND in these situations allows for appropriate error propagation by the system to other applications that may be accessing the same SmartBatch pipe data set as DFSORT or ICETOOL.

If DFSORT or ICETOOL detects an error before determining that a SmartBatch pipe data set is being used or before opening the SmartBatch pipe data set, appropriate error propagation may not occur. This can cause another application to go into a permanent wait for a SmartBatch pipe data set.

## User Abends from DFSORT Panels

If you receive a user abend with a code of 400, 401, or 402, a failure has occurred in the DFSORT Panels interface with ISPF shown Table 4.

Table 4. User Abends from DFSORT Panels

Abend Code	Problem
400	<p>An unexpected return code was received from an ISPF service after DFSORT Panels called that service. Possible causes are:</p> <ul style="list-style-type: none"><li>• A DFSORT run-time abend (system or user) occurred after foreground execution was selected from DFSORT Panels. DFSORT run-time abends can cause problems with the ISPF environment, with unpredictable results.</li></ul> <p>To avoid DFSORT error message user abends when using DFSORT Panels, make sure that your site has selected ERET=RC16 for the ICEMAC TSO environment, and that the ABEND option is not specified at run time.</p> <ul style="list-style-type: none"><li>• An ISPF service that was called by DFSORT Panels did not complete as expected.</li><li>• An internal logic error in DFSORT Panels resulted in an incorrect call to an IBM service.</li></ul>
401	<p>An ISPF SELECT service severe error (SELECT return code of 20) occurred after DFSORT Panels used the ISPF SELECT service to invoke a user-defined data entry routine. See <i>DFSORT Installation and Customization</i> for more information about customizing DFSORT Panels.</p>
402	<p>An ISPF SELECT service severe error (SELECT return code of 20) occurred after DFSORT Panels used the ISPF SELECT service to invoke a user-defined statement coder routine. See <i>DFSORT Installation and Customization</i> for more information about customizing DFSORT Panels.</p>

If you receive a user abend with a code from 1600 through 1699, DFSORT Panels detected an error in its internal logic. The abend is issued in this case to prevent infinite loops and to aid diagnosis of the problem. Please report such problems to your IBM representative.

For information about other user abends displayed by ISPF, see *ISPF Dialog Management Guide*.

## Common Problems Reported by DFSORT Users

The situations below are problems that are frequently reported to IBM personnel.



- System messages indicate DFSORT is using more storage than specified in the DFSORT parameters.

DFSORT treats its storage parameters (for example, TMAXLIM, MAXLIM, SIZE, MAINSIZE) as flexible limits, and may, in fact, use more storage than the specified limits in some cases to optimize performance or prevent termination. To avoid storage fragmentation, DFSORT might request all storage allocated to it above and below 16MB virtual, then release all storage exceeding the specified limits.

DFSORT messages ICE092I and ICE093I list an estimate of the total amount of storage used by DFSORT. DFSORT message ICE156I lists an estimate of the storage used by DFSORT above 16MB virtual. The difference between the two is an estimate of the storage used by DFSORT below 16MB virtual. Message ICE231I lists an estimate of the storage used exclusively for OUTFIL processing above and below 16MB virtual.
- Elapsed time for executing sort jobs appears excessive.

To achieve the shortest elapsed time for sort jobs, DFSORT tries to select the Blockset technique. Examine message ICE143I to see which technique was used. If DFSORT did not select Blockset, code a SORTDIAG DD DUMMY statement in the JCL to request a set of diagnostic messages and re-run the job. Diagnostic message ICE800I reports a reason code indicating why DFSORT did not select the Blockset technique. Correct the condition, if possible.

Excessive elapsed time can also be the result of forcing DFSORT to sort a large data set in a relatively small amount of storage. Ensure that you provide sufficient storage for the data to be sorted as discussed in *DFSORT Application Programming Guide* and *DFSORT Tuning Guide*.

## Common Errors

---

## Part 2. Messages and Codes

<b>Chapter 2. Message Considerations</b> . . . . .	15
Printing Messages and Control Statements . . . . .	15
Printing EFS Messages and Control Statements . . . . .	16
Writing Messages to the Master Console . . . . .	16
Finding DFSORT Control Statement Coding Errors . . . . .	17
<b>Chapter 3. Informational and Error Messages</b> . . . . .	19
<b>Chapter 4. Diagnostic Messages.</b> . . . . .	69
<b>Chapter 5. Informational Messages for Debugging</b> . . . . .	71
<b>Chapter 6. Return Codes and User Abends</b> . . . . .	73
DFSORT Return Codes. . . . .	73
ICETOOL Return Codes . . . . .	73
ICEGENER Return Codes. . . . .	74



---

## Chapter 2. Message Considerations

This chapter discusses the types of messages DFSORT produces and explains how to print messages and control statements, write messages to the master console, and find control statement coding errors.

DFSORT produces three types of messages:

- **Critical error** messages, which report and describe problems requiring programmer action to correct (explained in “Chapter 3. Informational and Error Messages” on page 19).
- **Informational** messages, which provide information only and require no programmer action (explained in “Chapter 3. Informational and Error Messages” on page 19).
- **Diagnostic** messages, which provide additional information useful in debugging and tuning (explained in “Chapter 4. Diagnostic Messages” on page 69).

Different messages might be issued for the same error situation, depending upon which DFSORT technique is used.

See *DFSORT Installation and Customization* for information on the ICEMAC installation options discussed in this section. PARM options and program control statement options are explained in *DFSORT Application Programming Guide*.

---

## Printing Messages and Control Statements

You can select the type of messages you want written to the message data set during installation or at run time. Messages ICE097I and ICE158A are written only to the master console.

If ICEMAC option NOMSGDD=QUIT is in effect, you must provide a DD statement using the name of the alternate message data set specified during installation or at run time, or a SYSOUT DD statement. Otherwise, DFSORT terminates with a return code of 20.

The following list outlines the hierarchy for message and control statements for the Blockset and Peerage/Vale sorting techniques. SORTDIAG DD, MSGPRT, and LIST, as they are used here, represent all methods of specifying the appropriate function. For example, FLAG (I) in the EXEC PARM field is equivalent to MSGPRT=ALL, and DIAGSIM=YES in ICEMAC is equivalent to specifying a SORTDIAG DD statement for every job.

### **SORTDIAG DD** statement

Prints all messages (including diagnostic messages); prints control statements. A message data set DD statement should be provided.

When SORTDIAG DD is specified, Blockset error messages that are normally suppressed are printed. This may result in an error message being printed for a run that is actually successfully completed by another technique (as indicated by a return code of 0 or 4).

### **MSGPRT=ALL** and **LIST** in effect

Prints all messages except diagnostic messages; prints control statements. A message data set DD statement should be provided.

## Message Considerations

### **MSGPRT=ALL** and **NOLIST** in effect

Prints all messages except diagnostic messages; does not print control statements. A message data set DD statement should be provided.

### **MSGPRT=CRITICAL** and **LIST** in effect

Prints only critical error messages; prints control statements. A message data set DD statement should be provided.

Control statements are not printed if **MSGPRT=CRITICAL** is in effect and **DFSORT** is dynamically invoked using the 24-bit parameter list.

### **MSGPRT=CRITICAL** and **NOLIST** in effect

Prints only critical error messages; does not print control statements. A message data set DD statement should be provided.

### **MSGPRT=NONE** in effect

Does not print messages; does not print control statements.

---

## Printing EFS Messages and Control Statements

Both the Extended Function Support (EFS) program and its exit routines (EFS01 and EFS02) can send messages to DFSORT for printing. The EFS program can send messages to DFSORT for printing at Major Call 1, 2, 3, or 4. EFS01 and EFS02 can only send messages to DFSORT for printing at Major Call 4. DFSORT prints EFS messages to the message data set, but does not write messages to the master console.

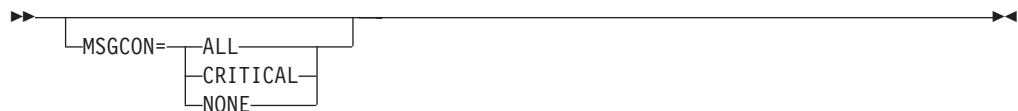
Use the **MSGPRT** option to control the printing of messages returned by the EFS program. Use the **LISTX** and **NOLISTX** options to control the printing of control statements returned by the EFS program; use these two options in the same way you use **LIST** and **NOLIST** to control printing of DFSORT control statements. **LISTX** and **NOLISTX** can be used independently of **LIST** and **NOLIST**. When a **SORTDIAG** DD statement is specified, **LISTX** is used. If no verb request list is returned at Major Call 1, **NOLISTX** is used.

See *DFSORT Application Programming Guide* for more information about Extended Function Support.

---

## Writing Messages to the Master Console

You can select the type of messages to be written to the master console with ICEMAC option **MSGCON**:



**ALL** Specifies that all messages except option-in-effect messages (ICE128I through ICE133I) and diagnostic messages (ICE800I through ICE999I) are written to the master console.

### **CRITICAL**

Specifies that only critical messages are written to the master console.

**NONE** Specifies that messages are not written to the master console.

Console message choices are independent of the message data set choices. Messages ICE0971 and ICE158A are written only to the master console. Inclusion of a SORTDIAG DD statement has no effect on console messages.

Because of the addition of the job name and the step name, some DFSORT messages can be truncated from the right when they are written to the master console, if you change the text in ICEMSGs. See *DFSORT Installation and Customization* for more information on how to change the text in ICEMSGs.

Messages returned by an EFS program cannot be written to the master console.

---

## Finding DFSORT Control Statement Coding Errors

DFSORT scans control statements for syntax and content errors, such as invalid operators, operands, or operand values, missing operands, and missing or invalid delimiters.

As explained in “Printing Messages and Control Statements” on page 15 and “Printing EFS Messages and Control Statements” on page 16, DFSORT prints messages and control statements as requested by the SORTDIAG DD statement, MSGPRT, LIST, and LISTX operands. The following paragraphs assume that messages and control statements are to be printed.

DFSORT scans each control statement for errors, and prints an error message for the first error detected. Scanning stops at the first error, and then continues with the next control statement. If DFSORT detects an error, it usually terminates after all the control statements are scanned. However, in some cases, detection of an error by the Blockset technique results in the use of another technique that could avoid termination, and, in other cases, Blockset terminates immediately without completing control statement scanning.

If an error is associated with a specific operator, operand or value for a control statement in DFSPARM, SYSIN, and SORTCNTL, DFSORT prints a marker (\$) directly below the control statement near the error. If an error is associated with a specific control statement, DFSORT prints the error message directly below the marker line or control statement.

If an error is associated with a specific control statement returned by an Extended Function Support (EFS) program, the EFS program can return a message to be printed with that control statement. The associated message must be returned at the same time the control statement is returned to DFSORT. The EFS program can also return an offset pointing to the error found in the control statement. DFSORT uses this offset to place a marker (\$) below the control statement. Any message associated with the control statement is printed directly below the marker line or control statement. See the *DFSORT Application Programming Guide* for more information about Extended Function Support.





---

## Chapter 3. Informational and Error Messages

DFSORT messages can appear within your printed output or on the master console. The formats are:

For printed messages

ICE $nnn$ s  $c$  text

For console messages

ICE $nnn$ s  $c$   $jjjjjjj$ . $sssssss$  text

### Where Represents

$nnn$  Message number.

$s$  Severity code. Values for  $s$  are:

$A$  Critical error message; requires programmer action.

$I$  Informational message; usually requires no programmer action.

$c$  Diagnostic trace character. This character is usually not needed, but might be requested by your IBM representative for diagnostic purposes.

$jjjjjjj$  Job name.

$sssssss$   
Step name.

$text$  Message text.

---

**ICE000I** - CONTROL STATEMENTS FOR  
5740-SM1, DFSORT REL  $rr.r - hh:mm$   
ON  $day mmm dd, yyyy -$

**Explanation:** Identifies the following for this DFSORT run:

- Release level ( $rr.r$ )
- Time as hours ( $hh$ ) and minutes ( $mm$ )
- Day of the week ( $day$ )
- Date as month ( $mmm$ ), day of the month ( $dd$ ) and year ( $yyyy$ )

**System Action:** None.

**Programmer Response:** None.

---

**ICE001A** TEXT BEGINS IN WRONG COLUMN

**Explanation:** Critical. A continuation line following a line broken at a comma did not begin within columns 2 through 71; or a continuation line following a line broken at column 71 (with a nonblank entry in 72) did not begin in column 16.

**System Action:** The program terminates.

**Programmer Response:** Check continuation lines for text beginning in a wrong column.

---

**ICE002I** DUPLICATE OR CONFLICTING  $xxxxxxx$   
STATEMENT

**Explanation:** The same operation definer, or mutually exclusive operation definers (SORT and MERGE, or INCLUDE and OMIT), appeared more than once in the same source (for example, SORTCNTL).

**System Action:** DFSORT does not analyze duplicate or conflicting statements. The first one encountered is used.

**Programmer Response:** No action is necessary, if the parameters on the duplicate (second and subsequent) control statements are not needed. Otherwise, consolidate the parameters from the duplicate control statements into a single control statement.

---

**ICE003A** CONTINUATION LINE MISSING

**Explanation:** Critical. A continuation line was indicated by the previous line ending with a comma, or with a

## Informational and Error Messages

nonblank entry in column 72, but no line followed.

**System Action:** The program terminates.

**Programmer Response:** Check for an overflow of parameters into column 72 or a missing continuation line.

---

### ICE005A STATEMENT DEFINER ERROR

**Explanation:** Critical. A control statement did not contain one of the acceptable operation definers (ALTSEQ, DEBUG, END, INCLUDE, INREC, MERGE, MODS, OMIT, OPTION, OUTFIL, OUTREC, RECORD, SORT, SUM). You can also receive this message for continuation lines after a line that has an error.

**System Action:** The program terminates.

**Programmer Response:** Check all statements for incorrect, misplaced, or misspelled operation definers. Check that no definer begins in column 1 (in which case it was treated as a label). If you have a label, check that it begins in column 1 (otherwise, it was treated as an operation definer). Check that no definer or DFSPARM PARM option is continued using the continuation column.

---

### ICE006A OPERAND DEFINER ERROR

**Explanation:** Critical. The first operand of a control statement did not begin on the same line as the operation definer, or an operand or operand value was not valid.

**System Action:** The program terminates.

**Programmer Response:** Check for statements that contain invalid operands, invalid operand values, or no operands.

---

### ICE007A SYNTAX ERROR

**Explanation:** Critical. A control statement contained an error in syntax.

**System Action:** The program terminates.

**Programmer Response:** Check the control statements for syntax errors. Some of the more common syntax errors are:

- Unbalanced parenthesis
- Missing comma
- Embedded blank
- Invalid format type
- Invalid operator
- Invalid constant
- Continuation of DFSPARM PARM options using the continuation column.
- Symbol used where it is not allowed

---

### ICE008A FIELD VALUE EXCEEDS MAXIMUM ALLOWABLE CHARACTERS

**Explanation:** Critical. A specified parameter value was too long. Most parameter values have a maximum length of eight characters, but some can have longer or shorter maximum lengths.

**System Action:** The program terminates.

**Programmer Response:** Check control statements for parameter values that are too long. Refer to the description of each parameter to verify the maximum length for its values.

---

### ICE010A NO SORT OR MERGE CONTROL STATEMENT

**Explanation:** Critical. All control statements have been processed and either:

- No SORT or MERGE control statement or OPTION COPY statement was found, or
- A 24-bit parameter list did not contain a SORT or MERGE control statement.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the 24-bit parameter list contains a SORT or MERGE control statement. If you are not using a 24-bit parameter list, make sure you have supplied a SORT or MERGE control statement or an OPTION COPY statement.

---

### ICE011A DUPLICATE OR CONFLICTING OPERANDS ON THE OPTION STATEMENT

**Explanation:** Critical. On an OPTION control statement, one of the following errors was found:

- A keyword was specified twice.
- A keyword and a variation of it were both specified. CKPT and CHKPT are variations, as are FILSZ and SIZE.
- A keyword and its opposite were both specified. EQUALS and NOEQUALS are examples of this.

**System Action:** The program terminates.

**Programmer Response:** Check the OPTION control statement for the errors indicated in the explanation and correct the errors.

---

### ICE012A MISSING FIELDS OPERAND DEFINER

**Explanation:** Critical. A SORT, MERGE, INREC, OUTREC, or SUM control statement did not contain the FIELDS operand.

**System Action:** The program terminates.

**Programmer Response:** Check for a SORT, MERGE, INREC, OUTREC, or SUM control statement that does not have the FIELDS operand.

---

**ICE013A    INVALID SORT OR MERGE  
STATEMENT OPERAND**

**Explanation:** Critical. An invalid keyword operand was detected on a SORT or MERGE control statement.

**System Action:** The program terminates.

**Programmer Response:** Make sure the SORT or MERGE control statement does not contain an invalid keyword operand.

---

**ICE014A    DUPLICATE SORT OR MERGE  
STATEMENT OPERAND**

**Explanation:** Critical. On a SORT or MERGE control statement, one of the following errors was found:

- A keyword was specified twice.
- A keyword and a variation of it were both specified. CKPT and CHKPT are variations, as are FILSZ and SIZE.
- A keyword and its opposite were both specified. EQUALS and NOEQUALS are examples of this.

**System Action:** The program terminates.

**Programmer Response:** Check the SORT or MERGE control statement for the errors indicated in the explanation and correct the errors.

---

**ICE015A    VARIABLE RECORD TOO SHORT**

**Explanation:** Critical. This message was issued for one of the following reasons:

- An input data set contained an invalid RDW or BDW.
- A record passed to DFSORT from user exit E15 or E35 contained an invalid RDW.
- A variable-length record was too short to contain all fields or, if Blockset was not selected, shorter than L4.
- A record did not contain the first byte of the first control field when VLSHRT was in effect and Blockset was not selected.

**System Action:** The program terminates.

**Programmer Response:** If the records are valid but too short to contain all fields, use the VLSCMP option (INCLUDE, OMIT) or VLSHRT option (INCLUDE, OMIT, SORT, MERGE) to prevent DFSORT from terminating. Otherwise, take one or more of the following actions:

- Check that the input data set contains valid BDWs.
- Check that the input data set and all records passed from exits contain variable-length records with valid RDWs.
- Decrease L4 if it is too large.
- Check the input data set and all records inserted at user exits E15 and E35 to see that all records contain all fields.
- Remove any records that are too short.
- Check your E15 and E35 routines and correct any errors.

If ICETOOL was used, check each ON field of the failing operator for incorrectly specified displacement (position) or length.

You can use the ICETOOL STATS operator with ON(VLEN) to show the length of the shortest variable length record. You can use the ICETOOL DISPLAY operator with ON(VLEN) to show the length of each variable length record.

---

**ICE016A    INVALID FIELDS OPERAND VALUE**

**Explanation:** Critical. A FIELDS operand on a SORT or MERGE control statement contained an invalid number of values.

**System Action:** The program terminates.

**Programmer Response:** Check for valid formats of the FIELDS operand:

**FIELDS**=(location,length,format,order...)

or

**FIELDS**=(location,length,order...),**FORMAT**=format

---

**ICE017A    CONTROL FIELD DISPLACEMENT OR  
LENGTH VALUE ERROR**

**Explanation:** Critical. An invalid length or displacement (position) value was specified in a control field, or VLSHRT was in effect and the maximum length plus displacement was exceeded for a control field.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the length and position values in the FIELDS operand of a SORT or MERGE control statement were specified correctly. Make sure that the position plus length of each field did not exceed 4093, and that bit positions and lengths were specified for binary fields only, and did not exceed 7. If VLSHRT was specified for a variable-length application, make sure that the position plus length of each field did not exceed 4086; control field manipulation or overlapping can reduce the limit of 4086. If EQUALS was in effect and Blockset was not selected, the maximum length value plus position value that you can specify is reduced by 4.

---

**ICE018A    INVALID FORMAT**

**Explanation:** Critical. This message was issued for one of the following reasons:

1. More than 112 control fields were specified, and Blockset was not selected.
2. A SORT, MERGE, OUTFIL, SUM, INCLUDE, or OMIT statement contained an invalid or missing format type.
3. For a SORT or MERGE control statement, the format was invalid for the length specified.
4. CSF, FS, Y2x or PD0 format was specified for SORT or MERGE and Blockset was not selected.

## Informational and Error Messages

5. D2 format was specified in the INCLUDE or OMIT parameter of an OUTFIL statement.

**System Action:** The program terminates.

**Programmer Response:** For cases 1 through 3, check that each format type is valid for the control statement specified.

For case 4, rerun the job with a SORTDIAG DD statement to get message ICE800I, which indicates the reason Blockset could not be used. If possible, remove the condition preventing the use of Blockset.

For case 5, use an INCLUDE or OMIT statement or change the D2 format to a format that is valid for the INCLUDE or OMIT parameter of an OUTFIL statement.

---

### ICE019A MEMBER NAME NOT SPECIFIED FOR *ddname*

**Explanation:** Critical. A data set indicated by the specified ddname is partitioned, but the member name was missing.

**System Action:** The program terminates.

**Programmer Response:** Ensure that a member name is specified for each partitioned data set or PDSE defined for the indicated ddname. If the ddname is a concatenation of data sets, then ensure that each partitioned data set or PDSE in the concatenation has a member name specified with it.

---

### ICE020A INVALID RECORD STATEMENT OPERAND

**Explanation:** Critical. An invalid keyword was found in a RECORD control statement.

**System Action:** The program terminates.

**Programmer Response:** Check for invalid keywords.

---

### ICE021A NO TYPE OPERAND

**Explanation:** Critical. RECORD TYPE was not specified for a tape work data set sort or Conventional merge application that uses one of the following:

- VSAM input
- an E15 exit that supplies all of the input records
- an E32 exit that supplies all of the input records.

**System Action:** The program terminates.

**Programmer Response:** Specify RECORD TYPE=F or RECORD TYPE=V as appropriate.

---

### ICE022A RECORD FORMAT NOT F, V OR D

**Explanation:** Critical. The TYPE operand of a RECORD control statement contained an invalid value.

**System Action:** The program terminates.

**Programmer Response:** Check that the TYPE operand value on the RECORD statement is F, FB, V, VB, D, or DB. Check also for a conflict between the SORTIN/SORTOUT DCB RECFM parameter and the RECORD control statement.

---

### ICE023A NO LENGTH OPERAND

**Explanation:** Critical. The input record length was not available. A RECORD statement with an L1 or L2 value is required when an E15 or E32 user exit supplies all of the input records, or when a temporary empty VSAM data set defined without a RECSZ value is used for input.

This message can also be issued if input processing is bypassed because of a previous error.

**System Action:** The program terminates.

**Programmer Response:** Check for:

- A missing RECORD statement
- A RECORD statement without a LENGTH operand or without an L1 value
- A missing SORTIN DD statement or SORTINnn DD statement
- A temporary empty input VSAM data set defined without a RECSZ value

---

### ICE024A RECORD LENGTH VALUE ERROR

**Explanation:** Critical. An incorrect or unsupported value was associated with either:

- The LENGTH operand of a RECORD control statement, or
- The input or output length values obtained from SORTIN/SORTINnn or SORTOUT/OUTFIL respectively.

**System Action:** The program terminates.

**Programmer Response:** Some of the more common errors are:

- Entry errors. Length values must not contain nonnumeric characters, negative numbers, more than eight characters, a nonprintable character, and so forth.
- Specified or calculated minimum length (L4) was greater than maximum length (L2) or average length (L5).
- Average length (L5) is greater than maximum length (L2).
- No LENGTH was specified either for the operand or the logical record length on the SORTIN DD statement.
- All records are supplied by an E15 or E32 user exit and no L1 or L2 operand is specified.
- The input record length (L1), the maximum length (L2), or the output record length (L3) exceeds the maximum record length that DFSORT can handle.
- The OUTFIL input record length was greater than 32767.

- SOLRF was in effect and the OUTREC or INREC record length was greater than 32767, or was greater than 32760 for fixed-length output records or 32756 for variable-length output records.

---

### ICE025A RECORD COUNT OFF

**Explanation:** Critical. The program has compared the count of input records and output records (shown in message ICE054I), taken into account the number of records inserted or deleted (shown in message ICE055I), if any, and found a discrepancy.

The message is issued when the entire output data set has been written. The message is suppressed if NOCHECK was in effect and you had an E35 user exit and no SORTOUT or OUTFIL data set.

**System Action:** The program terminates.

**Programmer Response:** The most probable cause is that you have not specified a SORTOUT or OUTFIL data set, have specified E35, and from your E35 routine have passed a return code of 8 (do not return) too early, when there are still output records remaining. If this is the case, you can avoid receiving this message by specifying OPTION NOCHECK. If this is not the case, examine any user exit routines (especially E15 and E35) for possible return code or other errors.

If a COBOL-invoking program contains an output procedure, make sure the RETURN statement is iterated until the AT END condition is executed. This occurs the next time the RETURN statement is executed *after* the last record was returned to you.

If the iteration of the RETURN is controlled by a PERFORM statement, the PERFORM logic should be controlled by the execution of the AT END clause of the RETURN statement.

It is possible but less likely that the error was caused by an internal sort problem.

---

### ICE026I SMF RECORD NOT WRITTEN TO THE SMF DATA SET (RC=*nn*)

**Explanation:** A nonzero return code was returned from system management facilities (SMF). *nn* is the return code from the SMFWTM macro, in decimal.

**System Action:** The SMF type 16 record is not successfully written.

**Programmer Response:** Have your system programmer determine whether SMF is properly initialized on your system by referring to *SPL: System Management Facilities* for the meaning of the return code from SMFWTM.

---

### ICE027A END OF [*ddname*] FIELD BEYOND MAXIMUM RECORD LENGTH

**Explanation:** Critical. If *ddname* is blank, then:

- The OUTREC or INREC length exceeded 32767 bytes, or
- if INREC was specified, a SORT, MERGE, OUTREC, or SUM field was defined as extending beyond the maximum reformatted record length.
- otherwise, a SORT, MERGE, INREC, OUTREC, SUM, INCLUDE, or OMIT field was defined as extending beyond the maximum input record length

Otherwise, *ddname* indicates the first data set in the OUTFIL group for which either:

- an OUTFIL field was defined as extending beyond the maximum OUTFIL input record length, or
- the OUTFIL OUTREC length exceeded 32767 bytes

**System Action:** The program terminates.

**Programmer Response:** Check all relevant control statements for incorrectly specified field position or length values. Check the record statement for incorrectly specified length values.

If an EFS field is specified in the SORT or MERGE control statement, make sure the position begins within the record.

If ICETOOL is used, check each ON field of the failing operator for incorrectly specified position or length values, and for a DISPLAY, OCCUR, RANGE, STATS, UNIQUE, or VERIFY operator, check that the LRECL of the FROM data set is at least 4.

---

### ICE028A TOO MANY EXITS

**Explanation:** Critical. A MODS statement specified more than the maximum number of user exits.

**System Action:** The program terminates.

**Programmer Response:** Make sure that routines are specified for valid user exits only, and that each exit is associated with only one routine. Exits that can be specified in the MODS statement are E11, E15, E16, E17, E18, E19, E31, E35, E37, E38, E39, and E61.

**Note:** For a merge-only application, only exits E31, E35, E37, E38, E39, and E61 can be specified.

---

### ICE029A IMPROPER EXIT

**Explanation:** Critical. This message was issued for one of the following reasons:

- An incorrect user exit was specified on a MODS control statement.
- A user exit in the sort or intermediate merge phase of the program was specified for a merge application.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the MODS control statement does not contain errors that resulted



## Informational and Error Messages

in the specification of an invalid program user exit number. Numbers that can be specified are E11, E15, E16, E17, E18, E19, E31, E35, E37, E38, E39, and E61.

**Note:** For a merge-only application, only exits E31, E35, E37, E38, E39, and E61 are valid.

---

### ICE030A MULTIPLY DEFINED EXITS

**Explanation:** Critical. This message was issued for one of the following reasons:

- A user exit was defined twice in a MODS control statement.
- An E15, E18, E35, or E39 user exit was specified both in the invoking program parameter list and in a MODS statement.

**System Action:** The program terminates.

**Programmer Response:** Check MODS statement for multiply defined user exits.

---

### ICE031A INVALID MODS OP CHAR

**Explanation:** Critical. A parameter of a MODS control statement contained an invalid character.

**System Action:** The program terminates.

**Programmer Response:** Check the parameters of the MODS control statement for a length field containing something other than numeric data, a source or name field beginning with something other than an alphabetic character, or containing a special character other than \$, @, or #.

---

### ICE032A EXIT E61 REQUIRED

**Explanation:** Critical. A SORT or MERGE control statement defined a control field to be modified by a user-written routine (this was done by specifying E for the control field sequence indicator), and user exit E61 was not activated by a MODS control statement.

**System Action:** The program terminates.

**Programmer Response:** Check for conflicts in the SORT or MERGE and MODS statements with respect to E and E61.

---

### ICE033A CONTROL FIELD SEQUENCE INDICATOR E REQUIRED

**Explanation:** Critical. User exit E61 was activated and no control fields were specified for user modification (an E control field sequence parameter was missing on a SORT or MERGE control statement).

**System Action:** The program terminates.

**Programmer Response:** Check for conflicts in the SORT or MERGE and MODS statements with respect to E and E61.

---

### ICE034A MODS STATEMENT OPERAND ERROR

**Explanation:** Critical. This message was issued for one of the following reasons:

- An incorrect number of parameters followed an operand defined on a MODS control statement.
- SYSIN was specified as the third parameter for a user exit whose fourth parameter was N or C.
- For those user exits pre-link-edited by the user (fourth parameter was N or C, or null), the third parameter (library ddname) was different for two or more exits.
- An invalid value was specified for the fourth parameter of the statement.
- E15=COB or HILEVEL=YES was specified and E15 was specified with T as the fourth parameter.
- E35=COB or HILEVEL=YES was specified and E35 was specified with T as the fourth parameter.
- S was specified for the fourth parameter, but the user exit was not E11 or E31.

**System Action:** The program terminates.

**Programmer Response:** Make sure that:

- Each exit operand of the MODS statement is specified using correct syntax.
- SYSIN is not specified for any user exit with the fourth parameter as N or C.
- T has not been specified as the fourth parameter for an E15 or E35 user exit along with E15=COB or E35=COB, respectively, or HILEVEL=YES.
- S has not been specified for the fourth parameter for an user exit other than E11 or E31.
- The dynamic link-edit parameters (SYSIN, T, and S) have not been specified for a copy application.
- The third parameter is specified if S or T is used for the fourth parameter.
- The third parameter is specified for a conventional merge or tape work data set sort.

---

### ICE035A DUPLICATE MODS ROUTINE OPERAND

**Explanation:** Critical. The same user-written routine was used for more than one user exit in a DFSORT program phase, or two or more routines had the same name.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the MODS control statement does not use duplicate names.

---

### ICE036I and ICE037I

**Explanation:** See "Chapter 4. Diagnostic Messages" on page 69.

---

### ICE038I      NMAX APPROXIMATELY = *n*

**Explanation:** This message gives an estimate of the maximum number of records that can be sorted using the intermediate storage and main storage available to DFSORT for the current application. The number replaces *n* in the text of the message as shown above. Nmax is calculated assuming that 2400-foot tapes are used. For variable-length records, the value is based on maximum record length.

**System Action:** None.

**Programmer Response:** None.

---

### ICE039A      INSUFFICIENT MAIN STORAGE - ADD AT LEAST *n*K BYTES

**Explanation:** Critical. There was not enough main storage available for DFSORT to run, or main storage was fragmented.

**System Action:** The program terminates.

**Programmer Response:** The message gives an estimate of how much more main storage is needed. Add at least that amount to the main storage already allocated to the program by recoding the REGION parameter or the MAINSIZE/SIZE value (see *DFSORT Application Programming Guide* for ways to specify this).

This message may sometimes be issued again even after you have provided more storage because DFSORT re-estimates the amount of storage it needs at several points in the program.

Storage requirements can be reduced by decreasing either the input block size or the number of intermediate storage areas. See also messages ICE092I, ICE093I, ICE156I and ICE231I.

Make sure that storage allocation is permitted above 16MB virtual and enough storage is available above 16MB virtual to load the DFSORT modules.

If you are using OUTFIL processing, increase the REGION value or decrease the number of OUTFIL data sets or their block sizes; do not increase the MAINSIZE/SIZE value.

---

### ICE040A      INSUFFICIENT WORK UNITS

**Explanation:** Critical. There were not enough work data sets to allow your application to run. This can occur when work data sets are on tape. In a merge-only application, this message can result from incorrectly specifying one or more input data sets (SORTIN01, and so forth) or from using an E32 user exit without specifying the number of data sets.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the DD statements do not contain errors. For a sort using tape

work files, the SORTWKdd numbers must be in sequence, starting with SORTWK01, and at least three work data sets must be assigned to the program. For a Conventional merge application, make sure that the numbers for the SORTINnn DD statements are in sequence, starting with SORTIN01.

---

### ICE041A      N GT NMAX

**Explanation:** Critical. The exact number of records specified in a FILSZ or SIZE operand was greater than the maximum sort capacity calculated by the program.

**System Action:** The program terminates.

**Programmer Response:** Check all FILSZ or SIZE operands for errors. If all operands are correct, check DD statements for an error in assigning work data sets. If DD statements are correct, assign more work data sets to the program.

---

### ICE042A      UNIT ASSIGNMENT ERROR *ddname*

**Explanation:** Critical. This message was issued for one of the following reasons:

1. Both a VSAM SORTINnn DD statement and a non-VSAM SORTINnn DD statement were specified for a merge application.
2. Input consisted of concatenated VSAM and non-VSAM data sets.
3. A dummy, spool or pipe data set, an HFS data set, or an HFS file, was specified as a work data set.
4. If *ddname* is DYNALLOC, the specified device type or number of work data sets was invalid.
5. DFSORT was invoked via the ICEGENER facility, but SYSIN was not DUMMY or the SYSIN *ddname* was changed. Control was automatically transferred to the IEBGENER system utility.
6. Both tape and DASD work devices were specified.
7. A SmartBatch pipe data set was specified for SORTIN, SORTINnn or SORTOUT, and Blockset was not selected.
8. An HFS file was specified for input or output, but the system does not support HFS files.
9. An HFS data set (not file) was specified for input or output.

**System Action:** The program terminates.

**Programmer Response:** For case 1, use either VSAM data sets or non-VSAM data sets for a merge application, but not both.

For case 2, use either a single VSAM data set or concatenated non-VSAM data sets for a copy or merge application.

For Case 3, remove or replace the invalid work data set.

For case 4, check that the device type specified is supported by the program and available at your site (see *DFSORT Application Programming Guide* for a list of direct access devices). Also check whether you have

## Informational and Error Messages

exceeded the maximum number of areas permitted for the storage type used.

For case 5, no action is necessary.

For case 6, specify either all tape or all DASD work data sets.

For case 7, rerun the job with a SORTDIAG DD statement to get message ICE800I, which indicates the reason Blockset could not be used. If possible, remove the condition preventing the use of Blockset.

For Case 8, remove the HFS file or ensure that the system supports HFS files.

For Case 9, remove the HFS data set or specify an HFS file.

---

### ICE043A INVALID DATA SET ATTRIBUTES SPECIFIED *ddname yyyyyyyy*

**Explanation:** Critical. *ddname* is the name of the DD statement in error; *yyyyyyy* is the error description.

One of the following conditions was encountered:

1. DD statements that defined input and output data sets contained information conflicting with each other, with information on the data set labels, or with the default values assumed for DCB subparameters by the program (see *DFSORT Application Programming Guide* for information on DCB subparameters).
2. The RECFM, LRECL or BLKSIZE was not available for an input data set.
3. A DD statement for input or output specified a cataloged DASD data set that does not exist on the volume pointed to by the catalog entry.
4. Input data sets for a merge contained different record lengths.
5. Blockset was not selected for a merge, and the largest block size was not specified for SORTIN01.
6. For a tape work data set sort, the first data set of the concatenation did not specify the largest block size.
7. A tape data set with the largest block size was not first in the SORTIN concatenation, BLKSIZE was not specified in the DD statement for that tape data set, and that tape data set was not DFSMSrmm-managed.
8. The record format of the input or output data set was U (undefined).
9. For a tape work data set sort of fixed length records, the LRECL of the input or output data set was larger than 32752 (or 32748 with EQUALS in effect).
10. LRECL=X was used for an input or output data set.
11. The track capacity of the SORTOUT device was smaller than the SORTOUT block size.
12. A variable-length VSAM data set was processed with the wrong RECFM.
13. The RECFM, LRECL and BLKSIZE were not available for a DUMMY or NULLFILE input data set.

**System Action:** The program terminates.

**Programmer Response:** For case 1, check DD statements for input and output data sets for conflict in the BLKSIZE (block size), RECFM (record format), and LRECL (logical record length) subparameters. Unless OUTFIL is used to convert variable input to fixed output, the output data set must be the same record type (fixed or variable) as the input data sets.

When using fixed-length records without E15 or E35 user exits, the following are true:

- For a merge application, SORTOUT LRECL (or RECORDSIZE for VSAM data sets) must not exceed either the SORTINnn LRECL (RECORDSIZE for VSAM data sets) or the reformatted record length.
- For a sort application, except with Blockset, SORTOUT LRECL (or RECORDSIZE for VSAM data sets) must not exceed SORTIN LRECL (or RECORDSIZE for VSAM data sets).

When using fixed-length records for a sort application, except with Blockset, the SORTOUT LRECL (or RECORDSIZE for VSAM data sets) must not exceed the reformatted record length.

For case 2, specify the RECFM, LRECL or BLKSIZE as appropriate on the input DD statement.

For case 3, check the volume serial numbers of the input data sets.

For case 4, ensure that all input data sets have the same record length.

For case 5, check that the SORTIN01 data set has the largest block size.

For case 6, ensure that the first data set of the concatenation has the largest block size.

For case 7, ensure that BLKSIZE is specified in the DD statement for the tape data set with the largest block size.

For case 8, ensure that the record format of the input and output data sets is not U (undefined).

For case 9, either reduce the input/output LRECL size, or replace the tape work data sets with DASD work data sets so the Blockset technique can be used.

For case 10, specify the actual maximum LRECL instead, or if the LRECL is greater than the maximum allowed, decrease it.

For case 11, ensure that the SORTOUT block size is appropriate for the SORTOUT device

For case 12, specify a RECORD statement with TYPE=F to process the VSAM records as fixed-length



or TYPE=V to process the VSAM records as variable-length, as appropriate.

For case 13, specify the RECFM, LRECL and BLKSIZE on the input DD statement for the DUMMY or NULLFILE input data set.

### ICE044I      EXIT E $nn$ INVALID OPTION

**Explanation:** An invalid input/output option was passed to DFSORT at user exit E18, E19, E38, or E39. The  $nn$  value in the above message text is replaced by the number of the user exit at which the error occurred.

**System Action:** The invalid option is ignored.

**Programmer Response:** Check the parameter list passed by the user-written routine against the table in "Appendix A. Valid Input/Output Options at User Exits—ICE044I" on page 113 before rerunning the application. An  $x$  in the table indicates an option that is allowed with the user exit in question.

### ICE045I      END SORT PH

**Explanation:** The sort (input) phase completed successfully. This message appears only when the BALN or POLY tape technique was used.

### ICE046A      SORT CAPACITY EXCEEDED - RECORD COUNT: $n$

**Explanation:** Critical. DFSORT was not able to complete processing with the intermediate storage available (Hiperspace or DASD work data sets). For work data sets with secondary allocation allowed, DFSORT overrides system B37 abends and continues processing; this message is issued only when no more space is available in Hiperspace or on any allocated work data set.

**Note:** DFSORT uses only the first volume of multi-volume work data sets.

The count  $n$  is either an approximation of the number of records or is the total number of records that DFSORT was able to read in before it used all of the available intermediate storage.

The amount of intermediate storage required can vary depending on many factors including:

- The amount of Hiperspace DFSORT is able to use at the time the sort is run
- The amount of main storage available
- The degree of randomness of the records to be sorted
- The values specified (or defaulted) for options such as FILSZ/SIZE, AVGRLEN, or DYNALLOC.
- The amount of padding required for short records when VLSHRT is in effect.

**System Action:** The program terminates.

**Programmer Response:** Take one or more of the following actions:

- If appropriate, increase the amount of main storage available to DFSORT using the options MAINSIZE/SIZE or the JCL option REGION. Increasing the amount of main storage available to DFSORT can help DFSORT use less intermediate storage. Avoid running a large sort in a small amount of main storage.
- If dynamic allocation was used, ensure that the values for the options DYNALLOC, FILSZ/SIZE, and AVGRLEN are appropriate. If necessary, specify these options. In particular, if your actual average input record length is significantly different from one-half of the LRECL, specify the actual average record length with AVGRLEN.
- If VLSHRT was in effect and the total size of all control fields was significantly larger than the average LRECL for the data set, you may be able to reduce the amount of work space required by reducing the total size of the control fields.
- If JCL work data sets were used, increase the amount of work space available to DFSORT.

### ICE047A      RECORD COUNT OFF, SPECIFIED $n$ , RECEIVED $n$

#### RCD COUNT OFF

**Explanation:** Critical. Installation option FSZEST=NO was specified or defaulted, and the number of records received was NOT equal to the value specified in the FILSZ= $x$  or SIZE= $y$  option in effect.

The SPECIFIED field contains the value specified in the FILSZ= $x$  or SIZE= $y$  option. The RECEIVED field contains the actual number of records in the input data sets (for SIZE= $y$ ) or the actual number of records to be sorted or merged (for FILSZ= $x$ ). See the explanation of SIZE= $y$  or FILSZ= $x$  on the OPTION statement in the *DFSORT Application Programming Guide* for details.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the FILSZ= $x$  or SIZE= $y$  value is accurate. Message ICE047A can be suppressed by specifying installation option FSZEST=YES, or by using FILSZ= $Ex$ , SIZE= $Ey$ , FILSZ= $Ux$ , or SIZE= $Uy$ , thus ensuring that the value will be treated as an estimate.

### ICE048I      NMAX EXCEEDED

**Explanation:** DFSORT exceeded the calculated sort capacity while processing the input data set, and user exit E16 was specified.

**System Action:** The user-written routine at E16 is entered.

**Programmer Response:** No response necessary. (The number of records sorted is equal to the NMAX calculated by DFSORT. See message ICE038I.)

## Informational and Error Messages

---

### ICE049I and ICE050I

**Explanation:** See "Chapter 4. Diagnostic Messages" on page 69.

---

### ICE051A UNENDING MERGE

**Explanation:** Critical. There was not enough main storage or work data set space available for DFSORT to run with.

**System Action:** The program terminates.

**Programmer Response:** Assign more intermediate storage or main storage and rerun the job. Reverse sequence files can require more space.

---

### ICE052I END OF DFSORT

**Explanation:** The program completed.

**System Action:** Returns control to the operating system or invoking program.

**Programmer Response:** None.

---

### ICE053A OUT OF SEQUENCE

**Explanation:** Critical. The current record leaving the intermediate or output phase was not in collating sequence with the last record blocked for output.

**System Action:** The program terminates.

**Programmer Response:** Contact IBM for programming support. If message ICE143I indicates the Blockset technique was selected, specifying OPTION NOBLKSET temporarily may allow you to bypass the DFSORT error.

---

### ICE054I RECORDS-IN: *n*, OUT: *n*

RCD IN *n*,OUT *n*

**Explanation:** This message lists the number of records read by DFSORT from the input data sets, and the number of records written to SORTOUT or used as input for OUTFIL processing. The numbers replace *n* in the text of the message as shown above. For a Conventional merge application, if an exact file size was not specified, the IN field is zero; if an exact file size was specified, it appears in the IN field.

If DEBUG CTRx was used and Blockset was not selected, the *n* values could be meaningless.

For a tape work data set sort or Conventional merge application, record counts that exceed 4294967295 will be displayed as asterisks.

**System Action:** None.

**Programmer Response:** If you are using user exit E15 or E35 and have any reason to suspect that you

are "losing" or "gaining" records, check message ICE055I. The sum of RECORDS IN plus INSERT should always be equal to the sum of RECORDS OUT plus DELETE. If it is not, you should also receive message ICE025A.

---

### ICE055I INSERT *n*, DELETE *n*

**Explanation:** The number of records inserted or deleted (prior to OUTFIL processing, if used) replaces the values shown as *n* in the above format.

If DEBUG CTRx was used and Blockset was not selected, the *n* values could be meaningless.

For a tape work data set sort or Conventional merge application, record counts that exceed 4294967295 will be displayed as asterisks.

**System Action:** None.

**Programmer Response:** See message ICE054I.

---

### ICE056A *ddname* NOT DEFINED

**Explanation:** Critical. A required DD statement was not supplied. This message appears for:

- Sort or copy applications without a SORTIN DD statement or E15 user exit
- Merge applications without a SORTINnn DD statement or E32 user exit
- Sort, merge, or copy applications without an E35 user exit, a SORTOUT DD statement or an OUTFIL data set.
- Tape work data set sort applications without a SORTIN DD statement or a SORTOUT DD statement
- The same OPTION SORTIN and OPTION SORTOUT *ddname*
- CKPT in effect with no work data set specified
- Conventional merge applications without a SORTIN01 DD statement (Conventional merge ignores SORTIN1) or a SORTOUT DD statement.
- A DD statement for a specified OUTFIL data set was not present.

**System Action:** The program terminates.

**Programmer Response:** Check the DD statements for errors.

---

### ICE057A SORTIN NOT SORTWK01

**Explanation:** Critical. A work data set other than SORTWK01 was assigned to the same tape drive as SORTIN.

**System Action:** The program terminates.

**Programmer Response:** Check the DD statements for errors.

---

---

### ICE058A      SORTOUT A WORK UNIT

**Explanation:** Critical. SORTOUT was specified on the same tape drive as a work data set.

**System Action:** The program terminates.

**Programmer Response:** Check the DD statements for errors.

---

### ICE059A      RECORD LENGTH INVALID FOR               {DEVICE}ddname}

**Explanation:** Critical. The intermediate record created by DFSORT from the input record was either less than 18 bytes or more than 32756 bytes when work units were tape (indicated by DEVICE) or was too large for the assigned work data set device with the indicated ddname. The error can exist for other devices in addition to the one indicated.

**System Action:** The program terminates.

**Programmer Response:** If the intermediate record is too small, redefine input record length to at least 18 bytes. If the length is too large, assign work data set devices with larger track capacity.

If EQUALS is in effect, the maximum record length is reduced by 4 bytes.

---

### ICE060I      INVALID ddname PARAMETER

**Explanation:** A PARM parameter specified in the ddname data set contained an error. If a parameter was specified more than once, the first entry was used (if valid).

**System Action:** Processing continues. Invalid and duplicate parameters are ignored.

**Programmer Response:** No action is necessary. For later runs, be sure that the optional parameters are valid. Valid PARM parameters are described in *DFSORT Application Programming Guide*.

---

### ICE061A      I/O ERROR, DD ddname, DEV address,               ECB completion code, CSW status bytes,               SENSE sense bytes

**Explanation:** Critical. This message was issued for one of the following reasons:

- The job control statements incorrectly specified record length or blocking information for the data set located on the device indicated by the DEV field in the message.
- A permanent error occurred during an I/O operation on the indicated device.

The most likely causes include:

- The output data set of a merge, copy, or ICEGENER application is the same as one of the input data sets.
- An output data set is the same as another output data set.

- Hardware-related error.

**System Action:** If no user options are specified, the program terminates.

**Programmer Response:** Make sure that the DD statement for the data set assigned to this device contains the correct DCB information. In a merge application, if the device in error holds an input data set, make sure that the DCB information (except for BLKSIZE) specified in the SORTIN01 DD statement correctly describes the data in this device.

If ICE061A contains ECB=41 and CSW=0006, and you receive an accompanying IOS000I simulated I/O error message, you could be experiencing a missing interrupt timeout on the corresponding channel due to a large data transfer. If this is the case, either reduce the value set for the IOMAXB installation option or raise the missing interrupt handler (MIH) value for the device associated with the error.

If the error persists, you can bypass it, in some cases, by specifying the NOBLKSET option temporarily to force DFSORT to use a different sorting technique.

---

### ICE062A      LINK-EDIT ERROR

**Explanation:** Critical. The linkage editor found a serious error.

**System Action:** The program terminates.

**Programmer Response:** Check that all the DD statements used by the linkage editor are included (SYSPRINT, SYSLIN, SYSUT1, and SYSLMOD), and that they are correct.

If the DD statements are correct, make sure that:

- All user routines in libraries are valid object decks or load modules.
- All user routines in the system input stream (SYSIN) are valid object decks.
- Modules to be link-edited together do not contain duplicate entry point names.
- A SORTLIB DD statement is specified if a tape work data set or conventional merge is used and the SORTLIB=SYSTEM installation option is specified.

---

### ICE063A      OPEN ERROR ddname

**Explanation:** Critical. An error occurred while trying to open the data set with the indicated ddname.

**System Action:** The program terminates.

**Programmer Response:** Check for any of the following:

- A missing or invalid DD statement
- Conflicting DCB information; for example, the records are fixed-blocked, but the block size is not a multiple of the record length
- Concatenated mixed DASD/tape or tape-only input without a DCB BLKSIZE parameter for the tape data set containing the largest block size

## Informational and Error Messages

- Concatenated, fixed-length input with different LRECL specifications
- Data sets required for dynamic link-editing cannot be opened.

---

### ICE064A DELETE ERR

**Explanation:** Critical. DFSORT was unable to delete either itself or a user exit routine. This message appears only when user exit routines are used.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the user exit routines are not modifying the DFSORT code and information areas, and rerun the job.

---

### ICE065A PROBABLE DECK STRUCTURE ERROR

**Explanation:** Critical. The end of the SYSIN data set was found before all needed user exit modules were read or the end of the SYSIN data set was not found after all specified modules were read.

**System Action:** The program terminates.

**Programmer Response:**

- Check that the MODS statement specifies the correct routines.
- Make sure the SYSIN data set contains all user exit routines specified by the MODS statement, and no others.
- Check for misplaced job control language statements, especially preceding a user exit routine on SYSIN.
- Make sure nothing follows the last object deck in SYSIN.

---

### ICE066I APROX RCD CNT *n*

**Explanation:** Sort capacity was reached. The count *n* is an approximation of the number of records DFSORT can handle with the available work data sets.

**System Action:** The program terminates.

**Programmer Response:** Respond as indicated in the accompanying message, ICE046A.

---

### ICE067I INVALID PARAMETER IN JCL EXEC PARM OR INVOKED PARMLIST

**Explanation:** An error was found in the PARM field parameters of the EXEC statement, or in the optional parameters of the parameter list passed to DFSORT using ATTACH, LINK, or XCTL. If a parameter was specified more than once, the first entry was used (if valid).

**System Action:** Processing continues. Invalid and duplicate parameters are ignored.

**Programmer Response:** No action is necessary. For later runs, be sure that the optional parameters are

valid. Valid parameters are described in *DFSORT Application Programming Guide*.

---

### ICE068A OUT OF SEQUENCE SORTIN $nn$

#### OUT OF SEQ SORTIN $nn$

**Explanation:** Critical. During a merge, a data set was found to be out of sequence.

If Blockset was selected:

- $nn$  is the data set identification (00 to 99).
- If input was supplied through user exit E32, then 00 signifies the first input file, 01 the second, and so on.

If Blockset was not selected:

- $nn$  is the data set identification (01 to 16).
- If input was supplied through user exit E32, then 01 signifies the first input file, 02 the second, and so on.

**System Action:** The program terminates.

**Programmer Response:** If a user-written routine was modifying the records, check the routine thoroughly. It should not modify control fields at user exit E35. If a user-written routine is not being used, make sure that all input data sets have been sorted on the same control fields, in the same order, and that they all have a similar format. Check whether you have also received message ICE072I.

If input was supplied through user exit E32, check your routine to make sure records are passed to the merge from the correct file.

If you were reading in variable-length VSAM records through user exit E32, check the format and accuracy of the record descriptor word (RDW) you built at the beginning of each record.

If Blockset was not selected, rerun the job with a SORTDIAG DD statement to get message ICE800I, which indicates the reason Blockset could not be used. If possible, remove the condition preventing the use of Blockset.

---

### ICE069A INVALID SIGN

**Explanation:** Critical. The first byte of signed numeric data with leading separate sign, or the last byte of signed numeric data with trailing separate sign, did not contain a valid sign character.

**System Action:** The program terminates.

**Programmer Response:** Check the description of data format in the FIELDS or FORMAT parameter of the SORT or MERGE statement.



---

**ICE070A EXIT xxxxxxxx NOT FOUND**

**Explanation:** Critical. This message was issued for one of the following reasons:

1. If xxxxxxxx is CSECT, the CSECT name of the user exit specified for a dynamically linked user exit was incorrect.
2. If xxxxxxxx is not CSECT, it is a member name specified in the MODS statement that could not be found in the appropriate library data set.

**System Action:** The program terminates.

**Programmer Response:** For case 1, change the CSECT name to be the same as the member name specified in the MODS statement.

For case 2:

- If you specified an exit ddname for the member, make sure the member is present in the library data set with that ddname.
- If you did not specify an exit ddname for the member, make sure the member is present in a library identified by the STEPLIB/JOBLIB or link library.

---

**ICE071A INVALID RETURN CODE FROM EXIT  
Enn [RC=x, CASE=id]**

**Explanation:** Critical. An E15, E16, E32, or E35 user exit passed back an invalid return code to DFSORT. The user exit can be part of an invoking program, such as a COBOL input or output procedure.

nn is the number of the user exit that passed back return code x.

x is the return code passed to DFSORT from the user exit; it prints as the decimal equivalent of the 32-bit value in register 15.

id is an identifier that helps you determine why return code x is invalid and what the valid return codes are for the situation.

**System Action:** The program terminates.

**Programmer Response:** Check the exit number (nn), return code (x), and case identification (id) against the table in "Appendix B. User Exit Return Codes—ICE071A" on page 115. Change your user exit to pass an appropriate valid return code (that is, one marked with an x in the table) for the identified situation. As shown in the table, ICEMAC option EXITCK=WEAK can be used to avoid this message and continue processing for certain invalid return codes from the E15 and E35 user exits.

In a COBOL-invoking program containing an input or output procedure, check that control passes through the end of the input procedure before the output procedure is entered, and through the output procedure before the sort is terminated.

Also check that the user exit did not overlay register 15, for example, by inadvertently including the register in a load multiple (LM) instruction.

For complete details of how return codes are interpreted in various situations for the E15, E16, E32, and E35 user exits, see the chapter on using your own user exit routines in the *DFSORT Application Programming Guide*. In particular, see the detailed tables in the section on the E15/E35 return codes and EXITCK.

---

**ICE072I FIELD NOT WITHIN MINIMUM RECORD LENGTH**

**Explanation:** A RECORD statement specified a minimum record length (L4) that cannot contain all fields specified in the SORT, MERGE, INCLUDE, OMIT, INREC, OUTREC, or SUM statement(s).

**System Action:** The L4 value is adjusted. Processing continues.

**Programmer Response:** None.

---

**ICE073A VARIABLE RECORD TOO LONG**

**Explanation:** Critical. This message was issued for one of the following reasons:

1. A variable length record was larger than the maximum length specified or defaulted.
2. The sum of the record length and the control field lengths was larger than 32767 bytes.
3. For a tape work data set sort, a variable length record was larger than 32752 bytes (or 32748 with EQUALS in effect).
4. OPTCD=Q was not specified on the DD statement for an ASCII tape input data set.
5. An invalid RDW or BDW was found.

**System Action:** The program terminates.

**Programmer Response:** For case 1, check the input both at E15 and E35, if used, and in SORTIN. Then delete the extra long records or increase the SORTIN or SORTOUT DD statement DCB LRECL value or the RECORD statement L1, L2, or L3 value, as appropriate.

If you are using INREC or OUTREC, check that a SORTOUT DD LRECL value or L3 value is at least as large as the reformatted output record length.

If you have VSAM records, remember that they are increased in length by the 4-byte record descriptor word (RDW) added when they enter DFSORT. If you are inserting or modifying records through E15, check the format of the RDW you are building at the beginning of each record.

For case 2, if the Blockset technique was selected, you may be able to run the application by specifying OPTION NOBLKSET.

You can use the ICETOOL STATS operator with ON(VLEN) to show the length of the longest variable length record. You can use the ICETOOL DISPLAY operator with ON(VLEN) to show the length of each variable length record.

For case 3, either reduce the record lengths, or replace

## Informational and Error Messages

the tape work data sets with DASD work data sets so the Blockset technique can be used.

For case 4, ensure that OPTCD=Q is specified on all ASCII tape input DD statements.

For case 5, ensure that all RDWs and BDWs are valid.

---

### ICE074I RECORD LENGTH L1 OR L3 OVERRIDDEN

**Explanation:** The L1 value for the LENGTH parameter of the RECORD statement was not the same as the LRECL value for SORTIN or SORTIN01, or the L3 value was not the same as the SORTOUT LRECL value. For VSAM, the equivalent of LRECL is maximum RECSZ.

**System Action:** Processing continues with the L value(s) overridden.

**Programmer Response:** For subsequent runs, check all the record lengths. Take special note of the L2 value. If you did not specify one, it defaulted to the value you specified for L1 (and was not overridden by the LRECL value). If the L2 value was too small, it could have caused your application to end at any of a number of points, and the error can be difficult to detect.

The input record type and maximum record length is shown in message ICE088I or ICE089I. If you have variable-length input records, check that the L1 value specified is actually a maximum.

The output record type and maximum record length is shown in message ICE090I or ICE091I. If you have variable-length output records, check that the L3 value specified is actually a maximum.

---

### ICE075A VSAM CB ERROR (nn) AT aaaaaa

**Explanation:** Critical. aaaaaa represents the storage address of the control block at which the error was detected. nn is the VSAM return code, in decimal, from a VSAM macro.

**System Action:** The program terminates, unless the error is detected during close, when the program tries to close all remaining VSAM data sets before terminating.

**Programmer Response:** Refer to *Macro Instructions for Data Sets* for the meaning of the error code and, if possible, take appropriate action.

---

### ICE076A VSAM INPUT ERROR i(nnn) ddname

**Explanation:** Critical. i is either P (physical) or L (logical), describing the type of error encountered for the data set with the indicated ddname. nnn is the VSAM feedback code (RPLERRCD) from a GET macro, in decimal.

**System Action:** The program terminates.

**Programmer Response:** Refer to *Using Data Sets* or *Macro Instructions for Data Sets* for the meaning of the

error code and, if possible, take appropriate action.

---

### ICE077A VSAM OUTPUT ERROR i(nnn) ddname<sup>1</sup>

**Explanation:** Critical. i is either P (physical) or L (logical), describing the type of error encountered. nnn is the VSAM feedback code (RPLERRCD) from a PUT macro, in decimal. The ddname of the data set in error is given, if available.

**System Action:** The program terminates.

**Programmer Response:** Refer to *Using Data Sets* or *Macro Instructions for Data Sets* for the meaning of the error code and, if possible, take appropriate action.

A VSAM logical error, reason code 8, can occur for a non-empty KSDS, RRDS, or VRRDS output data set if NORESET is in effect, or if the data set is defined without the REUSE option.

A VSAM logical error, reason code 12, can occur if the output data set is a KSDS and either the VSAM keys are not in ascending binary sequence or there are duplicate keys. If so, ensure that the keys are in ascending binary sequence and that there are no duplicate keys.

Refer to *DFSORT Application Programming Guide* for more information about VSAM data sets.

---

### ICE078A VSAM OPEN ERROR (nnn) ddname

**Explanation:** Critical. nnn is the VSAM OPEN ERROR return code (ACBERFLG) in decimal, for the data set with the indicated ddname.

**System Action:** The program terminates.

**Programmer Response:** Refer to *Macro Instructions for Data Sets* for the meaning of the error code and, if possible, take appropriate action.

Some of the more common reasons for a VSAM OPEN error are as follows:

- An empty data set is used for input, but NVSAMEMT is in effect.
- Concatenated VSAM data sets are used.
- The same VSAM data set defined without REUSE is used for both input and output, but RESET is in effect.

---

### ICE079A VSAM CLOSE ERROR (nnn) ddname

**Explanation:** Critical. nnn is the VSAM CLOSE ERROR return code (ACBERFLG), in decimal, for the data set with the indicated ddname.

**System Action:** The program terminates.

**Programmer Response:** Refer to *Using Data Sets* or *Macro Instructions for Data Sets* for the meaning of the error code and, if possible, take appropriate action.

## Informational and Error Messages

---

### ICE080I IN MAIN STORAGE SORT

**Explanation:** All records were sorted in main storage or data space. Work data sets were not used. Hiperspace was not used.

**System Action:** None.

**Programmer Response:** None.

---

### ICE081A FIRST COMMUNICATION AREA SPACE EXCEEDED

**Explanation:** Critical. The complexity of the application caused dynamic areas and routines to exceed the total fixed amount of storage allowed for them in the first communication area. This situation normally occurs due to specification of one or more of the following (in approximate order of importance):

- A large number of SORT, MERGE, INREC, or OUTREC fields
- Use of an E61 user exit
- Use of the VLSHRT option
- Use of the NOSZERO option
- Use of an ICETEXIT routine
- Very large records.
- An EFS program requesting statements with a large number of fields.

**System Action:** The program terminates.

**Programmer Response:** Reduce the complexity of the application. For example, reduce the number of SORT fields or specify OPTION NOVLSHRT, if possible.

---

### ICE082I CHECKPOINT CANCELLED

**Explanation:** No more work data set tracks were available; therefore, the tracks allocated for CKPT were given back to the sort work data sets.

**System Action:** The program continues, but no checkpoints are taken.

**Programmer Response:** Increase the work space allocation for the next run.

---

### ICE083A RESOURCES WERE UNAVAILABLE FOR DYNAMIC ALLOCATION OF WORK DATA SETS (xxxx)

**Explanation:** Critical. DFSORT attempted to dynamically allocate work data sets using the DYNALLOC/DYNALOC values in effect. Dynamic allocation failed; xxxx indicates the failure code returned by the system's dynamic allocation facility. See *Application Development Guide for Authorized Programs* for an explanation of the failure code.

Two commonly received return codes are:

**0218** The device type is defined to the system, but the requested device has insufficient space available. For example, the device type is

defined to the system but no devices of that type have been installed, or all devices of that type are mounted as private.

**021C** The device type is not defined to the system. For example, SYSDX was specified instead of SYSDA.

However, in a DFSMS environment, the return code is usually 970C regardless of the specific error encountered.

**System Action:** The program terminates.

**Programmer Response:** Ensure that work data sets can be dynamically allocated.

---

### ICE084I {BSAM|EXCP|VSAM} ACCESS METHOD USED FOR ddname

**Explanation:** This message identifies the access method used for the identified data set.

**System Action:** None.

**Programmer Response:** None, unless you have any problems reading or writing the data set. If EXCP was used, you can force DFSORT not to use EXCP by specifying BSAM on the DEBUG control statement or as an EXEC PARM.

---

### ICE085A SECOND COMMUNICATION AREA SPACE EXCEEDED – ADD AT LEAST nK BYTES TO REGION

**Explanation:** Critical. The complexity of the application caused dynamic areas and routines to exceed the total fixed amount of storage allowed for them in the second communication area. This situation normally occurs due to specification of one or more of the following (in approximate order of importance):

- A large number of SORT, MERGE, INREC, or OUTREC fields
- Use of an E61 user exit
- Use of the VLSHRT option
- Use of the NOSZERO option
- Use of an ICETEXIT routine
- A large number of SUM fields.
- A large number of OUTFIL OUTREC, HEADER1, TRAILER1, HEADER2, TRAILER2, or SECTIONS fields.

**System Action:** The program terminates.

**Programmer Response:**

- If n is not zero, increase the REGION value by at least nKB, and ensure that the MAINSIZE value in effect is at least 200KB.
- If n is zero and the MAINSIZE value in effect is less than 200KB, increase the MAINSIZE value to at least 200KB.
- If n is zero and the MAINSIZE value in effect is at least 200KB, reduce the complexity of the application.

## Informational and Error Messages

For example, reduce the number of sort fields or specify OPTION NOVLSHRT, if possible.

---

**ICE088I**     *jobname.stepname.procstepname*, INPUT  
LRECL=*n*, BLKSIZE=*n*, TYPE=*t*

**Explanation:** Gives current job name, step name, procedure step name, and information related to the input records when SORTIN or SORTIN*n* is present:

- LRECL is the largest logical record length of the input data sets.
- BLKSIZE is the largest block size (non-VSAM) or largest control interval size (VSAM) used by DFSORT.
- TYPE is the record format of one of the input data sets, and can be one or more of the following:
  - F - fixed-length input records
  - V - EBCDIC or ASCII variable-length input records
  - B - blocked records
  - S - standard for fixed-length records or spanned for variable-length records
  - A - ASA control characters.

**System Action:** None.

**Programmer Response:** None.

---

**ICE089I**     *jobname.stepname.procstepname*, INPUT  
LRECL=*n*, TYPE=*t*

**Explanation:** Gives the current job name, step name, procedure step name, and information related to the input records when all input is supplied by user exit E15 or E35 or when input is VSAM and DFSORT cannot obtain the control interval size:

- LRECL is the logical record length of the input records.
- TYPE is F for fixed-length input records, or V for EBCDIC or ASCII variable-length input records.

**System Action:** None.

**Programmer Response:** None.

---

**ICE090I**     OUTPUT LRECL=*n*, BLKSIZE=*n*,  
TYPE=*t* [(SDB)]

**Explanation:** Gives information related to the output records when SORTOUT is present:

- LRECL is the logical record length of the output records.
- BLKSIZE is the block size (non-VSAM) or control interval size (VSAM) of the output data set.
- TYPE is the record format of the output data set and can be one or more of the following:
  - F - fixed-length output records
  - V - EBCDIC or ASCII variable-length output records
  - B - blocked records
  - S - standard for fixed-length records or spanned for variable-length records
  - A - ASA control characters.

- (SDB) indicates that the system-determined optimum block size was used.

**System Action:** None.

**Programmer Response:** None.

---

**ICE091I**     OUTPUT LRECL=*n*, TYPE=*t*

**Explanation:** Gives information related to the output records when neither SORTOUT nor an OUTFIL data set is present (that is, when all output records are deleted via user exit E35):

- LRECL is the logical record length of the output records.
- TYPE is F for fixed-length output records or V for EBCDIC or ASCII variable-length output records.

**System Action:** None.

**Programmer Response:** None.

---

**ICE092I**     MAIN STORAGE = (*x,y,z*)

**Explanation:** Gives information related to the use of main storage for this DFSORT application:

- x**     The main storage value specified (subject to the SIZE/MAINSIZE limit), or supplied by default.
- y**     The main storage theoretically available to DFSORT, considering the MINLIM value specified when the program was installed.
- z**     The main storage actually available to DFSORT, after any other program took what it needed from the region (invoking program or exit routines).  
  
The value for z is not meaningful if the storage taken by the invoking program or user exit is greater than REGION or SIZE/MAINSIZE.

The x, y, and z references to main storage apply to the total storage above and below 16MB virtual.

**System Action:** None.

**Programmer Response:** None, unless DFSORT subsequently terminated abnormally. In that case, check the z value to see how much storage was really available to DFSORT. If space was the problem, you have probably also received message ICE039A. If storage was heavily fragmented, the result might instead be a system 80A abend in either DFSORT or one of your own routines.

You can need considerably more than the normal minimum if the region is fragmented.



---

**ICE093I      MAIN STORAGE = (MAX,y,z)**

**Explanation:** Gives information related to the use of main storage for this DFSORT application:

- MAX**    The value MAX was in effect.
- y**        The main storage theoretically available to DFSORT, considering the TMAXLIM or MAXLIM values specified when the program was installed.
- z**        The main storage actually available to DFSORT, after any other program took what it needed from the region (invoking program or user exit routines). RESALL and RESINV are not considered.
- The value for z is not meaningful if the storage taken by the invoking program or user exit is greater than REGION or SIZE/MAINSIZE.

The y and z references to main storage apply to the total storage above and below 16MB virtual.

**System Action:** None.

**Programmer Response:** None, unless DFSORT subsequently terminated abnormally. In that case, check the z value to see how much storage was really available to DFSORT. If space was the problem, you probably also received message ICE039A. If storage was heavily fragmented, the result might instead be a system 80A abend in either DFSORT or one of your own routines.

You can need considerably more than the normal minimum if the region is fragmented.

---

**ICE094I      SMF RECORD NOT WRITTEN**

**Explanation:** A DFSORT SMF record was requested, but system management facilities (SMF) was not active in the system.

**System Action:** Writing of the SMF record to the DFSORT SMF data set is bypassed.

**Programmer Response:** Have your system programmer determine if SMF is properly initialized on your system.

---

**ICE095A      INVALID OPTION STATEMENT  
                  OPERAND**

**Explanation:** Critical. An OPTION control statement contained an invalid keyword operand.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the OPTION control statement does not contain an invalid keyword operand. For valid keywords, refer to *DFSORT Application Programming Guide*.

---

**ICE096I      SUCCESSFUL RECOVERY FROM B37  
                  ABEND(S) FOR WORK DATA SET(S)**

**Explanation:** DFSORT successfully recovered from one or more system B37 abends. These abends occurred when DFSORT attempted to allocate more DASD space than was available on one or more work data sets.

**System Action:** Processing continues.

**Programmer Response:** None.

---

**ICE097I      SORT RECOVERING FROM B37  
                  ABEND ON SORTWK DATASET**

**Explanation:** A system B37 abend occurred when DFSORT attempted to allocate more DASD space than was available on a work data set. This message is issued only to the master console.

**System Action:** Processing continues.

**Programmer Response:** None.

---

**ICE098I      AVERAGE RECORD LENGTH = n  
                  BYTES**

**Explanation:** n is the number of bytes in the variable-length records (including the record descriptor word) divided by the number of sorted records. The number of sorted records includes all records received, added, or deleted before the E35 user exit was entered.

**System Action:** None.

**Programmer Response:** None.

---

**ICE099A      BLDL FAILED FOR ddname DATA SET**

**Explanation:** Critical. An error was detected during member verification when the identified data set was defined as a member of a partitioned data set or a PDSE. The BLDL system function was used for verification. The most likely cause is that the member was not found in the data set.

**System Action:** The program terminates.

**Programmer Response:** Ensure that the specified partitioned data set or PDSE member exists.

---

**ICE101A      xxxxxxxxxxxx STATEMENT NOT  
                  SUPPORTED FOR TECHNIQUE USED**

**Explanation:** Critical. The control statement indicated by xxxxxxxxxxxx is not supported for the technique used (tape work data set sort or Conventional merge). xxxxxxxxxxxx is replaced by INCLUDE/OMIT, SUM, INREC, or OUTREC.

**System Action:** The program terminates.

**Programmer Response:** Rerun the job with a SORTDIAG DD statement to get message ICE800I.

## Informational and Error Messages

ICE800I indicates the reason Blockset could not be used. If possible, remove the condition preventing the use of Blockset. Alternatively, remove the indicated statement.

---

### ICE102A MISSING COND OPERAND DEFINER

**Explanation:** Critical. An INCLUDE or OMIT control statement did not contain a logical expression definition.

**System Action:** The program terminates.

**Programmer Response:** Check the INCLUDE or OMIT control statement for lack of a logical expression definition (COND operand).

---

### ICE103A INVALID INCLUDE OR OMIT STATEMENT OPERAND

**Explanation:** Critical. An INCLUDE or OMIT control statement contained an invalid keyword operand.

**System Action:** The program terminates.

**Programmer Response:**

- Make sure that the INCLUDE or OMIT control statement does not contain an invalid keyword operand.
- Make sure that the D2 format is not specified in the FORMAT=f operand.

---

### ICE104A INVALID INREC OR OUTREC STATEMENT OPERAND

**Explanation:** Critical. An INREC or OUTREC control statement contained an invalid keyword operand.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the INREC or OUTREC control statement does not contain an invalid keyword operand.

---

### ICE105A INVALID SORT, MERGE, OR SUM STATEMENT OPERAND

**Explanation:** Critical. A SORT, MERGE, or SUM control statement contained an invalid keyword operand.

**System Action:** The program terminates.

**Programmer Response:**

- Make sure that the SORT, MERGE, or SUM control statement does not contain an invalid keyword operand.
- Make sure that FIELDS=COPY and FORMAT=f are not both specified for the SORT or MERGE control statement.
- Make sure that FIELDS=NONE and FORMAT=f are not both specified for the SUM control statement.
- Make sure that the D1 format is not specified in the FORMAT=f operand.

---

### ICE106A DUPLICATE INCLUDE OR OMIT STATEMENT OPERAND

**Explanation:** Critical. An INCLUDE or OMIT control statement contained a duplicate keyword operand.

**System Action:** The program terminates.

**Programmer Response:** Check the INCLUDE or OMIT control statement for a duplicate keyword operand.

---

### ICE107A DUPLICATE INREC OR OUTREC STATEMENT OPERAND

**Explanation:** Critical. An INREC or OUTREC control statement contained a duplicate keyword operand.

**System Action:** The program terminates.

**Programmer Response:** Check the INREC or OUTREC control statement for a duplicate keyword operand.

---

### ICE108A DUPLICATE SUM STATEMENT OPERAND

**Explanation:** Critical. A SUM control statement contained a duplicate keyword operand.

**System Action:** The program terminates.

**Programmer Response:** Check the SUM control statement for a duplicate keyword operand.

---

### ICE109A SUM FIELD DISPLACEMENT OR LENGTH VALUE ERROR

**Explanation:** Critical. A sum field definition on a SUM control statement contained an invalid length or displacement (position) value.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the length and position values in the FIELDS operand of the SUM control statement were specified correctly. For BI and FI, length must be 2, 4, or 8 bytes; for PD, length must be 1 through 16 bytes; for ZD, length must be 1 through 18 bytes; for FL, length must be 4, 8 or 16 bytes. Make sure that the position plus length of each field does not exceed 4093.

---

### ICE110I VERIFY NOT USED WITH SUM

**Explanation:** A SUM control statement was specified and VERIFY was in effect, but the Blockset technique was not selected. VERIFY with SUM is not supported by the other techniques.

**System Action:** Processing continues with NOVERIFY in effect.

**Programmer Response:** Rerun the job with a SORTDIAG DD statement to get message ICE800I.

ICE800I indicates the reason Blockset could not be used. If possible, remove the condition preventing the use of Blockset.

---

### ICE111A REFORMATTING FIELD ERROR

**Explanation:** Critical. The FIELDS parameter of an INREC or OUTREC statement, or the OUTREC parameter of an OUTFIL statement, contained an invalid column, separator, position, length, keyword, pattern, sign, constant or value. Some common errors are::

- A 0 value was used.
- A null value was used where it was not permitted.
- A null string, pattern, or sign was used.
- A column was greater than 32752, or was followed by another column.
- A position plus length was greater than 32753.
- The length for a hexadecimal field was greater than 16376.
- A repetition factor was greater than 4095 for a separator, or a character or hex constant was longer than 256 bytes.
- An invalid digit or an odd number of digits was specified for a hexadecimal constant.
- The length for a Y2 format field was not 2 for Y2C, Y2Z, Y2P, Y2S or Y2PP, or 1 for Y2D, Y2B or Y2DP, or 3–6 for Y2T, Y2W, Y2TP or Y2WP, or 2–3 for Y2U, Y2X, Y2UP or Y2XP, or 3–4 for Y2V, Y2Y, Y2VP or Y2YP.
- The length for an edit field was less than 2 bytes or greater than 8 bytes for PD0, or was greater than 4 bytes for BI or FI, 8 bytes for PD, 15 bytes for ZD, or 16 bytes for CSF/FS.
- More than 15 digits or 22 characters were specified in an edit pattern.
- SIGNz (where z is not S) was specified with Mn or without EDIT or EDxy.
- x, y, or z in EDxy or SIGNz were the same character.
- The value for LENGTH was greater than 22.
- NOMATCH was specified after position and length rather than after position, length, and CHANGE.
- The length for a lookup input field was greater than 64 with character or hexadecimal find-constants, or greater than 1 with bit constants.
- The length for a lookup output field was greater than 64.
- The length for a find-constant was greater than the lookup input field length.
- A find-constant was not a character, hexadecimal, or bit constant.
- The length for a set-constant or NOMATCH field was greater than the lookup output field length.
- An invalid character was specified in a find bit constant, or the number of bits for a find bit constant was not 8.
- A set-constant was not a character or hexadecimal constant.
- The length for a sequence number was greater than 16.
- The value for START was greater than 100000000000.

- The value for INCR was greater than 10000000.

**System Action:** The program terminates.

**Programmer Response:** Correct the invalid value.

---

### ICE112I EQUALS NOT USED WITH SUM

**Explanation:** A SUM control statement was specified and EQUALS was in effect, but the Blockset technique was not selected. EQUALS with SUM is not supported by the other techniques.

**System Action:** Processing continues with NOEQUALS in effect.

**Programmer Response:** Rerun the job with a SORTDIAG DD statement to get message ICE800I. ICE800I indicates the reason Blockset could not be used. If possible, remove the condition preventing the use of Blockset.

---

### ICE113A COMPARISON FIELD ERROR

**Explanation:** Critical. The COND parameter of an INCLUDE or OMIT statement, or the INCLUDE or OMIT parameter of an OUTFIL statement, contained an invalid position, length, format, constant or mask.

**System Action:** The program terminates.

**Programmer Response:** Make sure that all fields and constants are specified correctly. Make sure that the position plus the length of each field does not exceed 4093. Make sure that a shift-out (X'0E') is not embedded within double-byte data in a character constant.

---

### ICE114A INVALID COMPARISON

**Explanation:** Critical. One of the following situations exists:

- An INCLUDE or OMIT statement or the INCLUDE or OMIT parameter of an OUTFIL statement contained an invalid field-to-field, field-to-mask, or field-to-constant comparison.
- FORMAT=SS was specified after COND in an INCLUDE or OMIT statement.
- Locale processing was required, but a character (CH) field to binary (BI) field comparison was specified. Locale processing does not support the comparison of a CH to BI field.

**System Action:** The program terminates.

**Programmer Response:** Make sure that all of the comparisons are valid. If you specified FORMAT=SS after COND in an INCLUDE or OMIT statement, respecify it before COND. DFSORT's locale processing might eliminate the need for CH to BI comparisons. See *Application Programming Guide* for information relating to locale processing. If a CH to BI comparison is needed, specify run time option LOCALE=NONE.

## Informational and Error Messages

---

### ICE115A      INSUFFICIENT MAIN STORAGE

**Explanation:** Critical. Storage was fragmented, the reserved storage value was too large, or the sizes of the user exits were too large compared to the total storage available to DFSORT.

Main storage refers to the storage *below* 16MB virtual.

**System Action:** The program terminates.

**Programmer Response:** Add more main storage in 32KB increments by recoding the REGION parameter or the MAINSIZE/SIZE value. See *DFSORT Application Programming Guide* for ways to specify MAINSIZE/SIZE.

If routines are used at user exits, their size should be added to this minimum value. For efficient sorting, allow at least 50 percent more storage than the minimum required.

This message can be issued again even after you have provided more storage because DFSORT re-estimates the amount of storage it needs at several points in the program.

Storage requirements can be reduced by decreasing either the input block size or the number of work data sets. See also message ICE092I or ICE093I.

---

### ICE116A      SORT CAPACITY EXCEEDED

**Explanation:** Critical. DFSORT was not able to complete processing with the available DASD or tape work data sets. For DASD work data sets with secondary allocation allowed, DFSORT overrode system B37 abends and continued processing; this message was issued only when no space was available on any allocated work data set.

**Note:** DFSORT uses only the first volume of multi-volume work data sets.

The amount of work data set space required can vary depending on many factors, including:

- The amount of main storage available
- The degree of randomness of the records to be sorted
- The values specified (or defaulted) for options such as FILSZ/SIZE and DYNALLOC.

**System Action:** The program terminates.

**Programmer Response:** Take one or more of the following actions:

- If appropriate, increase the amount of main storage available to DFSORT by using the options MAINSIZE/SIZE or the JCL option REGION. Increasing the amount of main storage available to DFSORT can help DFSORT use less work data set space. Avoid running a large sort in a small amount of main storage.

- If dynamic allocation was used, check that the values for options DYNALLOC and FILSZ/SIZE were appropriate. If necessary, specify these options.
- If JCL DASD work data sets were used, increase the amount of DASD work space available to DFSORT.
- If tape work data sets were used, check that all reels contained full-length tapes. (A bad tape may appear short because of many write errors.) If all reels contained full-length tapes, rerun the application and specify more work data sets.

---

### ICE117A      I/O ERROR, *jobname, stepname, unit address, device type, ddname, operation attempted, error description, last seek address or block count, access method.* (SYNADAF)

**Explanation:** Critical. This message was issued for one of the following reasons:

- The job control statements incorrectly specified record length or blocking information for the data set located on the device indicated by the "unit address" field in the message.
- A permanent error occurred during an I/O operation on the indicated device.

The most probable cause is a hardware-related error.

**System Action:** If no user options are specified, the program terminates.

**Operator Response:** If the "error description" field in the message does not contain

"WRNG. LEN. RECORD"

run the job again with the indicated unit offline, using an alternative unit or volume in its place at run time.

**Programmer Response:** Make sure that the DD statement for the data set assigned to this device contains the correct DCB information. In a merge application, if the device in error holds an input data set, make sure that the DCB information (except for BLKSIZE) specified in the SORTIN01 DD statement correctly describes the data in this device.

---

### ICE119A      SUM FIELD OVERLAPS CONTROL FIELD

**Explanation:** Critical. A sum field in a SUM control statement overlapped a control field in a SORT or MERGE control statement.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the sum fields in the FIELDS operand of the SUM control statement do not overlap control fields in the FIELDS operand of the SORT or MERGE control statement.



---

**ICE120I      EXIT E<sub>n</sub> IGNORED**

**Explanation:** User exit E14 or E2<sub>n</sub> was specified in the MODS statement, as identified in the message by Enn. Exit Enn is not supported and cannot be entered.

**System Action:** The Enn operand is checked for correct syntax, but the user exit is not used.

**Programmer Response:** Optional. Correct the MODS statement by removing the identified Enn operand.

---

**ICE121A      FILE SIZE IS TOO LARGE**

**Explanation:** Critical. The amount of data to be sorted exceeded a DFSORT implementation limit for the maximum number of records that can be sorted. DFSORT cannot handle the amount of data regardless of environmental factors, such as the amount of work data set space provided. Specific implementation limits include:

1. Variable-length records - 4294967295 records
2. EQUALS in effect - 4294967295 records
3. VLSHRT in effect - 2147483647 records
4. Blockset technique not selected - 2147483647 records

**System Action:** The program terminates.

**Programmer Response:** For cases 1 through 3, perform the sort on subsets of the data set containing less than the indicated maximum number of records. Then merge the sorted subsets into a single data set.

For case 4, rerun the job with a SORTDIAG DD statement to get message ICE800I, which indicates the reason Blockset could not be used. If possible, remove the condition preventing the use of Blockset. Alternatively, perform the sort on subsets of the data set containing less than the indicated maximum number of records. Then merge the sorted subsets into a single data set.

---

**ICE122A      *keyword* VALUE INVALID**

**Explanation:** Critical. The numeric value specified for the indicated keyword exceeded the DFSORT processing limit of 562949953421311 records (Blockset selected) or 2147483647 records (Blockset could not be used).

**System Action:** The program terminates.

**Programmer Response:** Specify a numeric value for the indicated keyword that is less than the processing limit.

---

**ICE123I      CKPT OR CHKPT OPTION IGNORED**

**Explanation:** CKPT or CHKPT was specified in the SORT or OPTION control statement for a sort application, IGNCKPT=YES was specified during installation, and the Blockset technique was selected. The Blockset technique does not support the automatic

checkpoint/restart facility, so the CKPT or CHKPT option was ignored.

**System Action:** The program continues, but no checkpoints are taken.

**Programmer Response:** Specifying NOBLKSET on the OPTION control statement will allow you to take checkpoints, except in cases where only the Blockset technique can be used (for example, when an OUTFIL control statement is specified).

---

**ICE124A      SUM FIELD OVERLAPS RECORD DESCRIPTOR WORD**

**Explanation:** Critical. A sum field in a SUM control statement overlapped the record descriptor word (RDW) for variable-length record processing.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the sum fields in the FIELDS operand of the SUM control statement do not overlap the RDW.

---

**ICE125A      SUM FIELD OVERLAPS SUM FIELD**

**Explanation:** Critical. A sum field in a SUM control statement overlapped another sum field.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the sum fields in the FIELDS operand of the SUM control statement do not overlap.

---

**ICE126A      INCONSISTENT [*ddname*] REFORMATTING FIELDS FOUND**

**Explanation:** Critical. The FIELDS parameter of an INREC or OUTREC statement (*ddname* is blank) or the OUTREC parameter of an OUTFIL statement (*ddname* indicates the first data set in the OUTFIL group), contained fields that were inconsistent with the output record format (fixed or variable) or with other fields. Note that OUTFIL data sets for which CONVERT is specified have fixed-length record format. The inconsistency is one of the following:

- The first field was not an unedited input field, containing the record descriptor word (RDW), for variable record format, that is, the first field was not 1,*n* where *n* was equal to or greater than 4.
- A single field containing only bytes from the RDW was specified for variable record format.
- The first position was specified without a corresponding length.
- The last position was specified without a corresponding length, for fixed record format.
- The last position was specified without a corresponding length, for one but not all of the INREC statement, OUTREC statement, or OUTREC parameter (without CONVERT), for variable record format.

## Informational and Error Messages

- The last position was specified with a corresponding length, for one but not all of the INREC statement, OUTREC statement, or OUTREC parameter (without CONVERT), for variable record format.
- A column overlapped the previous output field in the reformatted record.
- CONVERT was specified without an OUTREC parameter for OUTFIL variable record format to fixed record format conversion.
- VLFILL=C'x' or VLFILL=X'yy' was specified without an OUTREC parameter for OUTFIL variable length record processing.
- An OUTREC parameter did not specify any input or separation fields, for example, OUTREC=(//)

**System Action:** The program terminates.

**Programmer Response:** Correct the inconsistency.

---

**ICE127I**    **OPTIONS: OVFL0=s,PAD=s,TRUNC=s,SPANINC=s,VLSCMP=x, SZERO=x, RESET=x,VSAMEMT=x**

---

**ICE128I**    **OPTIONS: SIZE=n,MAXLIM=n,MINLIM=n,EQUALS=x,LIST=x,ERET=a,MSGDDN=b**

---

**ICE129I**    **OPTIONS: VIO=x,RESDNT=c,SMF=e,WRKSEC=x,OUTSEC=x,VERIFY=x,CHALT=x,DYNALOC=d,ABCODE=j**

---

**ICE130I**    **OPTIONS: RESALL=n,RESINV=n,SVC=g,CHECK=x,WRKREL=x,OUTREL=x,CKPT=x,STIMER=x,COBEXIT=f**

---

**ICE131I**    **OPTIONS: TMAXLIM=n,ARESALL=n,ARESINV=n,OVERRGN=n,CINV=x,CFW=x, DSA=n**

---

**ICE132I**    **OPTIONS: VLSHRT=x,ZDPRINT=x, IEXIT=x,TEXIT=x,LISTX=x,EFS=h,EXITCK=k,PARMDDN=q,FSZEST=x**

---

**ICE133I**    **OPTIONS: HIPRMAX=p,DSPSIZE=r,ODMAXBF=n,SOLRF=x,VLLONG=x, VSAMIO=x**

---

**Explanation:** Messages ICE127I through ICE133I indicate the options in effect for a Blockset sort, copy, or merge application or a Peerage/Vale sort application.

When DFSORT is invoked by JCL, these options can come from the EXEC PARM field, the SYSIN data set, the DFSPARM data set, or the installation defaults. When DFSORT is invoked dynamically, these options can come from the SORTCNTL data set, the DFSPARM

data set, the invocation parameter list, or the installation defaults.

DFSORT can change the options due to conflicting specifications or for performance reasons. Some options are not meaningful for all applications (for example, HIPRMAX applies to a sort application, but not to a copy or merge application).

For information on where the options can be specified and on the order of override, see *DFSORT Application Programming Guide*. Message values are as follows:

**x**    Y for YES; N for NO  
**n**    A decimal value  
**a**    RC16 or ABEND  
**b**    Data set ddname or SYSOUT  
**c**    ALL for eligible DFSORT modules loaded into LPA; NONE for eligible DFSORT modules not loaded into LPA. DFSORT automatically determines the RESDNT status; you cannot override this status nor specify it as an option.  
**d**    N for NO, or (y,n) where y is the dynamic allocation device name and n is a decimal value  
**e**    SHORT, FULL, or NO  
**f**    COB1 or COB2  
**g**    n for SVC number n, or 109A for SVC 109 using the alternate SVC module  
**h**    EFS program name or NONE  
**j**    MSG or nnn for ABCODE number  
**k**    S for STRONG; W for WEAK  
**m**    ALL, NOWRK, or NONE  
**p**    A decimal value or OPTIMAL  
**q**    Control statement/PARM option DDNAME or DFSPARM  
**r**    A decimal value or MAX  
**s**    RC0, RC4 or RC16

**System Action:** None.

**Programmer Response:** None.

**Notes:**

1. Do not expect the values in ICE127I through ICE133I to be identical to the installation option values selected at your site. You can use the ICETOOL DEFAULTS operator to show all the installation option values.
2. Messages ICE131I and ICE133I are printed only for Blockset.

---

**ICE134I**    **NUMBER OF BYTES SORTED: n**

**Explanation:** n is the total number of bytes "sorted"; that is, the number of bytes in records for which control field processing was performed. The following insert/delete/alter processing is performed before control field processing and is taken into account when determining n: SORTIN, SKIPREC, STOPAFT, E15, and INCLUDE/OMIT. Other processing that can affect the number of bytes actually manipulated by DFSORT (for example, INREC) is done after control field processing, and is not taken into account when determining n.

If the input data includes incomplete spanned segments, they will be eliminated before "sorting". However, these incomplete spanned segments may be taken into account when determining n, in order to avoid impacting performance. If you want to ensure that n does not reflect incomplete spanned segments, you can eliminate them with copy, and sort the resulting data.

**System Action:** None.

**Programmer Response:** None.

**ICE136I SPACE FOR *ddname* WAS READJUSTED TO CYLINDERS**

**Explanation:** The indicated work data set was not allocated in cylinders. DFSORT adjusted its usage of the allocated space to cylinders and could have skipped several tracks in the primary extent(s) of the data set.

**System Action:** None.

**Programmer Response:** To optimize performance and minimize wasted DASD space for future runs, change your JCL to use cylinder allocation for the indicated work data set.

**ICE137A VSAM INPUT ERROR *i(nnn)* text from VSAM SYNAD**

**ICE137A (CONT.) text from VSAM SYNAD continued**

**Explanation:** Critical. *i* is either P (physical) or L (logical), describing the type of error encountered. *nnn* is the VSAM feedback code from a GET macro, in decimal, and is followed by the VSAM SYNAD message.

**System Action:** The program terminates.

**Programmer Response:** Refer to *Using Data Sets or Macro Instructions for Data Sets* for the meaning of the error code and, if possible, take appropriate action.

**ICE138A VSAM OUTPUT ERROR *i(nnn)* text from VSAM SYNAD**

**ICE138A (CONT.) text from VSAM SYNAD continued**

**Explanation:** Critical. *i* is either P (physical) or L (logical), describing the type of error encountered. *nnn* is the VSAM feedback code from a PUT macro, in decimal, and is followed by the VSAM SYNAD message.

**System Action:** The program terminates.

**Programmer Response:** Refer to *Using Data Sets or Macro Instructions for Data Sets* for the meaning of the error code and, if possible, take appropriate action.

**ICE139A UNSUPPORTED VSAM DATA SET ATTRIBUTE FOR *ddname***

**Explanation:** Critical. An attribute specified for the VSAM data set with the indicated *ddname* is not supported by DFSORT. DFSORT supports the following attributes:

- If Blockset is selected: SPAN, KSDS, ESDS, RRDS, and VRRDS
- If Blockset is not selected: SPAN, KSDS, ESDS, and RRDS

**System Action:** The program terminates.

**Programmer Response:** Remove any unsupported attributes, other than VRRDS, from the indicated VSAM data set. To use the VRRDS attribute, you must remove any condition preventing the use of Blockset. Rerun the job with a SORTDIAG DD statement to get message ICE800I, which indicates the reason why Blockset could not be used.

**ICE140I END OF PARAMETERS FROM *ddname* - SYSIN OR *xxxx*CNTL/PARAMETER LIST CONTROL STATEMENTS FOLLOW**

**Explanation:** An alternate parameter data set was specified with the indicated *ddname*. This message separates the listing of its parameters (control statements or PARM parameters) from the listing of control statements specified from other sources:

- The SYSIN DD statement
- The SORTCNTL DD statement
- The invoking parameter list.

The data set's parameters are listed only if LIST is in effect (see message ICE128I).

A parameter in a statement other than the OPTION or DEBUG control statement completely overrides the same or corresponding control statement in other sources.

A DEBUG control statement in the alternate parameter data set overrides only the same or corresponding parameters in a DEBUG control statement in the other sources. This selective override does not affect the other parameters in a DEBUG control statement in the alternate parameter data set or the other sources.

An OPTION control statement in the alternate parameter data set overrides only the same or corresponding parameters in an OPTION, SORT, or MERGE control statement in the other sources. This selective override does not affect the other parameters in an OPTION, SORT, or MERGE control statement in the alternate parameter data set or the other sources.

See *DFSORT Application Programming Guide* for complete details on the order of override.

**System Action:** None.

## Informational and Error Messages

**Programmer Response:** None.

---

### ICE141A SPANNED RECORD ON *ddname* COULD NOT BE ASSEMBLED

**Explanation:** Critical. A spanned record on the indicated data set could not be properly assembled. This message was issued for one of the following reasons:

- A segment length was greater than the LRECL.
- A segment length was less than 4 bytes.
- The total length of segments was greater than the LRECL.
- Segments are blocked incorrectly for VBS data sets (that is, a first and a last segment are in the same block).

**System Action:** The program terminates.

**Programmer Response:** Check the indicated data set for incorrect spanned records.

---

### ICE142I *ddname* NOT FOUND - SYSOUT USED

**Explanation:** A DD statement for *ddname* (the specified alternate message data set) was not found. SYSOUT was used instead.

**System Action:** Processing continues; messages are written to SYSOUT.

**Programmer Response:** If you want to use an alternate message data set, provide a DD statement for *ddname*.

---

### ICE143I *t a* TECHNIQUE SELECTED

**Explanation:** *t* indicates the DFSORT technique chosen for the run, and *a* indicates the application chosen for the DFSORT technique. Message values are as follows:

- t*      BLOCKSET – DASD work data set sort, standard merge or copy  
         PEERAGE — DASD work data set sort  
         VALE — DASD work data set sort  
         CONVENTIONAL — tape work data set sort or nonstandard merge
- a*      SORT  
         MERGE  
         COPY

**System Action:** None.

**Programmer Response:** None.

---

### ICE144A TOO MANY SUM FIELDS

**Explanation:** Critical. The complexity of the application caused dynamic areas to exceed the storage allowed for them.

**System Action:** The program terminates.

**Programmer Response:** Reduce the number of summary fields.

---

### ICE145A SVC *nnn* IS AN INVALID USER SVC

**Explanation:** Critical. The DFSORT ICEMAC installation parameters for your site indicate that SVC *nnn* is the DFSORT SVC. SVC *nnn* is not compatible with the DFSORT release currently running. Either SVC *nnn* is for a release of DFSORT other than this one, or SVC *nnn* is a non-DFSORT SVC.

**System Action:** The program terminates.

**Programmer Response:** Report this message to your system programmer. It indicates that the DFSORT SVC was incorrectly installed.

---

### ICE146I END OF STATEMENTS FROM *xxxxCNTL* – PARAMETER LIST STATEMENTS FOLLOW

**Explanation:** When a data set specified with a DD statement for *xxxxCNTL* is present, this message separates the listing of its statements from the listing of the statements specified in the parameter list. The statements for *xxxxCNTL* (if any) and the parameter list (if any) are listed only if LIST is in effect (see message ICE128I).

A statement other than OPTION or DEBUG in the *xxxxCNTL* data set completely overrides the same or corresponding statement in the parameter list.

A DEBUG statement in the *xxxxCNTL* data set overrides only the same or corresponding operands in a DEBUG statement in the parameter list. This selective override does not affect the other operands in a DEBUG statement in the *xxxxCNTL* data set or parameter list.

An OPTION statement in the *xxxxCNTL* data set overrides only the same or corresponding operands in an OPTION, SORT, or MERGE statement in the parameter list. This selective override does not affect the other operands in an OPTION, SORT, or MERGE statement in the *xxxxCNTL* data set or parameter list. (OPTION statement operands LIST, NOLIST, LISTX, NOLISTX, EFS, MSGDDN, MSGPRT, SORTDD, SORTIN, and SORTOUT are not used if specified in the *xxxxCNTL* data set.)

For complete details on the order of override, see *DFSORT Application Programming Guide*.

**System Action:** None.

**Programmer Response:** None.

---

### ICE147A OPTION STATEMENT OPERAND SORTIN OR SORTOUT NOT ALLOWED

**Explanation:** Critical. The OPTION statement operand SORTOUT was specified for a Conventional merge, or the operand SORTIN or SORTOUT was specified for a



tape work data set sort, or ICETOOL was used for a tape work data set sort.

**System Action:** The program terminates.

**Programmer Response:** Remove the SORTIN or SORTOUT operand. You can use the SORTDD operand of the OPTION statement to provide alternate ddnames for SORTIN and SORTOUT.

If ICETOOL was used, do not use tape work data sets.

**ICE148A** *ddname* **CONCATENATION NOT ALLOWED**

**Explanation:** Critical. SORTINnn or SORTWKdd data sets were concatenated.

**System Action:** The program terminates.

**Programmer Response:** Use separate DD statements for SORTINnn or SORTWKdd data sets.

**ICE149A** **DFSORT IS NOT LICENSED FOR USE ON THIS SYSTEM – RETURN CODE *x* AND REASON CODE *y***

**Explanation:** SYS1.PARMLIB member IFAPRDxx does not define DFSORT as a feature licensed for use on this system, or an error was encountered while registering DFSORT for use on this system.

**System Action:** The program terminates.

**Programmer Response:** If the return code (*x*) is 12, the reason code (*y*) is 4, and you are licensed to use DFSORT, then you need to update the IFAPRDxx member of parmlib in one of the following ways:

- If you license DFSORT as a separate product, add this entry:

```
PRODUCT OWNER ('IBM CORP')
  NAME(DFSORT)
  ID(5740-SM1)
  VERSION(*) RELEASE(*) MOD(*)
  FEATURENAME(DFSORT)
  STATE(ENABLED)
```

- If you license DFSORT as a feature of OS/390, see *OS/390 Planning for Installation* for information about adding the entry to IFAPRDxx.

After you have added the entry to IFAPRDxx, you must activate it by using a SET PROD command or by IPLing the system.

For any other return and reason codes, contact your IBM representative.

**ICE150I** **VLSHRT NOT USED FOR SORT, MERGE, INCLUDE OR OMIT STATEMENT FIELDS**

**Explanation:** The VLSHRT option was in effect, and one or more of the following situations exists:

- An INREC, OUTREC, or SUM statement was specified.
- A copy application was specified without an INCLUDE or OMIT statement.
- More than one control field was specified, and Blockset was not selected.
- The application had EFS processing in effect, and an EFS01 or EFS02 user exit routine was supplied by the EFS program.
- Locale processing was required for sort or merge fields.

**System Action:** Processing continues; VLSHRT is not used for SORT, MERGE, INCLUDE, or OMIT statement fields, but is used for OUTFIL INCLUDE or OMIT parameter fields, if specified.

**Programmer Response:** None, unless message ICE015A or ICE218A was received, in which case you can do one of the following, as appropriate:

- specify VLSCMP (if records are too short to contain all INCLUDE or OMIT fields)
- use equivalent OUTFIL parameters for the application (for example, the OUTREC parameter of OUTFIL instead of the OUTREC statement)
- respecify your fields to be within the shortest record in the input data set

**ICE151A** **TOO MANY [*ddname*] COMPARISION FIELDS**

**Explanation:** Critical. For an INCLUDE or OMIT statement (*ddname* is blank) or the INCLUDE or OMIT parameter of an OUTFIL statement (*ddname* indicates the first data set in the OUTFIL group), the complexity of the application caused dynamic areas to exceed the storage allowed for them.

**System Action:** The program terminates.

**Programmer Response:** Reduce the number of INCLUDE/OMIT fields or the size of INCLUDE/OMIT constants. Alternatively, you may be able to avoid reducing the number of fields or size of constants by using one of the following techniques:

- If NOSZERO is in effect and you can treat numeric -0 and +0 values as signed (that is, different) for this application, use the SZERO option.
- Rewrite the INCLUDE/OMIT condition to use substring comparison tests (see *DFSORT Application Programming Guide* for details)
- Use multiple DFSORT steps or ICETOOL operators to "OR" sets of conditions together to create a MOD output data set. In the following example, a1 to an and b1 to bn represent relational conditions.

```
//MULT EXEC PGM=ICETOOL
//TOOLMSG DD SYSOUT=*
//DFSMSG DD SYSOUT=*
//IN DD DSN=...,DISP=SHR,...
//OUT DD DSN=...,DISP=MOD,...
//TOOLIN DD *
```

## Informational and Error Messages

```
COPY FROM(IN) TO(OUT) USING(SETA)
COPY FROM(IN) TO(OUT) USING(SETB)
//SETACNTL DD *
  INCLUDE COND=(a1,OR,
                a2,OR,
                .
                .
                an)
//SETBCNTL DD *
  INCLUDE COND=(b1,OR,
                b2,OR,
                .
                .
                bn)
```

In the example above, a1 to an and b1 to bn represent relational conditions.

---

### ICE152I OVERFLOW DURING SUMMATION - RC=*n*

**Explanation:** The application was successfully processed, but overflow occurred for one or more pairs of summary fields. DFSORT set a return code of 0 or 4 as indicated by *n*. DFSORT only set a return code of 4 if OVFL=RC4 was in effect and summary fields overflowed.

**System Action:** Summation continues for pairs of records for which overflow does not occur. Summation is not performed for pairs of records for which overflow would produce more than one record with the same control field. DFSORT passes back a return code of *n* unless it sets a higher return code for some other reason.

**Programmer Response:** If appropriate, redesign the records so that summary fields do not overflow, or, if possible, use INREC to increase the size of the summary fields (see *DFSORT Application Programming Guide* ).

---

### ICE153A COBOL EXIT NOT SUPPORTED FOR TECHNIQUE USED

**Explanation:** Critical. An E15 or E35 user exit written in COBOL was specified, but is not supported for the technique used (tape work data set sort or Conventional merge).

**System Action:** The program terminates.

**Programmer Response:** Rerun the job with a SORTDIAG DD statement to get message ICE800I. ICE800I indicates the reason Blockset could not be used. If possible, remove the condition preventing the use of Blockset. Alternatively, remove the COBOL user exit.

---

### ICE154A STOPAFT NOT SUPPORTED FOR TECHNIQUE USED

**Explanation:** Critical. STOPAFT was in effect but is not supported for the technique used (tape work data set sort).

**System Action:** The program terminates.

**Programmer Response:** Use DASD work data sets, if possible, or remove the STOPAFT option.

---

### ICE155I STOPAFT OR SKIPREC NOT APPLICABLE TO MERGE

**Explanation:** A MERGE control statement was specified and STOPAFT or SKIPREC was in effect. Neither STOPAFT nor SKIPREC can be used with MERGE.

**System Action:** Processing continues; STOPAFT or SKIPREC is ignored.

**Programmer Response:** None.

---

### ICE156I MAIN STORAGE ABOVE 16-MEGABYTE = (*y,z*)

**Explanation:** Supplies information on the amount of storage available above 16MB virtual:

- y** The maximum amount of main storage available to DFSORT above 16MB virtual.
- z** The actual amount of main storage available to DFSORT above 16MB virtual, after DFSORT has released the ARESALL and ARESINV space.

This message is only printed when Blockset is selected.

**System Action:** None.

**Programmer Response:** None.

---

### ICE157I EXEC PARM E15=COB OR E35=COB AND NO MODS EXIT

**Explanation:** A MODS statement contained one of the following:

- E15=COB without a corresponding E15 operand
- E35=COB without a corresponding E35 operand.

**System Action:** Processing continues. The EXEC parameter E15=COB or E35=COB is ignored.

**Programmer Response:** Specify an E15 or E35 operand on a MODS statement or remove the EXEC parameter.

---

**ICE158A    SYSOUT DD STATEMENT MISSING**

**Explanation:** Critical. Issued only to the master console to indicate that ICEMAC option NOMSGDD=QUIT was used and a message data set was required, but not specified. Corresponds to DFSORT return code 20.

**System Action:** The program terminates.

**Programmer Response:** Specify a DD statement for the message data set, using the ddname specified by MSGDDN (if any) or SYSOUT.

---

**ICE159A    MODULE ICECOB2 NOT FOUND**

**Explanation:** Critical. Module ICECOB2 was not installed as part of DFSORT. An E15 or E35 user exit routine written in COBOL was to be used and COBEXIT=COB2 was in effect. Module ICECOB2 is required for COBEXIT=COB2.

**System Action:** The program terminates.

**Programmer Response:** If VS COBOL II (or above) has become available on your system since this release of DFSORT was installed, ask your system programmer to update the DFSORT installation.

If VS COBOL II (or above) is not available on your system, COBEXIT=COB2 cannot be used.

---

**ICE160A    COPY FUNCTION COULD NOT BE USED - REASON CODE IS nn**

**Explanation:** Critical. The COPY function was specified but could not be processed. Reason code values are as follows:

- 1     The SORTIN or SORTOUT data set was BDAM.
- 2     FREE=CLOSE was specified (ICEGENER).
- 3     An attempt to open a data set caused a system error.
- 4     The SORTIN or SORTOUT data set resided on an unsupported device.
- 5     ASCII tapes had the following parameters:  
(LABEL=AL or OPTCD=Q) and RECFM=D and BUFOFF= L  
or  
(LABEL=AL or OPTCD=Q) and RECFM= D  
and BUFOFF= 0
- 6     An attempt to read the DSCB for the SORTIN data set caused an error.
- 7     An attempt to read the DSCB for the SORTOUT data set caused an error.
- 8     The SORTIN data set had keyed records (ICEGENER).

9     User labels were present (ICEGENER).

10    A MODS statement Exx operand had SYSIN in the third parameter or T or S in the fourth parameter (that is, dynamic link-editing was requested.)

**Note:** For ICEGENER, SYSUT1 is equivalent to SORTIN and SYSUT2 is equivalent to SORTOUT.

**System Action:** The program terminates.

**Programmer Response:** Correct the situation indicated by the reason code.

---

**ICE161A    COBEXIT=COB2 AND COBOL E15 AND E35 EXITS WERE FOUND**

**Explanation:** Critical. COBEXIT=COB2 was in effect; therefore, separately compiled COBOL E15 and E35 user exits were not allowed together for COPY processing.

**System Action:** The program terminates.

**Programmer Response:** When COBEXIT=COB2 is in effect, use either a COBOL E15 user exit or a COBOL E35 user exit for copy applications, but not both. If your exits are written in OS VS COBOL, make sure that COBEXIT=COB2 is not in effect.

---

**ICE162I    ICEIEXIT CHANGED ONE OR MORE OPTIONS IN EFFECT**

**Explanation:** During installation, the initialization exit (ICEIEXIT) was used to change one or more of the options that apply to your application. DFSORT used the changed options where appropriate. Messages ICE127I-ICE133I indicate which options DFSORT used.

**System Action:** None.

**Programmer Response:** No action is necessary. If appropriate, see your system programmer for an explanation of the changes made with the ICEIEXIT routine.

---

**ICE163A    DFSORT CANNOT FIND THE IEBGENER MODULE**

**Explanation:** Critical. This message, issued only to the master console, indicates the DFSORT ICEGENER facility attempted to transfer control to the IEBGENER module, but the DFSORT installation option for that name was the same as the name for the ICEGENER module, thus resulting in a loop.

**System Action:** The program terminates with user abend 163.

**Programmer Response:** Ensure that the name being used for the GENER installation option is an alias of the IEBGENER module and not an alias of the ICEGENER module.

## Informational and Error Messages

---

**ICE164A**     *xnnnn* ABEND OCCURRED DURING EXECUTION OF ICETEXIT

**Explanation:** Critical. ESTAE was in effect and an abend occurred after the installation exit ICETEXIT was called but before control was returned to DFSORT.

**x**            The abend type, either S for system or U for user

**n n n n**      The abend code

**System Action:** None.

**Programmer Response:** Notify your system programmer that the installation exit, ICETEXIT, has abended so that they can determine the cause of the abend and correct the error. You might need to rerun the job with a SYSUDUMP DD statement to get an abend dump. With ESTAE in effect, DFSORT tries to call ICETEXIT (if it has not been called already) after an earlier abend. In this case, only the information about the first abend appears in the dump. Rerun the job with DEBUG ABSTP to get the dump for the secondary abend.

---

**ICE165I**     TOTAL DATA SET TRACKS ALLOCATED: *a*, TRACKS USED: *b*

**Explanation:** This message gives work data set space usage information for the sort run. Message values are as follows:

- a**            Total primary and secondary DASD work space allocated and available throughout sort processing (in tracks). This space is not available for use by other applications.
- b**            Total primary and secondary DASD work space actually used during sort processing (in tracks).

**Note:** In some cases, DFSORT may free up JCL-specified work space if it is determined that the space is not needed for the sort.

**System Action:** None.

**Programmer Response:** None.

---

**ICE168I**     CENTURY WINDOW FOR Y2 FORMAT FIELDS IS FROM *ssss* TO *eeee*

**Explanation:** Indicates the starting year (*ssss*) and ending year (*eeee*) for the century window to be used in conjunction with the specified Y2 format fields.

**System Action:** None

**Programmer Response:** None

---

**ICE170I**     FORMAT OPERAND IGNORED

**Explanation:** Format values were found in both the FIELDS or COND operand and the FORMAT operand.

**System Action:** The format values in the FIELDS or COND operand are used; the format value in the FORMAT operand is ignored.

**Programmer Response:** Optional. Remove the FORMAT operand or the format values in the FIELDS or COND operand.

---

**ICE171I**     SORTOUT LRECL OF *x* IS DIFFERENT FROM SORTIN(NN) LRECL OF *y* - RC=*n*

**Explanation:** The application was successfully processed, but the SORTOUT LRECL (*x*) and SORTIN/SORTINnn LRECL (*y*) were not equal. DFSORT set a return code of 0 or 4 as indicated by *n*. DFSORT only set a return code of 4 if:

- PAD=RC4 was in effect and *x* was larger than *y* (LRECL padding), or
- TRUNC=RC4 was in effect and *x* was smaller than *y* (LRECL truncation).

**System Action:** Processing continues and DFSORT passes back a return code of *n* unless it sets a higher return code for some other reason.

**Programmer Response:** If appropriate, change the LRECL of SORTOUT or SORTIN/SORTINnn so that the LRECLs are equal.

---

**ICE176A**     *service* SERVICE ERROR (*x-y*)

**Explanation:** Critical. The indicated system service failed when DFSORT called it. Message values are as follows:

- x**            The return code from the service, in decimal.
- y**            The reason code from the service, in decimal, or zero if unavailable.

**System Action:** The program terminates.

**Programmer Response:** Contact IBM for programming support.

---

**ICE177A**     DASD CALCULATION SERVICES ERROR (*x-y*)

**Explanation:** Critical. DASD Calculation Services returned an unexpected return or reason code when DFSORT called it to determine the optimum block size for an output data set.

**System Action:** The program terminates.

**Programmer Response:** Contact IBM for programming support. You can bypass the use of system-determined block size for this application by specifying either SDB=NO or a BLKSIZE value on the DD statement for the output data set.



---

**ICE178I REALLOCATING VIO SORTWK DATA SETS TO REAL SORTDK DATA SETS USING UNIT=*device***

**Explanation:** SORTWKdd data sets specified in the JCL were allocated on virtual devices (VIO). Since the VIO=NO installation option was specified or defaulted, DFSORT will attempt to reallocate the SORTWKdd data sets as SORTDKdd data sets using devices of the type indicated by device.

**System Action:** Processing continues; DFSORT attempts to reallocate the data sets.

**Programmer Response:** If appropriate, allocate the SORTWKdd data sets on real devices to avoid reallocation.

---

**ICE179A RESOURCES WERE UNAVAILABLE FOR VIO REALLOCATION OF WORK DATA SETS (*xxxx*)**

**Explanation:** Critical. SORTWKdd data sets specified in the JCL were allocated on virtual devices (VIO). Since the VIO=NO installation option was specified or defaulted, DFSORT attempted to reallocate the SORTWKdd data sets as SORTDKdd data sets using a real device of the type indicated in message ICE178I. The reallocation failed; xxxx indicates the failure code returned by the system's dynamic allocation facility. See *Application Development Guide for Authorized Programs* for an explanation of the failure code.

Two commonly received return codes are:

- 0218** The real device type is defined to the system, but the requested device has insufficient space available. For example, the real device type is defined to the system but no devices of that type have been installed, or all devices of that type are mounted as private.
- 021C** The real device type is not defined to the system. For example, 3380 was specified instead of 3390.

However, in a DFSMS environment, the return code is usually 970C regardless of the specific error encountered.

**System Action:** The program terminates.

**Programmer Response:** Ensure that the VIO SORTWKdd data sets can be reallocated on the real device indicated in message ICE178I, or if appropriate, allocate the SORTWKdd data sets on real devices to avoid reallocation.

Alternatively, specify installation option VIO=YES (can degrade performance). See *DFSORT Installation and Customization* for details on the VIO installation option.

---

**ICE180I HIPERSPACE STORAGE USED = *n*K BYTES**

**Explanation:** n is the number of Kilobytes of Hiperspace storage used during this sort. If n is zero, Hiperspace storage was not used.

**System Action:** None.

**Programmer Response:** None.

---

**ICE181A HIPERSPACE READ ERROR (*xx*) [*yyyyyyyy*]**

**Explanation:** Critical. The HSPSERV SREAD service request for a Hiperspace read failed. xx is the HSPSERV return code (in decimal). yyyyyyyy is the reason code (set only when the return code is 8).

**System Action:** The program terminates.

**Programmer Response:** Refer to the HSPSERV service documentation in the *Application Development Reference for Authorized Programs* for a description of HSPSERV return codes and reason codes. If possible, take appropriate action.

The use of Hiperspace can be bypassed by specifying HIPRMAX=0 on the OPTION control statement.

---

**ICE182A HIPERSPACE WRITE ERROR (*xx*) [*yyyyyyyy*]**

**Explanation:** Critical. The HSPSERV SWRITE service request for a Hiperspace write failed. xx is the HSPSERV return code (in decimal). yyyyyyyy is the reason code (set only when the return code is 8).

**System Action:** The program terminates.

**Programmer Response:** Refer to the HSPSERV service documentation in the *Application Development Reference for Authorized Programs* for a description of HSPSERV SWRITE return codes and reason codes. If possible, take appropriate action.

The use of Hiperspace can be bypassed by specifying HIPRMAX=0 on the OPTION control statement.

---

**ICE183A CONDITIONS NOT MET TO USE THE SAME VSAM DATA SET FOR indd AND outdd**

**Explanation:** Critical. The indd DD statement and outdd DD statement both specified the same VSAM data set, but one of the following conditions prevented DFSORT from using the same VSAM data set for input and output:

- The application is a copy or merge
- NOVSAMIO is in effect
- NORESET is in effect
- The VSAM data set is not defined with REUSE

**System Action:** The program terminates.

## Informational and Error Messages

**Programmer Response:** If you want to sort the VSAM data set in-place, ensure that VSAMIO and RESET are in effect, and that the VSAM data set is defined with REUSE. Otherwise, ensure that the indd DD and outdd DD statements do not specify the same VSAM data set.

---

### ICE184I THE NUMBER OF RECORDS SORTED WITH EQUAL KEYS IS *n*

**Explanation:** *n* is the number of records sorted having equal control keys.

**System Action:** None.

**Programmer Response:** None.

---

### ICE185A DFSORT TERMINATED WITH *xnnnn* ABEND DURING PHASE *m y*

**Explanation:** Critical. A system error occurred or an abend was requested while ESTAE was in effect.

This message gives the abend code, the application mode (if known at the time of the error), and the phase where the error occurred.

**x** The abend type, either S for system or U for user

**nnnn** The abend code

**m** S, M, or C for a Blockset sort, merge, or copy, respectively. *m* is P when the Peerage/Vale technique was used. *m* can also be blank if the abend occurred before DFSORT determined the application type.

**y** Phase number 0, 1, 2, 3, or 4.

**System Action:** The program terminates.

**Programmer Response:** Use the information in the message and the abend dump to determine the cause of the abend, and take appropriate action. For system abends, see the appropriate systems codes book. For user abends, see "Chapter 6. Return Codes and User Abends" on page 73.

---

### ICE186A DFSORT CONTINUED AFTER ERROR AND OUTPUT IS USEABLE

**Explanation:** Critical. An error was detected or an abend occurred; however, the sort, copy, or merge was completed successfully. The SORTOUT data was written successfully or an E35 user exit disposed of all of the output records. The SORTOUT data set written by DFSORT was closed successfully.

**System Action:** The program terminates with the abend code.

**Programmer Response:** The SORTOUT data set can be used (provided the system did not delete it) even though the program terminated abnormally. Because of the error, one of the following DFSORT functions below was not completed successfully:

- The writing of final and statistical messages
- The writing of an SMF record, if appropriate
- The calling of a termination exit, if present
- The closing of data sets, releasing of space, and other cleanup functions.

---

### ICE187I DFSORT SVC LEVEL INCOMPATIBLE WITH DFSORT PROGRAM LEVEL

**Explanation:** The running level of DFSORT did not match the installed level of the DFSORT SVC. This incompatibility had no effect on the DFSORT run if the SVC was not needed. If the SVC was needed, the run could have terminated with a system abend code of 56D or other failure. DFSORT uses the SVC with certain DASD devices and to write the SMF record. See *DFSORT Installation and Customization* for more information about installing the SVC. Message ICE816I was also printed if DFSORT abended trying to call the SVC.

**System Action:** Processing continues.

**Programmer Response:** Ask your system programmer to check that the DFSORT SVC is installed properly for this level of DFSORT and that the correct routing code for the SVC is selected. If necessary, rerun the job with a SORTDIAG DD dummy statement to get message ICE816I.

---

### ICE188I DATA SPACE STORAGE USED = *nK* BYTES

**Explanation:** *n* is the number of Kilobytes of data space storage used during this sort. If *n* is zero, data space storage was not used.

**System Action:** None.

**Programmer Response:** None.

---

### ICE189A BLOCKSET REQUIRED BUT COULD NOT BE USED - REASON CODE IS *nn*

**Explanation:** Critical. Blockset was required for one of the following:

- LOCALE processing
- OUTFIL processing
- Y2x, Y2xx, PD0, FS or CSF format
- INREC or OUTREC processing with one of the following:
  - p,m,HEX
  - p,HEX
  - p,m,f
  - p,m,lookup
  - SEQNUM
- A VSAM extended addressability data set
- To set the SORTOUT LRECL from the L3 length (without E35, INREC or OUTREC), the OUTREC length or the INREC length, with SOLRF in effect

- an FL format sort field with NOSZERO in effect
- VLLONG in effect and SORTOUT present
- VSAMEMT in effect for a sort or merge with VSAM input
- The same VSAM data set was specified for both input and output
- An HFS file was specified for input or output
- A tape data set with a block size greater than 32760 bytes was specified for input or output
- SDB=LARGE or SDB=INPUT was in effect and DFSORT selected a block size greater than 32760 bytes for a tape output data set

However, Blockset could not be used due to the reason indicated by reason code nn. See message ICE800I for the meaning of nn.

**System Action:** The program terminates.

**Programmer Response:** Correct the situation indicated by the reason code so Blockset can be used. Alternatively, you can remove the source of the requirement to use Blockset. However, this will result in the use of a less efficient technique.

---

**ICE190A      ddname DYNALLOC CALL FAILED -  
RETURN CODE x AND REASON CODE  
y**

**Explanation:** Critical. For the data set with the associated ddname, the DYNALLOC SVC failed while attempting to obtain the pathname for an HFS file, or while attempting to determine if the data set was an HFS data set. x and y are the return code and reason code, respectively, from the DYNALLOC SVC.

**System Action:** The program terminates.

**Programmer Response:** Refer to the explanation for DYNALLOC Return Codes in *Authorized Assembler Services Guide* for more information about the indicated return code and reason code. If possible, take appropriate action.

---

**ICE191I      DFSORT SVC NOT AVAILABLE -  
PERFORMANCE MAY BE DEGRADED**

**Explanation:** An input, output, or work data set was on a cached DASD control unit, but the DFSORT SVC was either not installed on the system or installed incorrectly. DFSORT did not use its SVC to set the optimal caching mode for the data set, which could have lengthened DFSORT's elapsed time. Message ICE816I was also printed if DFSORT abended while trying to call the SVC.

**System Action:** Processing continues.

**Programmer Response:** Have your system programmer check to see that the DFSORT SVC has been installed properly for this level of the DFSORT program and that the correct routing code for the SVC

has been selected. If necessary, rerun the job with a SORTDIAG DD dummy statement to get message ICE816I.

---

**ICE192I      INVALID RETURN CODE FROM EXIT  
Enn WAS ACCEPTED**

**Explanation:** User exit Enn (E15 or E35) inappropriately passed DFSORT a return code of 0, 4, 12 or 20. Because EXITCK=WEAK was in effect, the invalid return code was accepted and interpreted as a valid return code. See *DFSORT Application Programming Guide* for an explanation of how DFSORT interprets E15 and E35 return codes when EXITCK=WEAK is in effect.

This message is issued once for E15 if it passes one or more inappropriate return codes to DFSORT or once for E35 if it passes one or more inappropriate return codes to DFSORT.

**System Action:** Processing continues.

**Programmer Response:** None.

---

**ICE193I      ICEAMx ENVIRONMENT IN EFFECT -  
ICEyyy INSTALLATION MODULE  
SELECTED**

**Explanation:** ICEAMx indicates the environment in effect for this run as follows:

- ICEAM1 - batch JCL directly invoked environment
- ICEAM2 - batch program invoked environment
- ICEAM3 - TSO directly invoked environment
- ICEAM4 - TSO program invoked environment

ICEyyy indicates the time-of-day installation module (ICETDx) or environment installation module (ICEAMx) selected for this run. The set of installation defaults for the selected installation module are used for the run.

An ICETDx module can only be selected if:

- it is specified by the ENABLE parameter of the ICEAMx module in effect for the run-time environment, and
- it specifies a day and time range that includes the day and time of the run (as listed in DFSORT's ICE000I message)

If an ICETDx module is not selected, the ICEAMx module in effect for the run-time environment is selected.

**Note:** You can use ICETOOL's DEFAULTS operator to list the ENABLE values for the ICEAMx modules, the day and time ranges for the ICETDx modules and all of the installation defaults for all of the ICEAMx and ICETDx modules.

**System Action:** None.

**Programmer Response:** None.

## Informational and Error Messages

---

### ICE195A    OVERFLOW DURING SUMMATION – RC=16

**Explanation:** Critical. OVFL=RC16 was in effect and a pair of summary fields overflowed.

**System Action:** The program terminates.

**Programmer Response:** Redesign the records so that summary fields do not overflow, or if possible, use INREC to increase the size of the summary fields (see *DFSORT Application Programming Guide*). Alternatively, specify the OVFL=RC0 or OVFL=RC4 run-time option, as appropriate, to allow summary overflow for this application.

---

### ICE196A    SORTOUT LRECL OF *x* IS DIFFERENT FROM SORTIN(NN) LRECL OF *y* - RC=16

**Explanation:** Either:

- PAD=RC16 was in effect and the SORTOUT LRECL (*x*) was larger than the SORTIN/SORTINnn LRECL (*y*), or
- TRUNC=RC16 was in effect and the SORTOUT LRECL (*x*) was smaller than the SORTIN/SORTINnn LRECL (*y*).

**System Action:** The program terminates.

**Programmer Response:** Change the LRECL of SORTOUT or SORTIN/SORTINnn so that the LRECLs are equal. Alternatively, specify the PAD=RC0, PAD=RC4, TRUNC=RC0, or TRUNC=RC4 run-time option, as appropriate, to allow different LRECLs for this application.

---

### ICE197I    ONE OR MORE INCOMPLETE SPANNED RECORDS DETECTED AND ELIMINATED – RC=*n*

**Explanation:** The application was successfully processed, but one or more incomplete spanned records was detected. DFSORT set a return code of 0 or 4, as indicated by *n*, according to whether SPANINC=RC0 or SPANINC=RC4 was in effect.

**System Action:** Processing continues and DFSORT passes back a return code of *n* unless it sets a higher return code for some other reason. Incomplete spanned records are eliminated and valid records are recovered.

**Programmer Response:** None.

---

### ICE198I    DEBUG EQUCOUNT CANNOT BE USED WITHOUT WORK DATA SET

**Explanation:** DEBUG EQUCOUNT was not used for this variable-length-record Blockset sort because no work data set was supplied.

**System Action:** Processing continues. DEBUG EQUCOUNT is not used.

**Programmer Response:** To use EQUCOUNT for this application, at least one work data set must be supplied.

---

### ICE200I    IDENTIFIER FROM CALLING PROGRAM IS *cccc*

**Explanation:** *cccc* is the identifier passed by the calling program for this invocation of DFSORT.

**System Action:** None.

**Programmer Response:** If appropriate, use *cccc* to relate this set of DFSORT messages to the call to DFSORT that passed this identifier.

---

### ICE201I    RECORD TYPE IS *x* – DATA STARTS IN POSITION *y*

**Explanation:** Indicates whether DFSORT processes the input records as fixed-length (without an RDW) or variable-length (with an RDW).

- For fixed-length record processing, *x* is F and *y* is 1. Since the input records are processed without an RDW, the data starts in position 1. Positions in your control statements should be specified accordingly.
- For variable-length record processing, *x* is V and *y* is 5. Since the input records are processed with an RDW in positions 1-4, the data starts in position 5. Positions in your control statements should be specified accordingly.

**System Action:** None.

**Programmer Response:** None, unless you are using VSAM input, or supplying all input with an E15 or E32 exit, and the selected record type is not what you want DFSORT to use. In that case, specify RECORD TYPE=F or RECORD TYPE=V as appropriate.

---

### ICE202I    SYSUT2 LRECL OF *x* IS DIFFERENT FROM SYSUT1 LRECL OF *y* – RC = *n*

**Explanation:** The ICEGENER application was successfully processed, but the SYSUT2 LRECL (*x*) and SYSUT1 LRECL (*y*) were not equal. ICEGENER set a return code of 0 or 4 as indicated by *n*. ICEGENER only set a return code of 4 if:

- ICEMAC option GNPAD=RC4 was specified and *x* was larger than *y* (LRECL padding), or
- ICEMAC option GNTRUNC=RC4 was specified and *x* was smaller than *y* (LRECL truncation).

**System Action:** Processing continues and ICEGENER passes back a return code of *n* unless it sets a higher return code for some other reason.

**Programmer Response:** If appropriate, change the LRECL of SYSUT1 or SYSUT2 so that the LRECLs are equal.



---

**ICE204A INCOMPLETE SPANNED RECORD  
DETECTED ON *ddname* – RC=16**

**Explanation:** Critical. SPANINC=RC16 was in effect and a spanned record in the indicated data set was incomplete.

**System Action:** The program terminates.

**Programmer Response:** Delete all incomplete spanned records from the indicated data set. Alternatively, specify the SPANINC=RC0 or SPANINC=RC4 run-time option, as appropriate, to allow DFSORT to eliminate incomplete spanned records and recover valid records.

---

**ICE205A CONTROL WAS NOT PASSED TO  
ICETEXIT**

**Explanation:** Critical. Control was not passed to ICETEXIT because an error occurred while collecting statistical information for the termination installation exit (ICETEXIT).

**System Action:** The program terminates.

**Programmer Response:** Notify your system programmer of the problem.

---

**ICE210I *xxxx* USED, LRECL = *n*, BLKSIZE = *n*,  
TYPE = *t* [(SDB)]**

**Explanation:** Gives attribute and access method information for the OUTFIL data set with the indicated *ddname*, as follows:

- *xxxx* shows the access method used: EXCP, BSAM, or VSAM.
- LRECL=*n* shows the logical record length.
- BLKSIZE=*n* shows the block size (non-VSAM) or control interval size (VSAM).
- TYPE=*t* shows the record format where *t* is one or more of the following format indicators:
  - F - fixed-length records
  - V - EBCDIC or ASCII variable-length records
  - B - blocked records
  - S - standard for fixed-length records or spanned for variable-length records
  - A - ASA control characters
- (SDB) shows that the system-determined optimum block size was used.

**System Action:** None.

**Programmer Response:** None.

---

**ICE211I OLD OUTFIL STATEMENT  
PROCESSING USED**

**Explanation:** This OUTFIL statement did not have any of the following parameters: FNAMES, FILES, STARTREC, ENDREC, INCLUDE, OMIT, SAVE, SPLIT, OUTREC, LINES, HEADER1, TRAILER1, HEADER2, TRAILER2, SECTIONS, or NODETAIL. For compatibility, this OUTFIL statement was treated as an

“old” OUTFIL statement and all of its parameters were ignored.

**System Action:** Processing continues, but OUTFIL data sets are not associated with this OUTFIL statement. If the Blockset technique is not selected, control statement errors could result from continuation of this OUTFIL statement.

**Programmer Response:** None, unless this is not an old OUTFIL statement, in which case valid parameters from the list above should be specified.

---

**ICE212A MATCH NOT FOUND FOR *ddname*  
CHANGE FIELD AT POSITION *p***

**Explanation:** Critical. The CHANGE parameter was specified without a NOMATCH parameter and an input field value did not match any of the find constants. The first data set in the OUTFIL group for which a match was not found is indicated by *ddname*. The position of a change field for which a match was not found is indicated by *p*.

**System Action:** The system terminates when the first input field is encountered for which a match is not found.

**Programmer Response:** Correct the lookup table specified with the CHANGE parameter, or use the NOMATCH parameter to specify a constant or input field to be used as the output field if no match is found. Use of a constant such as NOMATCH=(C'\*\*\*') can be helpful in identifying all input field values for which a match is not found.

---

**ICE213A INVALID OUTFIL STATEMENT  
OPERAND**

**Explanation:** Critical. An OUTFIL control statement contained an invalid keyword operand.

**System Action:** The program terminates.

**Programmer Response:** Make sure that the OUTFIL control statement does not contain an invalid keyword operand.

---

**ICE214A DUPLICATE OUTFIL STATEMENT  
OPERAND**

**Explanation:** Critical. On an OUTFIL control statement, a keyword was specified twice, or INCLUDE and OMIT, INCLUDE and SAVE, or OMIT and SAVE was specified.

**System Action:** The program terminates.

**Programmer Response:** Correct the keywords in error.

## Informational and Error Messages

---

### ICE215A SPECIFIED FEATURE IS ONLY SUPPORTED BY OUTFIL STATEMENT

**Explanation:** Critical. One of the following, which is only supported by the OUTFIL control statement, was specified on an INREC or OUTREC control statement:

- CONVERT parameter
- VLFILL=byte parameter
- / in FIELDS parameter

**System Action:** The program terminates.

**Programmer Response:** If CONVERT was specified, use CONVERT and OUTREC in an OUTFIL statement instead. If VLFILL=byte was specified, use VLFILL=byte and OUTREC in an OUTFIL statement instead. If / was specified, use / in OUTREC in an OUTFIL statement instead.

---

### ICE217A *n* BYTE VARIABLE RECORD IS LONGER THAN *m* BYTE MAXIMUM FOR [*ddname*]E15]E32]E35}

**Explanation:** Critical. A variable-length record was larger than the maximum length specified or defaulted. The values shown in the message are as follows:

- *n* is the length of the variable-length record
- *m* is the maximum length specified or defaulted
- *ddname* indicates that the variable-length record, read from a SORTIN (sort or copy) or SORTIN*nn* (merge) data set or to be written to the SORTOUT or OUTFIL data set, had a length greater than the LRECL specified or defaulted. For SORTIN*nn*, the *ddname* is shown as 'SORTIN*nn*'. For SORTIN, SORTOUT, or OUTFIL, the actual *ddname* is shown.
- E15 indicates that an E15 exit passed a variable-length record with a length greater than the L2 value specified or defaulted
- E32 indicates that an E32 exit passed a variable-length record with a length greater than the L1 value specified or defaulted
- E35 indicates that an E35 exit passed a variable-length record with a length greater than the L3 value specified or defaulted

**System Action:** The program terminates.

**Programmer Response:** If variable-length records are longer than the LRECL of the SORTOUT or OUTFIL data set and you want to truncate long records to the LRECL, use the VLLONG option. Keep in mind that the use of VLLONG causes data at the end of long records to be lost.

If variable-length records are longer than the LRECL of the SORTIN data set, the first data set in the concatenation does not have the largest LRECL as required. Either specify the data set with the largest LRECL first in the SORTIN concatenation, or specify LRECL=*n* as a parameter on the SORTIN DD statement where *n* is equal to or greater than the length of the longest record.

If variable-length records from an E15, E32, or E35 exit

are longer than the corresponding *Ln* value, correct the RDW or *Ln* value, as appropriate.

You can use the ICETOOL STATS operator with ON(VLEN) to show the length of the longest variable-length record. You can use the ICETOOL DISPLAY operator with ON(VLEN) to show the length of each variable-length record.

---

### ICE218A *n* BYTE VARIABLE RECORD IS SHORTER THAN *m* BYTE MINIMUM FOR [*ddname*] FIELDS

**Explanation:** Critical. A variable-length record was shorter than 4 bytes, or was too short to contain all specified fields. The values shown in the message are as follows:

- *n* is the length of the variable-length record
- *m* is 4 or the minimum length needed for all fields specified
- if *ddname* is blank, the short record was detected during non-OUTFIL processing. Otherwise, the first data set in the OUTFIL group for which the short record was detected is indicated by *ddname*.

**System Action:** The program terminates.

**Programmer Response:** If the *ddname* is not blank and a variable-length record was too short to contain all fields specified in an OUTFIL OUTREC statement, specify the VLFILL=C'x' or VLFILL=X'yy' operand for that OUTFIL statement.

If the *ddname* is blank and the records are valid but too short to contain all fields, use the VLSCMP option (INCLUDE, OMIT) or VLSHRT option (INCLUDE, OMIT, SORT, MERGE) to prevent DFSORT from terminating.

See the Programmer Response for message ICE015A for additional information.

---

### ICE219I DUPLICATE DDNAME *ddname* IGNORED

**Explanation:** The indicated *ddname* was specified previously in an OUTFIL statement in this source. The OUTFIL specifications are not used for this *ddname*, although they may be used for other *ddnames* in this OUTFIL statement which have not been previously specified in this source.

This message is only printed for the first 128 duplicate *ddnames* found.

**System Action:** Processing continues. The first OUTFIL *ddname* encountered in a source is used (unless overridden by the use of the same *ddname* in an OUTFIL statement in a higher source).

**Programmer Response:** No action is necessary if the OUTFIL processing specified for the duplicate *ddname* is not needed. Otherwise, correct the *ddname* conflicts.

---

**ICE220A**     *ddname* **COULD NOT BE USED - REASON CODE IS** *nn*

**Explanation:** Critical. The OUTFIL data set associated with *ddname* could not be processed. *nn* is the reason code value, as follows:

- 1     An attempt to OPEN the OUTFIL tape data set to retrieve its attributes failed.
- 2     The OUTFIL data set resided on an unsupported device.
- 3     An attempt to read the DSCB for the OUTFIL data set failed.
- 4     The OUTFIL data set was BDAM.

**System Action:** The program terminates.

**Programmer Response:** Correct the situation indicated by the reason code.

---

**ICE221A**     **INVALID FIELD OR CONSTANT FOUND IN** [*ddname*] **CONDITION** *n*

**Explanation:** Critical. An error was detected in the COND parameter of an INCLUDE or OMIT statement (*ddname* is blank), or in the INCLUDE or OMIT parameter of an OUTFIL statement (*ddname* indicates the first data set in the OUTFIL group). *n* indicates the number of the relational condition in which one of the following errors was detected:

- the length for a PD field was 256
- the length for a CSF field was greater than 16
- the length for a CSL, CST, ASL, or AST field was 1
- the decimal constant for an FI field was greater than 2147483647 or less than -2147483648
- the number of digits (including leading zeros) in the decimal constant for an FI field was greater than 15
- the length for a Y2 field was not 2 for Y2C, Y2Z, Y2P or Y2S, or 1 for Y2D or Y2B, or 3-6 for Y2T or Y2W, or 2-3 for Y2U or Y2X, or 3-4 for Y2V or Y2Y
- a Y2 field was compared to another Y2 field with a different number of non-year digits
- a Y2 field was compared to a Y constant with a different number of non-year digits
- a Y2 field other than Y2S, Y2T or Y2W was compared to Y'LOW', Y'BLANKS' or Y'HIGH'
- a Y2 field was compared to a decimal constant instead of to a Y constant

**System Action:** The program terminates.

**Programmer Response:** Correct the field length or constant in error in relational condition *n*.

---

**ICE222A**     *n* **BYTE FIXED RECORD LENGTH IS NOT EQUAL TO** *m* **BYTE LRECL FOR** *ddname*

**Explanation:** Critical. The LRECL specified or retrieved for the fixed-length OUTFIL data set was not equal to the computed length of the output records for that data set. You cannot use the LRECL value to pad

the OUTFIL records or to truncate the records produced by OUTREC parameter processing. The values shown in the message are as follows:

- *n* is the computed length of the output records for the OUTFIL group
- *m* is the specified or retrieved LRECL of the OUTFIL data set
- *ddname* indicates the OUTFIL data set for which padding or truncation was required

**System Action:** The program terminates.

**Programmer Response:** Either let DFSORT set the LRECL to the computed record length, or use the OUTREC operand of the OUTFIL statement to set the output record length to the specified LRECL. Remember to allow an extra byte for OUTFIL report data sets (for the ASA carriage control character).

---

**ICE223A**     **REPORT FIELD ERROR**

**Explanation:** Critical. The LINES, HEADER1, TRAILER1, HEADER2, TRAILER2, or SECTIONS parameter of an OUTFIL statement contained an invalid column, report element, position, length, keyword, pattern, sign, or constant. Some common errors are :

- A 0 value was used.
- A null value was used where it was not permitted.
- A null string, pattern, or sign was used.
- A column was greater than 32752, preceded / or n/ (new line), or was followed by another column.
- A column overlapped the previous output field in the report record (a missing new line (/ or n/) to end the current report record and start the next one can cause this error).
- A position plus length was greater than 32753.
- DATE=(abcd) was specified with a, b or c not M, D, Y or 4, with M, D, Y or 4 specified more than once, or with Y and 4 both specified.
- TIME=(abc) was specified with ab not 12 or 24.
- The length for an input field or section break field was greater than 256 bytes.
- A repetition factor was greater than 4095 for a blank or character string report element, or greater than 255 for a blank lines report element or a section skip line count.
- A character string was longer than 256 bytes.
- The length for a statistics field was greater than 4 bytes for BI or FI, 8 bytes for PD, 15 bytes for ZD, or 16 bytes for CSF/FS.
- More than 15 digits or 22 characters were specified in a statistics pattern.
- SIGNz (where z is not S) was specified with Mn or without EDIT or EDxy.
- x, y, or z in EDxy or SIGNz were the same character.
- The value for LENGTH was greater than 22.
- The value for LINES was greater than 255.
- A section break field was not followed by SKIP, HEADER3, or TRAILER3.
- A statistics field was specified in HEADER1, HEADER2, or HEADER3.

## Informational and Error Messages

- HEADER3, TRAILER3, SKIP, or PAGEHEAD was specified more than once after a section break field.

**System Action:** The program terminates.

**Programmer Response:** Correct the invalid value.

---

### ICE224A *ddname* CANNOT BE USED FOR A REPORT

**Explanation:** Critical. The OUTFIL report specified for the OUTFIL data set associated with *ddname* cannot be produced because either:

- the OUTFIL data set is VSAM, or
- the record format specified or retrieved for the OUTFIL data set does not contain 'A' (ASA carriage control characters).

**System Action:** The program terminates.

**Programmer Response:** If the OUTFIL data set is VSAM, it cannot be used for a report. If the OUTFIL data set is non-VSAM, either let DFSORT set the RECFM appropriately or specify a RECFM that includes 'A'.

---

### ICE225A *ddname* : LRECL = *n* IS INVALID

**Explanation:** Critical. The specified or computed LRECL (*n*), for the OUTFIL data set associated with *ddname*, is invalid for one of the following reasons:

- LRECL=X was specified (*n* will be 32768)
- *n* is less than the minimum allowed: 1 (not a report) or 2 (report) for a fixed-length record data set or 5 (not a report) or 6 (report) for a variable-length record data set
- *n* is greater than the maximum allowed: 32760 for a fixed-length record data set, 32756 for a variable-length record data set, 32767 for a spanned record data set or 32752 for a VSAM variable-length record data set

**System Action:** The program terminates.

**Programmer Response:** If LRECL=X was specified, specify the actual maximum LRECL instead. If the LRECL is less than the minimum allowed, increase it. If the LRECL is greater than the maximum allowed, decrease it.

---

### ICE226A BLKSIZE = *n* IS INVALID (LRECL = *m*)

**Explanation:** Critical. The specified or computed block size (*n*), for the OUTFIL data set associated with *ddname*, is invalid for one of the following reasons (*m* is the associated LRECL):

- *n* is less than the minimum allowed: 1 (not a report) or 2 (report) for a fixed-length record data set or 9 (not a report) or 10 (report) for a variable-length record data set
- *n* is less than *m* for a fixed-length record data set
- *n* is not a multiple of *m* for a fixed-length record data set

- *n* is not equal to *m* for an unblocked fixed-length record data set
- *n* is less than *m* + 4 for a variable-length record data set
- *n* is greater than the track capacity of the device on which the data set resides

**System Action:** The program terminates.

**Programmer Response:** Change the block size to a valid value.

---

### ICE227I *ddname* : DELETED = *n*, REPORT = *n*, DATA = *n*

**Explanation:** For the OUTFIL data set associated with *ddname*, lists the number of OUTFIL input records deleted (that is, not written to this data set) and the number of report and data records written.

**System Action:** None.

**Programmer Response:** None.

---

### ICE228I *ddname* : TOTAL IN = *n*, TOTAL OUT = *n*

**Explanation:** For the OUTFIL data set associated with *ddname*, lists the number of OUTFIL input records and the number of OUTFIL output records (report plus data) written.

**System Action:** None.

**Programmer Response:** None.

---

### ICE229A *ddname* : LINES = *n* IN EFFECT, BUT *m* LINES REQUIRED

**Explanation:** Critical. For the OUTFIL group whose first data set is associated with *ddname*, the value specified or defaulted for LINES (*n*) is too low to contain the number of lines (*m*) required for the maximum of:

- The HEADER1 lines
- The TRAILER1 lines
- The sum of all lines for HEADER2, TRAILER2, HEADER3s, and TRAILER3s, and the data lines and blank lines produced from an input record.

**System Action:** The program terminates.

**Programmer Response:** Set LINES=*n* to a value equal to or greater than *m*, or reduce the number of lines required.

---

### ICE230A *n* BYTE HEADER/TRAILER RECORD LENGTH EXCEEDS *m* BYTE LRECL FOR *ddname*

**Explanation:** Critical. The specified, retrieved, or computed LRECL for the OUTFIL data set was less than the computed length for the report records for that data set. The values shown in the message are as follows:



- n is the computed length of the report records for the OUTFIL group
- m is the specified, retrieved, or computed LRECL of the OUTFIL data set
- ddname indicated the OUTFIL data set for the report

**System Action:** The program terminates.

**Programmer Response:** Use the OUTREC parameter to force a length for the data records that is longer than any report record, and either let DFSORT compute and set the LRECL, or ensure that the computed LRECL is equal to the existing or specified LRECL. Remember to allow an extra byte in the LRECL for the ASA carriage control character.

For example, if your data records are 40 bytes, but your longest report record is 60 bytes, you could use an OUTREC parameter such as:

```
OUTREC=(1,40,80:X)
```

DFSORT will then set the LRECL to 81 (1 byte for the ASA carriage control character plus 80 bytes for the length of the data records), and pad the data records with blanks on the right.

---

**ICE231I      STORAGE USED FOR OUTFIL :  
                  BELOW 16M = x, ABOVE 16M = y**

**Explanation:** Gives information related to the main storage used exclusively for OUTFIL processing. x is the storage used below 16MB virtual and y is the storage used above 16MB virtual.

**System Action:** None.

**Programmer Response:** None.

---

**ICE232A      ddname: A REPORT CANNOT BE SPLIT**

**Explanation:** Critical. For the OUTFIL group whose first data set is associated with ddname, the SPLIT parameter was specified along with one or more report parameters (LINES, HEADER1, TRAILER1, HEADER2, TRAILER2, SECTIONS or NODETAIL). The records of a report cannot be split among a group of OUTFIL data sets.

**System Action:** The program terminates.

**Programmer Response:** Remove either the SPLIT parameter or the report parameters.

---

**ICE233A      OUTFIL STATEMENTS CANNOT BE  
                  PASSED TO OR FROM AN EFS  
                  PROGRAM**

**Explanation:** Critical. An OUTFIL statement was to be passed to an EFS program or an EFS program passed back an OUTFIL statement. DFSORT does not allow OUTFIL statements to be processed by an EFS program that requests them, and does not allow an EFS

program to change other statements passed to it into OUTFIL statements.

**System Action:** The program terminates.

**Programmer Response:** Change the EFS program such that it does not request OUTFIL statements or change other statements it requests into OUTFIL statements.

---

**ICE234A      STARTREC AND ENDREC VALUES  
                  ARE INCONSISTENT**

**Explanation:** Critical. The value specified for ENDREC is less than the value specified for STARTREC. The ending record cannot be lower than the starting record.

**System Action:** The program terminates.

**Programmer Response:** Specify an ENDREC value that is equal to or greater than the STARTREC value.

---

**ICE270I      PROCESSING SYMNames  
                  STATEMENTS**

**Explanation:** A SYMNames data set was specified indicating that symbol processing is to be performed. DFSORT or ICETOOL will process the SYMNames statements and show any SYMNames statements in error after ICE270I.

**System Action:** None.

**Programmer Response:** None.

---

**ICE271I      SYMNames IS EMPTY - SYMBOL  
                  SUBSTITUTION WILL NOT BE  
                  PERFORMED**

**Explanation:** A SYMNames data set was specified, but was empty (that is, it contained no statements). Symbol substitution will not be performed.

**System Action:** Processing continues without symbol substitution.

**Programmer Response:** If symbol substitution is needed, add appropriate SYMNames statements to the specified SYMNames data set or specify a non-empty SYMNames data set.

---

**ICE272A      SYMBOL, VALUE OR SYNTAX IS  
                  INVALID**

**Explanation:** Critical. The SYMNames statement has one of the following errors:

- The symbol starts with a number (0-9).
- The symbol or value contains an invalid character. The valid characters are uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), the number sign (#), the dollar sign (\$), the commercial at sign (@) and the underscore(\_).
- The symbol, keyword or value is null (for example, symbol,,5,CH).

## Informational and Error Messages

- The symbol, keyword or value is followed by or contains an invalid delimiter.
- p or m in p,m or p,m,f is 0 or greater than 32752 or contains a non-numeric character.
- q in POSITION,q is 0 or greater than 32752 or contains a non-numeric character.
- symbol in POSITION,symbol references a symbol which was not previously defined or which was previously defined without a valid position (for example, a symbol for a constant).
- n in SKIP,n is 0 or greater than 32752 or contains a non-numeric character.
- The decimal constant contains a non-numeric character other than a leading plus sign (+) or minus sign (-).
- An equal sign (=) is specified for p, m or f, but the *previous position*, *previous length* or *previous format*, respectively, was not established.
- f in p,m,f is not a valid format. The valid formats are AC, AQ, ASL, AST, BI, CH, CLO, CSF, CSL, CST, CTO, D1, D2, FI, FL, FS, LS, OL, OT, PD, PD0, SS, TS, Y2B, Y2C, Y2D, Y2DP, Y2P, Y2PP, Y2S, Y2T, Y2TP, Y2U, Y2UP, Y2V, Y2VP, Y2W, Y2WP, Y2X, Y2XP, Y2Y, Y2YP, Y2Z and ZD, and lowercase or mixed case variations.
- x in ALIGN,x is not a valid alignment. The valid alignments are H, F, D, h, f and d.
- The character, hexadecimal or bit constant does not have an ending apostrophe after the string.
- The hexadecimal constant is null (X'') or contains an odd number of digits (for example, X'123').
- The hexadecimal constant contains an invalid character. The valid characters are 0-9, A-F and a-f.
- The bit constant is null (B'') or contains a number of bits which is not a multiple of 8 (for example, B'1010').
- The bit constant contains an invalid character. The valid characters are . (period), 0 and 1.

**System Action:** The program terminates.

**Programmer Response:** Correct the symbol, value or syntax error.

---

### ICE273A SYMBOL OR VALUE IS TOO LONG

**Explanation:** Critical. The SYMNAMES statement has one of the following errors:

- The symbol is longer than 50 characters.
- The value contains a number longer than 15 significant digits.
- The constant contains a string longer than 64 characters.

**System Action:** The program terminates.

**Programmer Response:** Specify a symbol, value or string that is less than or equal to the allowed limit.

---

### ICE274A NEXT POSITION TO BE USED FOR \* IS LARGER THAN 32752

**Explanation:** Critical. The SYMNAMES statement specifies an asterisk (\*) for p requesting that the *next position* be used for p. However, the *next position* is greater than the limit for p of 32752.

Calculation of the *next position* will be discontinued once a SYMNAMES statement error has been detected. Therefore, ICE274A is only issued once and only if a previous SYMNAMES processing error was not detected.

**System Action:** The program terminates.

**Programmer Response:** Specify a value for p that is less than or equal to 32752. You can do this directly by specifying a number for p or indirectly by specifying \* for p and ensuring that the *next position* is less than or equal to 32752. See *DFSORT Application Programming Guide* for an explanation of how the *next position* is set.

---

### ICE275A DUPLICATE SYMBOL

**Explanation:** Critical. The symbol was used in a previous SYMNAMES statement. A symbol can only be used once in SYMNAMES.

**System Action:** The program terminates.

**Programmer Response:** Remove or rename one of the duplicate symbols.

---

### ICE276A RESERVED WORD - NOT ALLOWED FOR SYMBOL

**Explanation:** Critical. The SYMNAMES statement specifies a DFSORT/ICETOOL reserved word for the symbol. Reserved words cannot be used for symbols. The reserved words are as follows (uppercase only as shown): A, AC, ALL, AND, AQ, ASL, AST, BI, CH, CLO, COPY, COUNT, COUNT15, CSF, CSL, CST, CTO, D, DATE, D1, D2, E, F, FI, FL, FS, H, HEX, LS, Mn, Mnn, NONE, NUM, OL, OR, OT, PAGE, PAGEHEAD, PD, PD0, SEQNUM, SS, SUBCOUNT, SUBCOUNT15, TIME, TS, VALCNT, VLEN, X, Y2x, Y2xx, Z and ZD where n is 0-9 and x is any character.

**System Action:** The program terminates.

**Programmer Response:** Use a symbol that is not one of the reserved words, such as a lowercase or mixed case version of the word being used. For example, you could use Valcnt which is not a reserved word instead of VALCNT which is.

---

### ICE279A ONE OR MORE ERRORS ENCOUNTERED DURING SYMNAMES PROCESSING

**Explanation:** Critical. DFSORT or ICETOOL detected one or more errors while processing SYMNAMES statements.

**System Action:** The program terminates.

**Programmer Response:** Correct each error indicated by a previous error message (ICE272A through ICE276A).

### ICE280I ORIGINAL STATEMENTS FROM source FOLLOW

**Explanation:** This message is issued for each source specified. It is followed by a listing of the original DFSORT control statements before they are transformed by symbol processing. Source can be one of the following:

- DFSPARM or the ddname specified in ICEMAC option PARMDDN
- SYSIN
- xxxxCNTL
- PARMLIST

**System Action:** None.

**Programmer Response:** None.

### ICE281I ORIGINAL OPERATOR STATEMENTS FOLLOW

**Explanation:** This message is followed by a listing of the original ICETOOL operator statements before they are transformed by symbol processing. The operator statements listed are from TOOLIN or the calling program's parameter list as indicated by message ICE632I.

**System Action:** None.

**Programmer Response:** None.

### ICE282I PERFORMING SYMBOL SUBSTITUTION AS NEEDED

**Explanation:** DFSORT or ICETOOL successfully processed the SYMNAMES statements. ICETOOL will transform operators as necessary using symbol substitution. DFSORT will transform control statements as necessary using symbol substitution. Any operators or control statements with substitution errors will be shown after ICE282I.

**System Action:** None.

**Programmer Response:** None.

### ICE283A SYMBOL, SYNTAX OR DELIMITER ERROR

**Explanation:** Critical. During symbol substitution processing, one of the following errors was found in a control statement or operator:

- A specified symbol was not defined in SYMNAMES.

**Note:** Symbols are treated as case-sensitive: Frank, FRANK and frank are three **different** symbols. Be careful to use symbols in control statements and operators exactly as they are defined in SYMNAMES.

- A specified symbol was longer than 50 characters.
- A specified symbol started with an invalid character or with a delimiter. Examples:
 

```
SORT FIELDS=(.Sym1,A)
* .Sym1 is invalid. FIELDS=(Sym1,A) is valid.
INREC FIELDS=((Sym1)
* ((Sym1) is invalid. FIELDS=(Sym1) is valid.
```
- A specified symbol contained an invalid character. Examples:
 

```
SUM FIELDS=(New-Field)
* New-Field is invalid. New_Field is valid.
OUTREC FIELDS=(A%B)
* A%B is invalid.
```

**Note:** A symbol with an invalid character cannot be defined in SYMNAMES.

- A specified symbol was followed by an invalid delimiter. Examples:
 

```
SUM FIELDS=(Sym1 )
* (Sym1 ) is invalid. FIELDS=(Sym1) is valid.
OUTFIL OUTREC=(Sym1/)
* Sym1/ is invalid. OUTREC=(Sym1,/) is valid.
```
- A delimiter was misplaced. Examples:
 

```
VERIFY ON(,Sym1)
* (,Sym1) is invalid. ON(Sym1) is valid.
VERIFY ON((5,4,PD)
* ((5,4,PD) is invalid. ON(5,4,PD) is valid.
VERIFY ON(5,4,PD))
* (5,4,PD)) is invalid. ON(5,4,PD) is valid.
```
- A syntax error caused a value to be misinterpreted as an undefined symbol. Example:
 

```
OUTREC FIELDS=(1,4,ZD)
```

ZD is misinterpreted as an undefined symbol because p,m,f is not allowed in an OUTREC statement. The following is valid since p,m,f is allowed in the OUTREC operand of an OUTFIL statement:

```
OUTFIL OUTREC=(1,4,ZD)
```

**System Action:** The program terminates.

**Programmer Response:** Correct the symbol, syntax or delimiter error.

### ICE284A SYMBOL SUBSTITUTION CAUSES OPERAND TO OVERFLOW COLUMN 72

**Explanation:** Critical. The operand contained a symbol. Substitution of the value for the symbol caused the operand to exceed 72 characters if the line was not continued or 71 characters if the line was continued. ICETOOL operators must be contained in columns 1-72, so the transformed operand cannot be processed by ICETOOL.



## Informational and Error Messages

**System Action:** The program terminates.

**Programmer Response:** Use a symbol or value that will not cause the operand to overflow column 72. If necessary, change the value in the SYMNames statement for the symbol.

---

### ICE287A ONE OR MORE ERRORS ENCOUNTERED DURING SYMBOL SUBSTITUTION

**Explanation:** Critical. DFSORT or ICETOOL detected one or more errors during symbol substitution processing.

**System Action:** The program terminates.

**Programmer Response:** Correct each error indicated by a previous error message (ICE283A or ICE284A).

---

### ICE600I DFSORT ICETOOL UTILITY RUN STARTED

**Explanation:** Indicates the start of the ICETOOL run.

**System Action:** None.

**Programmer Response:** None.

---

### ICE601I DFSORT ICETOOL UTILITY RUN ENDED - RETURN CODE: *nn*

**Explanation:** Indicates the end of the ICETOOL run and the highest return code encountered. Message ICE602I gives the return code for each operation.

**System Action:** None.

**Programmer Response:** None if *nn* is 00. Otherwise, check the ICE602I messages for non-zero return codes and take the action indicated.

---

### ICE602I OPERATION RETURN CODE: *nn*

**Explanation:** Indicates the return code for this operation.

**System Action:** If *nn* is greater than 04, this operation was terminated due to an error.

**Programmer Response:** None if *nn* is 00. Otherwise:

- If *nn* is 04, match the identifier in message ICE606I or ICE627I for this operation to the same identifier in DFSORT message ICE200I (in the DFSMSG data set) to find the set of DFSORT messages for this operation. If appropriate, correct the condition that caused the return code of 4 indicated in the relevant DFSORT informational message.
- If *nn* is 12, check the error message(s) for this operation in the TOOLMSG data set and correct the indicated errors.
- If *nn* is 16, match the identifier in message ICE606I or ICE627I for this operation to the same identifier in DFSORT message ICE200I (in the DFSMSG data

set) to find the set of DFSORT messages for this operation. Correct the error indicated in the DFSORT error message.

---

### ICE603I INFORMATION PRINTED IN *ddname* DATA SET

**Explanation:** The information requested for this operator was printed in the list data set with the indicated *ddname*. If the operation was terminated, the list data set can be incomplete.

**System Action:** None.

**Programmer Response:** None.

---

### ICE604A ERROR IN KEYWORD, PARAMETER, OR DELIMITER

**Explanation:** Critical. The statement contained an error in an operand (keyword, parameter) or a delimiter was incorrect or missing. Some common errors are:

- A keyword or parameter was misspelled. Example: ALLDUP instead of ALLDUPS
- A keyword was used with an operator for which it is not valid. Example: NOSIGN was used with an operator other than VERIFY.
- A parameter or value was used with a keyword for which it is not valid. Example: VSAMTYPE(U) instead of VSAMTYPE(F) or VSAMTYPE(V).
- The cccc value for USING(cccc) was SYS $c$  or was not four characters. Example: USING(ABC) instead of USING(ABCD) or USING(SYS1) instead of USING(SYX1).
- A left or right parenthesis was missing. Example: FROM IN instead of FROM(IN)
- A blank was used inside parentheses. Example: FROM( IN) instead of FROM(IN)
- A continuation indicator (-) was used incorrectly. Example: TO(OUT1,- instead of TO(OUT1,OUT2) -
- Parameters were not separated by a comma or semicolon. Example: ON(3:5:ZD) instead of ON(3,5,ZD) or ON(3;5;ZD)
- A numeric value was specified incorrectly. Example: ON(0,3,ZD) instead of ON(1,3,ZD) or LIMIT(+1) instead of LIMIT(1)
- An operand that can be specified only once per operator was specified more than once. Example: COPY TO(OUT1) TO(OUT2) instead of COPY TO(OUT1,OUT2)
- Mutually exclusive operands were used. Example:

RANGE  
EQUAL(10) LOWER(15)  
instead of  
RANGE EQUAL(10)  
or  
RANGE LOWER(15)

- A numeric value was too low or too high. Example: LINES(9) or LINES(1000)

## Informational and Error Messages

- A string was not enclosed in apostrophes. Example: HEADER(Revenue), HEADER('Revenue), HEADER('Revenue"), or HEADER("Revenue") instead of HEADER('Revenue')
- A string exceeded the character limit allowed. Example:

```
TITLE('This string is longer
than the limit of 50 characters
for TITLE')
```

**Note:** In your ICETOOL statement, the entire operand must be on *one* line.

- A parameter was specified incorrectly. Example: DATE(DMY) instead of DATE(DMY.)
- NOHEADER was used when HEADER(NONE) was required. Example: HEADER('Name') NOHEADER instead of HEADER('Name') HEADER(NONE)
- Mutually exclusive items were specified within an operand. Example: DATE(YM4/) instead of DATE(YMD/) or DATE(DM4/)
- A formatting item was specified for an operator other than DISPLAY. Example:

```
OCCUR ON(1,5,ZD,A1) instead of
OCCUR ON(1,5,ZD)
```

- A symbol was specified where it is not allowed. Example: LINES(Max\_Lines) instead of LINES(50)

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the point at which the error was detected. Correct the error.

### ICE605A REQUIRED DD STATEMENT NOT FOUND: *ddname*

**Explanation:** Critical. A DD statement for the indicated *ddname* was not present. The *ddname* was required because:

- It was DFSMSG.
- It was TOOLIN and the ICETOOL Parameter List Interface was not used.
- It was specified in the FROM, TO, LIST, or DISCARD operand for this operator.
- USING(*xxxx*) was specified for this operator.

**System Action:** If the *ddname* was TOOLIN or DFSMSG, ICETOOL terminates without performing any operations. Otherwise, this operation is terminated.

**Programmer Response:** Supply a DD statement for the indicated *ddname*.

### ICE606I DFSORT CALL *nnnn* FOR {COPY|SORT} FROM *ddname1* TO {*ddname2*|E35 EXIT|OUTFIL} USING *xxxxCNTL* {COMPLETED|TERMINATED}

**Explanation:** Supplies information about a specific call to DFSORT for this operation as follows:

**nnnn** The call identifier for this call to DFSORT. You

can match *nnnn* to the same *nnnn* in DFSORT message ICE200I in the DFSMSG data set to find the set of DFSORT messages for this call to DFSORT.

**COPY** The DFSORT copy function was used.

**SORT** The DFSORT sort function was used.

#### **ddname1**

The *ddname* of the data set that was copied or sorted.

#### **ddname2**

The *ddname* of the data set resulting from the copy or sort.

#### **E35 EXIT**

ICETOOL's E35 user exit was used.

#### **OUTFIL**

Either OUTFIL data sets were produced or an E35 user exit routine was used to dispose of all the output records

#### **xxxxCNTL**

The *ddname* of the DFSORT control data set used for the sort or copy.

#### **COMPLETED**

DFSORT did not detect any errors preventing completion of this ICETOOL operation.

#### **TERMINATED**

DFSORT detected an error which prevented completion of this ICETOOL operation.

**System Action:** None.

**Programmer Response:** You can use the information in this message to determine the result of this call to DFSORT. If DFSORT terminated, use the call identifier to find the set of DFSORT messages for this call (in the DFSMSG data set), and correct the error indicated in the DFSORT error message.

### ICE607I STATISTICS FOR {(*p,m,f*)}(VLEN):

**Explanation:** Indicates the field to which the statistics in subsequent messages ICE608I and ICE609I apply. (*p,m,f*) or (VLEN) is a field you specified for this STATS operator.

**System Action:** None.

**Programmer Response:** None.

### ICE608I MINIMUM: *snnnnnnnnnnnnnnnn*, MAXIMUM: *snnnnnnnnnnnnnnnn*

**Explanation:** Indicates the minimum and maximum for the field indicated in the ICE607I message preceding this message. Each value consists of a + or - sign and 15 decimal digits (padded with zeros on the left as needed). If the values could not be determined due to an error (as indicated in a previous message), asterisks were printed for the values.

## Informational and Error Messages

**System Action:** None.

**Programmer Response:** None unless asterisks were printed for the values, in which case you should correct the error indicated by the previous error message for this operation.

---

**ICE609I AVERAGE:** *snnnnnnnnnnnnnnnn*,  
**TOTAL:** *snnnnnnnnnnnnnnnn*

**Explanation:** Indicates the average and total for the field indicated in the ICE607I message preceding this message. Each value consists of a + or - sign and 15 decimal digits (padded with zeros on the left as needed). If a value could not be determined due to an error (as indicated in a previous message), asterisks were printed for the value.

**System Action:** None.

**Programmer Response:** None unless asterisks were printed for a value, in which case you should correct the error indicated by the previous error message for this operation.

---

**ICE610I NUMBER OF UNIQUE VALUES FOR**  
**(p,m,f)(VLEN):** *nnnnnnnnnnnnnnnn*

**Explanation:** Indicates the count of unique values for this field, printed as 15 decimal digits (padded with zeros on the left as needed). (p,m,f) or (VLEN) is the field you specified for this UNIQUE operator.

**System Action:** None.

**Programmer Response:** None.

---

**ICE611A TOTAL FOR{(p,m,f)(VLEN)}**  
**OVERFLOWED 15 DECIMAL DIGITS**

**Explanation:** Critical. The total for the indicated field was greater than +999999999999999 or less than -999999999999999. (p,m,f) or (VLEN) is a field you specified for this STATS or DISPLAY operator.

**System Action:** This operation is terminated.

For a STATS operator, asterisks are printed in message ICE609I for the average and total for this field.

For a DISPLAY operator, asterisks are printed for this field in any BAVERAGE, BTOTAL, AVERAGE, or TOTAL lines requested.

**Programmer Response:** If you need the average or total for this field, use the STATS or DISPLAY operator for subsets of the data set, which does not cause the total to overflow. Use the statistics for the subsets to determine the needed statistics for the field.

---

**ICE612I NO ERRORS FOUND IN STATEMENT**

**Explanation:** Printed in SCAN mode to indicate that no errors were found in the statement.

**System Action:** DFSORT is not called in SCAN mode.

**Programmer Response:** If SCAN mode was entered due to an error while in STOP mode, correct the error. If SCAN mode was entered due to a MODE SCAN statement, replace it with a MODE STOP or MODE CONTINUE statement.

---

**ICE613A REQUIRED KEYWORD MISSING:**  
*keyword*

**Explanation:** Critical. The indicated keyword was required for this operator, but was not specified. The required keywords and their operands for each operator are:

- COPY - FROM and TO or USING
- COUNT - FROM
- DEFAULTS - LIST
- DISPLAY - FROM, ON, and LIST. BREAK if BTITLE, BTOTAL, BMAXIMUM, BMINIMUM, or BAVERAGE is specified
- MODE - STOP, CONTINUE, or SCAN
- OCCUR - FROM, LIST, and ON(p,m,f), ON(p,m,HEX), or ON(VLEN)
- RANGE - FROM, ON, and HIGHER, LOWER, EQUAL, or NOTEQUAL
- SELECT - FROM, TO, ON, and ALLDUPS, NODUPS, HIGHER, LOWER. EQUAL, FIRST or LAST
- SORT - FROM, and USING
- STATS - FROM and ON
- UNIQUE - FROM and ON
- VERIFY - FROM and ON

**System Action:** This operation is terminated.

**Programmer Response:** Supply the indicated keyword or operand.

---

**ICE614A INVALID OPERATOR**

**Explanation:** Critical. The first keyword in the statement was not a valid operator. The valid operators are: COPY, COUNT, DEFAULTS, DISPLAY, MODE, OCCUR (or OCCURS), RANGE, SELECT, SORT, STATS, UNIQUE, and VERIFY.

A common cause of this error is a missing dash (-) on the previous line to indicate continuation.

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the point at which the error was detected. If an invalid operator was used, replace it with a valid operator. If this is a continuation line, use a dash after the last operand on the previous line.

---

**ICE615A DDNAME USED MORE THAN ONCE IN STATEMENT:** *ddname*

**Explanation:** Critical. This message was issued for one of the following reasons:

- TOOLIN, TOOLMSG, or DFSMSG was used as a *ddname* in a FROM, TO, LIST, or DISCARD operand for this operator.
- The indicated *ddname* was found more than once in a FROM, TO, LIST, or DISCARD operand for this operator.
- USING(*xxxx*) was specified and *xxxxCNTL* was used in a FROM or TO operand for this operator.

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the duplicate *ddname*. Replace it with a *ddname* which is unique for this operation and which is not TOOLMSG, DFSMSG, or TOOLIN.

---

**ICE616A STATEMENT DOES NOT END AT OR BEFORE COLUMN 72**

**Explanation:** Critical. The statement overran column 72, or a string contained unmatched quotes.

**System Action:** This operation is terminated.

**Programmer Response:** Recode the statement to end at or before column 72. Continuation can be indicated by a dash (-) after the operator or any operand. For example:

```

SORT FROM(INDD) -
      USING(ABCD) -
      TO(OUTPUT1,OUTPUT2,OUTPUT3)
    
```

Ensure that all quotes are matched as this message can also result from unmatched quotes in a string, for example, TITLE('Status Report').

---

**ICE617A RECORD COUNT OVERFLOWED 15 DECIMAL DIGITS**

**Explanation:** Critical. The number of records processed by ICETOOL for this operation was greater than 999999999999999.

**System Action:** This operation is terminated.

**Programmer Response:** Perform the operation on subsets of the data set containing less than 999999999999999 records.

---

**ICE618A INVALID (*p,m,f*) VALUE - RECORD:**  
*nnnnnnnnnnnnnnnnnn*, **HEX VALUE:** *h...h*

**Explanation:** Critical. Identifies an invalid decimal value in a specified field. One of the following was found:

- An invalid digit (A-F) in a field specified for a DISPLAY, RANGE, STATS, or VERIFY operator.

- An invalid sign (0-9) in a field specified for an OCCUR or VERIFY operator (NOSIGN was not specified).

(*p,m,f*) is a field you specified for this operator.

The invalid value was found in record number *nnnnnnnnnnnnnnnnnn* (prints as 15 decimal digits padded with zeros on the left as needed). For a DISPLAY, RANGE, STATS, or VERIFY operator, *nnnnnnnnnnnnnnnnnn* is the input record number (that is, the relative record number). For an OCCUR operator, *nnnnnnnnnnnnnnnnnn* is the sorted record number, and thus may not be useful.

*h...h* is the invalid value in hexadecimal.

**System Action:**

- For a DISPLAY operator with an invalid BREAK value, this operation is terminated.
- For a DISPLAY operator with an invalid ON value, asterisks are printed for this value in the data line and in any statistics lines requested. If this incorrect value causes the limit for decimal values to be reached, this operation is terminated after the current record is printed in the list data set. Otherwise, processing continues.
- For an OCCUR or RANGE operator, this operation is terminated.
- For a STATS operator, asterisks were printed in message ICE608I for minimum and maximum, and in message ICE609I for average and total for this field.
- For a VERIFY operator, if this bad value caused the limit for invalid decimal values to be reached, this operation is terminated. Otherwise, processing continues.

**Programmer Response:** Correct the invalid digit or sign in the identified field. The VERIFY or DISPLAY operator can be used to print all the invalid values and their relative record numbers.

---

**ICE619A INVALID LENGTH, FORMAT, OR COMBINATION FOR *operator* OPERATION**

**Explanation:** Critical. One of the following conditions was detected in the parameters of an ON or BREAK operand:

- The format was invalid. Example: ON(10,2,FL) or BREAK(10,2,FL)
- The format was not allowed for this operator. Example: VERIFY ON(10,2,BI)
- The length was not within the range allowed for the format and operator. Example:

```

STATS ON(10,8,BI)
    
```

- ON(VLEN) was specified for a VERIFY operator. Example: VERIFY ON(VLEN)
- ON(NUM) was specified for an operator other than DISPLAY. Example: STATS ON(NUM)



## Informational and Error Messages

- ON(VALCNT) was specified for an operator other than OCCUR. Example: DISPLAY ON(VALCNT)
- ON(p,m,HEX) was specified for an operator other than DISPLAY or OCCUR. Example:

UNIQUE ON(5,4,HEX)

- The length was not within the range allowed for ON(p,m,HEX). Example: DISPLAY ON(5,51,HEX)
- BREAK(p,m,HEX) was specified. Example:  
BREAK(5,4,HEX)

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the point at which the error was detected. Correct the error.

---

### ICE620A SUM OF POSITION AND LENGTH GREATER THAN *max*

**Explanation:** Critical. A specified field extended beyond the maximum position (*max* - 1) allowed for this operation.

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the point at which the error was detected. Change the position or length so that the field ends at or before position *max* - 1.

---

### ICE621I DFSORT DETECTED AN ERROR AFTER E35 EXIT PROCESSING COMPLETED

**Explanation:** A call to DFSORT for this operation resulted in a return code of 16. However, because the error was detected by DFSORT after ICETOOL's E35 user exit completed its processing, the error did not prevent the completion of this ICETOOL operation.

**System Action:** None.

**Programmer Response:** None required, but you can examine the DFSORT messages in DFSMSG to determine the error detected by DFSORT and correct it if appropriate.

---

### ICE622A EXPECTED CONTINUATION LINE NOT FOUND

**Explanation:** Critical. This message was issued for one of the following reasons:

- Continuation was indicated by a dash (-) after the last operand in the last line of the TOOLIN data set, but the continuation line was not found.
- A parameter list statement area did not contain an operator statement.
- Continuation was indicated by a dash (-) in the last line of a parameter list statement area, but the continuation line was not found.

**System Action:** This operation is terminated.

**Programmer Response:** Supply the continuation line.

---

### ICE623A MAXIMUM NUMBER OF *keyword* KEYWORDS EXCEEDED

**Explanation:** Critical. Too many keywords of the indicated type were specified for this operator. The maximum number of HEADER fields is 20 for a DISPLAY operator or 10 for an OCCUR operator. The maximum number of ON fields for each operator is:

- DISPLAY - 20
- OCCUR - 10
- RANGE - 1
- SELECT - 10
- STATS - 10
- UNIQUE - 1
- VERIFY - 10

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the point at which the error was detected. Reduce the number of indicated keywords for this operator to the maximum allowed. If necessary, use additional operator(s) to handle all the required fields.

---

### ICE624A MAXIMUM NUMBER OF TO DDNAMES EXCEEDED

**Explanation:** Critical. Too many TO ddnames were specified for this operator. The maximum number of TO ddnames for each operator is:

- COPY - 10
- SELECT - 1
- SORT - 10

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the point at which the error was detected. Reduce the number of TO ddnames for this operator to the maximum allowed. Use additional operator(s) to handle all the data sets required.

---

### ICE625A OPEN FAILED FOR *ddname* DATA SET

**Explanation:** Critical. The data set with the indicated *ddname* could not be opened. This occurred either for the data set specified for TOOLIN, SYMNames or SYMNOUT, or for the data set whose *ddname* was specified in the LIST operand for this operator.

**System Action:** If the *ddname* is TOOLIN, SYMNames, or SYMNOUT, ICETOOL terminates without performing any operations. Otherwise, this operation is terminated.

**Programmer Response:** Ensure that the attributes of the data set (record format, record length, block size) conform to the rules for that type of data set. See the ICETOOL chapter (TOOLIN or LIST operand data set) or Symbols chapter (SYMNames or SYMNOUT) in *DFSORT Application Programming Guide* for information on the rules.

---

**ICE626A      LIMIT FOR INVALID VALUES REACHED**

**Explanation:** Critical. The limit for the number of invalid decimal values specified by the LIMIT operand (or defaulted to 200) was reached for this DISPLAY or VERIFY operator.

**System Action:** This operation is terminated.

**Programmer Response:** Correct the invalid decimal values indicated in the ICE618A message(s), or set a higher LIMIT value.

---

**ICE627I      DFSORT CALL *nnnn* FOR  
                  {COPY}SORT} FROM *ddname1* TO  
                  {*ddname2*}E35 EXIT]OUTFIL}  
                  {COMPLETED]TERMINATED}**

**Explanation:** Supplies information about a specific call to DFSORT for this operation as follows:

**nnnn**    The call identifier for this call to DFSORT. You can match *nnnn* to the same *nnnn* in DFSORT message ICE200I in the DFSMSG data set to find the set of DFSORT messages for this call to DFSORT.

**COPY**    The DFSORT copy function was used.

**SORT**    The DFSORT sort function was used.

**ddname1**  
          The *ddname* of the data set that was copied or sorted.

**ddname2**  
          The *ddname* of the data set resulting from the copy or sort.

**E35 EXIT**  
          ICETOOL's E35 user exit was used.

**OUTFIL**  
          OUTFIL data sets were produced

**COMPLETED**  
          DFSORT did not detect any errors preventing completion of this ICETOOL operation.

**TERMINATED**  
          DFSORT detected an error which prevented completion of this ICETOOL operation.

**System Action:** None.

**Programmer Response:** You can use the information in this message to determine the result of this call to DFSORT. If DFSORT terminated, use the call identifier to find the set of DFSORT messages for this call (in the DFSMSG data set), and correct the error indicated in the DFSORT error message.

---

**ICE628I      RECORD COUNT: *nnnnnnnnnnnnnnnn***

**Explanation:** Indicates the number of records processed by ICETOOL (prints as 15 decimal digits padded with zeros on the left as needed).

If ICETOOL completed the operation successfully, this count reflects the number of records in the input data set or in the subset of the input data set selected by DFSORT statements (for example, INCLUDE).

If ICETOOL did not complete the operation successfully, this count reflects the number of records processed before an error was detected that caused ICETOOL to terminate processing of this operation.

**System Action:** None.

**Programmer Response:** None.

---

**ICE629A      HIGHER AND LOWER VALUES  
                  EXCLUDE ALL RECORDS**

**Explanation:** Critical. The values specified for HIGHER and LOWER excluded all records from the range. For example, HIGHER(5) and LOWER(6) define a range of 5 < value < 6, which excludes every value.

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the point at which the error was detected. Specify values for HIGHER and LOWER which define a valid range.

---

**ICE630I      MODE IN EFFECT: *mode***

**Explanation:** Indicates the processing mode to be used for subsequent operators as follows:

- STOP mode - set at the beginning of the run and when a MODE STOP statement is processed. Stops subsequent operations if an error is detected (by entering SCAN mode).
- CONTINUE mode - set when a MODE CONTINUE statement is processed. Continues with subsequent operations if an error is detected.
- SCAN mode - set when an error is encountered in STOP mode and when a MODE SCAN statement is processed. ICETOOL statements are checked for errors, but DFSORT is not called to perform the operations.

**System Action:** None.

**Programmer Response:** If SCAN mode was set due to an error while in STOP mode, correct the error.

---

**ICE631I      NUMBER OF VALUES IN RANGE FOR  
                  {(p,m,f)}(VLEN): *nnnnnnnnnnnnnnnn***

**Explanation:** Indicates the count of values within the specified range for this field (prints as 15 decimal digits padded with zeros on the left as needed). (p,m,f) or (VLEN) is the field you specified for this RANGE operator.

## Informational and Error Messages

**System Action:** None.

**Programmer Response:** None.

---

### ICE632I SOURCE FOR ICETOOL STATEMENTS: *source*

**Explanation:** Indicates whether ICETOOL statements were processed from TOOLIN or from the calling program's PARMLIST (parameter list). Statements are processed from TOOLIN unless the ICETOOL Parameter List Interface is used.

**System Action:** None.

**Programmer Response:** None.

---

### ICE633A RETURN AREA IS *n* BYTES, BUT *m* BYTES ARE REQUIRED

**Explanation:** The length of the return area (in bytes) as indicated in the calling program's parameter list was too small to contain all of the information to be returned for this operation. A return area of *m* bytes is needed.

**System Action:** This operation is terminated.

**Programmer Response:** Provide a return area of *m* bytes (or more) for this operation and set the return area length appropriately.

---

### ICE634A VALUE FOR (*p,m,f*) EXCEEDS 15 DIGITS

**Explanation:** Critical. A value for the indicated floating sign field had 16 digits, exceeding the limit of 15 digits.

**System Action:** This operation is terminated.

**Programmer Response:** Limit all floating sign (FS/CSF) data values for this operation to 15 digits (a sign followed by 15 digits is allowed). The DISPLAY operator with CH format can be used to visually identify the values with 16 digits.

---

### ICE635A NUMBER OF HEADER KEYWORDS DOES NOT MATCH NUMBER OF ON KEYWORDS

**Explanation:** Critical. There was not a one-to-one correspondence between the HEADER keywords and the ON keywords. For example, two HEADER keywords were specified with three ON keywords.

**System Action:** This operation is terminated.

**Programmer Response:** Specify one HEADER('string') or HEADER(NONE) keyword for each ON keyword.

---

### ICE636A INVALID VALUE IN DFSORT INSTALLATION MODULE ICEyyy

**Explanation:** Critical. A value in the indicated DFSORT installation module (ICEyyy) could not be interpreted by ICETOOL because the module was not at the same level as the ICETOOL modules. This indicates a serious problem with ICEyyy and can be indicative of problems with the other installation modules as well. Any or all of the values printed can be incorrect.

**Note:** ICETOOL cannot detect all possible invalid values in the ICEyyy modules; it only prints this message for the first invalid value it detects.

**System Action:** '> > INVALID < <' is printed in the list data set for the invalid value, and all subsequent processing for this operator is terminated.

**Programmer Response:** Have your system programmer check that the ICEMAC macro and the DFSORT modules were installed correctly and that the installation modules (ICEAM1, ICEAM2, ICEAM3, ICEAM4, ICETD1, ICETD2, ICETD3, and ICETD4) were generated correctly and are at the same level as the DFSORT modules.

---

### ICE637A *ddname* RECORD LENGTH OF *n* BYTES EXCEEDS MAXIMUM WIDTH OF *m* BYTES

**Explanation:** Critical. The calculated record length for the indicated list data set was greater than 2048 or the maximum width specified. *n* is the total bytes required in the list data set record for the carriage control character, the column widths (resulting from specified ON, HEADER, PLUS, BLANK, TOTAL, BREAK, BTITLE and BTOTAL operands), and the blanks between fields. *m* is the value specified for the WIDTH operand, or 2048 if WIDTH was not specified.

**System Action:** This operation is terminated.

**Programmer Response:** If *m* is less than 2048, either remove the WIDTH operand and let ICETOOL set the width, or if you need to set the WIDTH explicitly, increase its value to *n* or greater.

If *m* is 2048, take one or more of the following actions:

- Use formatting items or the PLUS or BLANK operand.
- Reduce the length of one or more HEADER strings.
- Reduce the length of one or more ON fields. For example, if an ON(1,8,PD) field always has zeros in bytes 1 through 3, use ON(4,5,PD) instead.
- Reduce the number of ON fields, especially if the BTOTAL or TOTAL operand is used.



---

**ICE638I      NUMBER OF RECORDS RESULTING  
FROM CRITERIA: nnnnnnnnnnnnnnn**

**Explanation:** Indicates the count of records produced as a result of the specified criteria, for example, ALLDUPS (prints as 15 decimal digits padded with zeros on the left as needed).

- For an OCCUR operator, the count indicates the total number of records in the list data set.
- For a SELECT operator, the count indicates the total number of records in the outdd data set.

**System Action:** None.

**Programmer Response:** None.

---

**ICE639A      INSUFFICIENT MAIN STORAGE - ADD  
AT LEAST nK BYTES TO REGION**

**Explanation:** Critical. ICETOOL could not get the additional nKB of storage needed for this operation.

**System Action:** This operation is terminated.

**Programmer Response:** Increase the REGION by at least nKB. In some cases, more than nKB are required to meet all the storage requirements for ICETOOL and DFSORT for this operation.

---

**ICE640A      INVALID FORMATTING ITEM**

**Explanation:** Critical. An ON(p,m,f,formatting) operand for this DISPLAY operator contained an invalid formatting item as follows:

- The formatting item was not /K, /M, /G, /KB, /MB, /GB, L'string', F'string', T'string' or a valid mask (that is, A0, A1-A5, B1-B6, C1-C6, D1-D6, E1-E4, or F1-F5).
- /K, /M, /G, /KB, /MB, /GB, F'string' or a mask was specified for a character field.
- More than one of /K, /M, /G, /KB, /MB and /GB was specified, more than one mask was specified, T'string' was specified more than once, L'string' was specified more than once or F'string' was specified more than once.
- L", F" or T" was specified.

**System Action:** This operation is terminated.

**Programmer Response:** A \$ marks the point at which the error was detected. Correct the error.

---

**ICE641A      service SERVICE ERROR (x-y)**

**Explanation:** Critical. The indicated system service failed when ICETOOL called it. Message values are as follows:

- x**      The return code from the service, in decimal.
- y**      The reason code from the service, in decimal, or zero if unavailable.

**System Action:** This operation is terminated.

**Programmer Response:** Contact IBM for programming support.

---

**ICE642A      DFSORT RELEASE AND ICEyyy  
RELEASE DO NOT MATCH**

**Explanation:** Critical. The release level of the indicated DFSORT installation module (ICEyyy) does not match the release level of DFSORT (as shown in the DEFAULTS heading line). This indicates that the ICEMAC macro used to compile ICEyyy was not at the same release level as the DFSORT modules being used for the run.

**Note:** This message is only printed for the first ICEyyy with a release level that does not match the DFSORT release level.

**System Action:** '>> release <<' is printed in the list data set for the invalid release level, and all subsequent processing for this operator is terminated.

**Programmer Response:** If one or more installation modules (ICEAM1, ICEAM2, ICEAM3, ICEAM4, ICETD1, ICETD2, ICETD3 and ICETD4) were compiled with the wrong level of the ICEMAC macro, recompile them with the correct level of the ICEMAC macro. If the wrong level of DFSORT modules were used for the run, rerun with the correct level of DFSORT modules.

---

**ICE700A      EFS PROCESSING IS NOT  
SUPPORTED FOR THE TECHNIQUE  
SELECTED**

**Explanation:** Critical. EFS processing is not supported for the technique selected (tape work data set sort or Conventional merge).

**System Action:** The program terminates.

**Programmer Response:** Rerun the job with SORTDIAG DD statement to get the reason for the revert from message ICE800I. Remove the cause of the revert or do not use EFS processing by specifying EFS=NONE.

---

**ICE701A      EFS PROCESSING WAS TERMINATED  
BY pppppppp AT MAJOR CALL n**

**Explanation:** Critical. Termination of the DFSORT run was requested by EFS program pppppppp at Major Call 1, 2, 3, 4, or 5.

**System Action:** If the termination was requested at Major Call 1, the program terminates. If the termination was requested at Major Call 2 through 5, DFSORT scans all remaining control statements and then the program terminates.

**Programmer Response:** Refer to the message issued by EFS program pppppppp for the error description.

## Informational and Error Messages

---

**ICE702A**    **INVALID RETURN CODE FROM EFS PROGRAM** *pppppppp* **AT MAJOR CALL**  
*n*

**Explanation:** Critical. EFS program *pppppppp* returned an invalid return code to DFSORT at Major Call 1, 2, 3, 4, or 5.

**System Action:** If the invalid return code was returned at Major Call 1, the program terminates. If the invalid return code was returned at Major Call 2 through 5, DFSORT scans all remaining control statements and then the program terminates.

**Programmer Response:** Check the EFS program thoroughly and ensure that the return code is either 0 or 16.

---

**ICE703A**    **INVALID RETURN CODE FROM** *ppppp*  
**OF EFS PROGRAM** *pppppppp*

**Explanation:** Critical. The *ppppp*-generated user exit routine, EFS01 or EFS02, of EFS program *pppppppp* returned an invalid return code to DFSORT.

**System Action:** The program terminates.

**Programmer Response:** Check the EFS-generated user exit routines thoroughly and ensure that the return code returned by EFS01 is either 0 or 16 and that the return code returned by EFS02 is 0, 4, or 16.

---

**ICE704A**    **LOAD FAILED FOR EFS PROGRAM**  
*pppppppp*

**Explanation:** Critical. EFS program *pppppppp* could not be loaded.

**System Action:** The program terminates.

**Programmer Response:** Check that EFS program *pppppppp* is installed.

---

**ICE705A**    **EFS PROCESSING WAS TERMINATED**  
**BY** *ppppp* **OF EFS PROGRAM** *pppppppp*

**Explanation:** Critical. Termination of the DFSORT run was requested by a user exit routine generated by EFS program *pppppppp*.

*ppppp* is one of the following names:

**EFS01** The SORT/MERGE user exit routine.

**EFS02** The INCLUDE/OMIT user exit routine.

**System Action:** The program terminates.

**Programmer Response:** Refer to the message issued by the user exit routine generated by EFS program *pppppppp*.

---

**ICE706I**    **END OF SUPPLIED CONTROL**  
**STATEMENTS – EFS RETURNED**  
**CONTROL STATEMENTS FOLLOW**

**Explanation:** When control statements are specified in the invoking parameter list or control statements are specified with the DD statement for SYSIN or xxxxCNTL, this message separates the listing of those control statements from the listing of the control statements returned by an EFS program.

A control statement returned by an EFS program completely overrides the corresponding control statement in the parameter list, or in SYSIN or xxxxCNTL. (OPTION control statement operands EFS, LIST, NOLIST, LISTX, NOLISTX, MSGDDN, MSGPRT, SORTDD, SORTIN, SORTOUT, and USEWKDD are ignored if returned by an EFS program.)

For complete details on the order of override, see *DFSORT Application Programming Guide*.

**System Action:** None.

**Programmer Response:** None.

---

**ICE707A**    **EXIT E61 IS NOT SUPPORTED FOR**  
**EFS FIELDS**

**Explanation:** Critical. The SORT or MERGE control statement defined a control field to be processed both by an EFS program and an E61 user exit routine.

**System Action:** The program terminates.

**Programmer Response:** Check the SORT or MERGE control statement for errors resulting in the specification of a D1 field format type with an E field sequence type.

---

**ICE708A**    **AN EFS PROGRAM IS REQUIRED**

**Explanation:** Critical. A SORT/MERGE control field or an INCLUDE/OMIT compare field was to be processed by an EFS program, but the EFS program was not activated by the EFS option, nor was a verb request list returned at Major Call 1.

**System Action:** The program terminates.

**Programmer Response:** Check the SORT/MERGE or INCLUDE/OMIT control statements for errors resulting in the specification of a D1 or D2 field format type. Check that the EFS option has been specified.

---

**ICE709A**    *ppppp* **EXIT ROUTINE IS REQUIRED**

**Explanation:** Critical. A SORT/MERGE control field or an INCLUDE/OMIT compare field was to be processed by an EFS program, but the EFS program did not return an address to EFS01 or EFS02.

*ppppp* is one of the following names:

**EFS01** The SORT/MERGE user exit routine.

**EFS02** The INCLUDE/OMIT user exit routine.

**System Action:** The program terminates.

**Programmer Response:** Check the SORT/MERGE and INCLUDE/OMIT control statements for errors resulting in the specification of a D1 or D2 field format type. Check the EFS program and ensure that it returns an address to EFS01 or EFS02.

**ICE710A THE aaaaaaaa CONTROL STATEMENT WAS NOT SENT TO EFS PROGRAM  
pppppppp**

**Explanation:** Critical. The control statement aaaaaaaa requested by EFS program pppppppp contained a syntax error. DFSORT could not determine the true length of the requested control statement or option.

**System Action:** The program terminates.

**Programmer Response:** Respond as indicated in the accompanying message ICE007A.

**ICE711I END OF EFS RETURNED CONTROL STATEMENTS FROM xxxxCNTL – PARAMETER LIST CONTROL STATEMENTS FOLLOW**

**Explanation:** When returned control statements are specified in the invoking parameter list and with the DD statement for xxxxCNTL, this message separates the listing of requested control statements from both sources.

If all returned control statements came from either SORTCNTL or the parameter list, message ICE711I was not printed; only ICE706I was printed.

For complete details on the order of override, see *DFSORT Application Programming Guide*.

**System Action:** None.

**Programmer Response:** None.

**ICE712I END OF EFS RETURNED PARAMETERS FROM ddname – SYSIN OR xxxx CNTL/PARAMETER LIST CONTROL STATEMENTS FOLLOW**

**Explanation:** When returned parameters are specified with:

- The SYSIN DD statement
- The SORTCNTL DD statement, or
- The invoking parameter list

and are specified with the DD statement indicated by ddname, this message separates the listing of the requested data set's parameters from the former sources.

See *DFSORT Application Programming Guide* for details on the order of override.

**System Action:** None.

**Programmer Response:** None.

**ICE713A LANGUAGE ENVIRONMENT INITIALIZATION FAILED**

**Explanation:** Critical. LE/370 initialization failed.

**System Action:** The program terminates.

**Programmer Response:** Contact IBM for programming support.

**ICE714A LOAD FAILED FOR ACTIVE LOCALE  
name**

**Explanation:** Critical. Indicates that the designated active locale could not be loaded and gives its name.

**System Action:** The program terminates.

**Programmer Response:** If the indicated locale name is that of an IBM supplied locale, DFSORT must have access to the run-time library containing the dynamically loadable routines. For example, the data set containing them might be called SYS1.SCEERUN. If you are unsure of the name of the data set where the compiled locale load modules are installed at your location, contact your system administrator.

If the indicated name is that of a user defined locale, DFSORT must have access to the load library containing it.

**ICE715A LOCALE PROCESSING CONFLICTS WITH keyword**

**Explanation:** Critical. Locale processing cannot be used with the function or feature indicated by the keyword shown. See *Application Programming Guide* for details on locale processing and the function or feature indicated.

**System Action:** The program terminates.

**Programmer Response:** If D1, D2, or EFS is the indicated keyword, take one of the following actions:

- Specify run-time option EFS=NONE to use locale processing without EFS processing. (Locale processing might eliminate the need for an EFS program.)
- Specify run-time option LOCALE=NONE to use EFS processing without locale processing.

If CHALT is the indicated keyword, take one of the following actions:

- Specify run-time option NOCHALT to use locale processing without alternate sequence processing for CH fields. (Locale processing might eliminate the need for alternate sequence processing.) If alternate sequence processing is needed for a particular field, specify AQ for that field instead of CH.
- Specify run-time option LOCALE=NONE to use alternate sequence processing for CH fields without locale processing.

## Informational and Error Messages

If INREC is the indicated keyword, take one of the following actions:

- Use the OUTREC statement or the OUTREC operand of the OUTFIL statement instead of the INREC statement to allow locale processing.
- Specify run-time option LOCALE=NONE to use the INREC statement without locale processing.

If E61 is the indicated keyword, take one of the following actions:

- Eliminate the E61 exit, and change E to A or D for each field, to use locale processing without E61 processing. (Locale processing might eliminate the need for an EFS program.)
- Specify run-time option LOCALE=NONE to use E61 processing without locale processing.

---

### ICE716A A CONTROL FIELD CONTAINS AN INVALID CHARACTER

**Explanation:** Critical. Locale processing of a control field detected that a character in the control field is not defined in the active locale.

**System Action:** The program terminates.

**Programmer Response:** Ensure that the correct locale is the active locale in effect or correct the invalid character.

---

### ICE717A CONTROL FIELDS FOR LOCALE PROCESSING ARE TOO COMPLEX

**Explanation:** Critical. For a SORT or MERGE statement with control fields that require locale processing, the complexity of the application caused dynamic areas to exceed the storage allowed for them.

**System Action:** The program terminates.

**Programmer Response:** Reduce the number or length of SORT or MERGE control fields that require locale processing.

---

### ICE721A A COMPARE FIELD CONTAINS AN INVALID CHARACTER

**Explanation:** Critical. Locale processing of a compare field detected that a character in the compare field is not defined in the active locale.

**System Action:** The program terminates.

**Programmer Response:** Ensure that the correct locale is the active locale in effect or correct the invalid character.

---

### ICE722I ACTIVE LOCALE *name* WAS USED FOR LOCALE PROCESSING

**Explanation:** Indicates that DFSORT used locale processing and gives the name of the active locale. Locale processing is used for:

- SORT or MERGE character (CH) control fields

- INCLUDE or OMIT character (CH) compare fields
- INCLUDE or OMIT character or hexadecimal constant to character (CH) compare field comparisons.

**System Action:** None

**Programmer Response:** None

---

### ICE725A LANGUAGE ENVIRONMENT SERVICE ERROR (*nnnn*)

**Explanation:** Critical. Language Environment services returned an unexpected feedback code for a service call used by DFSORT. *nnnn* is the error message number of the feedback code.

**System Action:** The program terminates.

**Programmer Response:** Take one or more of the following actions, as appropriate:

- Check that all CH fields specified in SORT, MERGE, INCLUDE, or OMIT statements contain valid character data values.
- Contact IBM for programming support. The error message number can be used to reference LE/370 Run-Time Messages in *Locale Processing*.

---

### ICE726A LOAD FAILED FOR LOCALE PROCESSING

**Explanation:** Critical. DFSORT's locale processing routine could not be loaded.

**System Action:** The program terminates.

**Programmer Response:** Contact your system programmer to ensure that DFSORT's locale processing support has been installed.

---

### ICE727A EFS PROCESSING CANNOT BE USED WITH SYMBOL PROCESSING

**Explanation:** Critical. A SYMNMES data set was specified indicating that symbol processing is to be performed. However, an EFS program was also specified. EFS processing and symbol processing cannot both be used.

**System Action:** The program terminates.

**Programmer Response:** Use EFS processing or symbol processing, but not both.

---

## Chapter 4. Diagnostic Messages

The following messages are DFSORT diagnostic messages for use by IBM field support personnel.

**ICE036I**

**ICE037I**

**ICE049I**

**ICE050I**

**ICE820I**

**ICE821I**

**ICE822I**

**ICE823I**

**ICE824I**

**ICE825I**

**ICE826I**

**ICE827I**

**ICE855I**

**ICE857I**

**ICE858I**

**ICE859I**

**ICE860I**

**ICE880I**

**ICE881I**

**ICE882I**

**ICE883I**

**ICE884I**

**ICE885I**

**ICE886I**

**ICE887I**

**ICE888I**

**ICE889I**

**ICE890I**

**ICE891I**

**ICE892I**

**ICE893I**

**ICE894I**

**ICE895I**

**ICE896I**

**ICE898I**

**ICE899I**

**ICE900I**

**ICE901I**

**ICE902I**

**ICE910I**

**ICE996I**

**ICE997I**

**ICE998I**

**ICE999I**

---

## Chapter 5. Informational Messages for Debugging

The following diagnostic messages can appear when the SORTDIAG DD statement is present.

---

### ICE800I      **BLOCKSET TECHNIQUE COULD NOT BE USED – REASON CODE IS *nn***

**Explanation:** DFSORT's most efficient technique, Blockset, could not be used for this application. *nn* is the revert reason code. See the following list to determine the reason associated with this code.

- 1      A critical message was issued.
- 5      BDAM was indicated in the SORTIN DD statement.
- 6      BDAM was indicated in the SORTIN data set label.
- 7      BDAM was indicated in the SORTOUT DD statement.
- 10     A DUMMY or spool SORTWK*dd* data set was specified.
- 12     An attempt to OPEN a data set caused a system error.
- 16     The SORTIN or SORTOUT data set resided on an unsupported device.
- 17     A SORTWK*dd* data set resided on an unsupported device (such as tape).
- 22     ASCII tapes had the following parameters:  
(LABEL=AL or OPTCD=Q) and RECFM=D and BUFOFF[=L  
or  
(LABEL=AL or OPTCD=Q) and RECFM[=D and BUFOFF[=0.
- 23     An attempt to read the DSCB for the SORTIN data set caused an error.
- 24     An attempt to read the DSCB for the SORTOUT data set caused an error.
- 28     An E18 or E38 user exit was specified with a tape SORTIN(*nn*) data set.
- 47     The NOBLKSET option was specified.
- 48     The CKPT run-time option and the IGNCPT=NO installation option were specified.
- 58     Dynamic tape SORTWK*dd* data sets were requested.
- 63     Too many control fields were specified.
- 64     Too many control fields were specified.
- 65     A SORTWK*dd* data set was not present.

- 66     An INREC statement was specified and EFS01 was in effect.
- 74     An E61 user exit was specified and VLSHRT was in effect.
- 75     A MODS statement Exx operand had SYSIN in the third parameter or T or S in the fourth parameter (that is, dynamic link-editing was requested.).

**Note:** Blockset cannot be used to process BDAM data sets.

**System Action:** None.

**Programmer Response:** None. However, for improved performance, correct the situation indicated by the reason code so Blockset can be used.

---

### ICE802I      **t TECHNIQUE IN CONTROL**

**Explanation:** *t* indicates the DFSORT technique that was in control:

#### **BLOCKSET**

DASD work data set sort, standard merge, or copy

#### **PEER/VALE**

DASD work data set sort

#### **CONVENTIONAL**

Nonstandard merge or tape work data set sort.

The indicated technique was not necessarily the technique ultimately selected (see message ICE143I).

**System Action:** None.

**Programmer Response:** None.

---

### ICE804I      ***ddname* EXCP COUNT: *n***

**Explanation:** If the *ddname* is OUTFIL, lists the total number of EXCPs performed, using the EXCP access method, to all OUTFIL data sets. Otherwise, lists the number of EXCPs performed, using the EXCP access method, to the data set associated with the indicated *ddname*.

**System Action:** None.

**Programmer Response:** None.



## Informational Messages for Debugging

---

**ICE805I**      **JOBNAME:** *jobname* , **STEPNAME:**  
*stepname*

**Explanation:** Indicates the job name and step name for the DFSORT run.

**System Action:** None.

**Programmer Response:** None.

---

**ICE816I**      **DFSORT SVC CALL RESULTED IN**  
**Snnn ABEND**

**Explanation:** A call to the DFSORT SVC resulted in a system abend. nnn is the abend code. It occurred when the SVC was not available to DFSORT because of improper installation or because the wrong version of the SVC was installed. Because DFSORT continued the run as if the abend had not occurred, this message was issued to assist system programmers in determining the reason why the SVC was not available. DFSORT runs more efficiently with cache devices if the SVC is available. See *DFSORT Installation and Customization* for further information.

**System Action:** None.

**Programmer Response:** Ask your system programmer to check that the DFSORT SVC is installed properly for this level of DFSORT and that the correct routing code for the SVC is selected.

---

**ICE818I**      **LOCATE CALL FAILED**

**Explanation:** A call to the LOCATE SVC failed while attempting to obtain information about an input or output data set.

**System Action:** Processing continues without the information from LOCATE. This can result in one of the following situations:

- DFSORT may not be able to accurately estimate the input file size.
- DFSORT may not be able to obtain the input record length.
- An OPEN error may occur if a VSAM output data set is defined without REUSE, and RESET is in effect.

**Programmer Response:** None required. If DFSORT is unable to accurately estimate the input file size, specifying a FILSZ value may improve DFSORT's performance, especially when dynamically allocated work data sets are used. If DFSORT is unable to obtain the input record length, specify an L1 value. If a VSAM OPEN error occurs, specifying NORESET may eliminate the OPEN error.

---

## Chapter 6. Return Codes and User Abends

---

### DFSORT Return Codes

For successful completion, DFSORT passes back a return code of 0 or 4 to the operating system or the invoking program.

For unsuccessful completion due to an unsupported operating system, DFSORT passes back a return code of 24 to the operating system or the invoking program.

For unsuccessful completion with NOABEND in effect, DFSORT passes back a return code of 16 or 20 to the operating system or the invoking program.

For unsuccessful completion with ABEND in effect, DFSORT issues a user abend with the appropriate code as specified by ICEMAC option ABCODE (either the error message number or a number between 1 and 99).

The meanings of the return codes that DFSORT passes back (in register 15) are:

- 0**      **Successful completion.** DFSORT completed successfully.
- 4**      **Successful completion.** DFSORT completed successfully, and:
  - OVFLO=RC4 was in effect and summary fields overflowed, or
  - PAD=RC4 was in effect and the SORTOUT LRECL was larger than the SORTIN/SORTINnn LRECL (LRECL padding), or
  - TRUNC=RC4 was in effect and the SORTOUT LRECL was smaller than the SORTIN/SORTINnn LRECL (LRECL truncation), or
  - SPANINC=RC4 was in effect and one or more incomplete spanned records was detected.
- 16**     **Unsuccessful completion.** DFSORT detected an error that prevented it from completing successfully.
- 20**     **Message data set missing.** ICEMAC option NOMSGDD=QUIT was in effect and neither a message data set DD statement nor a SYSOUT DD statement was provided.
- 24**     **Unsupported operating system.** This operating system is not supported by this release of DFSORT. Only current or subsequent releases of the following systems are supported:
  - OS/390
  - MVS/ESA

---

### ICETOOL Return Codes

ICETOOL sets a return code for each operation it performs in STOP or CONTINUE mode and passes back the highest return code it encounters to the operating system or the invoking program.

For successful completion of all operations, ICETOOL passes back a return code of 0 or 4 to the operating system or the invoking program.

For unsuccessful completion due to an unsupported operating system, ICETOOL passes back a return code of 24 to the operating system or invoking program.

## Return Codes

For unsuccessful completion of one or more operations, ICETOOL passes back a return code of 12, 16, or 20 to the operating system or the invoking program.

The meanings of the return codes that ICETOOL passes back (in register 15) are:

- 0**      **Successful completion.** All operations completed successfully.
- 4**      **Successful completion.** All operations completed successfully. DFSORT passed back a return code of 4 for one or more operations. See "DFSORT Return Codes" on page "DFSORT Return Codes" on page 73 for details.
- 12**     **Unsuccessful completion.** ICETOOL detected one or more errors that prevented it from completing successfully. Messages for these errors were printed in the TOOLMSG data set.
- 16**     **Unsuccessful completion.** DFSORT detected one or more errors that prevented ICETOOL from completing successfully. Messages for these errors were printed in the DFSMSG data set.
- 20**     **Message data set error.** The TOOLMSG DD statement was not present or the TOOLMSG data set was not opened.
- 24**     **Unsupported operating system.** This operating system is not supported by this release of DFSORT. Only current or subsequent releases of the following systems are supported:
  - OS/390
  - MVS/ESA

---

## ICEGENER Return Codes

ICEGENER can use either IEBGENER or the DFSORT copy function. However, for unsuccessful completion due to an unsupported operating system, ICEGENER passes back a return code of 24 to the operating system or the invoking program, without using either IEBGENER or DFSORT copy.

If ICEGENER transfers control to IEBGENER, IEBGENER passes back its return code to the operating system or the invoking program.

If ICEGENER uses the DFSORT copy function:

- For successful completion, ICEGENER passes back a return code of 0 or 4 to the operating system or the invoking program.
- For unsuccessful completion with NOABEND in effect, ICEGENER passes back a return code of 12 (changed from 16) to the operating system or the invoking program.
- For unsuccessful completion with ABEND in effect, DFSORT issues a user abend with the appropriate code as specified by the ICEMAC option ABCODE (either the error message number or a number between 1 and 99).

The meanings of the return codes that ICEGENER passes back (in register 15) are:

- 0**      **Successful completion.** ICEGENER completed successfully.
- 4**      **Successful completion.** ICEGENER completed successfully, and:
  - ICEMAC option GNPAD=RC4 was specified and the SYSUT2 LRECL was larger than the SYSUT1 LRECL (LRECL padding), or
  - ICEMAC option GNTRUNC=RC4 was specified and the SYSUT2 LRECL was smaller than the SYSUT1 LRECL (LRECL truncation), or

- SPANINC=RC4 was in effect and one or more incomplete spanned records was detected.
- 12 Unsuccessful completion.** DFSORT detected an error that prevented ICEGENER from completing successfully.
- 24 Unsupported operating system.** This operating system is not supported by this release of DFSORT. Only current or subsequent releases of the following systems are supported:
- OS/390
  - MVS/ESA

## Return Codes

---

## Part 3. Diagnosis Guide

<b>Chapter 7. Resolving Failures in the DFSORT Program</b> . . . . .	79
Developing Your Keyword String . . . . .	80
Constructing Keyword Strings . . . . .	82
Component Identification Keyword Procedure . . . . .	83
FMID Keyword Procedure . . . . .	84
Type-of-Technique Keyword Procedure . . . . .	84
Type-of-Application Keyword Procedure . . . . .	85
Type-of-Failure Keyword Procedures . . . . .	85
Wait and Loop Keywords Procedure . . . . .	86
System Abend Keyword Procedure . . . . .	87
User Abend Keyword Procedure . . . . .	87
Message Keyword Procedure . . . . .	88
Incorrect-Output Keyword Procedure . . . . .	88
Performance Keyword Procedure . . . . .	89
Document Keyword Procedure . . . . .	89
Module Keyword Procedure . . . . .	89
Offset Keyword Procedure . . . . .	91
Message Variable Keywords . . . . .	92
ICE061A Variable Keywords . . . . .	92
ICE117A Variable Keywords . . . . .	93
<b>Chapter 8. Resolving Problems When Using DFSORT Panels</b> . . . . .	95
Obtaining Diagnostic Information . . . . .	95
Determining Where the Failure Occurred . . . . .	95
Resolving Failures in DFSORT Panels . . . . .	96
DFSORT Panels Abend Panel . . . . .	96
TSO Abend Message . . . . .	97
Resolving Failures in ISPF or ISMF . . . . .	98
Resolving Failures in the DFSORT Program . . . . .	99
Foreground Execution Failures . . . . .	100
Background Execution Failures . . . . .	100
Recording Information . . . . .	100
Inter-Module Traces . . . . .	101
ISMF Logging with SORTREC . . . . .	101
Using the Profile Application . . . . .	102
Directing Abend Processing . . . . .	102
Recovering from Abends . . . . .	102
Passing Control to ISPF or TSO . . . . .	102
Obtaining Diagnostic Messages . . . . .	103
<b>Chapter 9. Searching the IBM Software Support Database</b> . . . . .	105
<b>Chapter 10. Fixing or Bypassing the Problem</b> . . . . .	107
Fixing the Problem . . . . .	107
Bypassing the Problem . . . . .	107
<b>Chapter 11. Reporting a Problem</b> . . . . .	109





---

## Chapter 7. Resolving Failures in the DFSORT Program

To resolve a program failure, you describe the problem with a set of keywords, called a *keyword string*. The IBM Support Center (or you, if you have a tool such as Info Access) uses the keyword string to search an IBM software support database, such as the Software Support Facility or the Early Warning System, to determine whether an authorized program analysis report (APAR) has already been recorded.

An APAR contains information explaining how to fix or bypass the error. This might be a program temporary fix (PTF), an APAR fix, or a bypass.

If an APAR has not been recorded, the IBM Support Center personnel might ask you to obtain messages or a dump to provide information beyond the keyword string you developed.

Use Table 5 to resolve a program failure.

Table 5. Program Failure Resolution

Step	Action
1	<p>Read and follow the directions in “Chapter 1. Eliminating Common Sources of Error” on page 3, to first eliminate your own user exit routines, calling programs (if you have any), and other factors as a source of error.</p> <p>If the source of error is not your own routines or calling programs, go to step 2.</p>
2	<p>Follow the instructions in “Developing Your Keyword String” on page 80. Each of the keywords in the string describes one aspect of a program failure and makes the search argument more specific. The more precise your keyword string, the more selective the search, and the fewer problem descriptions there are to evaluate.</p> <p>You need messages or a dump to develop most keyword strings. If you did not receive messages or a dump, read “How to Get Messages” on page 3, and Table 1 on page 3.</p> <p>When you complete your keyword string, go to step 3.</p>
3	<p>Follow the instructions in “Chapter 9. Searching the IBM Software Support Database” on page 105. The search determines whether the problem has been reported previously.</p> <p>If the problem has already been reported, go to step 4.</p> <p>If the problem has not been reported, go to step 5.</p>
4	<p>If the problem has already been reported, you will be instructed to apply a PTF or APAR fix or to bypass the problem until a PTF or APAR fix is available. See “Chapter 10. Fixing or Bypassing the Problem” on page 107.</p>
5	<p>If the problem has not been reported, follow the instructions in “Chapter 11. Reporting a Problem” on page 109, to report the problem.</p>

### Developing Your Keyword String

This section helps you to develop a keyword string to describe your program failure. The keyword string is used to search an IBM software support database, explained in “Chapter 9. Searching the IBM Software Support Database” on page 105. The number of keywords within a keyword string depends on the type of failure being described. The types of failures include:

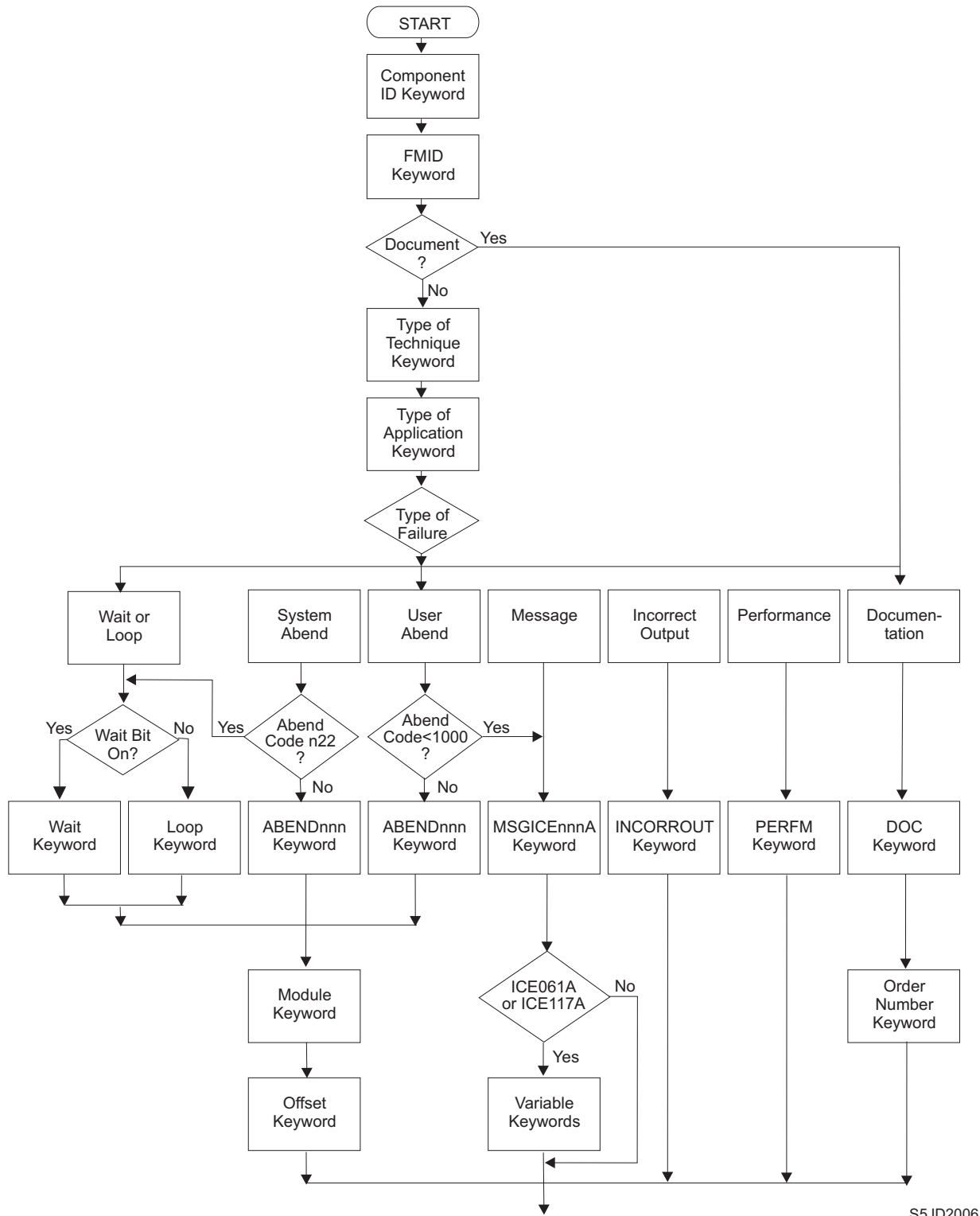
- System abend
- User abend
- Wait
- Loop
- Message
- Incorrect output
- Performance
- Document

Before contacting the IBM Support Center for assistance, ensure that your keyword string contains all the required keywords for your particular type of failure.

Figure 1 on page 81 shows the steps you need to follow, depending on the type of failure you describe. The following are the keywords you might need to use:

<b>Keyword</b>	<b>When required</b>
<b>Component Identification</b>	Always required
<b>FMID</b>	Always required
<b>Type-of-Technique</b>	Required for all except document errors
<b>Type-of-Application</b>	Required for all except document errors
<b>Type-of-Failure</b>	Always required
<b>Module</b>	Required for system abend, user abend, wait, and loop
<b>Offset</b>	Required for system abend, user abend, wait, and loop
<b>Message Variable</b>	Required for messages ICE061A and ICE117A
<b>Order Number</b>	Required for document error

Begin with “Constructing Keyword Strings” on page 82 and follow the procedures until the keyword string is developed and you are instructed to use it as a search argument.



S5JD2006

Figure 1. Keyword String Flowchart (Part 1 of 2)

## Constructing the Keyword String Correctly

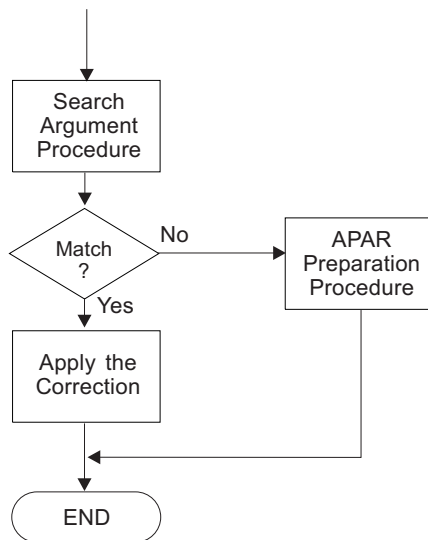


Figure 1. Keyword String Flowchart (Part 2 of 2)

---

## Constructing Keyword Strings

Use Figure 2 on page 83 to help you construct the keyword string to describe DFSORT program failures.

To use the diagram:

1. Choose the type of keyword.
2. Start from the left.
3. Continue to work from left to right.
4. When you encounter parallel elements in one line, make a choice.
5. Type uppercase letters and special characters exactly as given.
6. Replace lowercase letters with the appropriate data.
7. Your statement is complete when you reach zero, on the right side.

## Component Identification Keyword Procedure

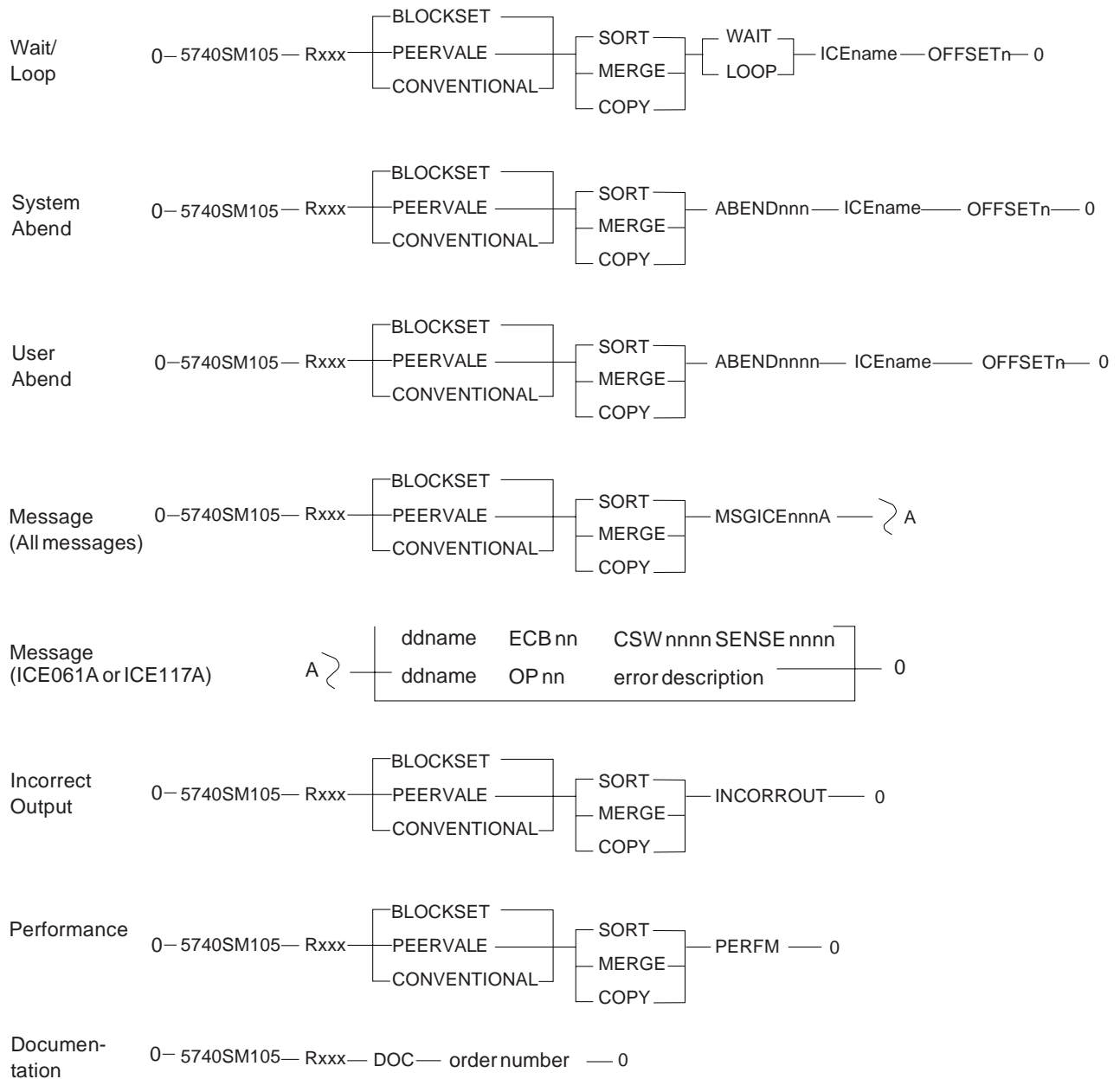


Figure 2. Keyword String Syntax. If you receive message ICE061A or ICE117A, fill in the second part of the message string. Otherwise, stop the keyword string at the message number.

## Component Identification Keyword Procedure

The DFSORT component identifier is the first keyword in the search argument. This number identifies the product within the IBM software support database.

1. The component identification keyword is nine characters. For DFSORT, this keyword is **5740SM105**.
2. The keyword string so far is:

**5740SM105**

3. Go to "FMID Keyword Procedure" on page 84, to determine the current release level keyword for DFSORT.

### FMID Keyword Procedure

The function modification identifier (FMID) keyword used with the component identification keyword narrows the search. The FMID shows the release of DFSORT.

1. Get the FMID from the control data set (CDS) list. To locate the FMID, find the module called ICEMAN in the list of module names. The FMID is on the same line as the ICEMAN module.

The FMID format is:

HSM1xxx

For example, the FMID for DFSORT Release 14.0 is HSM1G00.

2. Record the FMID keyword in the following format:

Rxxx

where xxx is the last three characters of the FMID.

For Release 14.0 problems, you now have the following keyword string:

5740SM105 RG00
----------------

G00 identifies the DFSORT base FMID HSM1G00.

To identify DFSORT feature-related FMID information, see *DFSORT Installation and Customization*.

3. If the problem seems to result from incorrect or missing information in a DFSORT publication, go to “Document Keyword Procedure” on page 89. If the problem is not in a DFSORT publication, go to “Type-of-Technique Keyword Procedure”.

---

### Type-of-Technique Keyword Procedure

The next keyword in the keyword string specifies the sort, merge, or copy technique being used when the failure occurred.

You need a message for this procedure. If you did not receive any messages, see “How to Get Messages” on page 3.

1. Find out which technique was used by looking at the diagnostic trace character in the *last* message you received from DFSORT.

There is a diagnostic trace character in all DFSORT messages. The format of a DFSORT message is:

ICEnns c text

#### Where Represents

**nnn** The error message number

**s** The severity code (which can be either A for error messages or I for information messages)

**c** The diagnostic trace character

**text** An explanation of the message



## Type-of-Technique Keyword Procedure

Each technique is associated with a set of diagnostic trace characters:

Trace character	Keyword
0 through 9 or A through I	BLOCKSET
J through Z	PEERVALE
— (dash)	CONVENTIONAL

For example, if you received message ICE085A 1, the 1 means the Blockset technique. The keyword string so far would be:

```
5740SM105 RG00 BLOCKSET
```

2. Go to “Type-of-Application Keyword Procedure”.

## Type-of-Application Keyword Procedure

The next keyword shows whether you were doing a sort, merge, or copy when the error occurred.

1. Record what you were doing when the error occurred:
  - If you were using DFSORT to *sort* data, add **SORT** to the keyword string.
  - If you were using DFSORT to *merge* data, add **MERGE** to the keyword string.
  - If you were using DFSORT to *copy* data, add **COPY** to the keyword string.

For example, if you were sorting data, the keyword string so far would be:

```
5740SM105 RG00 BLOCKSET SORT
```

2. Go to “Type-of-Failure Keyword Procedures”.

## Type-of-Failure Keyword Procedures

It is very important that you determine the type-of-failure keyword correctly. Use Table 6 to find the name and page

Table 6. Types of DFSORT Failures

Type of Failure	Description	Messages and Codes	Procedure Name and Page Numbers
Wait	Unexpected program suspension.	System completion code is n22 (where n is any hexadecimal digit).	“Wait and Loop Keywords Procedure” on page 86.
Loop	Uncontrolled program repetition.	System completion code is n22 (where n is any hexadecimal digit).	“Wait and Loop Keywords Procedure” on page 86.
System Abend	Abnormal termination of DFSORT.	System completion code is <b>not</b> n22 (where n is any hexadecimal digit), or zero.	“System Abend Keyword Procedure” on page 87.
User Abend	Abnormal termination of DFSORT.	User completion code is not zero.	“User Abend Keyword Procedure” on page 87.
Message	Errors identified by or associated with DFSORT error messages.	Error message is in the form ICEnnnA.	“Message Keyword Procedure” on page 88.

## Type-of-Failure Keyword Procedures

Table 6. Types of DFSORT Failures (continued)

Type of Failure	Description	Messages and Codes	Procedure Name and Page Numbers
Incorrect Output	Unexpected, missing, or incorrect output not associated with a message. (Use this keyword only when no other keywords seem appropriate.)	Not applicable	"Incorrect-Output Keyword Procedure" on page 88.
Performance	Performance degradation. (Use this keyword only when you are sure that this is not a wait or loop problem.)	Not applicable	"Performance Keyword Procedure" on page 89.
Documentation	Severe incorrect or missing information in DFSORT documentation.	Not applicable	"Document Keyword Procedure" on page 89.

## Wait and Loop Keywords Procedure

Because the wait and loop symptoms are often indistinguishable, the WAIT and LOOP keywords are included in the same type-of-failure procedure.

If your job enters a wait or loop, the system terminates the job and gives you a system abend code n22, where n is any hexadecimal digit. Common examples are 122, 222, and 322.

You need a dump for this procedure. If you did not receive a dump, see Table 1 on page 3.

1. Find the program status word (PSW) in the dump. See Figure 3.

system abend code		PSW	
JOB SORTA	STEP WAIT	TIME 162253	DATE 91039
COMPLETION CODE	SYSTEM = 322	ID = 000	CPUID = 230202333081
PSW AT ENTRY TO ABEND	07802000 001269D2	ILC 2	INTC 000A
ASCB 00F70730			
+0 ASCB C1E2C3C2	FWDP 00F67C88	BWDP 00F781E0	CMSF 00000000
+18 IDSF 00000000	TNEW 005AF300	CPUS 00000001	IDSQ 001D0019
+30 LDA 005EF548	RSM 00F705D8	CSCB 00F8BF48	TSB 00000000
+48 EWST 9707A4D3	XWST 19E75C60	JSTL 00000005	ECB 805ED7E0
+60 DUMP 005ECD90	FW1 FFFF0100	TMCH 00000000	ASXB 005EF300
+78 VSC 00000016	RCTP 005EDA68	LOCK 00000000	LSQH 00000000
+90 OUCB 00F8BE60	OUXB 00F68D48	FMCT 003B0000	XMPQ 00000000
+A8 MCC 00000000	JBNI 00FE07F0	JBNS 00F8BF58	SRQ 00000000
+C0 SMCT 00000007	SWTL 0000283B	SRBT 00000000	ATME 01673A00
+D8 TCBS 00000001	TCBL 00000000	WPRB 00FEEA08	SRDP 61FFFFF0
+F0 CMLC 00000000	SSCM 00000000	ASTE 005C4620	LTOV 005E6F80
+108 LXR 00000000	STKH 005EEA18	GOEL 005C4620	LQEL 002712F0
+120 FW3 00000000	GXL 00000000	EATT 00000000	EAT1 5120FC00
+138 FW4 00000000	RCMS 00000000	IOSC 00000002	PKXC 008001F4
+150 RESV 00000000	TCME 00000000		NSQA 00000000

Figure 3. Finding the System Abend Code and the PSW in the SYSUDUMP

2. Check whether the wait state bit (bit 14) of the PSW is on or off (see Figure 4 on page 87).

07802000 001269D2

 — PSW in the SYSUDUMP

To locate bit 14, divide the PSW into bits

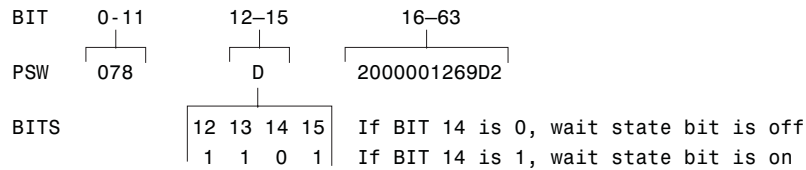


Figure 4. Finding the Wait State Bit in the PSW

3. If the wait state bit is on, use the keyword **WAIT**. If the wait state bit is off, use the keyword **LOOP**.
4. For the example in Figure 4, you would have the following keyword string:

5740SM105 RG00 BLOCKSET SORT LOOP

5. The next keyword is the module's name. Go to "Module Keyword Procedure" on page 89.

## System Abend Keyword Procedure

Use this procedure when DFSORT terminates abnormally and when you receive a nonzero (except n22) *system* completion code. If the completion code is n22 (where n is any hexadecimal digit), go to "Wait and Loop Keywords Procedure" on page 86.

You need a message for this procedure. If you did not receive any messages, refer to "How to Get Messages" on page 3.

1. Record the system abend code in the following format:

**ABENDnnn**

where nnn is the system abend code.

For example, if your system message contained ABEND=S0C1 U0000, you would have the following keyword string:

5740SM105 RG00 BLOCKSET SORT ABEND0C1

2. The next keyword is the module's name. Go to "Module Keyword Procedure" on page 89.

## User Abend Keyword Procedure

Use this procedure when DFSORT terminates abnormally and when you receive a nonzero *user* completion code.

You need a message for this procedure. If you did not receive any messages, refer to "How to Get Messages" on page 3.

1. Look at the user completion code.
  - If the code is less than 1000, it corresponds to a DFSORT error message. Go to "Message Keyword Procedure" on page 88.
  - If the code is equal to or greater than 1000, continue with step 2.
2. Record the user abend keyword in the following format:

## Type-of-Failure Keyword Procedures

ABENDnnnn

where nnnn is the user abend code.

For example, if your system completion message contained ABEND S000 U1004, your keyword string so far would be:

```
5740SM105  RG00  BLOCKSET  SORT  ABEND1004
```

3. The next keyword is the module's name. Go to "Module Keyword Procedure" on page 89.

## Message Keyword Procedure

Use message keywords when an error is described by, or associated with, a DFSORT error message.

1. Locate the DFSORT error message in the message data set.

The error message format is:

ICEnnnA c text

### Where Represents

**nnn** The error message number

**c** The diagnostic trace character

**text** An explanation of the error

2. Using the ICEnnnA part of the message, record the message keyword in the following format:

MSGICEnnnA

where nnn is the error message number.

For example, if you received the error message ICE061A, the keyword string would be:

```
5740SM105  RG00  BLOCKSET  SORT  MSGICE061A
```

3. If you received message ICE061A or ICE117A, go to "Message Variable Keywords" on page 92.  
If you did not receive either message ICE061A or ICE117A, your message keyword string is now complete. Go to "Chapter 9. Searching the IBM Software Support Database" on page 105.

## Incorrect-Output Keyword Procedure

Use the incorrect-output procedure when DFSORT:

- Generates unexpected output
- Fails to generate expected output
- Generates incorrect output

Do not use this procedure if the problem is associated with an error message. Use the message keyword procedure instead.

1. Use **INCORROUT** as your type-of-failure keyword.

## Type-of-Failure Keyword Procedures

You now have the following keyword string:

```
5740SM105 RG00 BLOCKSET SORT INCORROUT
```

2. The incorrect-output keyword string is now complete. Go to “Chapter 9. Searching the IBM Software Support Database” on page 105.

## Performance Keyword Procedure

Performance problems might be related to system tuning. Use this keyword when the performance problem cannot be corrected by system tuning.

1. Record the actual performance and the expected performance.
2. Use **PERFM** as your type-of-failure keyword.

You now have the following keyword string:

```
5740SM105 RG00 BLOCKSET SORT PERFM
```

3. The performance keyword string is now complete. Go to “Chapter 9. Searching the IBM Software Support Database” on page 105.

## Document Keyword Procedure

Use this keyword when a problem seems to be caused by incorrect or missing information in a DFSORT publication.

If the documentation problem is not severe, use the reader’s comment form (attached to the back of the manual that is causing the problem) to suggest corrections or improvements to the publication.

If you consider the problem a severe one, build the keyword string as follows:

1. Use **DOC** as your type-of-failure keyword.

You now have the following keyword string:

```
5740SM105 RG00 DOC
```

2. Use the order number on the cover of the document as your next keyword. The order number is of the form xxxx-yyyy-dd. Include xxxx, yyyy, and dd, but omit the hyphens. dd is the edition number.

For example, if you found a documentation error in *DFSORT Application Programming Guide Release 14*, which has the order number SC33-4035-18, your keyword string would be:

```
5740SM105 RG00 DOC SC33403518
```

3. The Document keyword string is now complete. Go to “Chapter 9. Searching the IBM Software Support Database” on page 105.

---

## Module Keyword Procedure

If your program ended abnormally or entered a wait or loop, your next keyword in the keyword string is the name of the module in control when the failure occurred.

You need a dump for this procedure. If you did not receive a dump, see Table 1 on page 3.

## Type-of-Failure Keyword Procedures

If you invoke DFSORT from a COBOL program compiled with VS COBOL, you might receive both a COBOL dump and a DFSORT dump. For this procedure, use only the DFSORT dump.

If you are using PL/I, you must specify the NOSPIE and NOSTAE macro at run time to ensure that you receive a DFSORT dump. If you do not, you might receive only a PL/I dump.

You need the abend address to obtain the module's name:

1. Find the program status word (PSW) in the dump, and note the instruction address, which is the last 31 bits of the PSW.

See Figure 5 for an MVS/ESA example. The last 31 bits of the PSW (that is, 0720CF52) is the instruction address.

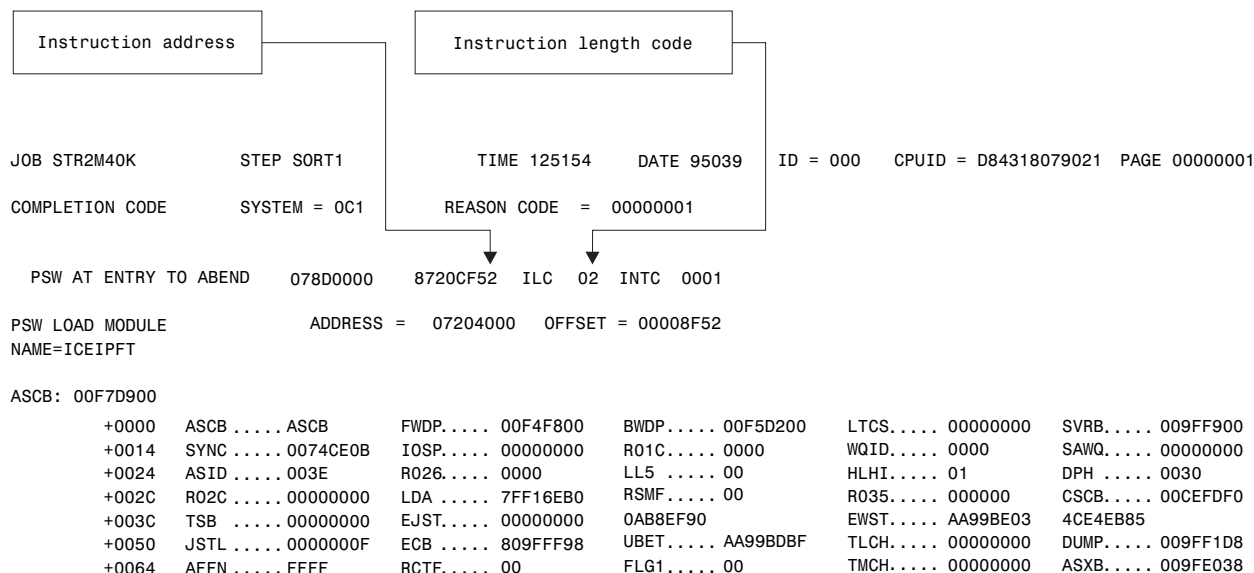


Figure 5. Finding the Abend Address

2. Obtain the abend address by subtracting the instruction length code (ILC) from the instruction address.

$$\begin{array}{r}
 0720CF52 \quad \text{instruction address} \\
 - \quad \quad 2 \quad \text{instruction length code} \\
 \hline
 0720CF50 \quad \text{abend address}
 \end{array}$$

3. Go to the abend address and scan backward until you find, on the right side of the dump, the closest name of the form:

ICName REL nn.n

where ICName is the module name.

In Figure 6 on page 91, ICEIPFT is the closest module name.

If you cannot find the module name, the problem might not be with DFSORT. In this case, call your IBM Support Center.



```

Abend address is 720CF50
0720CE00 D3C9C3C5 D5E2C5C4 40D4C1E3 C5D9C9C1 D3E24060 40D7D9D6 D7C5D9E3 E840D6C6 *LICENSED MATERIALS - PROPERTY OF*
0720CE20 40C9C2D4 40F5F7F4 F060E2D4 F1404DC3 5D40C3D6 D7E8D9C9 C7C8E340 C9C2D440 * IBM 5740 SM1 (C) COPYRIGHT IBM *
0720CE40 C3D6D9D7 4B40F1F9 F7F36B40 F1F9F9F8 4040C35D 40C3D6D7 E8D9C9C7 C8E340E6 *CORP. 1973. 1998 (C) COPYRIGHT W*
0720CE60 C1E3E2D6 D540D440 C3D6D5D5 C5D940C9 D5C34B40 F1F9F8F3 40404040 40404040 *ATSON M CONNER INC. 1983 *
0720CE80 40C1D3D3 40D9C9C7 C8E3E240 D9C5E2C5 D9E5C5C4 4B404040 C9C3C5C9 D7C6E340 * ALL RIGHTS RESERVED. ICEIPFT *
0720CEA0 D9C5D340 F1F44BF0 F0F861F2 F361F9F8 40F0F84B F3F740C2 C1E2C540 40404040 *REL 14.008/23/98 08.37 BASE *
0720CEC0 41C0F000 4190C800 41909800 D23ADB48 DB4992CF DB8392C9 D1829101 D9EE4780 *.{0...H...q.K.....k...ckIJbj.R...*
0720CEE0 C0C694BF D44558F0 D2989019 F2A45820 F30C5830 D4485830 30189801 20008C00 *(Fm.M..OKq..2u..3...M.....q.....*
0720CF00 000C5D00 C3981801 5B00F2FC 5A003014 50003014 96803008 5010F2FC 41100001 *..).Cq..$.2.!...&...o...&.2.....*
0720CF20 0A0B1810 58F0D298 5B00F2F8 5A003010 50003010 5010F2F8 91800445 47E0C0A2 *.....OKq$.28!...&...&.28j.M.. {s*
0720CF40 1B110A2E 58F0D298 8C000020 5000C39C 00005A00 F2F45A00 300C5000 300C5010 *.....OKq....).C...!.24!...&...&.*
0720CF60 F2F49819 F2A45810 D4485810 10089180 10404710 C0C641F0 C0C654F0 C3A04780 *24q.2u..M.....j....{F.O{F.OC....*
0720CF80 C0C69640 1040BFBF C3A44770 C0E60700 4100C0D8 47F0C0E0 C9C3C5E7 D7E4C240 *{Fo ..Cu..{W...{Q.0{ICEXPUB *
0720CFA0 1B110A08 18B050B0 D17C0700 4100C0F4 47F0C0FC C9C3C5E7 D7F2F440 1B110A08 *.....&.J@...{4.0{.ICEXP24 ....*
0720CFC0 18F005AF 917FD17C 4780C110 9680DAAB 45A0B004 41A0C120 50A0DA14 47F0C156 *.0..j"J@..A.o.....A.&....OA.*
0720CFE0 BFBF7284 47707016 45007014 C9C3C5E7 D7E4C240 0A094500 7022C9C3 C5E7D7F2 *...d.....ICEXPUB .....ICEXP2*

```

Figure 6. Finding the Module Name

- Use the module name as the keyword. For this example, the keyword string would be:

```
5740SM105 RG00 BLOCKSET SORT ABEND0C1 ICEIPFT
```

Check to see that the release level that appears after the module name is current. If the release level is not current, ensure DFSORT has been installed correctly.

- The next keyword shows the offset. Go to Figure 7 on page 92.

## Offset Keyword Procedure

If your program ended abnormally or entered a wait or loop, your next keyword in the keyword string after the name of the module is the offset keyword.

To determine the offset:

- Subtract the address of the first byte of the module from the abend address. The first byte of the module is the byte where the characters 'LICENSED' begin, just before the module name you found earlier by scanning the right side of the dump (as described in "Module Keyword Procedure" on page 89). If 'LICENSED' does not appear, the first byte is the beginning of the module name.

$$\begin{array}{r}
 0720CF50 \quad \text{abend address} \\
 -0720CE00 \quad \text{module's address} \\
 \hline
 150 \quad \text{offset}
 \end{array}$$

In the example in Figure 7 on page 92, the address of the module ICEIPFT is 0720CE00, and the abend address previously found is 0720CF50. Therefore, the offset is 150.

## Type-of-Failure Keyword Procedures

Module address is 720CE00				Finding the offset				Module name			
0720CE00	D3C9C3C5	D5E2C5C4	40D4C1E3	C5D9C9C1	D3E24060	40D7D9D6	D7C5D9E3	E840D6C6	*LICENSED MATERIALS - PROPERTY OF*		
0720CE20	40C9C2D4	40F5F7F4	F060E2D4	F1404DC3	5D40C3D6	D7E8D9C9	C7C8E340	C9C2D440	* IBM 5740 SM1 (C) COPYRIGHT IBM *		
0720CE40	C3D6D9D7	4B40F1F9	F7F36B40	F1F9F9F8	404DC35D	40C3D6D7	E8D9C9C7	C8E340E6	*CORP. 1973. 1998 (C) COPYRIGHT W*		
0720CE60	C1E3E2D6	D540D440	C3D6D5D5	C5D940C9	D5C34B40	F1F9F8F3	40404040	40404040	*ATSON M CONNER INC. 1983 *		
0720CE80	40C1D3D3	40D9C9C7	C8E3E240	D9C5E2C5	D9E5C5C4	4B404040	C9C3C5C9	D7C6E340	* ALL RIGHTS RESERVED. ICEIPFT *		
0720CEA0	D9C5D340	F1F44BF0	F0F861F2	F361F9F8	40F0F84B	F3F740C2	C1E2C540	40404040	*REL 14.008/23/98 08.37 BASE *		
0720CEC0	41C0F000	4190C800	41909800	D23ADB48	DB4992CF	DB8392C9	D1829101	D9EE4780	*.{0...H...q.K....k...ckIjbj.R...*		
0720CEE0	C0C694BF	D44558F0	D2989019	F2A45820	F30C5830	D4485830	30189801	20008C00	*(Fm.M..OKq..2u..3...M....q....*		
0720CF00	000C5D00	C3981801	5B00F2FC	5A003014	50003014	96803008	5010F2FC	41100001	*..).Cq..\$.2.!...&...o...&.2....*		
0720CF20	0A0B1810	58F0D298	5B00F2F8	5A003010	50003010	5010F2F8	91800445	47E0C0A2	*....0Kq\$.28!...&...&.28j.M.. {s*		
0720CF40	1B110A2E	58F0D298	8C000020	5000C39C	00005A00	F2F45A00	300C5000	300C5010	*....0Kq....).C...!.24!...&...&.*		
0720CF60	F2F49819	F2A45810	D4485810	10089180	10404710	C0C641F0	C0C654F0	C3A04780	*24q.2u..M....j....{F.0{F.OC....*		
0720CF80	C0C69640	1040BFBF	C3A44770	C0E60700	4100C0D8	47F0C0E0	C9C3C5E7	D7E4C240	*{Fo . . .Cu..{W....{Q.0{ICEXPUB *		
0720CFA0	1B110A08	18B050B0	D17C0700	4100C0F4	47F0C0FC	C9C3C5E7	D7F2F440	1B110A08	*....&.J@...{4.0{ICEXP24 ....*		
0720CF00	18F005AF	917FD17C	4780C110	9680DAAB	45A0B004	41A0C120	50A0DA14	47F0C156	*.0..j"J@..A.o.....A.&...0A.*		
0720CFE0	BFBF7284	47707016	45007014	C9C3C5E7	D7E4C240	0A094500	7022C9C3	C5E7D7F2	*...d.....ICEXPUB .....ICEXP2*		

Figure 7. Finding the Offset

- Record the offset keyword using the following format:

**OFFSET n**

where n is 1 to 4 digits and contains no leading zeros.

For example, the keyword string for a system abend 0C1 would look like this:

```
5740SM105 RG00 BLOCKSET SORT ABEND0C1 ICEIPFT OFFSET 150
```

- The keyword string is now complete. Go to “Chapter 9. Searching the IBM Software Support Database” on page 105.

## Message Variable Keywords

If your problem is for message ICE061A or ICE117A, you need to add a message variable keyword to your keyword string. Below are two sets of variable keywords. Add to your keyword string the set of variables that correspond to your message number.

### ICE061A Variable Keywords

- If you received message ICE061A, add to your keyword string the following four keywords from the message:
  - The ddname as follows:
    - For an input data set, use “SORTIN”
    - For a work data set, use “SORTWK”
    - For an output data set, use “SORTOUT”
  - ECBnn** (where nn is the completion code)
  - CSWnnnn** (where nnnn are the status bytes)
  - SENSEnnnn** (where nnnn are the sense bytes)

For example, if you received the message

```
ICE061A 0 I/O ERROR, DD SORTWK04, DEV 6C0, ECB 41, CSW 0020, SENSE 0000
```

the keyword string would be:

```
5740SM105 RG00 BLOCKSET SORT MSGICE061A SORTWK ECB41 CSW0020  
SENSE0000
```

## Type-of-Failure Keyword Procedures

2. The message keyword string is now complete. Go to “Chapter 9. Searching the IBM Software Support Database” on page 105.

### ICE117A Variable Keywords

1. If you received message ICE117A, add to your keyword string the following three keywords from the message:
  - a. The ddname as follows:
    - For an input data set, use “SORTIN”
    - For a work data set, use “SORTWK”
    - For an output data set, use “SORTOUT”
  - b. **OPnn** (where nn is the operation attempted)
  - c. The error description (In the message, the error description appears between the operation attempted and the last seek address or block count.)

For example, if you received the message

```
ICE117A J I/O ERROR NEW117, SORT1, 354, DA,  
SORTOUT, 0E OP, UNKNOWN COND., 00000007000100, EXCP
```

the keyword string would be:

```
5740SM105 RG00 PEERVALE SORT MSGICE117A SORTOUT OP0E UNKNOWN COND
```

Because ICE117A is issued only for the Peerage/Vale technique, the type-of-technique keyword in the above example is now PEERVALE.

2. The message keyword string is now complete. Go to “Chapter 9. Searching the IBM Software Support Database” on page 105.

## Type-of-Failure Keyword Procedures

---

## Chapter 8. Resolving Problems When Using DFSORT Panels

DFSORT Panels is an interface with which users can create and submit DFSORT jobs using interactive panels. DFSORT Panels is an entirely separate program from the DFSORT program that sorts, copies, and merges records.

- In background execution, DFSORT Panels invokes the DFSORT program by submitting a job stream a user generates within DFSORT Panels.
- In foreground execution, DFSORT Panels invokes the DFSORT program through the TSO CALL command, or a similar command.

DFSORT Panels is supported under ISPF, but you can also install it as a selection on an ISMF menu. The SORTREC line operator uses DFSORT processing invoked from the ISMF Data Set Application.

Failures you encounter when using DFSORT Panels can be caused by problems in the DFSORT Panels or the DFSORT program.

- Diagnose failures in the DFSORT Panels by following the steps covered in this section and *DFSMS/MVS Diagnosis Reference* (for ISMF) and in *ISPF and ISPF/PDF Diagnosis*.
- Diagnose failures in the DFSORT program by following the steps explained in “How to Use the Diagnosis Guide” on page ix.

---

### Obtaining Diagnostic Information

When you use DFSORT Panels, you can choose several options that supply dumps, messages, and diagnostic information in the event of an abend:

- Run with WTP message turned on in your TSO profile. This gives you the message identifier and an online symptom dump for error messages issued by components other than DFSORT Panels. See *TSO Command Reference* for information on setting up your TSO profile.
- Before invoking DFSORT Panels, allocate a SYSUDUMP, SYSABEND, or SYSMDUMP data set on TSO, so that you can have a full dump for an abend if you need one. For example, to dump to a new data set called UDUMP, use  
`ALLOC DD(SYSUDUMP) DS(UDUMP) MOD SPACE(5,5) CYLINDERS CATALOG`

To dump to a system output data set, use

```
ALLOC DD(SYSUDUMP) SYSOUT(Z) HOLD
```

See *JCL Reference* for more information on allocating a dump data set for background jobs.

- Run with the ISPF log turned on.
- Set the options in the DFSORT Panels user profile application to issue diagnostic information in the event of an abend. See “Using the Profile Application” on page 102 for details.

---

### Determining Where the Failure Occurred

This section explains how to determine whether the error is in the DFSORT Panels or the DFSORT program, and explains how to obtain additional information to help diagnose the failure.

## Determining Where the Failure Occurred

Answer the questions in Table 7 to locate the general cause of the abend.

Table 7. Abend Cause Location Procedure

Step	Action
1	Did your terminal display a DFSORT abend panel ( Figure 8 on page 97) or a TSO message indicating failure ( Figure 9 on page 97)? <ul style="list-style-type: none"><li>• If so, the abend is within the DFSORT Panels. Go to “Chapter 8. Resolving Problems When Using DFSORT Panels” on page 95.</li><li>• If not, go to step 2.</li></ul>
2	Did your terminal display an ISPF or ISMF abend panel? See Figure 11 and Figure 12 on page 99 for examples. <ul style="list-style-type: none"><li>• If so, the abend is within ISPF or ISMF. Go to “Resolving Failures in ISPF or ISMF” on page 98.</li><li>• If not, the abend is within the DFSORT program. Go to step 3.</li></ul>
3	Did you submit your job for foreground execution, and did you receive DFSORT error message ICE185A and a TSO abend symptom dump? <ul style="list-style-type: none"><li>• If so, go to “Foreground Execution Failures” on page 100.</li><li>• If not, go to step 4.</li></ul>
4	Did you submit your job for background execution and receive the standard DFSORT abend and error messages? <ul style="list-style-type: none"><li>• Go to “Background Execution Failures” on page 100.</li></ul>

---

## Resolving Failures in DFSORT Panels

DFSORT Panels is supported under ISPF, but you can also install it as a selection on an ISMF menu. The ISMF line operator SORTREC uses the DFSORT program to process records within the ISMF environment. Procedures for diagnosing failures in ISPF and ISMF are explained in *ISPF and ISPF/PDF Diagnosis and DFSMS/MVS Diagnosis Reference*. These documents explain how to construct a keyword string to search an IBM database for information about abends in ISPF and ISMF panel products.

If you are re-running a job to obtain diagnostic messages or a dump, read the information in “Obtaining Diagnostic Information” on page 95 to make sure that you have made any necessary changes to your TSO profile and have allocated any necessary data sets.

### DFSORT Panels Abend Panel

If DFSORT Panels terminates abnormally, a DFSORT Panels abend panel, shown on Figure 8 on page 97, informs you of the error and identifies the appropriate abend code. Use this code to begin building the keyword string, as explained in *DFSMS/MVS Diagnosis Guide* and *ISPF/PDF Diagnosis*.



```

                                DFSORT PANELS ABEND PANEL
COMMAND ===>

*****
*****
**
**
**          AN ABEND OCCURRED WHILE EXECUTING DFSORT PANELS          **
**
**          SYSTEM ABEND CODE: 0C4                                     **
**
**          DFSORT PANELS CANNOT CONTINUE                             **
**
**          PRESS THE ENTER KEY OR USE END TO TERMINATE DFSORT PANELS **
**          USE HELP TO DISPLAY A LIST OF COMMON ABEND CODES         **
**
*****
*****

```

Figure 8. DFSORT Panels Abend Panels Indicate the Problem Is in DFSORT Panels

## TSO Abend Message

When DFSORT Panels terminates abnormally and ISPF is in Test/Trace mode, a TSO message indicates the failure. See Figure 9.

Test/Trace mode is set automatically any time you invoke the ISPF/PDF Dialog Test facility; it remains set until you end ISPF.

```

* SPF SUBTASK ABEND *
PDF      ENDED DUE TO ERROR+
READY

```

Figure 9. TSO Messages Also Indicate That the Problem Is in DFSORT Panels

The plus sign (+) at the end of the message indicates that additional information is available.

1. To see the TSO message that describes the error further, enter a question mark (?).
2. When your terminal displays the second TSO error message, press Enter to obtain a dump of the abend to help diagnose the error.

If WTP is on in your TSO profile (see “Obtaining Diagnostic Information” on page 95), symptom dumps for the ISPF subtask (DFSORT Panels) and the main ISPF task are generated. See Figure 10 for an example.

## Resolving DFSORT Panel Failures

```
IEA995I SYMPTOM DUMP OUTPUT
ABEND CODE SYSTEM=0C4 TIME=11.33.40 SEQ=15485 CPU=0000 ASID=009F
PSW AT TIME OF ERROR 078D0000 822E9074 ILC 2 INTC 0D
ACTIVE LOAD MODULE=ICEPEP01 ADDRESS=022E8AF0 OFFSET=00000584
DATA AT PSW 022E906E - 8910000C 0A0D5880 A00CD507
GPR 0-3 822F9CA0 000C4000 022E28C6 00000011
GPR 4-7 00000001 022E2758 822FD928 822FD390
GPR 8-11 00000008 0001FE78 000218D0 022DA898
GPR 12-15 822E8B60 022DA898 00021620 00000008
END OF SYMPTOM DUMP
** LOGICAL SCREEN REQUEST FAILED - ABEND 0C4000 **
** CONTACT YOUR SYSTEMS PROGRAMMER OR DIALOG DEVELOPER **
*** SPF MAIN TASK ABEND ***
IEA995I SYMPTOM DUMP OUTPUT
ABEND CODE SYSTEM=0C4 TIME=11.33.48 SEQ=15486 CPU=0000 ASID=009F
PSW AT TIME OF ERROR 078D1000 8003BA6C ILC 2 INTC 0D
ACTIVE LOAD MODULE=ISPMMAIN ADDRESS=00039868 OFFSET=00002204
DATA AT PSW 0003BA66 - 00181610 0A0D98EC D09007FE
GPR 0-3 80000000 800C4000 0001B5D9 0003C05C
GPR 4-7 0003C03C 00000000 00000000 00000000
GPR 8-11 000158A8 00016620 0003B868 0003A868
GPR 12-15 80039868 0001B2A0 8003BA40 00000000
END OF SYMPTOM DUMP
READY
```

Figure 10. Two Symptom Dumps from TSO Appear on Your Screen

If a SYSUDUMP, SYSMDUMP, or SYSABEND data set was allocated before the error (see “Obtaining Diagnostic Information” on page 95), full dumps for the ISPF subtask and the main task are generated. If DISP=MOD was specified, the dump data set contains both dumps; otherwise, it just contains the second dump.

Use the first full dump, from the abending ISPF subtask (DFSORT Panels), to locate the name of the failing load module. The offsets identify the specific failing module within the load module.

If you did not have a dump data set allocated before the abend occurred, follow these steps:

1. Allocate a dump data set using the TSO ALLOC command (see “Obtaining Diagnostic Information” on page 95).
2. Invoke ISPF with the TEST parameter by using the ISPF TEST (or PDF TEST) command.
3. Specify N for RECOVER FROM ABENDS in your DFSORT Panels user profile.
4. Attempt to recreate the problem by going through your DFSORT Panels session again.

---

## Resolving Failures in ISPF or ISMF

If ISPF or ISMF terminates abnormally, a panel informs you of the error and identifies the appropriate abend code. Use this code to begin building the keyword string, as explained in *DFSMS/MVS Diagnosis Guide* and *ISPF and ISPF/PDF Diagnosis*. See Figure 11 on page 99 and Figure 12 on page 99 for examples.

```

----- ERROR RECOVERY -----
COMMAND ==>

* * * * *
* * * * *
* *           ISPF PROCESSOR ENDED ABNORMALLY           * *
* *
* *           Task ABEND code 0C4                       * *
* *
* *
* *
* *
* *
* * Press ENTER key to display primary option menu.     * *
* * Enter HELP command for list of common ABEND codes. * *
* *
* * * * *
* * * * *

```

Figure 11. ISPF Abend Panels Indicate the Problem Is in ISPF

```

                                ISMF ABEND PANEL
COMMAND ==>

*****
*****
**
**
**           AN ABEND OCCURRED WHILE EXECUTING ISMF           **
**
**           SYSTEM ABEND CODE: 0C4                           **
**
**           ISMF CANNOT CONTINUE                             **
**
**           PRESS THE ENTER KEY OR USE END TO TERMINATE ISMF **
**           USE HELP TO DISPLAY A LIST OF COMMON ABEND CODES **
**
*****
*****

```

Figure 12. ISMF Abend Panels Indicate the Problem Is in ISMF

## Resolving Failures in the DFSORT Program

The DFSORT program can fail during foreground or background execution.

## Resolving DFSORT Program Failures

### Foreground Execution Failures

If a foreground job fails because of a DFSORT program abend, you receive a TSO abend symptom dump.

More diagnostic information is available in the DFSORT program abend dump and from the TSO message output.

- To obtain a DFSORT program abend dump, follow these steps:
  1. Return to TSO and allocate a SYSUDUMP, SYSABEND, or SYSMDUMP data set to capture the abend dump. See “Obtaining Diagnostic Information” on page 95.
  2. Invoke DFSORT Panels again and re-run the job. The DFSORT program abend dump appears in the SYSUDUMP, SYSABEND, or SYSMDUMP data set, and you can begin the diagnosis sequence starting in “How to Use the Diagnosis Guide” on page ix.
- To obtain the TSO message output, change the ALLOC FILE(SYSOUT) statement in the ICEQFG01 CLIST (for SORT), ICEQFG02 CLIST (for COPY), or ICEQFG03 CLIST (for MERGE) to direct the SYSOUT output to a user data set or a system output data set. For example:

```
ALLOC FILE(SYSOUT) DS(DFSORT.OUTMSG) MOD SPACE (1,1) TRACKS CATALOG
```

All DFSORT Panels CLISTs are stored in a data set allocated to SYSPROC. To change the required CLIST, make a private copy with the modifications and place it in a private data set allocated to SYSPROC (with higher priority than the CLIST system data sets).

Be certain that the abend is not caused either by modifications made at your site to the CLISTs (ICEQFG01, ICEQFG02, ICEQFG03, ICEQFG04) invoked by DFSORT Panels or by any other modifications not supported by DFSORT or by DFSORT Panels.

### Background Execution Failures

With background jobs, you can diagnose failures not caused by DFSORT Panels using the diagnosis procedures in “Chapter 7. Resolving Failures in the DFSORT Program” on page 79.

If a dump data set is not present in the job stream, then specify a SYSUDUMP, SYSABEND, or SYSMDUMP DD statement on the DFSORT Execute Statement Entry Panel and resubmit the job to obtain the dump. See Table 1 on page 3 for more information.

The job can also be saved in a data set for forwarding to IBM personnel.

---

## Recording Information

DFSORT Panels uses the ISPF log to record information about the foreground and background execution of DFSORT program jobs and error conditions detected during processing.

To ensure that log entries are made, you must specify a primary allocation greater than zero in the ISPF log data set default options on the ISMF Log and List Defaults menu. If you specify zero as a primary allocation, no information about your session appears in the ISPF log. For more information about setting up ISPF log parameters, see *ISPF/PDF Guide*.

## Recording Information in the ISPF/ISMF Logs

The ISPF log routinely records:

- Successful foreground execution of a DFSORT job
- Successful submission of a JCL job stream for background execution
- Successful saving of JCL job stream in a data set

The log also contains entries for error conditions:

- Foreground execution failures
- Failures to submit a job stream because of SUBMIT errors
- Failures to save a JCL job stream in a data set because of DYNALLOC, OPEN, CLOSE, or DCB address errors
- Data set verification failures because of LOCATE, DYNALLOC, OPEN, BLDL, or CLOSE errors
- User-Defined Panels Interface (UDPI) errors
- ISPF service failures

## Inter-Module Traces

To receive a log record for every module that executes during a DFSORT Panels session, enter a **Y** in the LOG INTER-MODULE TRACE on the Logging/Abend Control Entry Panel in the Profile Application. This information can help diagnose loops or determine module flow when there is more than one possible path to an error.

Normally, set this option to **N**, because logging degrades performance.

## ISMF Logging with SORTREC

Table 8 shows DFSORT reason codes can appear in ISMF error messages in the ISPF log if the DFSORT SORTREC line operator processor detects an error. For more information on ISMF's use of the ISPF log, see *DFSMS/MVS Diagnosis Reference*.

Table 8. DFSORT Reason Codes

Reason Code	Description
RSN08000	OBJECT NAME (DATA SET NAME IN DATA SET APPLICATION, VOLUME SERIAL NUMBER IN VOLUME APPLICATION) IS BLANK
RSN08001	INPUT DATA SET NAME WAS NOT FOUND IN ANY CATALOG IN THE SPECIFIED SEARCH ORDER
RSN08002	DATA SET NAME IS GDG BASE NAME
RSN08003	DATA SET HAS UNEXPECTED CATALOG ENTRY TYPE
RSN08004	DYNAMIC ALLOCATION FAILED FOR INPUT DATA SET DURING SORTREC PROCESSING
RSN08005	AN INVALID LOGGING TYPE WAS PASSED TO THE ISMF LOG UTILITY
RSN08006	LOG UTILITY—A REQUEST WAS MADE TO LOG ABEND INFORMATION BUT THE POINTER TO THE RETRY WORK AREA (RWA) IS INVALID
RSN08007	LOG: MESSAGE NOT FOUND, RC=12
RSN08008	LOG: SEVERE ERROR, RC=20
RSN08009	LOG: INVALID RETURN CODE
RSN08010	A REQUEST WAS MADE TO LOG STATUS INFORMATION BUT NO ADDITIONAL MESSAGE IDS WERE SPECIFIED. AT LEAST ONE MESSAGE ID MUST BE SPECIFIED.

### Using the Profile Application

DFSORT Panels responds to abends based on:

- The value (**Y** or **N**) you specify in the RECOVER FROM ABENDS field of the Logging/Abend Control Entry Panel of your profile
- The way you invoke ISPF (in Normal or Test/Trace mode)
- The value (**Y** or **N**) you specify in the ISSUE DFSORT DIAGNOSTIC MESSAGES field of the DFSORT Control Entry Panel of your profile.

These options are summarized in the following sections. See *DFSORT Application Programming Guide* for complete details.

### Directing Abend Processing

Use the RECOVER FROM ABENDS field of the Logging/Abend Control Entry Panel to direct the DFSORT Panels to attempt to recover from abends or to transfer control to ISPF or TSO.

#### Recovering from Abends

If you specify **Y** in the RECOVER FROM ABENDS field of the Logging/Abend Control Entry Panel, DFSORT Panels attempts to recover from the abend and retain control. When an abend occurs, a DFSORT Panels symptom dump from the time of the failure is placed in the ISPF log. In some cases the symptom dump is sufficient to diagnose the problem. It contains information about:

- The application involved in the failure
- The function involved in the failure
- The current release level and function modification identifier (FMID) for DFSORT Panels
- The abend code
- The name of the module involved in the failure (normally ICEPSA01)
- The entry point address (EPA) of the failing module
- The program status word (PSW)
- The contents of the registers when the abend occurred

#### Passing Control to ISPF or TSO

If you specify **N** in the RECOVER FROM ABENDS field, DFSORT does not attempt to retain control when an abend occurs. The type of dump you receive depends upon whether the ISPF mode is set to Normal or Test/Trace:

- Normal mode  
DFSORT Panels passes control to ISPF. The ISPF error recovery panel appears and the log receives an ISPF symptom dump, consisting of:
  - The level of ISPF
  - The abend code
  - The PSW
  - The name of the module involved in the failure, if available
  - The EPA of the module involved in the failure
  - The EPA of the modules ISPUBS and ISTBLS
  - The contents of the registers when the abend occurred
- Test/Trace mode  
ISPF passes control to TSO. TSO displays an error message. If a SYSUDUMP, SYSABEND, or SYSMDUMP data set was allocated on TSO, you receive a dump. This method gives you the most accurate information on the abend.  
Test/Trace mode is set automatically any time you invoke the ISPF/PDF Dialog Test facility; it remains set until you end ISPF.

## Obtaining Diagnostic Messages

If you specify **Y** in the ISSUE DFSORT DIAGNOSTIC MESSAGES field of the DFSORT Control Entry Panel of your profile, DFSORT issues diagnostic messages to provide information about EXCP counts, buffer allocation, and processing techniques.

For background jobs, DFSORT generates a SORTDIAG DD statement to be inserted into your completed job stream. For foreground jobs, DFSORT generates the corresponding ALLOCATE command to achieve the same result under TSO.

Specify **N** in the field to direct DFSORT not to issue diagnostic messages.





---

## Chapter 9. Searching the IBM Software Support Database

After you complete your keyword string, you are ready to search the IBM Software Support Facility to determine if an APAR has already been recorded.

*Table 9. Steps to Search the IBM Software Support Facility*

Step	Action
1	<p>Search the IBM software support data base for your keyword string by using a search tool, such as Info Access, or by calling the IBM Support Center for assistance.</p> <p>If the problem has been reported before, a correction might already be available. This correction could be a PTF, an APAR fix, or a bypass.</p>
2	<p>If you find a set of APAR closing descriptions, compare the APAR closing descriptions in the IBM software support data base with your problem.</p>
3	<p>If you find an APAR that matches your problem, correct your problem by applying the PTF, APAR fix, or bypass described in the APAR. See "Chapter 10. Fixing or Bypassing the Problem" on page 107.</p>
4	<p>If you cannot find an appropriate APAR, try varying your search argument:</p> <ol style="list-style-type: none"><li>1. Drop one keyword at a time starting from the right until you get to the type-of-application keyword.</li><li>2. If your type-of-failure keyword is <b>LOOP</b>, <b>WAIT</b>, or <b>PERFM</b>, try searching with one or both of the other two. Sometimes what seems like a WAIT state might actually be a LOOP or a PERFM problem.</li></ol> <p>If you cannot find an appropriate APAR using either technique, go to "Chapter 11. Reporting a Problem" on page 109.</p>

## Searching the IBM Software Support Data Base

---

## Chapter 10. Fixing or Bypassing the Problem

---

### Fixing the Problem

Apply the appropriate PTF or APAR fix using SMP/E. If you are running DFSORT resident, you must re-IPL with CLPA to enable the fix, unless you are using Dynamic LPA.

For more information, see *SMP/E User's Guide*.

---

### Bypassing the Problem

This list describes some actions you can take that might allow you to bypass a problem:

- **If the problem is related to I/O for SORTIN, SORTOUT, or OUTFIL**, try using the BSAM run-time option.
- **If the problem is related to storage**, try adding 32KB by changing the REGION parameter or the MAINSIZE or SIZE run-time option. System 0C4 and DFSORT user 1nnn abends are sometimes related to storage problems and can be bypassed in this way.
- **If the problem is with the Blockset technique**, try using the NOBLKSET run-time option TEMPORARILY to force DFSORT to use a different technique. For more information, see "Type-of-Technique Keyword Procedure" on page 84.

**Note:** OPTION NOBLKSET should NOT be used as a permanent solution because it can significantly degrade performance.

## Fixing or Bypassing the Problem

---

## Chapter 11. Reporting a Problem

Follow these instructions to report a problem to the IBM Support Center.

Step	Action
1	<p>Gather the information.</p> <ol style="list-style-type: none"><li>1. Have the following information ready when you report a problem:<ul style="list-style-type: none"><li>• Customer number</li><li>• Current service level (PTF list and list of APAR fixes applied)</li><li>• The keyword string developed to search the IBM software support database</li><li>• Processor number (serial model)</li></ul></li><li>2. Also have the following documents ready to assist in analyzing the problem:<ul style="list-style-type: none"><li>• Source code (assembler or compiler listings) and load module for any user routines or invoking program.</li><li>• A listing of the job control language (JCL), DFSORT control statements, and all messages. You can use a SORTDIAG DD statement to obtain all messages, including diagnostic messages.</li><li>• A listing of the symbols information if a SYMNames DD statement was specified. You can produce this listing by specifying a SYMNOUT DD statement.</li><li>• A copy of the input data set in machine-readable form.</li><li>• A list of the installation defaults selected at your site. You can produce this list by using the DEFAULTS operator of ICETOOL with the same step, job, or link library used for the job associated with this APAR. See <i>DFSORT Application Programming Guide</i> for complete details of the DEFAULTS operator of ICETOOL.</li><li>• A dump from the failing run of DFSORT.</li></ul></li></ol>
2	<p>Contact the IBM Support Center.</p> <p>After gathering the information, contact the IBM Support Center for assistance in resolving the problem. If the difficulty is an IBM problem, the representative will assist you in preparing an APAR.</p>
3	<p>Submit APAR documentation.</p> <p>When submitting material for an APAR to IBM, carefully pack and clearly identify any tapes that contain application source programs, job stream data, data sets, or libraries.</p> <p>Each tape must have the following information attached and visible:</p> <ul style="list-style-type: none"><li>• The APAR number assigned by IBM</li><li>• A list of data sets on the tape (for example, application source program, JCL, data, and load modules)</li><li>• Data on how the tape was created:<ul style="list-style-type: none"><li>– The exact JCL listing or the list of commands used</li><li>– The record format and block size used for each data set</li><li>– Tape labeling</li><li>– The recording mode and density</li></ul></li></ul>





---

## Part 4. Appendixes



---

## Appendix A. Valid Input/Output Options at User Exits—ICE044I

An x in the table indicates an option that you can use with the user exit in question.

Option	E18	E19	E38	E39
SYNAD	x	x	x	x
EXLST	x <sup>1</sup>	x	x	x
EROPT	x		x	
EODAD	x			
BSAM EXLST	x		x <sup>2</sup>	x
VSAM PASSWORD	x		x <sup>2</sup>	x
<b>Notes:</b>				
1        Cannot be used if input is concatenated on unlike devices.				
2        For merge applications.				

## Valid Options at User Exits—ICE044I

## Appendix B. User Exit Return Codes—ICE071A

An x in the table indicates a valid return code that your user exit can pass back for the given situation. All other values (those not marked with an x and those not in the table) are invalid for the given situation.

End of input (EOI) is indicated by a zero in the first word of the parameter list passed to your exit.

Table 10. User Exit Return Codes - ICE071A

Enn	Case	EXITCK	Situation	Valid Return Codes					
				0	4	8	12	16	20*
E15	S1	STRONG	Without SORTIN			X	X	X	
E15	W1	WEAK	Without SORTIN	X	X	X	X	X	X
E15	S2	STRONG	With SORTIN before EOI	X	X	X	X	X	X
E15	W2	WEAK	With SORTIN before EOI	X	X	X	X	X	X
E15	S3	STRONG	With SORTIN after EOI			X	X	X	
E15	W3	WEAK	With SORTIN after EOI	X	X	X	X	X	X
E16	16	N/A	N/A	X	X	X			
E32	32	N/A	N/A			X	X	X	
E35	S0	STRONG	Without SORTOUT or OUTFIL before EOI		X	X		X	
E35	W0	WEAK	Without SORTOUT or OUTFIL before EOI	X	X	X	X	X	X
E35	S1	STRONG	Without SORTOUT or OUTFIL after EOI			X		X	
E35	W1	WEAK	Without SORTOUT or OUTFIL after EOI	X	X	X	X	X	X
E35	S2	STRONG	With SORTOUT or OUTFIL before EOI	X	X	X	X	X	X
E35	W2	WEAK	With SORTOUT or OUTFIL before EOI	X	X	X	X	X	X
E35	S3	STRONG	With SORTOUT or OUTFIL after EOI			X	X	X	
E35	W3	WEAK	With SORTOUT or OUTFIL after EOI	X	X	X	X	X	X

**Note:** \* Only applies to COBOL E15 and COBOL E35

## User Exit Return Codes—ICE071A

---

## Appendix C. Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights or other legally protectible rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, programs, or services, except those expressly designated by IBM, are the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594, U.S.A.

---

## Programming Interface Information

This book primarily documents information that is NOT intended to be used as a Programming Interface of DFSORT.

This book also documents intended Programming Interfaces that allow the customer to write programs to obtain the services of DFSORT. This information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:

┌ **Programming Interface Information** \_\_\_\_\_

Programming Interface Information

└ **End of Programming Interface Information** \_\_\_\_\_

---

## Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

DFSMS/MVS  
DFSMSrmm  
DFSORT  
Hiperspace  
IBM

Language Environment  
MVS/ESA  
OS/390  
SmartBatch





---

## Summary of Changes

---

### New Programming Support for Release 13 (PTFs after April, 1996)

#### Additional Year 2000 Features

A new Y2S format can order and transform two-digit character or zoned decimal year data according to the century window, while handling binary zeros, blanks and binary ones in the year field as special indicators.

A new Y2B format can order and transform two-digit binary year data according to the century window.

FREE=CLOSE support for DFSPARM makes it possible to override the SORT statements generated by multiple COBOL SORT verbs in the same COBOL program.

#### OS/390 Registration

With OS/390 R2 and above, a check is performed to ensure that the DFSORT product is licensed for use, either as a feature of OS/390 or as a separate program product.

---

### New Programming Support for Release 13 (PTFs - April, 1996)

#### Year 2000 Features

New Y2C, Y2Z, Y2P and Y2D formats, in conjunction with a new Y2PAST installation and run-time option, allow you to handle two-digit year data in the following ways:

- Set the appropriate century window for your applications (for example, 1915-2014 or 1950-2049).
- Order two-digit character, zoned decimal, packed decimal or decimal year data according to the century window using Blockset SORT or MERGE (for example, order 96 representing 1996 before 00 representing 2000 in ascending sequence, or order 00 before 96 in descending sequence).
- Transform two-digit character, zoned decimal, packed decimal or decimal year data to four-digit character year data according to the century window using OUTFIL OUTREC (for example, transform 96 to 1996 and 00 to 2000).

A new PD0 format allows you to order and transform parts of packed decimal fields (for example, month and day in date fields) using SORT, MERGE and OUTFIL.

#### Performance Improvements for FLR and VLR Blockset Sorts

Performance improvements for FLR and VLR Blockset sorts include the following:

- Dataspace sorting can now be used for variable-length record sort applications.
- DFSORT data processing methods have been improved.
- Dynamic storage adjustment is a new feature that allows DFSORT to automatically use more storage than the TMAXLIM value for a Blockset sort application if DFSORT determines that doing so should improve performance. New installation option DSA=n has been added to enable you to specify the dynamic storage adjustment limit.

- The upper limit for the amount of main storage that can be specified and used by DFSORT has been raised from 32M to 2000M. Specifying more main storage can provide the following benefits:
  - It allows DFSORT to sort very large data sets more efficiently.
  - It allows more sort applications to be done entirely in main storage, eliminating the need for intermediate work space and greatly reducing the EXCP counts for those applications.
  - It increases the maximum amount of data DFSORT can process in a single sort application.
- New installation option IOMAXBF=n has been added to enable you to specify the upper limit for the amount of storage to be used for SORTIN and SORTOUT data set buffers, which in turn limits the amount of data that can be transferred in a single I/O operation.
- The upper limit for the number of JCL and dynamically allocated work data sets that can be specified and used by DFSORT's Blockset technique has been raised from 32 to 100. The use of more work data sets increases the maximum amount of data DFSORT can process in a single sort application.
- Changes to the DFSORT SVC provide caching selection enhancements that improve storage control caching performance, especially for SORTIN and SORTOUT devices.
- DFSORT can now use NOEQUALS for VLR Blockset applications if EQUALS=NO is specified at installation or NOEQUALS is specified at run-time. The use of NOEQUALS can improve performance and is recommended for applications for which the order of records that collate identically need not be preserved from input to output. To minimize migration concerns, the IBM-supplied default for the ICEMAC EQUALS option is the new value VLBLKSET, which is equivalent to EQUALS=YES for VLR Blockset applications and to EQUALS=NO for all other applications.

## Floating Point for SUM

FL format can now be used with the SUM control statement for short (4-byte), long (8-byte) and extended (16-byte) floating point data.

## Security Improvements

Changes to the DFSORT SVC provide security improvements that bring DFSORT up to B1 security standards.

## EXCPVR Processing Removed

To enhance DFSORT's protection of system integrity, EXCPVR processing will no longer be used. EXCPVR parameter values will continue to be accepted, but will have no effect on DFSORT processing. In general, the performance improvements provided by EXCPVR processing have diminished with newer technologies and will be more than offset by the performance improvements listed above. Please ignore any references to EXCPVR in this book; all such references will be deleted when the book is updated.

---

## New Device Support for Release 13 (PTFs)

The IBM 3590 Magnetic Tape Subsystem is supported for input, output and work data sets.

---

## New Programming Support for Release 13

### DFSORT's Performance Booster for The SAS\*\* System

DFSORT Release 13 provides significant CPU time improvements for SAS applications. To take advantage of this new feature, contact SAS Institute Inc. for details of the support they provide to enable this enhancement.

### Dynamic Hipersorting

Dynamic Hipersorting is a new, automatic feature that eliminates the unintended system paging activity and expanded storage and paging data set space shortages that sometimes resulted from a large amount of Hipersorting activity, especially from multiple concurrent Hipersorting applications.

Dynamic Hipersorting allows for more optimal DFSORT and system performance and provides installation options that allow you to customize HIPRMAX=OPTIMAL to your own criteria. With the advent of this feature, we recommend that you use HIPRMAX=OPTIMAL as your site default.

### Performance

Performance enhancements for DFSORT applications that use the Blockset technique include the following:

- Dataspace sorting, introduced in R12 for fixed-length record sort applications, now available for variable-length record sort applications (MVS/ESA only)
- Improved data processing methods for fixed-length record sort applications
- OUTFIL processing for producing multiple output data sets using a single pass over one or more input data sets.

### OUTFIL Processing

OUTFIL is a new DFSORT control statement that allows you to create one or more output data sets for a sort, copy, or merge application from a single pass over one or more input data sets. You can use multiple OUTFIL statements, with each statement specifying the OUTFIL processing to be performed for one or more output data sets. OUTFIL processing begins after all other processing ends (that is, after processing for exits, options, and other control statements). OUTFIL statements support a wide variety of output data set tasks, including:

- Creation of multiple output data sets containing unedited or edited records from a single pass over one or more input data sets.
- Creation of multiple output data sets containing different ranges or subsets of records from a single pass over one or more input data sets. In addition, records that are not selected for any subset can be saved in another output data set.
- Conversion of variable-length record data sets to fixed-length record data sets.
- Sophisticated editing capabilities such as hexadecimal display and control of the way numeric fields are presented with respect to length, leading or suppressed zeros, symbols (for example, the thousands separator and decimal point), leading and trailing positive and negative signs, and so on. Twenty-six pre-defined editing masks are available for commonly used numeric editing patterns, encompassing many of the numeric notations used throughout the world. In addition, a virtually unlimited number of numeric editing patterns are available via user-defined editing masks.

- Selection of a character or hexadecimal string for output from a lookup table, based on a character, hexadecimal, or bit string as input (that is, lookup and change).
- Highly detailed three-level (report, page, and section) reports containing a variety of report elements you can specify (for example, current date, current time, page number, character strings, and blank lines) or derive from the input records (for example, character fields, edited numeric input fields, record counts, and edited totals, maximums, minimums, and averages for numeric input fields).

## National Language Support

### Cultural Sort and Merge

DFSORT will allow the selection of an active locale at installation or run time and will produce sorted or merged records for output according to the collating rules defined in the active locale. This provides sorting and merging for single- or multi-byte character data based on defined collating rules which retain the cultural and local characteristics of a language.

### Cultural Include and Omit

DFSORT will allow the selection of an active locale at installation or run time and will include or omit records for output according to the collating rules defined in the active locale. This provides inclusion or omission for single- or multi-byte character data based on defined collating rules which retain the cultural and local characteristics of a language.

### UTFIL Reports

UTFIL allows date, time, and numeric values in reports to be formatted in many of the notations used throughout the world.

### ICETOOL Reports

ICETOOL's DISPLAY operator allows date, time, and numeric values in reports to be formatted in many of the notations used throughout the world.

## ICETOOL Enhancements

ICETOOL is now even more versatile as a result of enhancements to the existing operators. The improvements to ICETOOL include:

- Allowing more data to be displayed with the DISPLAY and OCCUR operators. DISPLAY now allows up to 20 fields (increased from 10) and a line length of up to 2048 characters (increased from 121). OCCUR now allows a line length of up to 2048 characters (increased from 121).
- More extensive formatting capabilities for numeric fields with the DISPLAY operator. Formatting items can be used to change the appearance of individual numeric fields in reports with respect to separators, decimal point, decimal places, signs, division, leading strings, floating strings and trailing strings. Thirty-three pre-defined editing masks are available for commonly used numeric editing patterns, encompassing many of the numeric notations used throughout the world. Leading and trailing strings can also be used with character fields.
- Display of the four-digit or two-digit year with the DISPLAY and OCCUR operators.
- Division of reports into sections with the DISPLAY operator, based on the values in a character or numeric break field. Statistics (total, maximum, minimum and/or average) can be displayed for each section as well as for the entire report.
- Automatic use of UTFIL processing for a list of TO ddnames with the COPY and SORT operators, resulting in creation of multiple TO (output) data sets from a single pass over the FROM (input) data set.

- Allowing OUTFIL statements to be specified in the USING data set in addition to or instead of the TO operand with the COPY and SORT operators.
- Allowing the active locale to be specified for the COPY, COUNT and SORT operators, in order to override the installation default for the active locale. Thus, multiple active locales can be used in the same ICETOOL job step for these operators.
- Allowing the last record for each unique field value to be kept with the SELECT operator.

## INCLUDE/OMIT Substring Search

INCLUDE and OMIT function enhancements provide powerful substring search capability to allow inclusion or omission of records when:

- A specified character or hexadecimal constant is found anywhere within a specified input field (that is, a constant is a substring within a field) or
- A specified input value is found anywhere within a specified character or hexadecimal constant (that is, a field is a substring within a constant).

## SMF Type-16 Record Enhancements

New fields, such as information pertaining to each DFSORT run about SORTIN, SORTINnn, SORTOUT and OUTFIL data sets, control statements, record counts, specified values for E15, E35, HIPRMAX, DSPSIZE, FILSZ, LOCALE and AVGRLEN, have been added to DFSORT's SMF type-16 record.

SMF=FULL, SMF=SHORT, and SMF=NO can now be specified in an OPTION statement in DFSPARM or the extended parameter list, to produce or suppress the SMF type-16 record for an individual application.

**Note:** The offsets of fields ICESPGN, ICEUSER, and ICEGROUP have changed in the Release 13 SMF record. If you have programs that reference those fields, recompile them using the Release 13 version of the ICESMF macro, before attempting to run them against Release 13 SMF records.

## Other Enhancements

Several ICEMAC installation options have been added or changed:

- The IBM-supplied default for EXCPVR has been changed from ALL to NONE.
- The IBM-supplied default for DYNAUTO has been changed from NO to YES.
- SDBMSG enables you to specify whether DFSORT should use the system-determined optimum block size for DFSORT message data sets and ICETOOL message and list data sets.
- LOCALE enables you to select an active locale.
- ODMAXBF enables you to specify the maximum buffer space DFSORT can use for each OUTFIL data set.
- EXPMAX enables you to specify the maximum total amount of available storage to be used for all Hipersorting applications.
- EXPOLD enables you to specify the maximum total amount of old expanded storage to be used at any one time by all Hipersorting applications.
- EXPRES enables you to specify the minimum amount of available expanded storage to be reserved by DFSORT for use by non-Hipersorting applications.

Several run-time options have been added or changed:

- LOCALE enables you to select an active locale.

- SMF enables you to specify whether DFSORT is to produce SMF type-16 records.
- ODMAXBF enables you to specify the maximum buffer space DFSORT can use for each OUTFIL data set.
- NZDPRINT enables you to indicate that positive ZD summation results are not to be converted to printable numbers (overrides ZDPRINT).
- HILEVEL=YES on the MODS statement enables you to indicate that the E15 and E35 routines are to be treated as COBOL exits.
- DEBUG options BUFFERS=ANY and BUFFERS=BELOW will now be recognized but not used.

DFSORT will now ignore any DD statements not needed for the application (for example, a SORTIN DD statement will be ignored for a merge application).

For unsuccessful completion due to an unsupported operating system, DFSORT, ICEGENER, and ICETOOL will now pass back a return code of 24 to the operating system or invoking program.

The installation initialization exit, ICEIEXIT, enables you to specify the maximum buffer space DFSORT can use for each OUTFIL data set.

The installation termination exit, ICETEXIT, contains additional fields such as a flag to indicate that OUTFIL processing was used.

For INREC and OUTREC:

- The upper limit for columns and the end of fields has been raised from 32000 to 32752.
- 1: before the RDW field of variable-length records will be accepted and ignored.

For INCLUDE and OMIT, COND=ALL, COND=(ALL), COND=NONE, and COND=(NONE) enable you to include or omit all records.

The L2 value from the RECORD statement will be used if the L1 value is not specified when an E15 or E32 user exit passes all of the input records.

When input is a VSAM data set and output is a non-VSAM data set with RECFM not specified, DFSORT will now set the output RECFM as blocked rather than unblocked, when doing so will allow the use of the system-determined optimum block size for output.

---

## **New Programming Support for Release 12 (PTFs)**

ICEGENER, copy, and Blockset sort and merge can now be used when a tape output data set is specified with DISP=MOD or DISP=OLD, without specifying the RECFM, LRECL, or BLKSIZE in the DD statement.

Sequential striping is supported for input and output data sets.

Compression is supported for input and output data sets.

BatchPipes/MVS input and output pipes are supported.



---

## **New Device Support for Release 12 (PTFs)**

Four-digit device numbers are supported.

The IBM 3390-9 DASD is supported for input, output, and work data sets, although it is not recommended for work data sets for performance reasons.

The IBM RAMAC Array DASD and RAMAC Array Subsystem are supported for input, output, and work data sets.

The IBM 3990 Model 6 control unit is supported.

The IBM cached 9343 control unit models are supported.



---

# Readers' Comments — We'd Like to Hear from You

**DFSORT**  
**Messages, Codes and Diagnosis Guide**  
**Release 14**

**Publication No. SC26-7050-04**

**Overall, how satisfied are you with the information in this book?**

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**How satisfied are you that the information in this book is:**

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Please tell us how we can improve this book:**

Thank you for your responses. May we contact you?  Yes  No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

---

Name

---

Address

---

Company or Organization

---

Phone No.



Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES

# BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation  
RCF Processing Department  
G26/M86 050  
5600 Cottle Road  
SAN JOSE, CA  
U.S.A. 95193-0001



Fold and Tape

Please do not staple

Fold and Tape





Program Number: 5740-SM1



Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

SC26-7050-04

