IBM SMP/E for z/OS and OS/390

Messages, Codes, and Diagnosis

GA22-7770-02

IBM SMP/E for z/OS and OS/390

Messages, Codes, and Diagnosis

GA22-7770-02

Note!
Before using this information and the product it supports, be sure to read the general information under “Notices” on page 313.

Third Edition, March 2002

This book replaces the previous edition, GA22-7770-01, which is now obsolete. Changes or additions to text and illustrations are indicated by a vertical line to the left of the change.

This edition applies to IBM SMP/E for z/OS and OS/390, V3R1, program number 5655-G44, and to all subsequent releases and modifications, unless otherwise indicated in new editions.

Changes or additions to text and illustrations are indicated by a vertical line to the left of the change.

Order IBM publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

IBM welcomes your comments. A form for readers’ comments may be provided at the back of this publication, or you may address your comments to the following address:

International Business Machines Corporation
Department 55JA, Mail Station P384
2455 South Road
Poughkeepsie, NY 12601-5400
United States of America

FAX (United States & Canada): 1+845+432+9405
FAX (Other Countries):
Your International Access Code +1+845+432-9405

IBMLink (United States customers only): IBMUSM10(MHVRCFS)
Internet e-mail: mhvrcfs@us.ibm.com
World Wide Web: http://www.ibm.com/servers/eserver/zseries/zos/webqs.html

If you would like a reply, be sure to include your name, address, telephone number, or FAX number.

Make sure to include the following in your comment or note:
- Title and order number of this book
- Page number or topic related to your comment

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.

© Copyright International Business Machines Corporation 1986, 2002. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About This Book v
How to Use This Publication v
Notations Used in This Publication v
Bibliography vi
Using LookAt to look up message explanations vi

Summary of Changes ix

Chapter 1. SMP/E Messages 1
Message Format. 1
Diagnostic Messages 1

Chapter 2. SMP/E Return Codes 269
ACCEPT Return Codes 269
ACCEPT Error Recovery 271
APPLY Return Codes 272
APPLY Error Recovery 274
CLEANUP Return Codes. 274
CLEANUP Error Recovery 275
GENERATE Return Codes 275
GENERATE Error Recovery 275
JCLIN Return Codes 275
JCLIN Error Recovery 276
LINK Return Codes. 276
LINK Error Recovery 277
LIST Return Codes 277
LIST Error Recovery 277
LOG Return Codes 277
RECEIVE Return Codes 277
RECEIVE Error Recovery 278
REJECT Return Codes 279
REJECT Error Recovery 279
REPORT Return Codes 279
REPORT Error Recovery. 280
RESTORE Return Codes 280
RESTORE Error Recovery 282
SET Return Codes 282
SET Error Recovery 282
UCLIN Return Codes 282
UCLIN Error Recovery 283
ZONECOPY Return Codes 283
ZONECOPY Error Recovery 283
ZONEDELETE Return Codes 283
ZONEDELETE Error Recovery 284
ZONEEDIT Return Codes 284
ZONEEDIT Error Recovery 284
ZONEEXPORT Return Codes 284
ZONEEXPORT Error Recovery 285
ZONEIMPORT Return Codes 285
ZONEIMPORT Error Recovery 285
ZONEMERGE Return Codes 285
ZONEMERGE Error Recovery 286
ZONERENAME Return Codes. 286

© Copyright IBM Corp. 1986, 2002

ZONERENAME Error Recovery 286

Chapter 3. Diagnosing SMP/E Problems 287
Is SMP/E the Problem? 287
Installation Problems 287
Messages and Return Codes 288
Performance Problems 288
Utility Problems 288
SMPCSI Problems 289
Exception SYSMOD Problems. 289
Handling SMP/E Problems without Calling IBM 290
Analyzing Errors 290
Resolving Problems with the SMP/E Region Size. 292
Resolving Space Problems (x37 Abends). 292
Resolving VSAM Access Problems 294
Handling SMP/E Abends 295
Preventing Problems with the SMP/E Dialogs 295
Reporting SMP/E Problems to IBM 295
Information for All Failure Types 296
Types of Failure Keywords 297
Reporting Abends 298
Reporting Loops and Waits 300
Reporting Message Problems 301
Reporting Incorrect Output 301
Reporting Performance Problems 301
Reporting Documentation Problems. 302
Reporting Dialog Problems 303
Collecting Documentation for SMP/E Problems 306
Obtaining Dumps When Abends Occur 306
Using the DEBUG Command 306

Appendix. Accessibility. 311
Using assistive technologies 311
Keyboard navigation of the user interface. 311

Notices 313
Trademarks. 314

Glossary 315

Index 329

About This Book

Use this publication when:

- You need information about an SMP/E message or return code. This publication will help you (1) identify the SMP/E message or return code, (2) understand the action taken by the system, and (3) respond to the message or return code as needed.
- You suspect a problem with SMP/E. This publication will assist you in searching for a solution and, if necessary, reporting the problem to the IBM Support Center.

How to Use This Publication

To interpret an SMP/E message or code, see these chapters:

- **Chapter 2, "SMP/E Return Codes".** This chapter lists return codes according to the command, interprets the return codes, and lists possible error conditions. In addition, except for the LOG command, each section outlines instructions for error recovery.
- **Chapter 1, "SMP/E Messages".** This chapter lists the SMP/E diagnostic messages (prefix GIM) along with the system action and your required response. Messages appear alphanumerically.

To diagnose an SMP/E problem, see:

- **Chapter 3, "Diagnosing SMP/E Problems".** This chapter contains procedures for determining whether you can solve a suspected SMP/E problem and, if not, for gathering the information needed to report the problem to IBM.

The pages in the messages chapters have running heads to help you find messages more quickly. The running head corresponds to the first and last messages included on the page.

Notations Used in This Publication

- Abbreviations used in this publication are:

  DLIB Distribution library

  PTF Program temporary fix

  SMP/E System Modification Program Extended
- Messages appear in all uppercase letters. The following symbols may appear:

  ' Apostrophe

  ***** Asterisk

  Blank

  , Comma

  = Equal sign

  ( ) Parentheses

  . Period
- *Italic* character strings represent variables. For example, *aaaaaaaa* is a variable. However, each message SMP/E generates has a specific value (for example, MODA).

© Copyright IBM Corp. 1986, 2002

Bibliography

This section tells you more about the SMP/E library.

- The IBM SMP/E for z/OS and OS/390, V3R1 publications are available as printable PDF files and BookManager-viewable softcopy at

  `http://www.ibm.com/servers/eserver/zseries/zos/bkserv/`
- Table 1 lists the IBM SMP/E for z/OS and OS/390, V3R1 publications and briefly describes each one.
- For information on z/OS publications and more information on the IBM SMP/E for z/OS and OS/390, V3R1 books, see *z/OS Information Roadmap*.

Table 1. Publications for IBM SMP/E for z/OS and OS/390, V3R1

Title	Description
SMP/E Messages, Codes, and Diagnosis, GA22-7770	Explains SMP/E messages and return codes and the actions to take for each; and how to handle suspected SMP/E problems.
SMP/E Commands, SA22-7771	Explains SMP/E commands and processing in detail.
SMP/E Reference, SA22-7772	Explains SMP/E modification control statements, data sets, exit routines, and programming interfaces in detail and provides additional SMP/E reference material.
SMP/E User's Guide, SA22-7773	Describes how to use SMP/E to install programs and service.

Using LookAt to look up message explanations

LookAt is an online facility that allows you to look up explanations for z/OS messages, system abends, and some codes. Using LookAt to find information is faster than a conventional search because in most cases LookAt goes directly to the message explanation.

You can access LookAt from the Internet at:

`http://www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/lookat.html`

or from anywhere in z/OS where you can access a TSO command line (for example, TSO prompt, ISPF, z/OS UNIX System Services running OMVS).

To find a message explanation on the Internet, go to the LookAt Web site and simply enter the message identifier (for example, `IAT1836` or `IAT*`). You can select a specific release to narrow your search. You can also download code from the *z/OS Collection*, SK3T-4269 and the LookAt Web site so you can access LookAt from a PalmPilot (Palm VIIx suggested).

To use LookAt as a TSO command, you must have LookAt installed on your host system. You can obtain the LookAt code for TSO from a disk on your *z/OS Collection*, SK3T-4269 or from the LookAt Web site. To obtain the code from the LookAt Web site, do the following:

1. Go to http://www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/lookat.html.
2. Click the **News** button.
3. Scroll to **Download LookAt Code for TSO and VM.**
4. Click the ftp link, which will take you to a list of operating systems. Select the appropriate operating system. Then select the appropriate release.
5. Find the **lookat.me** file and follow its detailed instructions.

To find a message explanation from a TSO command line, simply enter: **lookat** *message-id*. LookAt will display the message explanation for the message requested.

Note: Some messages have information in more than one book. For example, IEC192I has routing and descriptor codes listed in *z/OS MVS Routing and Descriptor Codes*. For such messages, LookAt prompts you to choose which book to open.

Bibliography

Summary of Changes

Summary of Changes
for GA22-7770-02
as Updated, March 2002

This revision reflects the deletion, addition, or modification of information to support miscellaneous maintenance items. A vertical bar (v) in the left margin indicates changes to the text and illustrations.

New Information

- These new messages have been added:
  - GIM21000E
  - GIM22200E
  - GIM32501E
- An appendix with z/OS product accessibility information has been added.

Changed Information

- These messages have been changed:
  - GIM28201S
  - GIM30231E
  - GIM56101S
  - GIM61801E
  - GIM65601S
  - GIM66801E
  - GIM66901E
  - GIM70711T
  - GIM70717I
- In all messages, variable names have been improved.

Moved Information

- Chapter Chapter 1, “SMP/E Messages” on page 1 has been moved ahead of Chapter 2, “SMP/E Return Codes” on page 269.

Deleted Information

- These messages have been deleted:
  - GIM44266S
  - GIM44500S
  - GIM53001S
  - GIM53101S
  - GIM53102S
  - GIM53103S
  - GIM53201E
  - GIM53401I
  - GIM53402I
  - GIM53501I
  - GIM53601I

© Copyright IBM Corp. 1986, 2002

- GIM53701I
- GIM53702T
- GIM53703I
- GIM53704I
- GIM53705T
- GIM53801E
- GIM55701W
- GIM55702S
- GIM57401S
- GIM69018S
- GIM69019S
- GIM69020I
- GIM69021E
- GIM69021I
- GIM69021S
- GIM69021T
- GIM69021W
- GIM69023S
- GIM69024I
- GIM69025I
- GIM69026T
- GIM69027S
- GIM69028S
- GIM69029W
- GIM69030W

• ″CONVERT Return Codes″ in Chapter 2, “SMP/E Return Codes” on page 269.

Summary of Changes for GA22-7770-01 SMP/E Version 3

The book contains information previously presented in GA22-7770-00, which applied to z/OS Version 1 Release 1.

New Information

• These new messages have been added:
  - GIM20202I
  - GIM20312E
  - GIM20313S
  - GIM20314S
  - GIM20315S
  - GIM20399S
  - GIM22800S
  - GIM26003S
  - GIM26004W
  - GIM27301S
  - GIM27605S

- GIM27650S
- GIM30237S
- GIM33001W
- GIM35967I
- GIM35968I
- GIM35969I
- GIM35970I
- GIM35971I
- GIM36800E
- GIM37150S
- GIM41700E
- GIM41701E
- GIM41702E
- GIM41703I
- GIM41704I
- GIM42100W
- GIM42500W
- GIM42800S
- GIM43500S
- GIM43500W
- GIM44015S
- GIM44016S
- GIM44017S
- GIM44500I
- GIM45200I
- GIM45201I
- GIM45201S
- GIM45500S
- GIM45700S
- GIM45800S
- GIM45900S
- GIM46000S
- GIM46100I
- GIM46200I
- GIM46700I
- GIM46701I
- GIM46702I
- GIM46703I
- GIM46800I
- GIM47000T
- GIM47100S
- GIM47400S
- GIM47500I
- GIM47501I
- GIM47600I

- GIM47601I
- GIM47700S
- GIM47800S
- GIM47801S
- GIM47900S
- GIM48000S
- GIM48100S
- GIM48200W
- GIM48200S
- GIM48600S
- GIM48900S
- GIM49000S
- GIM49100S
- GIM49200S
- GIM50903I
- GIM50904S
- GIM50905S
- GIM56403S
- GIM56404S
- GIM56405S
- GIM57302S
- GIM57303S
- GIM57304S
- GIM57802E
- GIM57802I
- GIM62602S
- GIM64605E
- GIM65402I
- GIM65403I
- GIM65404I
- GIM65405I
- GIM65406I
- GIM69070S
- GIM69186E
- GIM69199E
- GIM69264S

Changed Information

- These messages have been changed:
  - GIM20307T
  - GIM20550W
  - GIM20551S
  - GIM22402I
  - GIM22406I
  - GIM23401T

- GIM27301E
- GIM27301W
- GIM27701W
- GIM33503S
- GIM33600S
- GIM35601E
- GIM35701E
- GIM35801W
- GIM36600S
- GIM37100S
- GIM39202S
- GIM40901E
- GIM40902I
- GIM40902W
- GIM40903E
- GIM44005S
- GIM44330T
- GIM57702T
- GIM61701E
- GIM61705E
- GIM61903W
- GIM62001I
- GIM62701W
- GIM63201I
- GIM69158I
- GIM70715E

Moved Information

- Chapter Chapter 1, “SMP/E Messages” on page 1 has been moved ahead of Chapter 2, “SMP/E Return Codes” on page 269.

Deleted Information

- These messages have been deleted:
  - GIM22801E
  - GIM22801W

See Appendix B, ″Migration″, in *SMP/E User’s Guide* for information on the new and changed function in this release of SMP/E.

This book contains terminology, maintenance, and editorial changes, including changes to improve consistency and retrievability.

Chapter 1. SMP/E Messages

This chapter explains the format of SMP/E messages and lists the SMP/E diagnostic messages in alphanumeric order.

Message Format

SMP/E messages (except for WTO, WTP, and WTOR messages) are in the format GIM*nnnnnv yy text*. WTO, WTP, and WTOR messages are in the format GIM*nnnv yy text*.

- *n...n* is the message number.
- *v* is the severity level, as follows:

Severity Level	Return Code	Category	When Issued
I	00	Informational	– To show stages of SMP/E processing. – To accompany other messages as an explanation for errors or unusual conditions.
W	04	Warning	– Processing completed but SMP/E detected a possible error. Look at these messages to determine if SMP/E's action was appropriate.
E	08	Error	– SMP/E processing did not complete properly.At least one SYSMOD failed. For example, when APPLY processing of a SYSMOD stops,this is an error, and SMP/E issues a return code of at least 8.
S	12	Severe	– An entire SMP/E command failed. For example, when APPLY processing stops because of insufficient storage, SMP/E issues a return code of 12.
T	16	Terminating	– A situation occurred that forced SMP/E processing to stop.
T	20	Terminating	– An internal SMP/E error caused SMP/Eprocessing to stop. For example, there may be an interface problem between SMP/E modules or a data error on the SMPCSI data set.

SMP/E sets the severity level of a message and the return code at the same time. If a message does not cause SMP/E to set a return code, the severity level of that message is *I*.

- *yy* is the severity highlighting code, as follows:
  Blanks Severity I and W messages
  ** Severity E, S, and T messages

  Note: The *yy* value is not shown in this manual because it varies from message to message.
- *text* is the message text.

Diagnostic Messages

© Copyright IBM Corp. 1986, 2002

GIM000T **MESSAGE MODULE** *modname* **COULD NOT BE LOADED. SMP/E PROCESSING IS STOPPED. ABEND CODE:** *abncode* **REASON CODE:** *rsncode*

Explanation:

modname	name of the SMP/E message module
abncode	4-character system abend code
rsncode	2-character system reason code.

The SMP/E message module was not loaded. Possible reasons include:

- The LANGUAGE value on the GIMSMP EXEC statement was specified incorrectly, or specified an unsupported language. Therefore, no corresponding message module could be found.
- The indicated message module was not found in a link list library.

System Action: SMP/E processing stops.

Programmer Response: Verify that the LANGUAGE value on the GIMSMP EXEC statement is specified correctly, and that the language is supported by SMP/E.

Check that the message module was actually installed in the target library when SMP/E was installed.

See the *z/OS MVS System Codes* manual for an explanation of the system abend and reason codes.

GIM00001T **MODULE GIMKXENU COULD NOT BE LOADED. SMP/E PROCESSING IS STOPPED. ABEND CODE:** *abncode* **REASON CODE:** *rsncode***.**

Explanation:

abncode	4-character system abend code
rsncode	2-character system reason code.

The SMP/E constants module, GIMKXENU, was not loaded. Possible reasons include:

- The LANGUAGE value on the GIMSMP EXEC statement was specified incorrectly, or specified an unsupported language. Therefore, no corresponding report module could be found.
- GIMKXENU was not found in a link list library.

System Action: SMP/E processing stops.

Programmer Response: Check that GIMKXENU was actually installed in the target library when SMP/E was installed.

See the *z/OS MVS System Codes* manual for an explanation of the system abend and reason codes.

GIM00002W **THE CAUSER SYSMOD SUMMARY REPORT WILL NOT CONTAIN POSSIBLE CAUSE TEXT BECAUSE MODULE GIMPCENU WAS NOT LOADED.**

Explanation: The load module, GIMPCENU, that is needed to produce possible cause text in the Causer SYSMOD Summary Report was not loaded. As a result, the information provided by that load module will not appear in the report.

Possible reason includes: GIMPCENU was not found in a link list library.

System Action: Processing continues.

Programmer Response: Check to see if GIMPCENU was actually installed in the target library when SMP/E was installed.

GIM20101I *proctype* **PROCESSING FAILED FOR THE** *ddname* **LIBRARY.**

Explanation:

proctype	OPEN or CLOSE
ddname	ddname of the library being processed

System Action: The messages that follow show the action SMP/E takes.

Programmer Response: If OPEN failed, check for the following:

- A missing DD statement
- An incorrect data set name

Before running the job in the future, you may want to fix this problem.

GIM20101W *proctype* **PROCESSING FAILED FOR THE** *ddname* **LIBRARY.**

Explanation:

proctype	OPEN or CLOSE
ddname	ddname of the library being processed

System Action: The messages that follow show the action SMP/E takes.

Programmer Response: If OPEN failed, check for the following:

- A missing DD statement
- An incorrect data set name

If necessary, fix this problem and rerun the job.

GIM20101E *proctype* **PROCESSING FAILED FOR THE** *ddname* **LIBRARY.**

Explanation:

proctype	OPEN or CLOSE
ddname	ddname of the library being processed

System Action: The messages that follow show the action SMP/E takes.

Programmer Response: Do **one** of the following:

- If OPEN failed, check for the following:
  - A missing DD statement
  - An incorrect data set name

  Fix the problem and rerun the job.
- If CLOSE failed, rerun the job. If CLOSE continues to fail, SMP/E needs data set maintenance.

GIM20101S *proctype* **PROCESSING FAILED FOR THE** *ddname* **LIBRARY.**

Explanation:

proctype	OPEN or CLOSE
ddname	ddname of the library being processed

System Action: The messages that follow show the action SMP/E takes.

Programmer Response: Do **one** of the following:

- If OPEN failed, check for the following:
  - A missing DD statement
  - An incorrect data set name

  Fix the problem and rerun the job.
- If CLOSE failed, rerun the job. If CLOSE continues to fail, SMP/E needs data set maintenance.

GIM20101T *proctype* **PROCESSING FAILED FOR THE** *ddname* **LIBRARY.**

Explanation:

proctype	OPEN or CLOSE
ddname	ddname of the library being processed

System Action: The messages that follow show the action SMP/E takes.

Programmer Response: Do **one** of the following:

- If OPEN failed, check for the following:
  - A missing DD statement
  - An incorrect data set name

  Fix the problem and rerun the job.
- If CLOSE failed, rerun the job. If CLOSE continues to fail, SMP/E needs data set maintenance.

GIM20201W SMP/E COULD NOT OBTAIN ENOUGH STORAGE FOR PROCESSING.

Explanation: SMP/E could not allocate internal tables because of insufficient storage.

System Action: The messages that follow in the output listing show SMP/E's actions.

Programmer Response: If necessary, do **one** of the following, then rerun the job:

- Increase the REGION parameter on the EXEC statement.
- Decrease the number of SYSMODs that SMP/E is processing in this run.
- Decrease the PEMAX value in the OPTIONS entry being used.

GIM20201E SMP/E COULD NOT OBTAIN ENOUGH STORAGE FOR PROCESSING.

Explanation: SMP/E could not allocate internal tables because of insufficient storage.

System Action: The messages that follow in the output listing show SMP/E's actions.

Programmer Response: Do **one** of the following:

- Increase the REGION parameter on the EXEC statement.
- Decrease the number of SYSMODs that SMP/E is processing in this run.
- Decrease the PEMAX value in the OPTIONS entry being used.

Rerun the job.

GIM20201S SMP/E COULD NOT OBTAIN ENOUGH STORAGE FOR PROCESSING.

Explanation: SMP/E could not allocate internal tables because of insufficient storage.

System Action: The messages that follow in the output listing show SMP/E's actions.

Programmer Response: Do **one** of the following:

- Increase the REGION parameter on the EXEC statement.
- Decrease the number of SYSMODs that SMP/E is processing in this run.
- Decrease the PEMAX value in the OPTIONS entry being used.

Rerun the job.

GIM20201T SMP/E COULD NOT OBTAIN ENOUGH STORAGE FOR PROCESSING.

Explanation: SMP/E could not allocate internal tables because of insufficient storage.

System Action: The messages that follow in the output listing show SMP/E's actions.

Programmer Response: Do **one** of the following:

- Increase the REGION parameter on the EXEC statement.
- Decrease the number of SYSMODs that SMP/E is processing in this run.
- Decrease the PEMAX value in the OPTIONS entry being used.

Rerun the job.

GIM20202I *obtnfixd* **TOTAL BYTES ARE RESERVED FOR ICT FIXED DATA, OF WHICH,** *usedfixd* **BYTES ARE USED.** *obtnvard* **TOTAL BYTES ARE RESERVED FOR ICT VARIABLE DATA, OF WHICH,** *usedvard* **BYTES ARE USED.**

Explanation:

obtnfixd	amount of storage obtained for the ICT fixed data
usedfixd	amount of storage used for the ICT fixed data
obtnvard	amount of storage obtained for the ICT variable data
usedvard	amount of storage used for the ICT variable data

SMP/E could not allocate internal tables because of insufficient storage.

System Action: The messages that follow in the output listing show the actions taken by SMP/E.

Programmer Response: If the amount of storage used for the ICT fixed data area, *usedfixd*, is approximately 32MB, decrease the number of SYSMODs that SMP/E is processing in the run. Otherwise, do one or more of the following:

- Increase the REGION parameter on the EXEC statement
- Decrease the number of SYSMODs that SMP/E is processing in this run
- Decrease the PEMAX value in the OPTIONS entry being used.

Rerun the job. If storage is still insufficient, contact the IBM Support Center.

GIM20301S THERE IS A SYNTAX ERROR IN THE COMMAND AT COLUMN *colnum***.**

Explanation:

colnum	2-digit column number

SMP/E found a syntax error at the specified column in the command being processed. This message is immediately preceded by the line with the syntax error.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: Command processing stops.

Programmer Response: Check the operands specified on the command. Correct the error and rerun the job.

GIM20301T THERE IS A SYNTAX ERROR IN THE COMMAND AT COLUMN *colnum***.**

Explanation:

colnum	2-digit column number

SMP/E found a syntax error at the specified column in the command being processed. This message is immediately preceded by the line with the syntax error.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: SMP/E processing stops.

Programmer Response: Check the operands specified on the command. Correct the error and rerun the job.

GIM20302S THERE IS A SYNTAX ERROR IN THE MCS AT COLUMN *colnum***.**

Explanation:

colnum	2-digit column number

SMP/E found a syntax error at the specified column in the modification control statement (MCS) being processed. This message is immediately preceded by the line with the syntax error.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: SYSMOD processing stops.

Programmer Response: Check the operands specified on the MCS. Correct the error and rerun the job.

GIM20303E THERE IS A SYNTAX ERROR IN THE UCL STATEMENT AT COLUMN *colnum***.**

Explanation:

colnum	2-digit column number

SMP/E found a syntax error at the specified column in the UCL statement being processed.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: SMP/E ignores the UCL statement. Processing continues with the next UCL statement.

Programmer Response: Check the operands specified on the UCL statement. Correct the error and rerun the job.

GIM20304E THERE IS A SYNTAX ERROR IN THE LINKAGE EDITOR INPUT STATEMENT.

Explanation: During JCLIN processing, SMP/E found a syntax error on a linkage editor input statement for the job step it is scanning.

System Action: SMP/E stops scanning that job step. Scanning continues with the next job step.

Programmer Response: Check the linkage editor input statement. Correct the error and rerun the job.

GIM20305E THERE IS A SYNTAX ERROR IN THE ASSEMBLER INPUT STATEMENT.

Explanation: During JCLIN processing, SMP/E found a syntax error on an assembler input statement for the job step it is scanning.

System Action: SMP/E stops scanning that job step. Scanning continues with the next job step.

Programmer Response: Check the assembler input statement. Correct the error and rerun the job.

GIM20306E THERE IS A SYNTAX ERROR IN THE COPY INPUT STATEMENT.

Explanation: During JCLIN processing, SMP/E found a syntax error on a copy input statement for the job step it is scanning.

System Action: SMP/E stops scanning that job step. Scanning continues with the next job step.

Programmer Response: Check the copy input statement. Correct the error and rerun the job.

GIM20307T THERE IS A SYNTAX ERROR IN THE EXEC PARM STATEMENT AT CHARACTER *charpos***.**

Explanation:
charpos character position within the PARM value

SMP/E found a syntax error at the indicated character position within the PARM value on the EXEC statement for GIMSMP, GIMZIP, or GIMUNZIP.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: SMP/E processing stops.

Programmer Response: Check the parameters specified on the EXEC statement. Correct the error and rerun the job.

GIM20308E THERE IS A SYNTAX ERROR IN THE ZONEEDIT CHANGE STATEMENT AT COLUMN *colnum***.**

Explanation:
colnum 2-digit column number

SMP/E found a syntax error at the indicated column on a ZONEEDIT CHANGE statement.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: Command processing stops.

Programmer Response: Check the operands specified on the CHANGE statement. Correct the error and rerun the job.

GIM20308S THERE IS A SYNTAX ERROR IN THE ZONEEDIT CHANGE STATEMENT AT COLUMN *colnum***.**

Explanation:
colnum 2-digit column number

SMP/E found a syntax error at the indicated column on a ZONEEDIT CHANGE statement.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: Command processing stops.

Programmer Response: Check the operands specified on the CHANGE statement. Correct the error and rerun the job.

GIM20309W THERE IS A SYNTAX ERROR IN OPCODE MEMBER *membname* **AT COLUMN** *colnum***.**

Explanation:
membname name of a member in SMPPARM
colnum column number

During JCLIN processing, SMP/E found a syntax error at the indicated column in the OPCODE member being used.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond

column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: JCLIN processing continues with the next job step.

Programmer Response: Check the parameters specified on the SMPPARM statement. Correct the error and rerun the job.

GIM20310E THERE IS A DOUBLE-BYTE CHARACTER SYNTAX ERROR IN THE *command* **INPUT.**

Explanation:
command SMP/E command containing the error

The indicated input contained double-byte characters. However, the shift-in and shift-out characters created by the input device are not in the format expected by SMP/E. When double-byte characters are used, a shift-out character, X'0E', must precede the first double-byte character, and a shift-in character, X'0F', must follow the last one. There may have been an extra shift-in or shift-out character, or a required shift-in or shift-out character may have been missing.

System Action: SYSMOD processing stops.

Programmer Response: Reenter the input and rerun the job. If the error occurs again, there may be a problem with the input device you are using. Contact your hardware support group for help in solving this problem.

GIM20310S THERE IS A DOUBLE-BYTE CHARACTER SYNTAX ERROR IN THE *command* **INPUT.**

Explanation:
command SMP/E command containing the error

The indicated input contained double-byte characters. However, the shift-in and shift-out characters created by the input device are not in the format expected by SMP/E. When double-byte characters are used, a shift-out character, X'0E', must precede the first double-byte character, and a shift-in character, X'0F', must follow the last one. There may have been an extra shift-in or shift-out character, or a required shift-in or shift-out character may have been missing.

System Action: Command processing stops.

Programmer Response: Reenter the input and rerun the job. If the error occurs again, there may be a problem with the input device you are using. Contact your hardware support group for help in solving this problem.

GIM20311E THERE IS A SYNTAX ERROR IN THE CONTROL STATEMENT AT COLUMN *colnum***.**

Explanation:
colnum column number

SMP/E found a syntax error at the specified column in the command or MCS statement being processed. This message is immediately preceded by the line with the syntax error.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: SYSMOD processing stops.

Programmer Response: Check the operands specified on the command or MCS statement. Correct the error and rerun the job.

GIM20311S THERE IS A SYNTAX ERROR IN THE CONTROL STATEMENT AT COLUMN *colnum***.**

Explanation:
colnum column number

SMP/E found a syntax error at the specified column in the command or MCS statement being processed. This message is immediately preceded by the line with the syntax error.

Note: SMP/E checks only columns 1 through 72. If data, such as a period, is specified beyond column 72, SMP/E ignores it and indicates an error in the statement after the one containing that data.

System Action: Command processing stops.

Programmer Response: Check the operands specified on the command or MCS statement. Correct the error and rerun the job.

GIM20312E THERE IS A SYNTAX ERROR IN THE JCLIN CONTROL STATEMENT AT COLUMN *colnum***.**

Explanation:
colnum column number

During JCLIN processing, SMP/E found a syntax error on a JCLIN control statement (**//*SMPE-IF**, **//*SMPE-ELSE**, or **//*SMPE-END**) at the indicated column.

System Action: JCLIN processing fails for the current JCLIN input stream.

Programmer Response: Correct the error and rerun the SMP/E command.

GIM20313S THERE IS A SYNTAX ERROR IN THE CONTROL STATEMENT FROM DD STATEMENT *ddstmnt* **AT COLUMN** *colnum***.**

Explanation:

ddstmnt	ddname or DDDEF name
colnum	2-digit column number

There is a syntax error at the specified column in the data set associated with the identified ddname or DDDEF. This message is immediately preceded by the line with the syntax error.

System Action: Processing stops.

Programmer Response: Check the syntax of the control statements. Correct the error and rerun the job.

GIM20314S THERE IS A SYNTAX ERROR IN PACKAGE ATTRIBUTE FILE *pafname* **AT COLUMN** *colnum***.**

Explanation:

pafname	name of the package attribute file. If this name exceeds 300 characters in length, only the first 300 characters will appear in this message.
colnum	column number

There is a syntax error at the specified column in the package attribute file. This message is immediately preceded by the line with the syntax error.

System Action: Processing stops.

Programmer Response: Check that the correct package file has been specified in your SERVER data set. If the package file specification is correct, contact the supplier of the package.

GIM20315S THERE IS A SYNTAX ERROR IN SMPPARM MEMBER *member* **AT COLUMN** *number***.**

Explanation:

member	SMPPARM data set member name (GIMDDALC or GIMEXITS)
number	column number

SMP/E found a syntax error at the specified column in a control statement within the SMPPARM member being processed. This message is immediately preceded by the line with the syntax error.

Note: SMP/E checks only columns 1 through 72. If any data is specified beyond column 72, SMP/E ignores it and indicates an error in the statement following the one containing the error.

System Action: Command processing stops.

Programmer Response: Check the operands specified on the control statement (see *SMP/E Reference*. for syntax details). Correct the error and rerun the job .

GIM20399S *command* **PROCESSING FAILED BECAUSE SMPPTS SPILL DATA SETS ARE DEFINED. SPILL DATA SETS ARE SUPPORTED WITH APAR IR42960 on OS/390 RELEASE 5 AND HIGHER.**

Explanation:

command	an SMP/E command

Command processing failed because SMPPTS spill data sets are defined. SMP/E detected a JCL DD statement or DDDEF entry for the first spill data set, SMPPTS1. SMPPTS spill data can only be processed by the following

- OS/390 releases 5 and 6 with PTF UR52517. UR52517 includes APAR IR42960.
- OS/390 releases 7, 8, 9, and 10 with PTF UR52518. UR52518 includes APAR IR42960.
- z/OS release 1 with PTF UR52518 or z/OS release 2 or higher.
- SMP/E Version 3.

System Action: Command processing stops.

Programmer Response: Do any of the following:

- If possible, move all the members from the SMPPTS spill data sets to the primary SMPPTS data set. Then remove the JCL DD statement or DDDEF entry for SMPPTS1 and rerun the job.
- If all members from the SMPPTS spill data sets cannot fit in the primary SMPPTS data set, you may want to reduce the number of SYSMODs being saved in the SMPPTS and its spill data set, using a release of SMP/E that supports SMPPTS spill data sets, before attempting to move all members into the primary SMPPTS data set. The number of SYSMODs in the SMPPTS and its spill data sets can be reduced by doing a REJECT or by accepting SYSMODs which have already been applied.
- If it is impossible to fit all SYSMODs in a single SMPPTS data set, you need to use a level of SMP/E that supports spill data sets before the failing command can be processed.

GIM20400E THE DDNAME SPECIFIED BY THE *operand* **OPERAND OF THE** *command* **COMMAND COULD NOT BE ALLOCATED. A JCL COMMENT WILL BE GENERATED EACH TIME A** *jclstmt* **IS REQUIRED.**

Explanation:

operand the command operand specifying a ddname that could not be allocated
command SMP/E command being processed at the time of the failure
jclstmt the JCL statement that will be replaced by a comment in the generated JCL

SMP/E was unable to allocate a ddname specifying a data set that is needed for the generation of a JCL statement. An accompanying message indicates the reason SMP/E was unable to allocate the ddname.

System Action: SMP/E generates a JCL comment in place of the JCL statement in the generated JCL. Command processing continues, but the output job will not run unless you fix the error.

Programmer Response: Do **one** of the following:

- Use the information provided by the accompanying message to resolve the allocation failure, then rerun the command.
- Edit the JCL output to add the required JCL statement in place of the comment.

GIM20501I *command* **PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS** *rc***.**

Explanation:
command SMP/E command
rc the highest return code for the command being processed

If you were running the REJECT command, you will also see the following information:

```
REJECT STATISTICS

  SYSMODS REJECTED         - number  SYSMODS NOT REJECTED - number
  FMIDS DELETED            - number  FMIDS NOT DELETED    - number
  HOLDDATA DELETED         - number
  FEATURE ENTRIES REJECTED - number
  PRODUCT ENTRIES REJECTED - number
```

SYSMODS REJECTED
is the number of SYSMODs that were rejected.

SYSMODS NOT REJECTED
is the number of SYSMODs that were candidates but were not rejected. The reason appears in the REJECT Summary Report.

FMIDS DELETED
is the numbers of FMIDs that were deleted. This includes FMIDs specified on the DELETEFMID operand for NOFMID mode, or FMIDs that were deleted from the GLOBALZONE entry in other REJECT modes when functions were deleted.

FMIDS NOT DELETED
is the number of FMIDs specified on the DELETEFMID operand that were not deleted.

HOLDDATA DELETED
is the number of external HOLDDATA entries that were deleted.

FEATURE ENTRIES REJECTED
is the number of FEATURE entries that were rejected.

PRODUCT ENTRIES REJECTED
is the number of PRODUCT entries that were rejected.

System Action: The return code determines the system action.

Programmer Response: See the return codes for that command in Chapter 2, “SMP/E Return Codes” to determine the success or failure of the command.

If you were running the REJECT command, you also got statistics. These statistics do not require any specific action unless one of the values is not what you expected. For example, if one of the values is a zero and you expected it to be greater than zero, you might want to check the REJECT Summary Report.

GIM20502I SMP/E PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS *rc***. SMP/E IS AT LEVEL** *rr.ss***.**

Explanation:
rc return code for the SMP/E job step.
rr.ss the level of SMP/E, in terms of version, release, and service level. For example, SMP/E 31.*nn* is V3R1.0 service level *nn*.

SMP/E processing completed with the indicated return code. The *rr.ss* variable indicates the level of SMP/E that is installed.

System Action: The return code determines the system action.

Programmer Response: No response is required.

GIM20503I UNKNOWN PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS *rc***.**

Explanation:
rc return code for the processing that was done

SMP/E could not determine which command was being processed.

System Action: The return code determines the system action.

Programmer Response: Make sure that you have spelled the command correctly and have not included extra blanks in the command name. Then rerun the job.

GIM20550W *command* **COMMAND RESULTS MAY BE INCOMPLETE. UNSUPPORTED ENTRIES OR SUBENTRIES WERE FOUND IN** *zonetype* **ZONE** *zonename***. THIS DATA IS SUPPORTED ONLY BY** *system level* **SMP/E, OR HIGHER.**

Explanation:

command	SMP/E command
zonetype	SMP/E zone type
zonename	Zone name
system	z/OS or OS/390
level	Minimum version and release (*vv.rr*) of z/OS or OS/390 SMP/E required to process this zone. **Note:** SMP/E V3R1 is identical to z/OS V1R2 SMP/E.

SMP/E found entries or subentries created by a later level of SMP/E. The current level of SMP/E does not know how to properly handle the new information and therefore command results may be incomplete.

System Action: Processing continues.

Programmer Response: Nothing needs to be done, but you should consider migrating to the specified level of SMP/E.

GIM20551S *command* **COMMAND PROCESSING FAILED. UNSUPPORTED ENTRIES OR SUBENTRIES WERE FOUND IN** *zonetype* **ZONE** *zonename***. THIS DATA IS SUPPORTED ONLY BY** *system level* **SMP/E, OR HIGHER.**

Explanation:

command	SMP/E command
zonetype	SMP/E zone type
zonename	Zone name
system	z/OS or OS/390
level	Minimum version and release (*vv.rr*) of z/OS or OS/390 SMP/E required to process this zone. **Note:** SMP/E V3R1 is identical to z/OS V1R2 SMP/E.

SMP/E found entries or subentries created by a later level of SMP/E. The current level of SMP/E does not know how to properly handle the new information and therefore SMP/E command processing is being failed to prevent possible problems in the zone.

System Action: Command processing ends.

Programmer Response: Re-run the command using the specified SMP/E level or higher.

GIM20601E SYSMOD PROCESSING STOPPED BECAUSE OF THE RETURN CODE ISSUED BY THE EXIT ROUTINE.

Explanation: Because of the return code from an installation exit routine, SYSMOD processing stopped.

System Action: The indicated processing stops.

Programmer Response: Find out why the exit routine ended the request. Check whether the exit routine issued the correct return code for this request.

GIM20601S *command* **PROCESSING STOPPED BECAUSE OF THE RETURN CODE ISSUED BY THE EXIT ROUTINE.**

Explanation:

command	SMP/E command

Because of the return code from an installation exit routine, processing stopped for the current command.

System Action: The indicated processing stops.

Programmer Response: Find out why the exit routine ended the request. Check whether the exit routine issued the correct return code for this request.

GIM20601T SMP/E PROCESSING STOPPED BECAUSE OF THE RETURN CODE ISSUED BY THE EXIT ROUTINE.

Explanation: Because of the return code from an installation exit routine, SMP/E processing stopped.

System Action: The indicated processing stops.

Programmer Response: Find out why the exit routine ended the request. Check whether the exit routine issued the correct return code for this request.

GIM20701S *command* **PROCESSING STOPPED BECAUSE THE EXIT ROUTINE ISSUED AN UNDEFINED RETURN CODE (***rc***).**

Explanation:

command	SMP/E command
rc	return code issued by the exit routine

The installation exit routine issued an undefined return code.

System Action: Processing for the current command stops.

Programmer Response: Check the logic of the exit routine to make sure it returns only defined codes.

GIM20701T SMP/E PROCESSING STOPPED BECAUSE THE EXIT ROUTINE ISSUED AN UNDEFINED RETURN CODE (*rc*).

Explanation:

rc	return code issued by the exit routine

The installation exit routine issued an undefined return code.

System Action: SMP/E processing stops.

Programmer Response: Check the logic of the exit routine to make sure it returns only defined codes.

GIM20800E *command* PROCESSING FAILED FOR SYSMOD *sysmod* BECAUSE THE ++*mcsname* MCS WOULD CAUSE THE OWNERSHIP OF ELEMENT *element* TO CHANGE.

Explanation:

command	APPLY or ACCEPT
sysmod	SYSMOD ID
mcsname	element update MCS
element	element name

A SYSMOD containing an element update (++MACUPD, ++SRCUPD, or ++ZAP) attempted to change the ownership of the element with the ++VER VERSION operand. The ownership can be changed only if the SYSMOD provides a replacement for the element.

System Action: SYSMOD processing stops.

Programmer Response: Contact the packager of the SYSMOD.

GIM20900E SMP/E COULD NOT OBTAIN LINK-EDIT PARAMETERS FOR LOAD MODULE *loadmod* FOR SYSMOD *sysmod*. MEMBER *modname* WAS NOT FOUND IN *dataset* DATA SET *dsname* WHICH WAS ALLOCATED TO DDNAME *ddname*.

Explanation:

loadmod	LMOD being linked
sysmod	SYSMOD causing link-edit
modname	module name
dataset	SMPTLIB or LKLIB
dsname	data set name of SMPTLIB or LKLIB data set specified to contain the module
ddname	ddname to which data set was allocated

SMP/E expected the indicated module to be in the specified LKLIB or SMPTLIB data set, but could not find it there.

System Action: SYSMOD processing stops.

Programmer Response: Determine why the module is not in the indicated data set. Correct the error and rerun the job.

GIM21000E SYSMOD *sysmod* WAS NOT RECEIVED BECAUSE MORE THAN ONE ARCHIVE FILE WAS FOUND AS A MATCH FOR RELFILE *relfile* IN THE package PACKAGE.

Explanation:

sysmod	SYSMOD ID
relfile	RELFILE name
package	package id

An error occurred when the SMPTLIB data sets were being loaded for the indicated SYSMOD. The expected relfile name was found more than once in the GIMZIP package during either RECEIVE FROMNETWORK or RECEIVE FROMNTS command processing. It is possible that the packager created more than one dataset name for the same relfile with different high level qualifiers. SMP/E expects the last two parts of the data set name for a relative file to be ″sysmod-id.Fnnnn″. If the data set name contains high level qualifiers, SMP/E expects the data set name ″rfdsnpfx.sysmod_id.Fnnnn″ to exist in the GIMZIP package, where:

rfdsnpfx	is the prefix supplied by the RFPREPIX operand on the header MCS statement that is being processed. This part of the data set name is optional. If no RFDSNPFX value was specified, this prefix is not included in the data set name allocated by SMP/E.
sysmod_id	is the SYSMOD ID of the SYSMOD to which the file belongs. This part of the data set is required.
F*nnnn*	is the letter F followed by the number specified on the RELFILE operand of the corresponding MCS statement in the SYSMOD. This part of the data set name is required.

The relfile name was found more than once in the GIMZIP package. SMP/E will search the relfile names in the package for the the required qualifiers (either ″sysmod_id.Fnnnn″ or ″rfdsnpfx.sysmod_id.Fnnnn″). The qualifiers must be immediately before the trailing qualifiers ″pax.Z″, which are always present in the relfile names of the GIMZIP package. Qualifiers before the required qualifiers will be ignored.

For example, if GIMZIP processing encounters these two relfiles:

1. ’SERVERPAC.IBM.SYSMOD1.F1.pax.Z’
2. ’SERVERPAC.TWO.IBM.SYSMOD1.F1.pax.Z’

GIMZIP will ignore the "SERVERPAC." and "SERVERPAC.TWO." qualifiers on these data sets and therefore consider them to be duplicate data sets named "IBM.SYSMOD1.F1".

System Action: SMP/E does not receive the SYSMOD. SMP/E deletes the SMPTLIB data sets associated with the SYSMOD.

Programmer Response: The package is incorrect. Contact the provider of the package and inform them of the problem.

GIM21100E SYSMOD *sysmod* CONTAINS MORE THAN ONE ++RENAME STATEMENT WITH LMOD *loadmod*. THIS IS NOT ALLOWED.

Explanation:

sysmod	SYSMOD ID
loadmod	LMOD name

The indicated SYSMOD contains more than one ++RENAME statement that specifies the same LMOD as either the current name or the new name (TONAME) of the load module. Here is an example, with two ++RENAME statements specifying LMODB:

```
++RENAME (LMODA) TONAME(LMODB).
++RENAME (LMODB) TONAME(LMODC).
```

System Action: SYSMOD processing stops.

Programmer Response: Fix the SYSMOD so that it does not contain more than one ++RENAME statement with any given LMOD name. Then rerun the job.

GIM21200E THE NAME OF THE RELATED ZONE CANNOT BE GLOBAL.

Explanation: The RELATED subentry for the target or distribution zone was set to GLOBAL. This is invalid, because the name GLOBAL is reserved for the global zone.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Change the value of the RELATED zone name from GLOBAL to the name of a target or distribution zone. Then resubmit the job.

GIM21200S THE NAME OF THE RELATED ZONE CANNOT BE GLOBAL.

Explanation: The RELATED operand for the ZONECOPY, ZONEIMPORT, or ZONERENAME command was set to GLOBAL. This is invalid, because the name GLOBAL is reserved for the global zone.

System Action: Command processing stops.

Programmer Response: Change the value of the RELATED zone name from GLOBAL to the name of a target or distribution zone. Then resubmit the job.

GIM21300W SYSMOD *sysmod1* CONTAINS MORE THAN ONE ++HOLD MCS FOR REASON ID *reasonid* WITH SYSMOD ID *sysmod2*. ONLY THE HOLDDATA FROM THE LAST ++HOLD WITH THIS REASON ID AND SYSMOD ID IS SAVED.

Explanation:

sysmod1	SYSMOD ID
reasonid	system HOLD reason ID
sysmod2	SYSMOD ID

During RECEIVE processing, SMP/E encountered a SYSMOD containing inline ++HOLD statements. Two or more of these HOLDs were for the same reason ID and specified the same originating SYSMOD ID but had different comments. As SMP/E processes such inline ++HOLD statements, it overlays the previous one in the CSI. As a result, the CSI reflects only the last inline ++HOLD statement that was processed for the SYSMOD and does not indicate all the actions needed to resolve all the system HOLDs that were contained in the SYSMOD.

System Action: Processing continues.

Programmer Response: To see the inline ++HOLD statements that were overlaid, list the MCS entry for the SYSMOD. The output from this LIST command will help you determine what additional actions you need to take to resolve all the inline HOLDs for the SYSMOD.

GIM21401E *enttype* ENTRY *entname* COULD NOT BE STORED IN THE *library* LIBRARY.

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library being processed

SMP/E cannot store the directory entry for this entry. A previous message in SMPOUT shows the reason.

System Action: SYSMOD processing stops.

Programmer Response: Find the cause of the error from the previous messages. Correct the error and rerun the job.

GIM21402W *enttype* ENTRY *entname* COULD NOT BE STORED IN THE *zonename* ZONE.

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

SMP/E cannot store zone information for this entry. A previous message in SMPOUT shows the reason.

System Action: Processing continues.

Programmer Response: Find the cause of the error from the previous messages. If necessary, correct the error and rerun the job.

GIM21402E *enttype* **ENTRY** *entname* **COULD NOT BE STORED IN THE** *zonename* **ZONE.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

SMP/E cannot store zone information for this entry. A previous message in SMPOUT shows the reason.

System Action: SYSMOD processing stops.

Programmer Response: Find the cause of the error from the previous messages. Correct the error and rerun the job.

GIM21402S *enttype* **ENTRY** *entname* **COULD NOT BE STORED IN THE** *zonename* **ZONE.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

SMP/E cannot store zone information for this entry. A previous message in SMPOUT shows the reason.

System Action: Command processing stops.

Programmer Response: Find the cause of the error from the previous messages. Correct the error and rerun the job.

GIM21402T *enttype* **ENTRY** *entname* **COULD NOT BE STORED IN THE** *zonename* **ZONE.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

SMP/E cannot store zone information for this entry. A previous message in SMPOUT shows the reason.

System Action: SMP/E processing stops.

Programmer Response: Find the cause of the error from the previous messages. Correct the error and rerun the job.

GIM21500W **PREALLOCATED SMPTLIB DATA SET** *smptlib* **HAS A DSNTYPE OF** *dsntype1***. THIS DIFFERS FROM THE DSNTYPE OF** *dsntype2* **FOR THE CORRESPONDING RELFILE DATA SET** *relfile***.**

Explanation:

smptlib	the SMPTLIB data set name
dsntype1	the DSNTYPE (LIBRARY or PDS) of the SMPTLIB data set
dsntype2	the DSNTYPE (LIBRARY or PDS) of the RELFILE data set
relfile	the RELFILE data set name

During RECEIVE processing, SMP/E found a preallocated SMPTLIB data set. However, the DSNTYPE of the preallocated data set does not match the DSNTYPE of its associated RELFILE data set.

System Action: RECEIVE processing continues but if the RELFILE data set contains load modules or program objects, errors may occur when SMP/E tries to copy the RELFILE data set to the SMPTLIB data set.

Programmer Response: If an error occurred during the copy to this SMPTLIB, do **one** of the following:

- Reallocate the SMPTLIB data set with the correct DSNTYPE.
- Delete the SMPTLIB data set and allow SMP/E to allocate it with the appropriate DSNTYPE.

Then rerun the RECEIVE command.

GIM21601E **UPDATE PROCESSING FOR SYSMOD** *sysmod* **FAILED FOR MEMBER** *member* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *number***.**

Explanation:

sysmod	SYSMOD ID
member	element name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
number	utility sequence number

Update utility processing failed for the indicated element, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: SYSMOD processing stops if:

- The return code is greater than the return code you specified.
- The return code is greater than the default return code.

Programmer Response:

- Look at the output to find the cause of the error. You can use the utility sequence number as an index into the SYSPRINT output to find the error. The number is incremented for each utility call in an SMP/E run.

Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:

- The PRINT subentry specifies a DDDEF of DUMMY.
- The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

- If IEBUPDTE error message **MEMBER NAME NOT FOUND** was also issued, be sure that the element exists in the distribution or target library (DISTLIB or SYSLIB) shown in the element entry or the element MCS.

Correct the error and rerun the job.

GIM21602I UPDATE PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MEMBER** *member* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
member	element name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Update utility processing was successful for the indicated element, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues, as shown by the messages that follow in SMPOUT.

Programmer Response: None.

GIM21602W UPDATE PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MEMBER** *member* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
member	element name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Update utility processing was successful for the indicated element, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues, as shown by the messages that follow in SMPOUT.

Programmer Response: None.

GIM21603E UPDATE PROCESSING FOR SYSMOD *sysmod* **FAILED FOR MEMBER** *member* **IN THE** *library* **LIBRARY. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
member	member (element) name
library	ddname of the library
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Update processing failed for the indicated element (member).

System Action: SYSMOD processing stops for the indicated SYSMOD.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM21700W SMP/E COULD NOT DETERMINE THE DSNTYPE OF RELFILE DATA SET *relfile***.**

Explanation:

relfile	the RELFILE data set name

During RECEIVE processing, SMP/E could not determine the DSNTYPE of the indicated RELFILE data set. SMP/E uses the DSNTYPE of the RELFILE data set when allocating its corresponding SMPTLIB data set. Because the DSNTYPE of the RELFILE data set is unknown, SMP/E uses the DSNTYPE specified in the SMPTLIB DDDEF, if one is specified, when allocating the SMPTLIB data set. If a DSNTYPE is not specified on the SMPTLIB DDDEF, SMP/E does not specify a DSNTYPE for the SMPTLIB allocation.

System Action: RECEIVE processing continues but if the RELFILE data set contains load modules or program objects, and the DSNTYPEs of the RELFILE data set and the SMPTLIB data set do not match, errors may occur when SMP/E tries to copy the RELFILE data set to the SMPTLIB data set.

Programmer Response: If an error occurred during the copy to this SMPTLIB, preallocate the SMPTLIB data set with the correct DSNTYPE and rerun the RECEIVE command.

GIM21801I *enttype* **ENTRY** *entname* **WAS STORED IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library being processed

SMP/E successfully stored or restored the entry in the indicated library.

System Action: SYSMOD processing continues.

Programmer Response: None.

GIM21802I *enttype* **ENTRY** *entname* **WAS STORED IN THE** *zonename* **ZONE.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being updated

SMP/E successfully stored or restored the entry in the indicated zone.

System Action: SYSMOD processing continues.

Programmer Response: None.

GIM21803I **MEMBER** *member* **WAS STORED IN THE** *library* **LIBRARY.**

Explanation:

member	member name
library	ddname of the library being processed

SMP/E successfully stored the member in the indicated library.

System Action: SYSMOD processing continues.

Programmer Response: None.

GIM21901W *enttype* **ENTRY** *entname* **IN THE** *library* **LIBRARY WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library being processed

SMP/E could not create, update, or list the indicated entry because the SYSMOD entry needed a PEMAX value greater than the value in the current OPTIONS entry.

Note: This message may indicate that the entry has incorrect data. This may have been caused by an internal SMP/E error.

System Action: Processing stops for the indicated entry.

Programmer Response: Do **one** of the following:

- Use UCLIN to increase the PEMAX value in the OPTIONS entry.

  Note: Do not decrease the PEMAX value after SMP/E processes SYSMODs with a larger PEMAX, because existing SYSMOD entries may be too large for SMP/E to process.
- Make sure the OPTIONS entry containing the increased PEMAX value is the same as the OPTIONS entry named in the zone definition entry.
- If the entry has incorrect data, either correct the data or restore the data set and rerun the job.

GIM21901E *enttype* **ENTRY** *entname* **IN THE** *library* **LIBRARY WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library being processed

SMP/E could not create, update, or list the indicated entry because the SYSMOD entry needed a PEMAX value greater than the value in the current OPTIONS entry.

Note: This message may indicate that the entry has incorrect data. This may have been caused by an internal SMP/E error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Use UCLIN to increase the PEMAX value in the OPTIONS entry.

  Note: Do not decrease the PEMAX value after SMP/E processes SYSMODs with a larger PEMAX, because existing SYSMOD entries may be too large for SMP/E to process.

- Make sure the OPTIONS entry containing the increased PEMAX value is the same as the OPTIONS entry named in the zone definition entry.
- If the entry has incorrect data, either correct the data or restore the data set and rerun the job.

GIM21901S *enttype* **ENTRY** *entname* **IN THE** *library* **LIBRARY WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library being processed

SMP/E could not create, update, or list the indicated entry because the SYSMOD entry needed a PEMAX value greater than the value in the current OPTIONS entry.

Note: This message may indicate that the entry has incorrect data. This may have been caused by an internal SMP/E error.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:

- Use UCLIN to increase the PEMAX value in the OPTIONS entry.

  Note: Do not decrease the PEMAX value after SMP/E processes SYSMODs with a larger PEMAX, because existing SYSMOD entries may be too large for SMP/E to process.
- Make sure the OPTIONS entry containing the increased PEMAX value is the same as the OPTIONS entry named in the zone definition entry.
- If the entry has incorrect data, either correct the data or restore the data set and rerun the job.

GIM21902W *enttype* **ENTRY** *entname* **IN THE** *zonename* **ZONE WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

SMP/E could not create, update, list, or process the indicated entry, because the SYSMOD entry needs a PEMAX value greater than the value in the current OPTIONS entry.

Notes:

1. This message may indicate that the entry has incorrect data. This may have been caused by an internal SMP/E error.
2. During APPLY or ACCEPT processing, this message may be issued when SMP/E is building a candidate list for mass-mode processing. In this case, the entry name and entry type values in the message may refer to SYSMODs that are eventually eliminated during SYSMOD selection processing. The PEMAX value in the OPTIONS entry being used needs to be increased.

System Action: Processing stops for the indicated entry.

Programmer Response: Do **one** of the following, as appropriate:

- Use UCLIN to increase the PEMAX value in the OPTIONS entry.

  Note: Do not decrease the PEMAX value after SMP/E processes SYSMODs with a larger PEMAX, because existing SYSMOD entries may be too large for SMP/E to process.
- Make sure the OPTIONS entry containing the increased PEMAX value is the same as the OPTIONS entry named in the zone definition entry.
- If the entry has incorrect data, either correct the data or restore the data set and rerun the job.

GIM21902E *enttype* **ENTRY** *entname* **IN THE** *zonename* **ZONE WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

SMP/E could not create, update, list, or process the indicated entry, because the SYSMOD entry needs a PEMAX value greater than the value in the current OPTIONS entry.

Notes:

1. This message may indicate that the entry has incorrect data. This may have been caused by an internal SMP/E error.
2. During APPLY or ACCEPT processing, this message may be issued when SMP/E is building a candidate list for mass-mode processing. In this case, the entry name and entry type values in the message may refer to SYSMODs that are eventually eliminated during SYSMOD selection processing. The PEMAX value in the OPTIONS entry being used needs to be increased.

3. During RESTORE processing, this message may be issued if prior SYSMODs were applied using a higher PEMAX value. RESTORE processing checks to see whether the candidate SYSMOD has a dependency on any other SYSMOD not yet accepted.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following, as appropriate:

- Use UCLIN to increase the PEMAX value in the OPTIONS entry.

  Note: Do not decrease the PEMAX value after SMP/E processes SYSMODs with a larger PEMAX, because existing SYSMOD entries may be too large for SMP/E to process.
- Make sure the OPTIONS entry containing the increased PEMAX value is the same as the OPTIONS entry named in the zone definition entry.
- If the entry has incorrect data, either correct the data or restore the data set and rerun the job.

GIM21902S *enttype* **ENTRY** *entname* **IN THE** *zonename* **ZONE WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

SMP/E could not create, update, list, or process the indicated entry, because the SYSMOD entry needs a PEMAX value greater than the value in the current OPTIONS entry.

Notes:

1. This message may indicate that the entry has incorrect data. This may have been caused by an internal SMP/E error.
2. During APPLY or ACCEPT processing, this message may be issued when SMP/E is building a candidate list for mass-mode processing. In this case, the entry name and entry type values in the message may refer to SYSMODs that are eventually eliminated during SYSMOD selection processing. The PEMAX value in the OPTIONS entry being used needs to be increased.

System Action: Command processing stops.

Programmer Response: Do **one** of the following, as appropriate:

- Use UCLIN to increase the PEMAX value in the OPTIONS entry.

  Note: Do not decrease the PEMAX value after SMP/E processes SYSMODs with a larger PEMAX, because existing SYSMOD entries may be too large for SMP/E to process.
- Make sure the OPTIONS entry containing the increased PEMAX value is the same as the OPTIONS entry named in the zone definition entry.
- If the entry has incorrect data, either correct the data or restore the data set and rerun the job.

GIM21903E **ENTRY** *entname* **IN THE** *zonename* **ZONE WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.**

Explanation:

entname	entry name
zonename	name of the zone being processed

SMP/E could not create, update, or list the indicated entry because the SYSMOD entry needed a PEMAX value greater than the value in the current OPTIONS entry. (SMP/E could not determine the entry type.)

Note: This message may indicate that the entry has incorrect data. This may have been caused by an internal SMP/E error.

System Action: Depending on the return code, processing stops for the SYSMOD or the command.

Programmer Response: Do **one** of the following:

- Use UCLIN to increase the PEMAX value in the OPTIONS entry.

  Note: Do not decrease the PEMAX value after SMP/E processes SYSMODs with a larger PEMAX, because existing SYSMOD entries may be too large for SMP/E to process.
- Make sure the OPTIONS entry containing the increased PEMAX value is the same as the OPTIONS entry named in the zone definition entry.
- If the entry has incorrect data, either correct the data or restore the data set and rerun the job.

GIM21904E **ENTRY** *entname* **IN THE** *library* **LIBRARY WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.**

Explanation:

entname	entry name
library	ddname of the library being processed

SMP/E could not create, update, or list the indicated entry because the SYSMOD entry needed a PEMAX value greater than the value in the current OPTIONS

entry. (Because of an internal SMP/E error, SMP/E could not determine the entry type.)

Note: This message may indicate that the entry has incorrect data. This may have been caused by an internal SMP/E error.

System Action: Depending on the return code, processing stops for the SYSMOD or the command.

Programmer Response: Do **one** of the following:

- Use UCLIN to increase the PEMAX value in the OPTIONS entry.

  Note: Do not decrease the PEMAX value after SMP/E processes SYSMODs with a larger PEMAX, because existing SYSMOD entries may be too large for SMP/E to process.
- Make sure the OPTIONS entry containing the increased PEMAX value is the same as the OPTIONS entry named in the zone definition entry.
- If the entry has incorrect data, either correct the data or restore the data set and rerun the job.

GIM22000I AN ERROR MAY OCCUR WHEN RELFILE DATA SET *relfile* IS COPIED TO THE SMPTLIB DATA SET *smptlib*.

Explanation:

relfile	the RELFILE data set name
smptlib	the SMPTLIB data set name

This message follows warning message GIM21500W, GIM21700W, or GIM22500W. During RECEIVE processing, one of the following conditions occurred:

- SMP/E found a preallocated SMPTLIB data set. However, the DSNTYPE of the preallocated data set does not match the DSNTYPE of its associated RELFILE data set.
- SMP/E could not determine the DSNTYPE of the RELFILE data set.
- SMP/E determined that the DSNTYPE of the RELFILE was LIBRARY, but the system does not support the LIBRARY value for DSNTYPE.

System Action: RECEIVE processing continues but if the RELFILE data set contains load modules or program objects, and the DSNTYPE of the SMPTLIB data set does not match the DSNTYPE of the RELFILE data set, errors may occur when SMP/E tries to copy the RELFILE data set to the SMPTLIB data set.

Programmer Response: Refer to programmer response for the preceding GIM21500W, GIM21700W, or GIM22500W message.

GIM22100E THE NAME OF THE TARGET OR DISTRIBUTION ZONE CANNOT BE GLOBAL.

Explanation: The zone name specified for ZONEINDEX was set to GLOBAL. This is invalid, because the name GLOBAL is reserved for the global zone.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Change the value of the target or distribution zone name to something other than GLOBAL. Then resubmit the job.

GIM22200E SYSMOD *sysmod* WAS NOT RECEIVED BECAUSE AN ARCHIVE FILE FOR RELFILE *relfile* WAS NOT FOUND IN THE *package* PACKAGE.

Explanation:

sysmod	SYSMOD ID
relfile	RELFILE name
package	package id

An error occurred when the SMPTLIB data sets were being loaded for the indicated SYSMOD. The expected relfile name was not found in the GIMZIP package during either RECEIVE FROMNETWORK or RECEIVE FROMNTS command processing. It is possible that the packager gave an incorrect dataset name for relfile. SMP/E expects the last two parts of the data set name for a relative file to be ″sysmod-id.Fnnnn″. If the data set name contains high level qualifiers, SMP/E expects to find in the GIMZIP package the data set name ″rfdsnpfx.sysmod_id.Fnnnn″, where:

rfdsnpfx	is the prefix supplied by the RFPREPFX operand on the header MCS statement that is being processed. This part of the data set name is optional. If no RFDSNPFX value was specified, this prefix is not included in the data set name allocated by SMP/E.
sysmod_id	is the SYSMOD ID of the SYSMOD to which the file belongs. This part of the data set is required.
F*nnnn*	is the letter F followed by the number specified on the RELFILE operand of the corresponding MCS statement in the SYSMOD. This part data set name is required.

The relfile name was not found in the GIMZIP package. SMP/E will search the relfile names in the package for the the required qualifiers (either ″sysmod_id.Fnnnn″ or ″rfdsnpfx.sysmod_id.Fnnnn″). The qualifiers must be immediately before the trailing qualifiers ″pax.Z″, which are always present in the relfile names of the GIMZIP

package. Qualifiers before the required qualifiers will be ignored. This message is issued if a match is not found. The data set name may be missing, or improperly named.

For example, suppose that the expected relfile name is ″IBM.SYSMOD1.F1″. In this case, SMP/E would consider ″SERVERPAC.IBM.SYSMOD1.F1.pax.Z″ to be a match for the expected relfile name (the ″SERVERPAC.″ qualifier is ignored), whereas SMP/E would **not** consider ″SERVERPAC.POKIBM.SYSMOD1.F1.pax.Z″ to be a match for the expected relfile name (because ″POKIBM″ does not match ″IBM″ and is part of a required qualifier).

System Action: SMP/E does not receive the SYSMOD, and RECEIVE processing for the SYSMOD stops. The SMPTLIBs that have already been allocated are deleted.

Programmer Response: The RELFILE name may have been accidentally deleted, never included, or improperly named by the packager. Notify the provider of the package that there is a problem.

GIM22300E JCLIN PROCESSING FAILED BECAUSE THE JCL STATEMENT DID NOT BEGIN IN COLUMN 1.

Explanation: A JCL statement did not start in column 1 when it should have.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM22401I *enttype entname* **WAS DELETED FROM THE** *library* **LIBRARY.**

Explanation:

enttype	element type
entname	element name
library	ddname of the library being processed

SMP/E successfully deleted the element from the indicated library.

System Action: Processing continues.

Programmer Response: None.

GIM22402I *enttype1 entname1* **(ALIAS OF** *enttype2 entname2***) WAS DELETED FROM THE** *library* **LIBRARY.**

Explanation:

enttype1	entry type of the alias
entname2	entry name of the alias
enttype2	entry type of the element
entname2	entry name of the element
library	ddname of the library being processed

SMP/E successfully deleted the element from the indicated library. Note that if an alias is being deleted and the alias value is greater than 300 characters in length, only the first 300 characters are placed in the message.

System Action: Processing continues.

Programmer Response: None.

GIM22403I MEMBER *member* **WAS DELETED FROM THE** *library* **LIBRARY.**

Explanation:

member	member name
library	ddname of the library being processed

SMP/E successfully deleted the member from the indicated library.

System Action: Processing continues.

Programmer Response: None.

GIM22404I *enttype* **ENTRY** *entname* **WAS DELETED FROM THE** *zonename* **ZONE.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

SMP/E successfully deleted the entry from the indicated zone.

System Action: Processing continues.

Programmer Response: None.

GIM22405I FILE *filename* **WAS REMOVED FROM THE** *library* **LIBRARY.**

Explanation:

filename	file name
library	ddname of the library being processed

SMP/E successfully removed the file from the indicated library.

System Action: Processing continues.

Programmer Response: None.

GIM22406I FILE *filename* **(LINK FOR** *enttype entname***) WAS REMOVED FROM THE** *library* **LIBRARY.**

Explanation:

filename	name of link
enttype	entry type of element
entname	entry name of element
library	ddname of library being processed

SMP/E successfully removed the file from the indicated library. Note that if the file name is greater than 300 characters in length, only the first 300 characters are placed in the message.

System Action: Processing continues.

Programmer Response: None.

GIM22407I MEMBER *member* (SIDE DECK FOR LMOD *loadmod*) WAS DELETED FROM THE *library* LIBRARY.

Explanation:

member	member name
loadmod	load module name
library	ddname of the library

SMP/E successfully deleted the member from the indicated library.

System Action: Processing continues.

Programmer Response: None.

GIM22408I FILE *filename* (SYMBOLIC LINK FOR *enttype entname*) WAS REMOVED FROM THE *library* LIBRARY.

Explanation:

filename	name of symbolic link
enttype	entry type of element
entname	entry name of element
library	ddname of library being processed

SMP/E successfully removed the file from the indicated library. Note that if the length of the file name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: Processing continues.

Programmer Response: None.

GIM22500W SMP/E CANNOT ALLOCATE SMPTLIB DATA SET *smptlib* WITH A DSNTYPE OF LIBRARY.

Explanation:

smptlib	the SMPTLIB data set name

During RECEIVE processing, SMP/E determined that the SMPTLIB data set should be allocated with a DSNTYPE of LIBRARY. However, the system does not have DSNTYPE support.

System Action: SMP/E attempts to allocate the SMPTLIB data set as a PDS. RECEIVE processing continues but if the RELFILE contains load modules or program objects, errors may occur when SMP/E tries to copy the RELFILE to the SMPTLIB data set.

Programmer Response: If an error occurred during the copy to this SMPTLIB, install this product from a system that supports the LIBRARY DSNTYPE.

GIM22601I *command* PROCESSING FAILED FOR SYSMOD *sysmod*.

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

A preceding message describes the reason for the failure. SMP/E found the error for one SYSMOD only. SMP/E will continue to process other SYSMODs.

Note: You may find additional information in the SMPLOG data set.

System Action: Processing continues.

Programmer Response: Check previous messages to determine the cause of error.

GIM22601E *command* PROCESSING FAILED FOR SYSMOD *sysmod*.

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

A preceding message describes the reason for the failure. SMP/E found the error for one SYSMOD only. SMP/E will continue to process other SYSMODs.

Note: You may find additional information in the SMPLOG data set.

System Action: SYSMOD processing stops.

Programmer Response: Do the following:
1. Check previous messages to determine the cause of error.
2. Correct the error and rerun the job.

GIM22602E *command* PROCESSING WAS INCOMPLETE FOR SYSMOD *sysmod*.

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

A preceding message describes the reason for processing to be incomplete. SMP/E found the error for this SYSMOD only. SMP/E continues to process other SYSMODs.

Note: You may find additional information in the SMPLOG data set.

System Action: Processing stops for the SYSMOD or the command.

Programmer Response: Do the following:
1. Check previous messages to determine the cause of error.

2. Check the error and rerun the job.

GIM22603E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. SYSMOD** *sysmod* **WOULD HAVE UPDATED THE** *library* **LIBRARY, BUT THE** *library* **LIBRARY IS OUT OF SPACE.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID
library	ddname of the library

Processing for the indicated SYSMOD failed because the SYSMOD has unprocessed elements for the indicated library, which ran out of space.

System Action: SYSMOD processing stops for the indicated SYSMOD.

Programmer Response: Increase the size of the out-of-space library and rerun the command for the indicated SYSMOD.

GIM22701I *command* **PROCESSING WAS SUCCESSFUL FOR SYSMOD** *sysmod***.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

Command processing was successful for SYSMOD *sysmod*.

System Action: None.

Programmer Response: None.

GIM22800S **REPLY FROM THE FTP SERVER AT** *location* **WAS NOT RECEIVED.**

Explanation:

location	Host name or IP address specified in SERVER data set

SMP/E processing timed out before receiving a reply from the indicated FTP server. The server may not be operational or the connection may have been lost.

System Action: RECEIVE FROMNETWORK processing terminates.

Programmer Response: Verify that the FTP server on the indicated host is operational and rerun the RECEIVE.

GIM22901S **THE COMMAND WAS NOT PROCESSED BECAUSE OF A PREVIOUS ERROR.**

Explanation: SMP/E did not process a command because of an error described in a previous message. SMP/E did, however, check the command for syntax errors.

System Action: Processing continues with the next command.

Programmer Response: Correct the error in the command and rerun the job.

GIM23101E **THERE IS A SYNTAX ERROR IN A ZAP CONTROL STATEMENT FOR MODULE** *modname* **IN SYSMOD** *sysmod***.**

Explanation:

modname	module name
sysmod	SYSMOD ID

SMP/E detected a syntax error in the superzap statement for the module in the indicated SYSMOD. The statement with the error may be in the line before this message.

System Action: If the return code is nonzero, SMP/E processing stops. Otherwise, processing continues.

Programmer Response: Correct the syntax error and rerun the job.

GIM23300W **ALIAS** *alias* **FOR** *enttype entname* **WAS NOT INSTALLED INTO THE LIBRARY** *library* **BECAUSE THE LIBRARY IS SEQUENTIAL.**

Explanation:

alias	alias name for a data element
enttype	data element type
entname	data element name
library	library ddname

The indicated alias for the specified data element can not be installed in the library because the library has a sequential data set organization.

System Action: Processing continues.

Programmer Response: Verify that the data element's destination library should be sequential and that the alias need not exist in the library. If the library was incorrectly allocated with a sequential data set organization, reallocate the library and rerun the job. If the command is APPLY or ACCEPT, rerun the job with the REDO operand.

GIM23401T **PROGRAM** *progname* **WAS REQUIRED FOR SMP/E PROCESSING BUT WAS NOT AVAILABLE.**

Explanation:

progname	program name

SMP/E could not find the indicated program, which it needs for processing.

System Action: Step processing stops. SMP/E

processing also stops if you specified an incorrect utility name in the global zone UTILITY entry being used.

Programmer Response: Do **one** of the following:

- If the program is available on your system, but not contained within the link list or link pack area, add a JOBLIB or STEPLIB for the library containing the program, and rerun the job.
- If the identified program is a utility, check that you have specified the utility name correctly in the global zone UTILITY entry being used. Correct it and rerun the job.

GIM23402T A BLDL FAILED FOR PROGRAM *progname* **THAT IS REQUIRED FOR SMP/E PROCESSING. THE RETURN CODE WAS** *rc***.**

Explanation:

progname	program name
rc	return code

SMP/E issued a BLDL macro instruction for program *progname* that failed with return code *rc*.

System Action: Step processing and SMP/E processing stops.

Programmer Response: Refer to the BLDL macro instruction documentation for an explanation of the return code.

GIM23403T TOKEN SERVICES ARE REQUIRED FOR GIMAPI PROCESSING BUT ARE NOT AVAILABLE.

Explanation: During GIMAPI processing, the following token services are used:

- IEANTCR
- IEANTDL
- IEANTRT

One of these services could not be loaded.

System Action: GIMAPI processing stops.

Programmer Response: Execute the GIMAPI program on a system that supports token services.

GIM23500I LIBRARY *library* **WAS NOT COMPRESSED BECAUSE IT IS A SEQUENTIAL DATA SET.**

Explanation:

library	ddname of library being processed

The indicated library was specified in the COMPRESS operand, but SMP/E does not compress sequential libraries.

System Action: Processing continues.

Programmer Response: Do not specify the indicated library in the COMPRESS list.

GIM23600E *command* **PROCESSING TO THE** *library* **LIBRARY FAILED FOR** *enttype entname* **IN SYSMOD** *sysmod***. DATE** *yy.ddd* **— TIME** *hh:mm:ss*

Explanation:

command	I/O or STOW
library	ddname of the library to which the element was to be installed
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

SMP/E command processing failed for the indicated element.

System Action: SMP/E does not store the member. Messages that follow this message show SMP/E's action.

Programmer Response: For I/O errors, fix the error and rerun the job. For STOW errors, increase the directory block allocation for the indicated library and rerun the job.

GIM23700E SMP/E PROCESSING TO THE *library* **LIBRARY FAILED FOR** *enttype entname* **IN SYSMOD** *sysmod***. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **- TIME** *hh:mm:ss*

Explanation:

library	ddname of the library to which the element was to be installed.
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

SMP/E command processing failed for the indicated element.

System Action: SYSMOD processing stops.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM23701E ZAP VER PROCESSING FOR SYSMOD *sysmod* FAILED FOR MODULE *modname* IN LMOD *loadmod* IN THE *library* LIBRARY. THE RETURN CODE (*rc*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *seqno*.

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

ZAP utility processing failed for the indicated module, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: SYSMOD processing stops if:
- The return code is greater than the return code you specified.
- The return code is greater than the default return code.

Programmer Response:

1. Make sure the default utility return code or the return code you specified in the UTILITY entry is the one you want to use. Make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. Check the output from the ZAP utility to find the cause of the error. The sequence number may be used as an index into SYSPRINT output to find the error. The sequence number is incremented for each utility call in an SMP/E run.

   Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:
   - The PRINT subentry specifies a DDDEF of DUMMY.
   - The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.
3. Correct the error and rerun the job.

GIM23702I ZAP VER PROCESSING FOR SYSMOD *sysmod* WAS SUCCESSFUL FOR MODULE *modname* IN LMOD *loadmod* IN THE *library* LIBRARY. THE RETURN CODE WAS *rc*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *seqno*.

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

ZAP utility processing was successful for the indicated module, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Programmer Response: None.

GIM23702W ZAP VER PROCESSING FOR SYSMOD *sysmod* WAS SUCCESSFUL FOR MODULE *modname* IN LMOD *loadmod* IN THE *library* LIBRARY. THE RETURN CODE WAS *rc*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *seqno*.

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Because the return code from zap utility processing did not exceed the maximum acceptable return code, zap processing was considered successful for the indicated module, which was part of SYSMOD *sysmod*. However, because the return code from zap processing was not zero, zap processing may not have produced results that you consider acceptable. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Programmer Response: Check the zap SYSPRINT

output to determine whether the results are what you consider acceptable.

- If the results are acceptable, no further action is necessary.
- If the results are not acceptable, take the post-installation steps needed to achieve acceptable results.

GIM23703E ZAP REP PROCESSING FOR SYSMOD *sysmod* **FAILED FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE (***rc***) EXCEEDED THE ALLOWABLE VALUE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

ZAP utility processing failed for the indicated module, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: SYSMOD processing stops if:

- The return code is greater than the return code you specified.
- The return code is greater than the default return code.

Programmer Response:

1. Make sure the default utility return code or the return code you specified in the UTILITY entry is the one you want to use. Make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. Check the output from the ZAP utility to find the cause of the error. The sequence number may be used as an index into SYSPRINT output to find the error. The sequence number is incremented for each utility call in an SMP/E run.

   Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:

   - The PRINT subentry specifies a DDDEF of DUMMY.
   - The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.
3. Correct the error and rerun the job.

GIM23704I ZAP REP PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Zap utility processing was successful for the indicated module, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Programmer Response: None.

GIM23704W ZAP REP PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Because the return code from zap utility processing did not exceed the maximum acceptable return code, zap processing was considered successful for the indicated module, which was part of SYSMOD *sysmod*. However, because the return code from zap processing was not zero, zap processing may not have produced results

that you consider acceptable. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Programmer Response: Check the zap SYSPRINT output to determine whether the results are what you consider acceptable.

- If the results are acceptable, no further action is necessary.
- If the results are not acceptable, take the post-installation steps needed to achieve acceptable results.

GIM23800E SMP/E PROCESSING FROM THE *library* **LIBRARY TO THE** *library* **LIBRARY FAILED FOR** *enttype entname* **IN SYSMOD** *sysmod***. THERE IS AN INCONSISTENCY IN THE DATA ATTRIBUTES. REASON CODE IS** *rc***. DATE** *yy.ddd* **— TIME** *hh:mm:ss*

Explanation:

library	ddname of the library from which the data was to be copied
library	ddname of the library to which the data was to be installed
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
rc	reason for the failure
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

SMP/E command processing failed for the indicated element. The reason codes and their meanings are:

01 A data set with a record format of VS or VBS must be sequential.

02 The input and output data sets must have either a partitioned or sequential organization.

03 The record format of the input data set is not supported by SMP/E. The supported formats are F, FB, FS, FBS, FA, FBA, FM, FBM, V, VB, VA, VBA, VM, VBM, VS, and VBS.

04 A data set with record format VS must have a record length at least as large as that of the data set from which it is copying the element.

05 A data set with record format VBS must have a block size equal to that of the data set from which it is copying the element.

06 The record format of the output data set is not compatible with the record format of the input data set. Compatible record formats are:

INPUT	OUTPUT
F, FB	F, FB, FS, FBS, FA, FBA, FM, FBM, V, VB, VA, VBA, VM, VBM
FS, FBS	F, FB, FS, FBS, FA, FBA, FM, FBM, V, VB, VA, VBA, VM, VBM
FA, FBA	F, FB, FS, FBS, FA, FBA, V, VB, VA, VBA
FM, FBM	F, FB, FS, FBS, FM, FBM, V, VB, VM, VBM
V, VB	F, FB, FS, FBS, FA, FBA, FM, FBM, V, VB, VA, VBA, VM, VBM
VA, VBA	F, FB, FS, FBS, FA, FBA, V, VB, VA, VBA
VM, VBM	F, FB, FS, FBS, FM, FBM, V, VB, VM, VBM
VS	VS (sequential)
VBS	VBS (sequential)

07 The record length and block size of the library are not compatible. If the data set is fixed format, the block size must be evenly divisible by the record length. If the data set is variable format, the block size must be at least 4 larger than the record length.

08 The input data is not completely defined (that is, its data set organization, record format, or record length is zero).

09 The output data set organization is not defined.

System Action: SYSMOD processing stops.

Programmer Response: Correct the inconsistent data set attribute.

GIM23801E COPY PROCESSING TO THE *library* **LIBRARY FAILED FOR MODULE** *modname* **IN LMOD** *loadmod* **IN SYSMOD** *sysmod***. THE RETURN CODE (***rc***) EXCEEDED THE ALLOWABLE VALUE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

library	ddname of the library into which the element was to be copied
modname	module name
loadmod	load module name
sysmod	SYSMOD ID
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes

ss	seconds
seqno	utility sequence number

Copy processing failed for the indicated module, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

During a single invocation of the copy utility, load modules may be copied for several SYSMODs. Copy processing may fail for some load modules but be successful for others. Therefore, some load modules may have been successfully copied, even though an error code was issued.

- This message shows that all the modules or load modules that SMP/E handled during this calling of the copy utility failed, although only one module may have an error.
- The message may also appear for modules within a SYSMOD that SMP/E never copied because other modules in the SYSMOD were in error.

System Action: SYSMOD processing stops if:
- The return code is greater than the return code you specified.
- The return code is greater than the default return code.

Programmer Response:

1. Make sure the default utility return code or the return code you specified in the UTILITY entry is the one you want to use. Make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. If the copy failed, check the utility output to find the error. The sequence number may be used as an index into SYSPRINT output to find the error. The sequence number is incremented for each utility call in an SMP/E run.

   Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:
   - The PRINT subentry specifies a DDDEF of DUMMY.
   - The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.
3. Correct the error and rerun the job.

GIM23802I **COPY PROCESSING TO THE** *library* **LIBRARY WAS SUCCESSFUL FOR MODULE** *modname* **IN LMOD** *loadmod* **IN SYSMOD** *sysmod***. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

library	ddname of the library into which the element was to be copied
modname	module name
loadmod	load module name
sysmod	SYSMOD ID
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Copy processing was successful for the indicated module, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Programmer Response: None.

GIM23802W **COPY PROCESSING TO THE** *library* **LIBRARY WAS SUCCESSFUL FOR MODULE** *modname* **IN LMOD** *loadmod* **IN SYSMOD** *sysmod***. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

library	ddname of the library into which the element was to be copied
modname	module name
loadmod	load module name
sysmod	SYSMOD ID
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Because the return code from copy processing did not exceed the maximum acceptable return code, copy processing was successful for the indicated module, which was part of SYSMOD *sysmod*. However, because the return code from copy processing was not zero, copy processing may not have produced results that you consider acceptable. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Programmer Response: Check the copy SYSPRINT output to determine whether the results are what you consider acceptable.

- If the results are acceptable, no further action is necessary.

- If the results are not acceptable, take the post-installation steps needed to achieve acceptable results.

GIM23803E **COPY PROCESSING TO THE** *library* **LIBRARY FAILED FOR MODULE** *modname* **IN LMOD** *loadmod* **IN SYSMOD** *sysmod***. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy***.***ddd* **- TIME** *hh***:***mm***:***ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

library	ddname of the library into which the element was to be copied
modname	module name
loadmod	load module name
sysmod	SYSMOD ID
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Copy processing failed for the indicated module, which is part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: SYSMOD processing stops.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM23901E **LINK-EDIT PROCESSING FOR SYSMOD** *sysmod* **FAILED FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE (***rc***) EXCEEDED THE ALLOWABLE VALUE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Link-edit processing failed for the indicated module, which was part of SYSMOD *sysmod*. The return code from the link edit utility is greater than:

- The highest acceptable return code for load module *loadmod* specified in the load module's RETURN CODE subentry,
- The highest acceptable return code you specified in the link edit UTILITY entry, or
- SMP/E's default highest acceptable return code.

The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: SYSMOD processing stops.

Programmer Response:

1. If the indicated load module does not contain a RETURN CODE subentry within its LMOD entry, then ensure SMP/E's default highest acceptable return code or the highest acceptable return code you specified in the link edit UTILITY entry is the one you want to use. Also make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. Check the linkage editor output to determine the cause of the error. The utility sequence number may be used as an index into SYSPRINT output to find the error. The sequence number is incremented for each utility call in an SMP/E run.

   Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:

   - The PRINT subentry specifies a DDDEF of DUMMY.
   - The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.
3. Correct the error and rerun the job.

GIM23902E **LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING FAILED FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE (***rc***) EXCEEDED THE ALLOWABLE VALUE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes

ss	seconds
seqno	utility sequence number

Link-edit processing failed for the indicated module, which was included by load module build processing. The return code from the link edit utility is greater than:

- The highest acceptable return code for load module *loadmod* specified in the load module's RETURN CODE subentry,
- The highest acceptable return code you specified in the link edit UTILITY entry, or
- SMP/E's default highest acceptable return code.

The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: SYSMOD processing stops for the SYSMODs named in the message GIM23901E. These SYSMODs contain the modules that caused load module build processing to be performed.

Programmer Response:

1. If the indicated load module does not contain a RETURN CODE subentry within its LMOD entry, then ensure SMP/E's default highest acceptable return code or the highest acceptable return code you specified in the link edit UTILITY entry is the one you want to use. Also make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. Check the output from the linkage editor to determine the error. The utility sequence number may be used as an index into SYSPRINT output to find the error. The sequence number is incremented for each utility call in an SMP/E run.

   Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:
   - The PRINT subentry specifies a DDDEF of DUMMY.
   - The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.
3. Fix the error and rerun the job.

GIM23903I LINK-EDIT PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Because the return code from link-edit processing did not exceed the maximum acceptable return code, link-edit processing was considered successful for the indicated module, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Programmer Response: No action is needed.

GIM23903W LINK-EDIT PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Because the return code from link-edit processing did not exceed the maximum acceptable return code, link-edit processing was considered successful for the indicated module, which was part of SYSMOD *sysmod*. However, because the return code from link-edit processing was not zero, link-edit processing may not have produced results that you consider acceptable. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

Note: Product program directories often state the expected link-edit return codes for their load modules during SMP/E processing. Many times the return code is expected to be 4 or 8 because post-SMP/E link-edit work is required (for example, the load modules may require interface routines or compiler library routines).

For this reason, the SMP/E default acceptable link-edit return code may have been left as or set to 8. This allows the SYSMODs to be installed,

but requires you to check the actual link-edit return code in the GIM23903W messages.

System Action: Processing continues.

Programmer Response: Check the link-edit SYSPRINT output (and product program directories, if appropriate) to determine whether the results are what you consider acceptable.

Note: If the return code is not zero, this may simply mean that you need to do post-SMP/E link-edit work. In that case, the product program directory should explain whether you should expect nonzero return codes and what additional actions you may need to take.

If the results are not acceptable, follow the actions described in the programmer response for GIM23902E.

GIM23904I LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING WAS SUCCESSFUL FOR MODULE *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Link-edit processing was successful for the indicated module, which was included by load module build processing. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Message GIM23903I or GIM23903W names the SYSMOD containing the modules that caused load module build processing to be done.

Programmer Response: None.

GIM23904W LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING WAS SUCCESSFUL FOR MODULE *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Because the return code from link-edit processing did not exceed the maximum acceptable return code, link-edit processing was successful for the indicated module, which was included by load module build processing. However, because the return code from link-edit processing was not zero, link-edit processing may not have produced results that you consider acceptable. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Message GIM23903I or GIM23903W names the SYSMOD containing the modules that caused load module build processing to be done.

Programmer Response: Check the link-edit SYSPRINT output to determine whether the results are what you consider acceptable.

- If the results are acceptable, no further action is necessary.
- If the results are not acceptable, take the post-installation steps needed to achieve acceptable results.

GIM23905E LINK-EDIT PROCESSING FOR SYSMOD *sysmod* **FAILED FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Link-edit processing failed for the indicated module in the indicated load module.

System Action: SYSMOD processing stops for the indicated SYSMOD.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM23906E LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING FAILED FOR MODULE *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqno***.**

Explanation:

modname	module name
loadmod	load module name
library	ddname of the library being processed
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number

Link-edit processing failed for the indicated module, which was included by load module build processing. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: SYSMOD processing stops.

Message GIM23901E names the SYSMOD containing the modules that caused load module build processing to be done.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM23907W LINK-EDIT PROCESSING FAILED FOR MOD *modname* **IN LMOD** *loadmod* **IN ZONE** *xzone* **IN SYSLIB** *syslib***. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *syslib***. THE** *syslib* **LIBRARY RAN OUT OF SPACE.**

Explanation:

modname	MOD name
loadmod	LMOD name
xzone	cross-zone name for load module
syslib	SYSLIB ddname
abendcde	abend code

The library ran out of space during link-edit processing for the indicated module and load module.

Note: Date, time, and sequence number information is not generated for this utility completion message, because the volume of utility output for the cross-zone processing of the APPLY and RESTORE commands is expected to be small.

System Action: Command processing continues.

Programmer Response: Increase the size of the indicated library, and use the LINK command to update the LMOD with the indicated MOD.

GIM23908E LINK PROCESSING FAILED FOR LMOD *loadmod* **IN SYSLIB** *syslib***. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *syslib***. THE** *syslib* **LIBRARY RAN OUT OF SPACE.**

Explanation:

loadmod	LMOD name
syslib	SYSLIB value
abendcde	abend code

LINK processing failed because the indicated library ran out of space.

Note: Date, time, and sequence number information is **not** generated for this utility completion message, because the volume of utility output for the LINK command is expected to be small.

System Action: The return code determines the system action.

Programmer Response: Increase the size of the indicated library, and rerun the LINK command for this load module.

GIM23911E LINK-EDIT PROCESSING FOR SYSMOD *sysmod* **FAILED FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE (***rc***) EXCEEDED THE ALLOWABLE VALUE. DATE** *yy.ddd* **— TIME** *hh:mm:ss* **— SEQUENCE NUMBER** *seqno* **— SYSPRINT FILE** *sysprint***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number
sysprint	ddname of logical SYSPRINT

Link-edit processing failed for the indicated module, which was part of SYSMOD *sysmod*. The utility

sequence number matches the sequence number on the utility's SYSPRINT output, which is contained in file sysprint.

System Action: SYSMOD processing stops if:
- The return code is greater than the return code you specified.
- The return code is greater than the default return code.

Programmer Response:
1. Make sure the default utility return code or the return code you specified in the UTILITY entry is the one you want to use. Make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. Check the linkage editor output to determine the cause of the error. The utility sequence number may be used as an index into the output contained in file sysprint to find the error. The sequence number is incremented for each utility call in an SMP/E run.

   Note: The allocation for the sysprint file is modeled after SYSPRINT or the ddname specified in the PRINT subentry of the active UTILITY entry for the link-edit utility. If you did not get any utility output, check to see if the applicable DD statement is allocated to a proper SYSOUT type. Some typical reasons for not getting any utility output are that
   - The ddname used as a model specifies DUMMY.
   - The ddname used as a model specifies a SYSOUT class that suppresses output.
3. Correct the error and rerun the job.

GIM23912E LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING FAILED FOR MODULE *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE (***rc***) EXCEEDED THE ALLOWABLE VALUE. DATE** *yy.ddd* **— TIME** *hh:mm:ss* **— SEQUENCE NUMBER** *seqno* **— SYSPRINT FILE** *sysprint***.**

Explanation:

modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number
sysprint	ddname of logical SYSPRINT

Link-edit processing failed for the indicated module, which was included by load module build processing. The utility sequence number matches the sequence number on the utility's SYSPRINT output, which is contained in file sysprint.

System Action: SYSMOD processing stops if:
- The return code is greater than the return code you specified.
- The return code is greater than the default return code.

Message GIM23911E names the SYSMOD containing the modules that caused load module build processing to be done.

Programmer Response:
1. Make sure the default utility return code or the return code you specified in the UTILITY entry is the one you want to use. Make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. Check the output from the linkage editor to determine the error. The utility sequence number may be used as an index into output contained in *sysprint* to find the error. The sequence number is incremented for each utility call in an SMP/E run.

   Note: The allocation for the sysprint file is modeled after SYSPRINT or the ddname specified in the PRINT subentry of the active UTILITY entry for the link-edit utility. If you did not get any utility output, check to see if the applicable DD statement is allocated to a proper SYSOUT type. Some typical reasons for not getting any utility output are that
   - The ddname used as a model specifies DUMMY.
   - The ddname used as a model specifies a SYSOUT class that suppresses output.
3. Fix the error and rerun the job.

GIM23913I LINK-EDIT PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **— TIME** *hh:mm:ss* **— SEQUENCE NUMBER** *seqno* **— SYSPRINT FILE** *sysprint***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour

mm	minutes
ss	seconds
seqno	utility sequence number
sysprint	ddname of logical SYSPRINT

Because the return code from link-edit processing did not exceed the maximum acceptable return code, link-edit processing was considered successful for the indicated module, which was part of SYSMOD *sysmod*. The utility sequence number matches the sequence number on the utility's SYSPRINT output, which is contained in file sysprint.

System Action: Processing continues.

Programmer Response: No action is needed.

GIM23913W LINK-EDIT PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **— TIME** *hh:mm:ss* **— SEQUENCE NUMBER** *seqno* **— SYSPRINT FILE** *sysprint***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number
sysprint	ddname of logical SYSPRINT

Because the return code from link-edit processing did not exceed the maximum acceptable return code, link-edit processing was considered successful for the indicated module, which was part of SYSMOD *sysmod*. However, because the return code from link-edit processing was not zero, link-edit processing may not have produced results that you consider acceptable. The utility sequence number matches the sequence number on the utility's SYSPRINT output, which is contained in file sysprint.

Note: Product program directories often state the expected link-edit return codes for their load modules during SMP/E processing. Many times the return code is expected to be 4 or 8 because post-SMP/E link-edit work is required (for example, the load modules may require interface routines or compiler library routines).

For this reason, the SMP/E default acceptable link-edit return code may have been left as or set to 8. This allows the SYSMODs to be installed, but requires you to check the actual link-edit return code in the GIM23913W messages.

System Action: Processing continues.

Programmer Response: Check the link-edit SYSPRINT output in file sysprint (and product program directories, if appropriate) to determine whether the results are what you consider acceptable.

Note: If the return code is not zero, this may simply mean that you need to do post-SMP/E link-edit work. In that case, the product program directory should explain whether you should expect nonzero return codes and what additional actions you may need to take.

If the results are not acceptable, follow the actions described in the programmer response for GIM23912E.

GIM23914I LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING WAS SUCCESSFUL FOR MODULE *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc***. DATE** *yy.ddd* **— TIME** *hh:mm:ss* **— SEQUENCE NUMBER** *seqno* **— SYSPRINT FILE** *sysprint***.**

Explanation:

modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number
sysprint	ddname of logical SYSPRINT

Link-edit processing was successful for the indicated module, which was included by load module build processing. The utility sequence number matches the sequence number on the utility's SYSPRINT output, which is in file *sysprint*.

System Action: Processing continues.

Message GIM23913I or GIM23913W names the SYSMOD containing the modules that caused load module build processing to be done.

Programmer Response: None.

GIM23914W LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING WAS SUCCESSFUL FOR MODULE *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc*. **DATE** *yy.ddd* **— TIME** *hh:mm:ss* **— SEQUENCE NUMBER** *seqno* **— SYSPRINT FILE** *sysprint*.

Explanation:

modname	module name
loadmod	load module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number
sysprint	ddname of logical SYSPRINT

Because the return code from link-edit processing did not exceed the maximum acceptable return code, link-edit processing was successful for the indicated module, which was included by load module build processing. However, because the return code from link-edit processing was not zero, link-edit processing may not have produced results that you consider acceptable. The utility sequence number matches the sequence number on the utility's SYSPRINT output, which is in file *sysprint*.

System Action: Processing continues.

Message GIM23913I or GIM23913W names the SYSMOD containing the modules that caused load module build processing to be done.

Programmer Response: Check the link-edit SYSPRINT in file *sysprint* output to determine whether the results are what you consider acceptable.

- If the results are acceptable, no further action is necessary.
- If the results are not acceptable, take the post-installation steps needed to achieve acceptable results.

GIM23915E LINK-EDIT PROCESSING FOR SYSMOD *sysmod* **FAILED FOR MODULE** *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library*. **THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **— TIME** *hh:mm:ss* **— SEQUENCE NUMBER** *seqno* **— SYSPRINT FILE** *sysprint*.

Explanation:

sysmod	SYSMOD ID
modname	module name
loadmod	load module name
library	ddname of the library
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number
sysprint	ddname of logical SYSPRINT

Link-edit processing failed for the indicated module in the indicated load module.

System Action: SYSMOD processing stops for the indicated SYSMOD.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM23916E LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING FAILED FOR MODULE *modname* **IN LMOD** *loadmod* **IN THE** *library* **LIBRARY. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library*. **THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **— TIME** *hh:mm:ss* **— SEQUENCE NUMBER** *seqno* **— SYSPRINT FILE** *sysprint*.

Explanation:

modname	module name
loadmod	load module name
library	ddname of the library being processed
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqno	utility sequence number
sysprint	ddname of logical SYSPRINT

Link-edit processing failed for the indicated module, which was included by load module build processing. The utility sequence number matches the sequence number on the utility's SYSPRINT output in file sysprint.

System Action: SYSMOD processing stops.

Message GIM23911E names the SYSMOD containing the modules that caused load module build processing to be done.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM24001E ASSEMBLER PROCESSING FOR SYSMOD *sysmod* FAILED FOR MODULE *modname* IN THE *library* LIBRARY. THE RETURN CODE (*rc*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss*.

Explanation:

sysmod	SYSMOD ID
modname	module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

Assembler processing failed for the indicated module, which was part of SYSMOD *sysmod*.

System Action: SYSMOD processing stops if:
- The return code is greater than the return code you specified.
- The return code is greater than the default return code.

Programmer Response:

1. Make sure the default utility return code or the return code you specified in the UTILITY entry is the one you want to use. Make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. Check the assembler output to determine the cause of the error. The time and date stamp on this message may be used as an aid in paging through assembler output by comparing this time stamp against the time stamp issued by the assembler at the top of each SYSPRINT page.

   Notes:

   a. The time stamp on the message might not exactly match the time stamp on the assembler listing, especially if you ran a big assembler job. The message will probably show a later time stamp than the assembler output. The time stamp will still help you find the information you are looking for.

   b. If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:
      - The PRINT subentry specifies a DDDEF of DUMMY.
      - The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

      If SYSPRINT is specified correctly, the problem could be in the UTILITY entry for the assembler. Specifying NOLIST in the PARM subentry for the assembler suppresses utility output. To get your output, delete NOLIST.
3. Correct the error and rerun the job.

GIM24002E ASSEMBLER PROCESSING FAILED FOR MODULE *modname* IN THE *library* LIBRARY WHICH IS NEEDED TO BUILD LOAD MODULE *loadmod*. THE RETURN CODE (*rc*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss*.

Explanation:

modname	module name
library	ddname of the library being processed
loadmod	load module name
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

Assembler processing failed for the indicated module, which is needed by load module build processing to complete the indicated load module.

System Action: SYSMOD processing stops.

Programmer Response:

1. Make sure the default utility return code or the return code you specified in the UTILITY entry is the one you want to use. Make sure the OPTIONS entry you are using contains the name of that UTILITY entry.
2. Check the assembler output to determine the cause of the error. The time and date stamp on this message may be used as an aid in paging through assembler output by comparing this time stamp against the time stamp issued by the assembler at the top of each SYSPRINT page.

   Notes:

   a. The time stamp on the message might not match the time stamp on the assembler listing, especially if you ran a big assembler job. The message will probably show a later time stamp than the assembler output. The time stamp will still help you find the information you are looking for.

   b. If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:
      - The PRINT subentry specifies a DDDEF of DUMMY.

- The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

If SYSPRINT is specified correctly, the problem could be in the UTILITY entry for the assembler. Specifying NOLIST in the PARM subentry for the assembler suppresses utility output. To get your output, delete NOLIST.

3. Correct the error and rerun the job.

GIM24003I ASSEMBLER PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MODULE** *modname* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc*. **DATE** *yy.ddd* **- TIME** *hh:mm:ss*.

Explanation:

sysmod	SYSMOD ID
modname	module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

Assembler processing was successful for the indicated module, which was part of SYSMOD *sysmod*.

System Action: Processing continues.

Programmer Response: None.

GIM24003W ASSEMBLER PROCESSING FOR SYSMOD *sysmod* **WAS SUCCESSFUL FOR MODULE** *modname* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc*. **DATE** *yy.ddd* **- TIME** *hh:mm:ss*.

Explanation:

sysmod	SYSMOD ID
modname	module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

Because the return code from assembler processing did not exceed the maximum acceptable return code, assembler processing was successful for the indicated module, which was part of SYSMOD *sysmod*. However, because the return code from assembler processing was not zero, assembler processing may not have produced results that you consider acceptable.

System Action: Processing continues.

Programmer Response: Check the assembler SYSPRINT output to determine whether the results are what you consider acceptable.

- If the results are acceptable, no further action is necessary.
- If the results are not acceptable, take the post-installation steps needed to achieve acceptable results.

GIM24004I ASSEMBLER PROCESSING FOR LOAD MODULE BUILD PROCESSING WAS SUCCESSFUL FOR MODULE *modname* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc*. **DATE** *yy.ddd* **- TIME** *hh:mm:ss*.

Explanation:

modname	module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

Assembler processing was successful for the indicated module, which was included by load module build processing.

System Action: Processing continues.

Programmer Response: None.

GIM24004W ASSEMBLER PROCESSING FOR LOAD MODULE BUILD PROCESSING WAS SUCCESSFUL FOR MODULE *modname* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rc*. **DATE** *yy.ddd* **- TIME** *hh:mm:ss*.

Explanation:

modname	module name
library	ddname of the library being processed
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

Because the return code from assembler processing did not exceed the maximum acceptable return code, assembler processing was successful for the indicated module, which was included by load module build processing. However, because the return code from assembler processing was not zero, assembler processing may not have produced results that you consider acceptable.

System Action: Processing continues.

Programmer Response: Check the assembler SYSPRINT output to determine whether the results are what you consider acceptable.

- If the results are acceptable, no further action is necessary.
- If the results are not acceptable, take the post-installation steps needed to achieve acceptable results.

GIM24005E **ASSEMBLER PROCESSING FOR SYSMOD** *sysmod* **FAILED FOR MODULE** *modname* **IN THE** *library* **LIBRARY. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **- TIME** *hh:mm:ss***.**

Explanation:

sysmod	SYSMOD ID
modname	module name
library	ddname of the library
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

Assembler processing failed for the indicated module.

System Action: SYSMOD processing stops for the indicated SYSMOD.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM24006E **ASSEMBLER PROCESSING FAILED FOR MODULE** *modname* **IN THE** *library* **LIBRARY WHICH IS NEEDED TO BUILD LOAD MODULE** *loadmod***. ABEND** *abendcde* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **- TIME** *hh:mm:ss***.**

Explanation:

modname	module name
library	ddname of the library being processed
loadmod	load module name
abendcde	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

Assembler processing failed for the indicated module, which is needed by load module build processing to complete the indicated load module, because the library ran out of space.

System Action: SYSMOD processing stops.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM24100I **SMP/E PROCESSING TO THE** *library* **LIBRARY WAS SUCCESSFUL FOR** *enttype entname* **IN SYSMOD** *sysmod***. DATE** *yy.ddd* **- TIME** *hh:mm:ss*

Explanation:

library	ddname of the library to which the element was to be installed.
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

The indicated element was successfully processed.

System Action: Processing continues.

Programmer Response: None.

GIM24200E **SYSTEM ABEND** *abendcde* **OCCURRED WITH A REASON CODE OF** *rsncode* **DURING SMP/E PROCESSING. THE** *ddname* **LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT IS SEQUENTIAL.**

Explanation:

abendcde	abend code
rsncode	abend reason code
ddname	ddname of the library

The indicated library ran out of space when it was being processed by SMP/E. The library is not eligible for retry processing because it has a sequential data set organization.

System Action: The element being processed and the SYSMOD supplying it are failed.

Programmer Response: Increase the size of the library being processed and rerun the job.

GIM24300E **LIBRARY** *library* **IS SEQUENTIAL YET** *enttype1 entname1* **IS ALREADY ASSIGNED TO IT. SMP/E PROCESSING HAS FAILED FOR** *enttype2 entname2* **IN SYSMOD** *sysmod***.**

Explanation:

library	ddname of the sequential library
enttype1	already assigned data element type
entname1	already assigned data element name
enttype2	data element type

entname2 data element name
sysmod name of SYSMOD supplying the data element

The data element being APPLIED, ACCEPTED, or RESTORED has the specified library, which is a sequential data set. Only one element can exist in a sequential data set, and there is already one assigned to it.

System Action: Processing fails for the SYSMODs containing the data elements.

Programmer Response: Check the DDDEF entry or DD statement for the specified library to ensure that it specifies the correct data set. If the data set is correct, verify that the data set was allocated with the correct organization. If it should have a partitioned organization, change the allocation and rerun the command.

If the data set should be sequential, verify that the DDNAME for the specified data element is correct. If it is incorrect, correct the SYSLIB or DISTLIB subentry, and rerun the command.

GIM24501E RESTORE PROCESSING FAILED FOR SYSMOD *sysmod* **BECAUSE OTHER SYSMODS WERE DELETED WHEN** *sysmod* **WAS APPLIED.**

Explanation:
sysmod SYSMOD ID

SMP/E cannot restore SYSMOD *sysmod* because SYSMOD *sysmod* is a function SYSMOD that deleted other SYSMODs when it was applied. (SYSMOD *sysmod* contains a ++VER DELETE MCS.) SMP/E cannot restore the elements from the deleted SYSMODs. Therefore, SMP/E cannot restore SYSMOD *sysmod*.

System Action: SYSMOD processing stops. This message will be followed by GIM37001S, which indicates that RESTORE processing has stopped.

Programmer Response: Do **one** of the following:

- If the SYSMOD was specified on the SELECT operand, delete it from the SELECT list.
- If the SYSMOD was included by the GROUP operand, delete the GROUP operand and specify the required SYSMODs on the SELECT operand instead.

Then rerun the RESTORE command.

GIM24502E RESTORE PROCESSING FAILED FOR SYSMOD *sysmod* **BECAUSE** *sysmod* **WAS PREVIOUSLY ACCEPTED.**

Explanation:
sysmod SYSMOD ID

SMP/E cannot restore SYSMOD *sysmod* because SYSMOD *sysmod* has been accepted and has updated elements in the distribution libraries. Therefore, SMP/E cannot use the elements in the distribution libraries to restore the target libraries. SMP/E can only restore SYSMODs that have been applied but not accepted.

System Action: SYSMOD processing stops. If this SYSMOD is a function, this message will be followed by GIM37001S, which indicates that RESTORE processing has stopped.

Programmer Response: Do **one** of the following:

- If the SYSMOD was specified on the SELECT operand, delete it from the SELECT list.
- If the SYSMOD was included by the GROUP operand, delete the GROUP operand and specify the required SYSMODs on the SELECT operand instead.

Then rerun the RESTORE command.

GIM24503E RESTORE PROCESSING FAILED FOR SYSMOD *sysmod* **BECAUSE** *sysmod* **WAS PREVIOUSLY DELETED.**

Explanation:
sysmod SYSMOD ID

SMP/E cannot restore SYSMOD *sysmod* because it was deleted when another SYSMOD was applied.

System Action: SYSMOD processing stops. If this SYSMOD is a function, this message will be followed by GIM37001S, which indicates that RESTORE processing has stopped.

Programmer Response: Do **one** of the following:

- If the SYSMOD was specified on the SELECT operand, delete it from the SELECT list.
- If the SYSMOD was included by the GROUP operand, delete the GROUP operand and specify the required SYSMODs on the SELECT operand instead.

Then rerun the RESTORE command.

GIM24504E RESTORE PROCESSING FAILED FOR SYSMOD *sysmod* **BECAUSE** *sysmod* **HAS NOT BEEN APPLIED.**

Explanation:
sysmod SYSMOD ID

SMP/E cannot restore SYSMOD *sysmod* because SYSMOD *sysmod* has not been applied. A SYSMOD can only be restored if it has been applied.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- If the SYSMOD was specified on the SELECT operand, delete it from the SELECT list.

- If the SYSMOD was included by the GROUP operand, delete the GROUP operand and specify the required SYSMODs on the SELECT operand instead.

Then rerun the RESTORE command.

GIM24505E RESTORE PROCESSING FAILED FOR SYSMOD *sysmod* **BECAUSE** *sysmoda* **WAS PREVIOUSLY SUPERSEDED BY SYSMOD** *sysmod***.**

Explanation:

sysmod	SYSMOD ID
sysmod	superseding SYSMOD ID

SMP/E cannot restore SYSMOD *sysmod* because it has never been applied. Instead, SYSMOD *sysmod*, which supersedes SYSMOD *sysmod*, has been applied. In this case, SMP/E cannot determine which SYSMODs it should restore.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- If the SYSMOD was specified on the SELECT operand, delete it from the SELECT list.
- If the SYSMOD was included by the GROUP operand, delete the GROUP operand and specify the required SYSMODs on the SELECT operand instead.
- If SYSMOD *sysmod* should be restored, add it to the SELECT operand.

Then rerun the RESTORE command.

GIM24506W SYSMOD *sysmod* **SELECTED FOR RESTORE PROCESSING IS SUPERSEDED.**

Explanation:

sysmod	SYSMOD ID

SYSMOD *sysmod* was specified on the SELECT operand but had not been applied. However, it was superseded by another SYSMOD that was included by RESTORE processing. Both SYSMODs were restored.

System Action: Processing continues.

Programmer Response: No action is required. If you want to put the superseding SYSMOD back on the target libraries, receive it again and reapply it.

GIM24601I *enttype* **ENTRY** *entname* **IS NEEDED FOR PROCESSING BUT IS NOT IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library being processed

Element *entname* is needed for command processing but does not exist in the indicated library.

System Action: Check the messages that follow in SMPOUT to determine the system action.

Programmer Response: Check SMPLOG to determine why SMP/E could not find the element in the library.

GIM24601W *enttype* **ENTRY** *entname* **IS NEEDED FOR PROCESSING BUT IS NOT IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library being processed

Element *entname* is needed for command processing but does not exist in the indicated library.

System Action: Check the messages that follow in SMPOUT to determine the system action.

Programmer Response: Check SMPLOG to determine why SMP/E could not find the element in the library.

GIM24601E *enttype* **ENTRY** *entname* **IS NEEDED FOR PROCESSING BUT IS NOT IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library being processed

Element *entname* is needed for command processing but does not exist in the indicated library.

System Action: Check the messages that follow in SMPOUT to determine the system action.

Programmer Response: Check SMPLOG to determine why SMP/E could not find the element in the library.

GIM24602W *enttype* **ENTRY** *entname* **IS NEEDED TO PROCESS SYSMOD** *sysmod* **BUT IS NOT IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
sysmod	SYSMOD ID
library	ddname of the library being processed

Element *entname* is needed for SYSMOD processing but does not exist in the indicated library.

System Action: Check the messages that follow in SMPOUT to determine the system action.

If the entry was for an assembly, the following occurs:
- SMP/E does not issue any further messages.
- SMP/E does not do assemblies for the SYSMOD.
- SYSMOD processing continues.

Programmer Response: Check SMPLOG to determine why SMP/E could not find the element in the library. For example, the indicated SYSMOD may not be applicable to your system.

GIM24602E *enttype* **ENTRY** *entname* **IS NEEDED TO PROCESS SYSMOD** *sysmod* **BUT IS NOT IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
sysmod	SYSMOD ID
library	ddname of the library being processed

Element *entname* is needed for SYSMOD processing but does not exist in the indicated library.

System Action: Check the messages that follow in SMPOUT to determine the system action.

If the entry was for an assembly, the following occurs:
- SMP/E does not issue any further messages.
- SMP/E does not do assemblies for the SYSMOD.
- SYSMOD processing continues.

Programmer Response: Check SMPLOG to determine why SMP/E could not find the element in the library. For example, the indicated SYSMOD may not be applicable to your system.

GIM24603E *enttype* **ENTRY** *entname* **IS NEEDED TO BUILD THE FOLLOWING LOAD MODULES FOR SYSMOD** *sysmod* **BUT IS NOT IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
sysmod	SYSMOD ID
library	ddname of the library being processed

Entry *entname* is needed to build load modules, but does not exist in the indicated library as expected.

Message GIM65905 follows this message and identifies the load modules that could not be built.

System Action: SYSMOD processing stops.

Programmer Response: SMP/E expected to find entry *entname*, but it does not exist. Check SMPLOG for indications of why the entry no longer exists in the library.

GIM24605E **SYSMOD** *sysmod* **WAS SELECTED FOR REJECT PROCESSING BUT IT WAS NOT FOUND IN THE SMPPTS DATA SET.**

Explanation:

sysmod	SYSMOD ID

SYSMOD *sysmod* cannot be rejected because it has not been received.

System Action: SYSMOD processing stops. Processing continues with the next SYSMOD.

Programmer Response: Specify the correct SYSMODs on the SELECT operand and rerun the command.

GIM24606E *enttype* **ENTRY** *entname* **IS NEEDED FOR PROCESSING BUT IS NOT IN THE** *zonename* **ZONE.**

Explanation:

enttype	entry type
entname	entry name
zonename	name of the zone being processed

Element *entname* is needed for command processing but does not exist in the indicated zone.

System Action: Check the messages that follow in SMPOUT to determine the system action.

Programmer Response: Check SMPLOG to determine why SMP/E could not find the element in the zone.

GIM24607E **THE** *enttype* **ENTRY IS NEEDED TO PROCESS SYSMOD** *sysmod* **BUT IS NOT IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
sysmod	SYSMOD ID
library	ddname of the library being processed

The indicated entry type is needed for SYSMOD processing but does not exist in the indicated library.

System Action: Check the messages that follow in SMPOUT to determine the system action.

Programmer Response: Check SMPLOG to determine why SMP/E could not find the entry in the library. For example, the indicated SYSMOD may not be applicable to your system.

GIM24608E SHELLSCR ENTRY *shellscr* IS NEEDED TO PROCESS *enttype entname* FOR SYSMOD *sysmod*, BUT SHELLSCR *shellscr* IS NOT IN THE *zonename* ZONE.

Explanation:

shellscr	SHELLSCR entry name
enttype	element type
entname	element name
sysmod	SYSMOD ID
zonename	zone name

The indicated element specifies that shell script *shellscr* is to be invoked to complete the element's installation. However, the SHELLSCR entry for *shellscr* does not exist in the indicated zone.

System Action: SYSMOD processing stops.

Programmer Response: Determine whether the SHELLSCR element is installed. If not, install it and rerun the job.

GIM24701W SMP/E COULD NOT OBTAIN LINK-EDIT PARAMETERS FOR LOAD MODULE *loadmod* FOR SYSMOD *sysmod*. DEFAULTS WERE USED.

Explanation:

loadmod	load module name
sysmod	SYSMOD ID

SMP/E tried to determine the link-edit parameters to be used for the indicated load module. (See the description of ACCEPT processing or APPLY processing, as appropriate, in *SMP/E Commands* for details on how SMP/E makes this determination.) Because no other values were available, SMP/E is using default link-edit parameters.

Note: This message is normal when SMP/E accepts elements into the distribution libraries for the first time.

System Action: SMP/E passes its default link-edit parameters to the link-edit utility. (See the description of the UTILITY entry in *SMP/E Reference* for a list of the default link-edit parameters.)

Programmer Response: If SMP/E default link-edit parameters are used, make sure they are appropriate for the load module being processed.

GIM24801W NO SYSMODS SATISFIED THE OPERANDS SPECIFIED ON THE *command* COMMAND.

Explanation:

command	SMP/E command (RECEIVE)

The operands that you specify on a command determine which SYSMODs SMP/E selects for processing. This command failed because SMP/E did not select any SYSMODs. Here is an example of when this can happen:

- You enter the **RECEIVE FORFMID(JXY01)** command. This tells SMP/E to receive all the specified SYSMODs in the SMPPTFIN data set. However, no SYSMODs are received. This can happen if either of the following is true:
  - The eligible SYSMODs have already been received.
  - No SYSMODs satisfied the FORFMID operand on the RECEIVE command.

For more information about the selection criteria for the RECEIVE command, see the related chapter in *SMP/E Commands*.

System Action: Command processing stops normally.

Programmer Response: Review the other messages that were issued for this command. Make sure you specified the correct operands on the command.

GIM24801S NO SYSMODS SATISFIED THE OPERANDS SPECIFIED ON THE *command* COMMAND.

Explanation:

command	SMP/E command

The operands that you specify on a command determine which SYSMODs SMP/E selects for processing. This command failed because SMP/E did not select any SYSMODs. Here are some examples of when this can happen:

- You enter the **APPLY APARS** command. This tells SMP/E to install all applicable APARs into the target zone. However, there are no applicable APARs to install. This can happen if any of the following are true:
  - There are no APARs in the global zone.
  - The SRELs or FMIDs for APARs in the global zone are not applicable to the target zone.
  - The APARs in the global zone have already been installed in the target zone.
- You enter the **REJECT** command, without any operands. This tells SMP/E to remove from the global zone all PTFs that have never been installed in any target or distribution zone. However, no PTFs meet this criterion. This can happen if either of the following is true:
  - There are no PTFs in the global zone.
  - The PTFs in the global zone have been installed in a target or distribution zone.
- You enter the **RESTORE S(UZ00001)** command. This tells SMP/E to bring the elements affected by SYSMOD UZ00001 back to their distribution library level. However, UZ00001 cannot be restored. This can happen if any of the following are true:

– UZ00001 has already been accepted into the distribution libraries.
– UZ00001 has not been installed in the target zone or has been installed in error (the ERROR indicator is set in the SYSMOD entry).

For more information about the selection criteria for each command, see the related chapter in *SMP/E Commands*.

System Action: Command processing fails. Processing continues with the next command.

Programmer Response: Review the other messages that were issued for this command. Make sure you specified the correct operands on the command.

GIM24901E SYSMOD *sysmod* FAILED BECAUSE THE SAME MODULE (*modname*) WAS SPECIFIED ON DIFFERENT TYPES OF ZAP NAME STATEMENTS.

Explanation:

sysmod	SYSMOD ID
modname	module name

The same module was specified on different NAME statements for the ZAP utility (NAME *csect* and NAME *lmod csect*). SMP/E cannot determine whether the module should be updated in all the load modules that contain it (as indicated by the NAME *csect* statement) or in a single load module (as indicated by the NAME *lmod csect* statement).

System Action: SYSMOD processing stops. Processing continues with the next SYSMOD.

Programmer Response: Correct the NAME statements and rerun the job.

GIM25000W LMOD *loadmod* CONTAINS *subtype* SUBENTRIES. *subtype* INFORMATION IS NOT INCLUDED IN FUNCTION *fmid* FROM THE EXISTING ZONE ENTRIES.

Explanation:

loadmod	load module name
subtype	subentry type
fmid	FMID

During BUILDMCS command processing, SMP/E found one of the following subentries in an LMOD entry:

MODDEL	Modules deleted from the load module
XZMOD	Cross-zone modules

The information associated with the subentries is not included in the MCS for the function generated by the BUILDMCS command.

System Action: Processing continues.

Programmer Response: If the information associated with the subentries should be included in the new environment, you can do the following:

- For MODDEL subentries, what you can do depends on the state of the deleted module in the new zone environment. If the deleted module does not exist in the new zone, then you can use UCLIN to add the MODDEL subentry to the LMOD entry after the superseding function is installed. If the deleted module does exist in the new zone, then you can add an INCLUDE statement in the JCLIN of the superseding function to add the module into the load module.
- For XZMOD subentries, you can use the LINK command to add the cross-zone module into the load module after the superseding function is installed into the new zone environment.

GIM25101S WHEN THE *operand* OPERAND IS SPECIFIED, THE *command* COMMAND MUST SPECIFY AT LEAST ONE *zonetype* ZONE.

Explanation:

operand	operand
command	SMP/E command (REJECT or REPORT)
zonetype	TARGET or DLIB

For the REJECT command, **one** of the following occurred:

- The command specified the PURGE operand, but did not specify a DLIB zone or a ZONESET that contains DLIB zones.
- The command specified the TARGETZONE operand, but did not specify a target zone or a ZONESET that contains target zones.

For the REPORT command, **one** of the following occurred:

- The command specified the DLIBZONE operand, but the specified ZONESET contains only target zones.
- The command specified the TARGETZONE operand, but the specified ZONESET contains only DLIB zones.

System Action: Command processing stops.

Programmer Response: Specify the required type of zone for the indicated operand. Then rerun the job.

GIM25201E THE INDICATED COMMAND IS INCOMPLETE. IT MAY BE MISSING DATA OR A DELIMITER, OR A PREVIOUS COMMAND MAY BE MISSING A PARENTHESIS.

Explanation: SMP/E found an incomplete command, and an end-of-file occurred before the end of the statement. SMP/E lists the command in error before this message.

System Action: The command is not processed. For SMP/E's action, check the messages that follow in SMPOUT.

Programmer Response: Check the job for the following:
- A missing comment delimiter (*/)
- A missing statement delimiter (.)
- A previous LOG command that is missing parentheses
- An end-of-file before the minimum data that is required to make a complete statement

Correct the error and rerun the job.

GIM25201T THE INDICATED COMMAND IS INCOMPLETE. IT MAY BE MISSING DATA OR A DELIMITER, OR A PREVIOUS COMMAND MAY BE MISSING A PARENTHESIS.

Explanation: SMP/E found an incomplete command, and an end-of-file occurred before the end of the statement. SMP/E lists the command in error before this message.

System Action: The command is not processed. For SMP/E's action, check the messages that follow in SMPOUT.

Programmer Response: Check the job for the following:
- A missing comment delimiter (*/)
- A missing statement delimiter (.)
- A previous LOG command that is missing parentheses
- An end-of-file before the minimum data that is required to make a complete statement

Correct the error and rerun the job.

GIM25301E *enttype* **ENTRY** *entname* **WAS NOT DELETED BECAUSE IT DOES NOT EXIST.**

Explanation:

enttype	entry type
entname	entry name or *prodid vv.rr.mm* for PRODUCT entries

SMP/E could not find the entry you want to delete in the specified zone.

System Action: SMP/E ignores the UCL statement. Processing continues with the next UCL statement.

Programmer Response: Correct the UCL statement and rerun the statement.

GIM25401E *value* **IS NOT DEFINED AS A ZONE OR A ZONESET IN THE GLOBAL ZONE.**

Explanation:

value	value specified as either a zone or ZONESET name

SMP/E tried to determine whether the specified name was a zone or a ZONESET. The name was not found to be defined as either a zone or a ZONESET in the global zone.

System Action: Command processing stops.

Programmer Response: Check where SMP/E encountered this name. It was either explicitly specified or was part of a ZONESET definition.

Define the zone or ZONESET in the global zone and rerun the job.

GIM25401S *value* **IS NOT DEFINED AS A ZONE OR A ZONESET IN THE GLOBAL ZONE.**

Explanation:

value	value specified as either a zone or ZONESET name

SMP/E tried to determine whether the specified name was a zone or a ZONESET. The name was not found to be defined as either a zone or a ZONESET in the global zone.

System Action: Command processing stops.

Programmer Response: Check where SMP/E encountered this name. It was either explicitly specified or was part of a ZONESET definition.

Define the zone or ZONESET in the global zone and rerun the job.

GIM25501I ENTRY *entname* **WAS UPDATED BY UCLIN.**

Explanation:

entname	entry name or *prodid vv.rr.mm* for PRODUCT entries

UCLIN processing for the entry was successful.

System Action: Processing continues with the next UCL statement.

Programmer Response: None.

GIM25601I THE SPECIFIED ENTRY WAS NOT UPDATED BECAUSE OF AN ERROR DURING UCLIN PROCESSING.

Explanation: UCLIN processing failed for the entry because of an error identified in a previous message. SMP/E did not change the entry.

System Action: Update processing stops.

Programmer Response: Fix the error and rerun the statement.

GIM25701E THE SPECIFIED CHANGES WERE NOT PROCESSED BECAUSE THEY WOULD LEAVE THE ENTRY INCOMPLETE. *subtype* IS REQUIRED.

Explanation:

subtype	subentry type

If SMP/E made the indicated update, the affected entry would have insufficient or inconsistent data. Additional data needed by SMP/E is indicated by *subtype*.

System Action: SMP/E ignores the UCL statement and does not change the entry. Processing continues with the next UCL statement.

Programmer Response: Check *SMP/E Reference* to find out which subentries are required for the entry you are updating. Provide the missing information in the SMPCSI or UCLIN input and rerun the statement.

GIM25801E *command1* PROCESSING FAILED BECAUSE SMP/E REACHED THE END-OF-FILE WITHOUT FINDING THE *command2* COMMAND.

Explanation:

command1	command that failed (UCLIN or ZONEEDIT)
command2	command that was not found (ENDUCL or ENDZONEEDIT)

Before SMP/E found the ENDUCL or ENDZONEEDIT statement, an end-of-file occurred on SMPCNTL.

System Action: SMP/E ignores the current UCL or ZONEEDIT statement, if any, and stops command processing.

Programmer Response: Add the ENDUCL or ENDZONEEDIT statement to the input data stream and rerun the statements that SMP/E did not process.

GIM25901E SMP/E COULD NOT ADD *subtype* SUBENTRY *subentry* TO *enttype* ENTRY *entname* BECAUSE THE NUMBER OF ELEMENTS IN THE ENTRY EXCEEDED THE CURRENT PEMAX VALUE.

Explanation:

subtype	subentry type
subentry	subentry name
enttype	entry type
entname	entry name

As SMP/E tried to add a subentry to the specified entry, the number of elements in the entry exceeded the maximum number allowed (PEMAX or fixed value).

System Action:

- For UCLIN processing, SMP/E ignores the UCL statement, and processing continues with the next UCL statement.
- For RECEIVE, APPLY, ACCEPT and RESTORE processing, the SYSMOD associated with the specified entry stops processing.

  You may see this happen for the global zone during RECEIVE processing. In this case, SMP/E added the named FMID entry to the global zone but, for some reason, could not receive the function SYSMOD. To get around this problem, you could have used UCLIN to delete the FMID of the SYSMOD that was not received.

Programmer Response: Do **one** of the following:

- For UCL processing, do either of the following:
  - Reduce the number of subentries you specified in the UCL statement.
  - Increase the PEMAX value in the OPTIONS entry being used.
- For RECEIVE, APPLY, ACCEPT, and RESTORE processing, increase the PEMAX value in the OPTIONS entry being used.

  If this situation occurs for the global zone, do the following:

  1. Increase the PEMAX value in the OPTIONS entry being used.
  2. Run UCLIN against the GLOBALZONE entry to add FMIDs that SMP/E received but did not add to the GLOBALZONE entry.

Make sure the OPTIONS entry containing the increased PEMAX value is the OPTIONS entry named in the zone definition entry or on the SET command, if appropriate.

GIM25902E SMP/E COULD NOT ADD *subtype* SUBENTRY *subentry* TO *enttype* ENTRY *entname* IN THE SMPSCDS DATA SET BECAUSE THE NUMBER OF ELEMENTS IN THE ENTRY EXCEEDED THE CURRENT PEMAX VALUE.

Explanation:

subtype	subentry type
subentry	subentry name
enttype	entry type
entname	entry name

During APPLY processing, SMP/E was adding BACKUP entries to the SMPSCDS data set. As SMP/E tried to add a subentry to the specified entry, the number of elements in the entry exceeded the maximum number allowed (PEMAX or fixed value.)

System Action: SYSMOD processing stops.

Programmer Response: Increase the PEMAX value in the OPTIONS entry being used. Make sure the OPTIONS entry containing the increased PEMAX value is the OPTIONS entry named in the zone definition

entry or on the SET command, if appropriate.

GIM25903E SMP/E COULD NOT ADD ++ASSIGN SOURCEID *sourceid* TO SYSMOD ENTRY *sysmod* BECAUSE THE NUMBER OF ELEMENTS IN THE ENTRY EXCEEDED THE CURRENT PEMAX VALUE.

Explanation:

sourceid	SOURCEID value
sysmod	SYSMOD ID

As SMP/E tried to add a SOURCEID value to the specified SYSMOD entry, the number of elements in the entry exceeded the maximum number allowed (PEMAX or fixed value.)

System Action: Processing stops for the ++ASSIGN MCS. Processing continues with the next MCS.

Programmer Response: Do the following:
1. Increase the PEMAX value in the OPTIONS entry being used.
2. Run UCLIN against the global zone SYSMOD entry to add the SOURCEID value.

GIM26001W THE *operand* OPERAND WAS NOT PROCESSED BECAUSE IT IS NOT ALLOWED FOR THE SMPTLIB DATA SET.

Explanation:

operand	CATALOG, DELETE, KEEP, MOD, NEW, OLD, or SHR

The SMPTLIB DDDEF entry cannot contain an initial or final disposition. SMP/E will determine the appropriate disposition based on the command being processed.

System Action: UCLIN processing continues. SMP/E ignores the disposition operand.

Programmer Response: None.

For more information about SMPTLIB data sets, see *SMP/E Reference*.

GIM26002E UCLIN PROCESSING FAILED BECAUSE THE *operand* OPERAND IS NOT ALLOWED FOR THE SMPTLIB DATA SET.

Explanation:

operand	DATASET, PATH, or SYSOUT

The SMPTLIB DDDEF entry cannot contain a data set name, SYSOUT class, or PATH subentry.

System Action: UCLIN processing stops.

Programmer Response: If you want to define a data set name for the SMPTLIB data set, you may specify a data set prefix (DSPREFIX) in either the SMPTLIB DDDEF entry or in the OPTIONS entry that will be in effect. Delete the DATASET, SYSOUT, or PATH value and rerun the UCL statement.

For more information about SMPTLIB data sets, see *SMP/E Reference*.

GIM26003S THERE IS AN ERROR IN A GIMDDALC CONTROL STATEMENT. THE *operand* OPERAND IS NOT ALLOWED FOR THE SMPTLIB DATA SET.

Explanation:

operand	control statement operand

A GIMDDALC control statement for SMPTLIB data sets cannot contain the indicated operand. Only the SPACE, TRACKS, and DIR operands are allowed for SMPTLIB.

System Action: Command processing stops.

Programmer Response: Correct the control statement and rerun the job. For information on the GIMDDALC control statement and the SMPTLIB data set, see *SMP/E Reference*.

GIM26004W THE *operand* OPERAND IS BEING IGNORED SINCE TRANSFERONLY WAS SPECIFIED ON THE FROMNETWORK OPERAND.

Explanation:

operand	name of operand

The named operand is being ignored because the TRANSFERONLY operand was specified.

System Action: Processing continues.

Programmer Response: If the network package is successfully transferred, rerun the RECEIVE command with the FROMNTS operand specified along with the named operand that was ignored to get the desired results for the RECEIVE command.

GIM26101E THE *operand* OPERAND VALUE SPECIFIED ON THE UCL DEL STATEMENT DOES NOT MATCH THE EXISTING VALUE (*value*) IN THE ENTRY.

Explanation:

operand	value specified on the UCL DEL statement.
value	subentry value in the existing entry (*prodid vv.rr.mm* for PRODSUP subentries)

The operand on the UCL DEL command did not match the subentry value in the existing entry.

System Action: Processing stops for the UCL statement.

Programmer Response: Specify the correct value and rerun the UCL statement.

GIM26201I A PREVIOUS ERROR FORCED JCLIN SCANNING TO FAIL.

Explanation: An error explained in a previous message caused the JCLIN command to stop scanning SMPJCLIN.

System Action: JCLIN processing stops.

Programmer Response: Fix the previous error and rerun the job.

GIM26301E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE ASSEM OR SRC ENTRY NAME COULD NOT BE DETERMINED.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

This message is issued when SMP/E is attempting to build an ASSEM entry or SRC entry and cannot determine the name to use for the entry after checking all the possible sources of information. For details on the checking SMP/E does, see the sections on building ASSEM entries and SRC entries in the JCLIN command chapter of *SMP/E Commands*.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26301S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE ASSEM OR SRC ENTRY NAME COULD NOT BE DETERMINED.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

This message is issued when SMP/E is attempting to build an ASSEM entry or SRC entry and cannot determine the name to use for the entry after checking all the possible sources of information. For details on the checking SMP/E does, see the sections on building ASSEM entries and SRC entries in the JCLIN command chapter of *SMP/E Commands*.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26302E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. AN UNKNOWN ERROR OCCURRED FOR A MODULE IN A LINK-EDIT STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

An undefined error occurred during linkage-editor processing for a module. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26302S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. AN UNKNOWN ERROR OCCURRED FOR A MODULE IN A LINK-EDIT STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

An undefined error occurred during linkage-editor processing for a module. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26303E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THERE IS A SYNTAX ERROR IN THE COPY CONTROL STATEMENT.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

There is a syntax error in a copy control statement that was being processed. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: See *z/OS DFSMSdfp Utilities*, SC26-7414, for the correct syntax of copy control statements. Fix the error and rerun the job.

GIM26303S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THERE IS A SYNTAX ERROR IN THE COPY CONTROL STATEMENT.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

There is a syntax error in a copy control statement that was being processed. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: See *z/OS DFSMSdfp Utilities*, SC26-7414, for the correct syntax of copy control statements. Fix the error and rerun the job.

GIM26304E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. SMP/E DID NOT RECOGNIZE THE CHARACTERS ON THE MOD OPERAND IN A LINK-EDIT STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

SMP/E did not recognize the value specified for the MOD operand on a linkage editor control statement. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: See *z/OS DFSMS Program Management*, SC27-1130, for the correct syntax of linkage editor control statements. Fix the error and rerun the job.

GIM26304S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. SMP/E DID NOT RECOGNIZE THE CHARACTERS ON THE MOD OPERAND IN A LINK-EDIT STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

SMP/E did not recognize the value specified for the MOD operand on a linkage editor control statement. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: See *z/OS DFSMS Program Management*, SC27-1130, for the correct syntax of linkage editor control statements. Fix the error and rerun the job.

GIM26305E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE MACRO NAME MUST BE FROM 1 TO 8 CHARACTERS LONG.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

When SMP/E was processing assembler statements, it found a macro name with an incorrect length. The

length must be from 1 to 8 characters. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26305S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE MACRO NAME MUST BE FROM 1 TO 8 CHARACTERS LONG.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

When SMP/E was processing assembler statements, it found a macro name with an incorrect length. The length must be from 1 to 8 characters. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26306E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. AN I/O ERROR OCCURRED IN THE LAST LINE PROCESSED BY SMP/E.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

An I/O error occurred when SMP/E was processing the utility control statements. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job, if necessary.

GIM26306S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. AN I/O ERROR OCCURRED IN THE LAST LINE PROCESSED BY SMP/E.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

An I/O error occurred when SMP/E was processing the utility control statements. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job, if necessary.

GIM26307E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THERE IS A SYNTAX ERROR IN A LINK-EDIT CONTROL STATEMENT.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

There is a syntax error in a link-edit control statement that was being processed. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: See *z/OS DFSMS Program Management*, SC27-1130, for the correct syntax of linkage editor control statements. Fix the error and rerun the job.

GIM26307S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THERE IS A SYNTAX ERROR IN A LINK-EDIT CONTROL STATEMENT.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

There is a syntax error in a link-edit control statement that was being processed. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: See *z/OS DFSMS Program Management*, SC27-1130, for the correct syntax of linkage editor control statements. Fix the error and rerun the job.

GIM26308E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. SMP/E RAN OUT OF STORAGE WHILE PROCESSING AN ASSEMBLER STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

SMP/E exceeded its allocated storage while it was processing an assembler step. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Increase the REGION size on the EXEC statement for GIMSMP. Then rerun the job.

GIM26308S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. SMP/E RAN OUT OF STORAGE WHILE PROCESSING AN ASSEMBLER STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

SMP/E exceeded its allocated storage while it was processing an assembler step. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Increase the REGION size on the EXEC statement for GIMSMP. Then rerun the job.

GIM26309E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE MODULE NAME WAS NOT SPECIFIED ON A SYSLMOD DD STATEMENT OR ON A NAME STATEMENT.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

The name of the module being processed is not on the SYSLMOD DD statement or on a link-edit NAME statement. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26309S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE MODULE NAME WAS NOT SPECIFIED ON A SYSLMOD DD STATEMENT OR ON A NAME STATEMENT.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

The name of the module being processed is not on the SYSLMOD DD statement or on a link-edit NAME statement. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26310E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE PGM OPERAND WAS SPECIFIED ON THE EXEC STATEMENT, BUT NO SYSLMOD DD STATEMENT WAS FOUND.

Explanation:

stepname step name or NONAME
jobname job name or NONAME

SMP/E expected to find a SYSLMOD DD statement because PGM was specified on the EXEC statement. However, there was no SYSLMOD DD statement. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26310S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE PGM OPERAND WAS SPECIFIED ON THE EXEC STATEMENT, BUT NO SYSLMOD DD STATEMENT WAS FOUND.

Explanation:

stepname step name or NONAME
jobname job name or NONAME

SMP/E expected to find a SYSLMOD DD statement because PGM was specified on the EXEC statement. However, there was no SYSLMOD DD statement. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26311E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. SMP/E RAN OUT OF STORAGE WHILE PROCESSING A LINK-EDIT STEP.

Explanation:

stepname step name or NONAME
jobname job name or NONAME

SMP/E exceeded its allocated storage while it was processing a link-edit step. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Increase the REGION size on the EXEC statement for GIMSMP. Then rerun the job.

GIM26311S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. SMP/E RAN OUT OF STORAGE WHILE PROCESSING A LINK-EDIT STEP.

Explanation:

stepname step name or NONAME
jobname job name or NONAME

SMP/E exceeded its allocated storage while it was processing a link-edit step. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Increase the REGION size on the EXEC statement for GIMSMP. Then rerun the job.

GIM26312E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE DSNAME OPERAND WAS NOT SPECIFIED ON THE EXEC STATEMENT FOR A LINK-EDIT STEP.

Explanation:

stepname step name or NONAME
jobname job name or NONAME

The DSNAME operand was not specified on the EXEC statement for a link-edit procedure. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26312S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE DSNAME OPERAND WAS NOT SPECIFIED ON THE EXEC STATEMENT FOR A LINK-EDIT STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

The DSNAME operand was not specified on the EXEC statement for a link-edit procedure. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26313E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. A NAME STATEMENT WAS NOT SPECIFIED FOR THE MODULE, OR THE MOD OPERAND WAS NOT SPECIFIED ON THE EXEC STATEMENT FOR A LINK-EDIT STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

When SMP/E was processing a link-edit step, it encountered a module that was not specified on a link-edit NAME statement or on the MOD operand of the EXEC statement for the link-edit procedure. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26313S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. A NAME STATEMENT WAS NOT SPECIFIED FOR THE MODULE, OR THE MOD OPERAND WAS NOT SPECIFIED ON THE EXEC STATEMENT FOR A LINK-EDIT STEP.

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME

When SMP/E was processing a link-edit step, it encountered a module that was not specified on a link-edit NAME statement or on the MOD operand of the EXEC statement for the link-edit procedure. This message is preceded by the command in error and its description.

- If NONAME appears as the step name or job name, the error is in the step or job.
- If the error is on the EXEC statement of the failing step, this message may show the step name or job name that precedes the one in error.

System Action: None.

Programmer Response: Fix the error and rerun the job.

GIM26314E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE LIBRARYDD COMMENT CARD IS MISSING AFTER PATH WAS SPECIFIED ON THE *ddname* DD STATEMENT.

Explanation:

stepname	step name
jobname	job name
ddname	ddname

A link-edit job step contained a DD statement specifying the PATH operand. SMP/E expected the next statement immediately after that DD statement to be a LIBRARYDD comment statement. However, this was not the case. Either the LIBRARYDD comment statement was actually missing, or other statements (such as another DD statement) came between it and the DD statement specifying the PATH operand.

Here is an example where the LIBRARYDD comment statement is missing after the SYSLMOD DD statement.

```
//STEP1 EXEC PGM=IEWBLINK,PARM='RENT,REUS'
//SYSLMOD DD PATH='/path_name1/'
//AOS12   DD DSN=SYS1.AOS12,DISP=SHR <-- need LIBRARYDD
//SYSLIN  DD *
   INCLUDE AOS12(MOD00001)
   INCLUDE AOS12(MOD00002)
   ENTRY MOD00001
   NAME LMOD01(R)
/*
```

This is JCL updated to include the missing LIBRARYDD comment:

```
//STEP1 EXEC PGM=IEWBLINK,PARM='RENT,REUS'
//SYSLMOD DD PATH='/path_name1/'
//*LIBRARYDD=BPXLOAD1
//AOS12   DD DSN=SYS1.AOS12,DISP=SHR
//SYSLIN  DD *
   INCLUDE AOS12(MOD00001)
   INCLUDE AOS12(MOD00002)
   ENTRY MOD00001
   NAME LMOD01(R)
/*
```

System Action: SYSMOD processing stops.

Programmer Response: Include the appropriate LIBRARYDD comment statement immediately after the DD statement specifying the PATH operand, and rerun the job.

GIM26314S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. THE LIBRARYDD COMMENT CARD IS MISSING AFTER PATH WAS SPECIFIED ON THE *ddname* DD STATEMENT.

Explanation:

stepname	step name
jobname	job name
ddname	ddname

SMP/E was processing the JCLIN command. A link-edit job step contained a DD statement specifying the PATH operand. SMP/E expected the next statement immediately after that DD statement to be a LIBRARYDD comment statement. However, this was not the case. Either the LIBRARYDD comment statement was actually missing, or other statements (such as another DD statement) came between it and the DD statement specifying the PATH operand.

Here is an example where the LIBRARYDD comment statement is missing after the SYSLMOD DD statement.

```
//STEP1 EXEC PGM=IEWBLINK,PARM='RENT,REUS'
//SYSLMOD DD PATH='/path_name1/'
//AOS12   DD DSN=SYS1.AOS12,DISP=SHR <-- need LIBRARYDD
//SYSLIN  DD *
   INCLUDE AOS12(MOD00001)
   INCLUDE AOS12(MOD00002)
   ENTRY MOD00001
   NAME LMOD01(R)
/*
```

This is JCL updated to include the missing LIBRARYDD comment:

```
//STEP1 EXEC PGM=IEWBLINK,PARM='RENT,REUS'
//SYSLMOD DD PATH='/path_name1/'
//*LIBRARYDD=BPXLOAD1
//AOS12   DD DSN=SYS1.AOS12,DISP=SHR
//SYSLIN  DD *
   INCLUDE AOS12(MOD00001)
   INCLUDE AOS12(MOD00002)
   ENTRY MOD00001
   NAME LMOD01(R)
/*
```

System Action: Command processing stops.

Programmer Response: Include the appropriate LIBRARYDD comment statement immediately after the DD statement specifying the PATH operand, and rerun the job.

GIM26315E AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. NO VALUE WAS SPECIFIED ON THE LIBRARYDD COMMENT STATEMENT.

Explanation:

stepname	step name
jobname	job name

A link-edit job step contained a DD statement specifying the PATH operand. The next statement immediately after that DD statement was a LIBRARYDD comment statement, as SMP/E requires, but no ddname was specified on the comment.

System Action: SYSMOD processing stops.

Programmer Response: Include the appropriate ddname on the LIBRARYDD comment, and rerun the job.

GIM26315S AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *stepname* IN JOB *jobname*. NO VALUE WAS SPECIFIED ON THE LIBRARYDD COMMENT STATEMENT.

Explanation:

stepname	step name
jobname	job name

SMP/E was processing the JCLIN command. A link-edit job step contained a DD statement specifying the PATH operand. The next statement immediately after that DD statement was a LIBRARYDD comment statement, as SMP/E requires, but no ddname was specified on the comment.

System Action: Command processing stops.

Programmer Response: Include the appropriate ddname on the LIBRARYDD comment, and rerun the job.

GIM26401E THE *dddef* DDDEF ENTRY CANNOT HAVE MORE THAN ONE VOLUME SERIAL NUMBER.

Explanation:

dddef	name of the DDDEF entry

A UCL statement for the indicated DDDEF entry specified more than one volume serial number. This is not allowed. Only an SMPTLIB DDDEF entry can have more than one volume serial number.

System Action: Processing for the UCL statement stops.

Programmer Response: Specify one volume serial number, and rerun the statement.

GIM26501W THE *subentry* SUBENTRY WAS NOT CHANGED.

Explanation:

subentry subentry specified on the IF or CHANGE statement

No changes were made by the ZONEEDIT command because the specified subentry does not exist.

System Action: Processing continues.

Programmer Response: If you want to make the change, specify the correct subentry on the IF or CHANGE statement, then rerun the ZONEEDIT command. Otherwise, do nothing.

GIM26601I LINKAGE EDITOR INPUT CONTAINS AN ERROR.

Explanation: During JCLIN processing, SMP/E found an error in a link-edit statement being scanned. A prior message indicates the error.

System Action: JCLIN processing stops.

Programmer Response: Fix the error and rerun the job.

GIM26602I COPY INPUT CONTAINS AN ERROR.

Explanation: During JCLIN processing, SMP/E found an error in a copy statement being scanned. A prior message indicates the error.

System Action: JCLIN processing stops.

Programmer Response: Fix the error and rerun the job.

GIM26603I ASSEMBLER INPUT CONTAINS AN ERROR.

Explanation: During JCLIN processing, SMP/E found an error in an assembler statement being scanned. A prior message indicates the error.

System Action: JCLIN processing stops.

Programmer Response: Fix the error and rerun the job.

GIM26701E DIRECTORY SPACE FOR THE *library* LIBRARY WAS EXCEEDED DURING AN ATTEMPT TO STORE *enttype* ENTRY *entname*.

Explanation:

library ddname of the library being processed
enttype entry type
entname entry name

When SMP/E tried to store the element in the indicated library, it exceeded the number of directory blocks allocated to the library.

System Action: SMP/E does not store the member. Messages that follow this message show SMP/E's action.

Programmer Response: Increase the directory block allocation for the indicated library and rerun the job.

GIM26701I DIRECTORY SPACE FOR THE *library* LIBRARY WAS EXCEEDED DURING AN ATTEMPT TO STORE *enttype* ENTRY *entname*.

Explanation:

library ddname of library being processed
enttype entry type
entname entry name

When SMP/E tried to store the element in the indicated library, it exceeded the number of directory blocks allocated to the library.

System Action: SMP/E does not store the member. Messages that follow this message show the actions taken by SMP/E.

Programmer Response: Increase the directory block allocation for the indicated library and rerun the job if necessary.

GIM26701S DIRECTORY SPACE FOR THE *library* LIBRARY WAS EXCEEDED DURING AN ATTEMPT TO STORE *enttype* ENTRY *entname*.

Explanation:

library ddname of the library being processed
enttype entry type
entname entry name

When SMP/E tried to store the element in the indicated library, it exceeded the number of directory blocks allocated to the library.

System Action: SMP/E does not store the member. Messages that follow this message show SMP/E's action.

Programmer Response: Increase the directory block allocation for the indicated library and rerun the job.

GIM26801E AN I/O ERROR OCCURRED DURING AN ATTEMPT TO STORE *elmtype* *elmname* ON THE *library* LIBRARY.

Explanation:

elmtype element type

elmname element name
library ddname of library

An I/O error occurred when SMP/E tried to store the element on the indicated library.

System Action: The element is not stored. Processing ends for the indicated element.

Programmer Response: Fix the error and rerun the job.

GIM26801T AN I/O ERROR OCCURRED DURING AN ATTEMPT TO STORE *enttype entname* ON THE *library* LIBRARY.

Explanation:
enttype entry type
entname entry name
library ddname of the library being processed

An I/O error occurred when SMP/E tried to store the entry on the indicated library.

System Action: SMP/E does not store the entry. SMP/E processing ends, as shown by the messages that follow this message.

Programmer Response: Fix the error and rerun the job.

GIM26901T AN I/O ERROR OCCURRED DURING BLDL PROCESSING FOR *enttype entname* ON THE *library* LIBRARY.

Explanation:
enttype entry type
entname entry name
library ddname of the library being processed

A BLDL operation caused an I/O error on the indicated library.

System Action: SMP/E processing fails.

Programmer Response: Fix the error and rerun the job.

GIM26902T AN I/O ERROR OCCURRED DURING BLDL PROCESSING FOR *enttype entname* ON THE *zonename* ZONE.

Explanation:
enttype entry type
entname entry name
zonename name of the zone being processed

A BLDL operation caused an I/O error on the indicated zone.

System Action: SMP/E processing fails.

Programmer Response: Fix the error and rerun the job.

GIM27001I THE *library* LIBRARY WAS NOT PROTECTED BECAUSE RACF IS NOT ACTIVE OR NOT INSTALLED.

Explanation:
library ddname of the library being processed

PROTECT was specified in the DDDEF entry for the indicated data set so that it would be RACF-protected. SMP/E successfully allocated the data set. However, because RACF is not active or installed, the data set was not RACF-protected.

Note: No data set name was specified in the DDDEF entry or DD statement for the library. Therefore, this message shows the ddname instead of the system-generated data set name.

System Action: Processing continues.

Programmer Response:

- If the data set you allocated is new and will be kept after the job step (such as the SMPTLIB data sets), and you want the data set to be RACF-protected, do **one** of the following:
  - Activate RACF, if it was not active.
  - Install RACF and rerun the job.
  - RACF-protect the data set outside of SMP/E.
- If the data set will not be kept after the job step, or if the data set does not have to be protected, no action is required.

GIM27002I DATA SET *dataset* WAS NOT PROTECTED BECAUSE RACF IS NOT ACTIVE OR NOT INSTALLED.

Explanation:
dataset data set name of the library being processed

PROTECT was specified in the DDDEF entry for the indicated data set so that it would be RACF-protected. SMP/E successfully allocated the data set. However, because RACF is not active or installed, the data set was not RACF-protected.

System Action: Processing continues.

Programmer Response:

- If the data set you allocated is new and will be kept after the job step (such as the SMPTLIB data sets), and you want the data set to be RACF-protected, do **one** of the following:
  - Activate RACF, if it was not active.
  - Install RACF and rerun the job.
  - RACF-protect the data set outside of SMP/E.
- If the data set will not be kept after the job step, or if the data set does not have to be protected, no action is required.

GIM27101E SMPTLIB *smptlib* WAS NOT PROTECTED BECAUSE THE USER IS NOT DEFINED TO RACF.

Explanation:

smptlib data set name of the SMPTLIB data set

During RECEIVE processing, SMP/E attempted to allocate the indicated SMPTLIB data set with RACF protection because PROTECT was specified in the DDDEF entry. However, the user who submitted the job was not defined to RACF, so the data set could not be protected.

System Action: Command processing stops.

Programmer Response:

- If you want the data set to be RACF-protected, define the user to RACF.
- If the data set does not have to be protected, remove the PROTECT operand from the SMPTLIB DDDEF.

Fix the error and rerun the RECEIVE command.

GIM27201S THE *operand* OPERAND IS NOT ALLOWED IN REJECT MASS MODE.

Explanation:

operand DELETEFMID or TARGETZONE

DELETEFMID and TARGETZONE cannot be specified during mass mode REJECT processing.

- DELETEFMID is only allowed for NOFMID mode.
- TARGETZONE is only allowed for PURGE mode.

System Action: REJECT processing stops.

Programmer Response:

- If you want the processing indicated by the DELETEFMID or TARGETZONE operands, see *SMP/E Commands* for an explanation of these operands and the modes of REJECT processing.
- Remove DELETEFMID or TARGETZONE from the REJECT command.

Fix the error and rerun the REJECT command.

GIM27301W THE ELEMENT OR JCLIN DATA (TEXT OR OBJECT) IS MISSING.

Explanation:

During command processing, one of the following occurred:

- SMP/E could not find either inline text or object records that were to follow an SMP/E element MCS.
- SMP/E could not find any input in the JCLIN input file. This file may be the FROMDS, RELFILE, SMPJCLIN, or TXLIB data set, or it may refer to data packaged inline after the ++JCLIN MCS.

System Action: Processing continues with the next MCS.

Programmer Response: Do **one** of the following:

- Add an object deck or text deck after the element MCS statements for the elements that are not in a FROMDS, LKLIB, RELFILE, or TXLIB data set.
- Add JCL to the JCLIN input data set.

Then rerun the command.

GIM27301E THE ELEMENT OR JCLIN DATA (TEXT OR OBJECT) IS MISSING.

Explanation:

During command processing, one of the following occurred:

- SMP/E could not find either inline text or object records that were to follow an SMP/E element MCS.
- SMP/E could not find any input in the JCLIN input file. This file may be the FROMDS, RELFILE, SMPJCLIN, or TXLIB data set, or it may refer to data packaged inline after the ++JCLIN MCS.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Add an object deck or text deck after the element MCS statements for the elements that are not in a FROMDS, LKLIB, RELFILE, or TXLIB data set.
- Add JCL to the JCLIN input data set.

Then rerun the command.

GIM27301S THE ELEMENT OR JCLIN DATA (TEXT OR OBJECT) IS MISSING.

Explanation:

During command processing, one of the following occurred:

- SMP/E could not find either inline text or object records that were to follow an SMP/E element MCS.
- SMP/E could not find any input in the JCLIN input file. This file may be the FROMDS, RELFILE, SMPJCLIN, or TXLIB data set, or it may refer to data packaged inline after the ++JCLIN MCS.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Add an object deck or text deck after the element MCS statements for the elements that are not in a FROMDS, LKLIB, RELFILE, or TXLIB data set.
- Add JCL to the JCLIN input data set.

Then rerun the command.

GIM27401E **I/O ERROR -** *jobname*, *stepname*, *address*, *device*, *lib*, *action*, *description*, *access_method*, *optional_PDSE_data* **I/O ERROR -** *jobname*, *stepname*, *address*, *device*, *lib*, *action*, *description*, *rbn*, *access_method*, *optional_PDSE_data* **I/O ERROR -** *jobname*, *stepname*, *address*, *device*, *lib*, *action*, *description*, *track_address*, *block_number*, *access_method*, *optional_PDSE_data*

Explanation: An I/O error occurred when SMP/E was processing the indicated data set. The message corresponds to the SYNADAF information listed in *z/OS DFSMS Macro Instructions for Data Sets*, SC26-7408.

If the data set being analyzed is a PDSE, the message also includes the following information:
1. Concatenation number
2. TTR token of member
3. Relative record number
4. SMS return code
5. SMS reason code

System Action: The messages following this message show SMP/E's action.

Programmer Response: Check for correct DCB information on the input DD statement (especially for a non-labeled tape). Fix the error and rerun the job.

GIM27401T **I/O ERROR -** *jobname*, *stepname*, *address*, *device*, *lib*, *action*, *description*, *access_method*, *optional_PDSE_data* **I/O ERROR -** *jobname*, *stepname*, *address*, *device*, *lib*, *action*, *description*, *rbn*, *access_method*, *optional_PDSE_data* **I/O ERROR -** *jobname*, *stepname*, *address*, *device*, *lib*, *action*, *description*, *track_address*, *block_number*, *access_method*, *optional_PDSE_data*

Explanation: An I/O error occurred when SMP/E was processing the indicated data set. The message corresponds to the SYNADAF information listed in *z/OS DFSMS Macro Instructions for Data Sets*, SC26-7408.

If the data set being analyzed is a PDSE, the message also includes the following information:
1. Concatenation number
2. TTR token of member
3. Relative record number
4. SMS return code
5. SMS reason code

System Action: The messages following this message show SMP/E's action.

Programmer Response: Check for correct DCB information on the input DD statement (especially for a non-labeled tape). Fix the error and rerun the job.

GIM27501I **YOU MAY WANT TO REJECT FUNCTION** *fmid* **BECAUSE IT NO LONGER APPLIES TO THE GLOBAL ZONE.**

Explanation:

fmid	FMID of the function SYSMOD

During the NOFMID mode of REJECT processing, SMP/E found that the indicated dependent function was defined by an FMID subentry in the global zone. Because its FMID was not specified on DELETEFMID, the dependent function was not deleted. However, the base function to which this dependent function applies no longer has an FMID subentry in the global zone. Without the base function, there is no need for the dependent function.

System Action: REJECT processing continues. The dependent function is not rejected.

Programmer Response: Because the base function for the dependent function is no longer defined in the global zone, the dependent function is probably not required either. Do **one** of the following:

- If you want to reject the dependent function, run the **REJECT NOFMID DELETEFMID(***fmid***)** command to reject the function and the SYSMODs that apply to it.
- If you do not want to reject the dependent function, no action is required.

GIM27601E *operand1* **AND** *operand2* **ARE MUTUALLY EXCLUSIVE OPERANDS.**

Explanation:

operand1	first operand
operand2	second operand

The indicated operands are not allowed on the same UCL statement.

System Action: Processing stops for the UCL statement.

Programmer Response: Fix the error and rerun the job.

GIM27601S *operand1* **AND** *operand2* **ARE MUTUALLY EXCLUSIVE OPERANDS.**

Explanation:

operand1	first operand
operand2	second operand

The indicated operands are not allowed on the same MCS statement.

System Action: SYSMOD processing stops.

Programmer Response: Fix the error and rerun the job.

GIM27601T *operand1* **AND** *operand2* **ARE MUTUALLY EXCLUSIVE OPERANDS.**

Explanation:

operand1	first operand
operand2	second operand

The indicated operands are not allowed on the same command.

System Action: Command processing stops.

Programmer Response: Fix the error and rerun the job.

GIM27602E **A SPECIFIED OPERAND IS NOT ALLOWED FOR THE** *dataset* **DATA SET.**

Explanation:

dataset	data set name

An operand specified on a UCL statement is not allowed for the library that is being updated.

System Action: SMP/E does not update the member being processed in the indicated data set.

Programmer Response: See the description of the UCLIN command in *SMP/E Commands* to see which operands are allowed for the data set being processed.

Fix the error and rerun the job.

GIM27605S *variable* **IS NOT A VALID SUBSTITUTION VARIABLE WITHIN A FIREWALL COMMAND.**

Explanation:

variable	substitution variable

A substitution variable that was specified within a firewall command is not supported. The supported substitution variables are &ACCOUNT;, &PORT;, &PW;, &REMOTE_ACCOUNT;, &REMOTE_HOST;, &REMOTE_PORT;, &REMOTE_PW;, &REMOTE_USER;, AND &USER;.

System Action: Processing stops.

Programmer Response: Correct the error and rerun the job.

GIM27650S *keyword1* **AND** *keyword2* **ARE MUTUALLY EXCLUSIVE.**

Explanation:

keyword1	first keyword
keyword2	second keyword

The file attribute file cannot contain both of the indicated keywords.

System Action: Processing stops.

Programmer Response: Correct the error and rerun the job

GIM27701W *enttype* **ENTRY** *entname* **DOES NOT EXIST. THE REPLACE OPERATION HAS BEEN CHANGED TO AN ADD.**

Explanation:

enttype	entry type
entname	entry name or *prodid vv.rr.mm* for PRODUCT entries

SMP/E could not find the entry specified on the UCL REP statement. It assumed you wanted to add the entry.

System Action: Processing continues.

Programmer Response: Do **one** of the following:

- If you did want to add the indicated entry, no action is required.
- If you did not want to add the indicated entry, do the following:
  - Check whether the UCL REP statement specifies the correct entry name.
  - Check whether the preceding SET BDY command specifies the correct zone.
  - Check that the ZONEINDEX subentry in the global zone specifies the correct SMPCSI data set.

Fix the error and rerun the statement.

GIM27801S **THE EXCLUDEZONE OPERAND SPECIFIED ALL THE ZONES IN THE ZONEINDEX SUBENTRIES. NO ZONES ARE LEFT TO CHECK.**

Explanation:

During REJECT processing, SMP/E found that EXCLUDEZONE listed all the zones specified by zone indexes in the global zone. There were no zones left for SMP/E to check.

System Action: REJECT processing stops. Processing continues with the next command.

Programmer Response: Do **one** of the following:

- Look at the zones listed in the EXCLUDEZONE operand. If you want SMP/E to check any of the zones listed there during REJECT processing, remove that zone from the EXCLUDEZONE operand.
- If you are trying to force SMP/E to reject SYSMODs, use BYPASS with APPLYCHECK and ACCEPTCHECK in SELECT mode to skip the check processing.

Then rerun the command.

GIM27901S A VSAM ERROR OCCURRED DURING AN ATTEMPT TO ACCESS AN SMPCSI DATA SET – VPLFUNCT = *vplfunct*.

Explanation:
vplfunct VPLFUNCT value

A VSAM error occurred when SMP/E tried to access an SMPCSI data set.

System Action: Generally, SMP/E will continue processing. However, system action depends on the severity of the VSAM error.

Programmer Response: Follow these steps:

1. Look up any other messages you got in conjunction with this message. You may be able to solve the problem without obtaining additional information.
2. Report the problem to the IBM Support Center.

Chapter 3, "Diagnosing SMP/E Problems" contains more information about how to handle VSAM problems.

GIM28002I SMPTLIB *smptlib* WAS NOT DELETED BECAUSE VOLUME INFORMATION WAS NOT SPECIFIED.

Explanation:
smptlib SMPTLIB data set name

The command being processed required SMP/E to delete the indicated SMPTLIB data set. However, SMP/E could not find the data set through the catalog, and no volume information was provided in an SMPTLIB DDDEF entry or DD statement.

System Action: Command processing continues.

Programmer Response: Decide whether it is really necessary to delete the SMPTLIB data set. If so, delete the data set outside of SMP/E.

GIM28101E SYSMOD *sysmod* CONTAINS MORE THAN ONE MCS FOR ELEMENT *elmname*. ONLY ONE IS ALLOWED.

Explanation:
sysmod SYSMOD ID
elmname element name

During RECEIVE processing, SMP/E found two MCS statements that specified the same element in one SYSMOD.

System Action: SYSMOD processing stops.

Programmer Response: Fix the MCS statements so that they do not specify the same element. Then rerun the job.

GIM28201S ZONE *zonename* WAS NOT PROCESSED BECAUSE IT IS A *zonetype* ZONE AND *operand* WAS SPECIFIED.

Explanation:
zonename zone name
zonetype DLIB or TARGET
operand TARGETZONE or DLIBZONE

During REPORT processing, SMP/E found that a zone specified on the FORZONE operand is the wrong type of zone for the zone type that was specified. For example, TARGETZONE was specified, but the indicated zone is a distribution zone, or DLIBZONE was specified and the indicated zone is a target zone.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:

- Make sure all the zones specified on the FORZONE operand are target zones if TARGETZONE was specified, or are distribution zones if DLIBZONE was specified.
- Do not specify FORZONE. In this case, SMP/E will automatically process only the zones that are the same type as one you specified.

GIM28301E ALLOCATION FAILED FOR *dataset*, WHICH IS REQUIRED IN ORDER TO PROCESS LOAD MODULE *loadmod*. THE SYSMOD IS *sysmod*.

Explanation:
dataset ddname of the required data set
loadmod load module name
sysmod SYSMOD ID

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

Note: For specific information about the allocation failure and how to correct the problem, see the explanations of the previous allocation failure messages.

GIM28302W ALLOCATION FAILED FOR *dataset*, WHICH IS REQUIRED FOR COMPRESS PROCESSING.

Explanation:
dataset ddname of the required data set

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

Note: For specific information about the allocation failure and how to correct the problem, see the explanations of the previous allocation failure messages.

GIM28303E ALLOCATION FAILED FOR *dataset*, WHICH IS REQUIRED IN ORDER TO PROCESS SYSMOD *sysmod*.

Explanation:

dataset	ddname of the required data set
sysmod	SYSMOD ID

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

Note: For specific information about the allocation failure and how to correct the problem, see the explanations of the previous allocation failure messages.

GIM28304I ALLOCATION FAILED FOR SMPLOGA.

Explanation: The SMPLOGA data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the SMPLOGA DD statement.
- Create an SMPLOGA DDDEF entry.
- Correct the DDDEF entry for SMPLOGA.

Then rerun the job.

GIM28305T THE SMPCSI DD STATEMENT IS MISSING.

Explanation: The SMPCSI DD statement is missing or specifies the wrong ddname. This DD statement points to the SMPCSI data set that contains the global zone.

System Action: SMP/E processing fails.

Programmer Response: Do **one** of the following:
- Add the SMPCSI DD statement.
- Specify the SMPCSI data set name on the CSI operand of the EXEC statement for GIMSMP.
- Correct the ddname.

Then rerun the job.

GIM28306I ALLOCATION FAILED FOR *dataset*.

Explanation:

dataset	ddname of the required data set

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

GIM28306W ALLOCATION FAILED FOR *dataset*.

Explanation:

dataset	ddname of the required data set

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

GIM28306E ALLOCATION FAILED FOR *dataset*.

Explanation:

dataset	ddname of the required data set

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

GIM28306S ALLOCATION FAILED FOR *dataset***.**

Explanation:

dataset	ddname of the required data set

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

GIM28306T ALLOCATION FAILED FOR *dataset***.**

Explanation:

dataset	ddname of the required data set

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

GIM28307E ALLOCATION FAILED FOR *dataset***, WHICH IS REQUIRED IN ORDER TO PROCESS ELEMENT** *element* **IN SYSMOD** *sysmod***.**

Explanation:

dataset	ddname of the required data set
element	element name
sysmod	SYSMOD ID

The indicated data set could not be allocated.

System Action: Subsequent messages in SMPOUT show SMP/E's action.

Programmer Response: Do **one** of the following:
- Add the required DD statement.
- Create a DDDEF entry for the data set.
- Correct the DDDEF entry for the indicated data set.

Then rerun the job.

Note: For specific information about the allocation failure and how to correct the problem, see the explanations of the previous allocation failure messages.

GIM28308E ALLOCATION FAILED FOR *smptload* **WHICH IS REQUIRED IN ORDER TO PROCESS** *elmtype* **ELEMENT** *elmname* **IN SYSMOD** *sysmod***.**

Explanation:

smptload	SMPTLOAD
elmtype	element type
elmname	element name
sysmod	name of SYSMOD containing element

An inline element is an unloaded member and its aliases from a partitioned data set (for example, an inline element for a ++PROGRAM MCS). The original partitioned data set was either a PDS or a PDSE.

The reloading of the element must be done to a data set of the same kind (PDS or PDSE) as the originating data set for the element. If the target or distribution library that is to ultimately receive the element is not of the same type as the originating data set for the element, SMP/E allocates a temporary data set of the proper type to the SMPTLOAD DD statement and invokes the copy utility to reload the element prior to copying it into its target or distribution library.

Since the allocation of the SMPTLOAD data set failed, SMP/E is unable to reload the element to a data set of the proper type. Hence SMP/E processing for the element fails.

System Action: Processing stops for the element and the SYSMOD containing the element.

Programmer Response: Message GIM54701E has been issued previously in the SMP/E output when the allocation error for SMPTLOAD was first encountered. Find this message and determine from it why the allocation failed. Correct the problem and try to install the SYSMOD containing the element again.

GIM28309I ALLOCATION FAILED FOR *ddname* **WHICH WAS MODELED AFTER** *model***.** *ddname* **IS NEEDED IN ORDER TO PROCESS** *elmtype* **ELEMENT** *elmname* **IN SYSMOD** *sysmod***.**

Explanation:

ddname	ddname
model	model ddname
elmtype	element type
elmname	element name
sysmod	name of SYSMOD supplying element

ddname is needed in order to process *elmtype* element *elmname*. The allocation of *ddname* was attempted using *model* as a model. This allocation attempt failed.

System Action: Processing stops for both the element and the SYSMOD supplying the element.

Programmer Response: See the explanations for

previous allocation failure messages that identify why the allocation could not be done.

GIM28310I INFORMATION RETRIEVAL FAILED FOR *ddname*. INFORMATION ABOUT LIBRARY *ddname* IS NEEDED BEFORE SMP/E CAN INSTALL *elmtype* ELEMENT *elmname* FROM SYSMOD *sysmod* INTO *ddname*.

Explanation:

ddname	ddname
elmtype	element type
elmname	element name
sysmod	name of SYSMOD supplying element

Information about library *ddname*, such as DSNTYPE, is needed to process *elmtype* element *elmname*. Because this information could not be obtained, SMP/E was not able to process element *elmname*.

System Action: Processing stops for both the element and the SYSMOD supplying the element.

Programmer Response: See the explanations for previous information retrieval failure messages that indicate why the information about the data set could not be obtained.

GIM28401E THE LOAD MODULE ON THE ZAP NAME STATEMENT FOR MODULE *modname* IN SYSMOD *sysmod* IS NOT AN LMOD SUBENTRY IN THE MOD ENTRY FOR *modname*.

Explanation:

modname	module name
sysmod	SYSMOD ID

The load module name specified on a superzap NAME statement is not in the target zone as a valid load module for the specified module.

System Action: SYSMOD processing stops. Processing continues with the next SYSMOD.

Programmer Response: Fix or remove the load module name from the superzap NAME statement and rerun the job.

Note: If a valid load module name was specified on the ALIAS operand, use the name that is in the target zone and **not** in the ALIAS operand.

GIM28501W COMPRESS PROCESSING FAILED FOR THE *library* LIBRARY BECAUSE IT IS AN UNMOVABLE DATA SET.

Explanation:

library	ddname of the library

SMP/E cannot compress the indicated library because it has location-dependent data.

System Action: The indicated library is not compressed. Processing continues with the next library.

Programmer Response: None.

GIM28600W LMOD *loadmod* CONTAINS MODULES NOT OWNED BY FMID *fmid*. THESE MODULES ARE NOT INCLUDED IN FUNCTION *fmid*, BUT WILL BE IDENTIFIED IN THE JCLIN FOR LMOD *loadmod*.

Explanation:

loadmod	load module name
fmid	FMID

The specified load module contains modules not associated with the FMID being processed by the BUILDMCS command. ++MOD statements are created only for the modules associated with the FMID being processed, but the load module will be completely defined in the JCLIN with an INCLUDE statement for **all** component modules.

SMP/E does not have enough information to determine if the FMID being processed defined the load module with JCLIN. Therefore, the JCLIN is created by the BUILDMCS command.

System Action: Processing continues.

Programmer Response: If the FMID being processed should not contain JCLIN to define the specified load module, you must edit the JCLIN to remove the definition of this load module.

GIM28701S THE COMMAND WAS NOT PROCESSED BECAUSE A RETURN CODE SPECIFIED ON THE RC OPERAND WAS EXCEEDED.

Explanation: The RC operand, which determines the processing of the current command, was used to specify a return code for another SMP/E command. SMP/E found that the return code for that command was greater than the specified limit.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Look at the SMPOUT data to find what command caused SMP/E to bypass the current command. Fix any errors and decide whether to rerun the job.

GIM28801I TARGET ZONE *zonename* WAS NOT UPDATED BECAUSE JCLIN PROCESSING FAILED.

Explanation:
zonename target zone name

An error shown by a previous message caused JCLIN processing to end without updating the target zone.

System Action: Processing continues.

Programmer Response: Fix the error and rerun JCLIN processing.

GIM29000W SYSMOD *sysmod1* IS *action* IN ERROR, BUT WILL BE SUPERSEDED BY FUNCTION *sysmod2*.

Explanation:
sysmod1 SYSMOD ID
action APPLIED or ACCEPTED
sysmod2 Function SYSMOD ID

SYSMOD *sysmod1* is applied or accepted in error. However, during BUILDMCS command processing, it will be superseded by function SYSMOD *sysmod*.

System Action: Processing continues.

Programmer Response: Since the indicated SYSMOD is applied or accepted in error, the information in the target or distribution zone associated with the SYSMOD may be incorrect. You should verify that the superseding function SYSMOD created by the BUILDMCS command is correct, or APPLY or ACCEPT the indicated SYSMOD correctly and then rerun the BUILDMCS command.

GIM29100E SYSMOD *sysmod* IS NOT A FUNCTION. SYSMODS SPECIFIED ON THE FORFMID OPERAND MUST BE FUNCTIONS.

Explanation:
sysmod SYSMOD ID

A SYSMOD ID specified on the FORFMID operand of the BUILDMCS command was not a function.

System Action: BUILDMCS processing continues with the next FMID.

Programmer Response: None.

GIM29201E THE MEMBER NAME ON THE UPDATE CONTROL STATEMENT DOES NOT MATCH THE MEMBER NAME ON THE ELEMENT UPDATE MCS.

Explanation: The member name on the control statement for the update utility did not match the name on the element update MCS.

System Action: RECEIVE processing stops for the SYSMOD.

Programmer Response: Correct the update control statement and rerun the job.

GIM29300E SYSMOD *sysmod* WAS NOT PROCESSED BECAUSE *sysmod* IS NOT DEFINED IN ZONE *zonename*.

Explanation:
sysmod SYSMOD ID
zonename zone name

A SYSMOD specified on the FORFMID operand of the BUILDMCS command was not defined in the zone specified on the SET command.

System Action: BUILDMCS processing continues with the next SYSMOD.

Programmer Response: None.

GIM29400E SYSMOD *sysmod1* WAS NOT PROCESSED BECAUSE *sysmod1* WAS DELETED BY *sysmod2*.

Explanation:
sysmod1 SYSMOD ID
sysmod2 SYSMOD ID

A SYSMOD specified on the FORFMID operand of the BUILDMCS command was deleted by another SYSMOD.

System Action: BUILDMCS processing continues with the next SYSMOD.

Programmer Response: None.

GIM29500E SYSMOD *sysmod1* WAS NOT PROCESSED BECAUSE *sysmod1* WAS SUPERSEDED BY *sysmod2*.

Explanation:
sysmod1 SYSMOD ID
sysmod2 SYSMOD ID

A SYSMOD specified on the FORFMID operand of the BUILDMCS command was superseded by another SYSMOD.

System Action: BUILDMCS processing continues with the next SYSMOD.

Programmer Response: None.

GIM29600E SYSMOD *sysmod* WAS NOT PROCESSED BECAUSE *sysmod* WAS IN ERROR.

Explanation:
sysmod SYSMOD ID

A SYSMOD specified on the FORFMID operand of the BUILDMCS command was in error.

System Action: BUILDMCS processing continues with the next SYSMOD.

Programmer Response: None.

GIM29700E SYSMOD *sysmod* CONTAINS ELEMENTS OR JCLIN USING THE RELFILE AND FROMDS OPERANDS. ONLY ONE OF THESE OPERANDS IS ALLOWED IN THE SAME SYSMOD.

Explanation:
sysmod SYSMOD ID

The indicated SYSMOD is packaged using both the RELFILE and FROMDS packaging methods. This means some elements or the JCLIN uses the RELFILE operand, and other elements or the JCLIN uses the FROMDS operand. Only one of these packaging methods may be used within a single SYSMOD.

System Action: RECEIVE processing stops for the indicated SYSMOD.

Programmer Response: Fix the problem and run the RECEIVE again.

GIM29800E CURRENT PATH NAME WAS NOT CHANGED BECAUSE THE NEW NAME EXCEEDS THE 255-CHARACTER MAXIMUM.

Explanation: A path name can have up to 255 characters, excluding any delimiting single quotes. The new name would have been too long, so the path name was not changed.

System Action: The CHANGE statement is not processed.

Programmer Response: Specify a new path name that is less than or equal to 255 characters and rerun the job.

GIM29900S THE PATH TO_VALUE MUST END WITH A SLASH (″/″).

Explanation: The path name specified as the to_value on the CHANGE statement did not end with the required delimiter, a slash (″/″). The path name of the to_value must end in a slash when a full path name is specified or when a wildcard character (*) is used in the from_value and the character immediately preceding the wildcard character is a slash.

System Action: For the CHANGE statement of the ZONEEDIT command, this CHANGE statement and remaining CHANGE statements, up to the ENDZEDIT command, are not processed.

Programmer Response: Fix the CHANGE statement and rerun the job.

GIM30000E AN UNKNOWN COMMAND NAME WAS PASSED TO GIMAPI.

Explanation: The string supplied as the name of the command to GIMAPI is not one of the supported commands.

System Action: Command processing stops.

Programmer Response: Fix the command name and rerun the calling program.

GIM30100E THE CSI PARAMETER OF THE QUERY COMMAND SPECIFIES A DATA SET NAME THAT EXCEEDS 44 CHARACTERS.

Explanation: The CSI parameter specifies a data set name that has more than 44 characters.

System Action: Command processing stops.

Programmer Response: Supply a valid global CSI data set name and rerun the calling program.

GIM30201E *command* PROCESSING FAILED FOR SYSMOD *sysmod*. ALL SYSMODS THAT WOULD HAVE DELETED *sysmod* HAVE FAILED.

Explanation:
command SMP/E command
sysmod SYSMOD ID

SMP/E could not process the indicated SYSMOD because it was supposed to be deleted, but all the SYSMODs that were supposed to delete it failed.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to elements of the SYSMOD. Then fix the error and rerun the job.

GIM30202E *command* PROCESSING FAILED FOR SYSMOD *sysmod*. ALL SYSMODS THAT WOULD HAVE SUPERSEDED *sysmod* HAVE FAILED.

Explanation:
command SMP/E command
sysmod SYSMOD ID

SMP/E could not process the indicated SYSMOD because it was supposed to be superseded, but all the SYSMODs that were supposed to supersede it failed.

System Action: SYSMOD processing stops. SMP/E

does not attempt to process any other elements in the SYSMOD.

Programmer Response: Do the following:

1. See subsequent messages to find the associated SYSMODs that caused the failure.
2. Examine the Causer SYSMOD Summary report to determine why those SYSMODs failed.

Then fix the error and rerun the job.

GIM30203I *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. PROCESSING FOR AN ELEMENT DELETED BY** *sysmod* **FAILED.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E could not process the indicated SYSMOD because an element that was supposed to be deleted was not. The element was supposed to be deleted because one of the following is true:

- The element is part of a function that was specified on the ++VER DELETE operand in the indicated SYSMOD.
- The DELETE operand was specified on the element MCS for the element.

System Action: SYSMOD processing stops.

Programmer Response: Examine SMPOUT for messages that describe the failure. Then fix the error and rerun the job.

GIM30204E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. REQUIRED SYSMODS WERE EXCLUDED.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E could not process the indicated SYSMOD because required SYSMODs were specified on the EXCLUDE operand. If the indicated SYSMOD defined requisites, the excluded SYSMODs would have satisfied those requisites. If the indicated SYSMOD was held for an error reason ID, the excluded SYSMODs would have resolved the reason ID.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Remove the required SYSMODs from the EXCLUDE operand. Then rerun the job.

GIM30206E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. HOLD REASON IDS WERE NOT RESOLVED.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E could not process the indicated SYSMOD because it was held, and one or more of the reason IDs was not resolved. Message GIM359*xx*, which follows this message, names the reason ID and the type of hold.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: See the programmer response for the GIM359*xx* messages that follow.

GIM30208I *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. INLINE JCLIN PROCESSING FAILED.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E could not process the indicated SYSMOD because processing failed for the associated inline JCLIN. Previous messages indicate the reason for the failure. (Inline JCLIN is JCLIN data associated with a ++JCLIN MCS. This data may be packaged inline after the MCS, or it may be in a FROMDS, RELFILE, or TXLIB data set.)

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for previous messages that describe the error. Then fix the error and rerun the job.

GIM30209E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. REQUIRED SYSMODS FAILED OR WERE MISSING.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E could not process the indicated SYSMOD because required SYSMODs failed or were missing. If the indicated SYSMOD defined requisites, the required SYSMODs would have satisfied those requisites. If the indicated SYSMOD was held for an error reason ID, the required SYSMODs would have resolved the reason ID. Message GIM359*xx*, which follows this message, names the required SYSMODs that failed or were missing.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Do **one** of the following:

- Examine SMPOUT for messages relating to requisites of the SYSMOD.
- Use the SYSMOD Status report to find the associated SYSMOD that caused the failure.

Fix the error and rerun the job.

GIM30211E **RESTORE PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE IT IS A CROSS-ZONE REQUISITE FOR OTHER SYSMODS.**

Explanation:

sysmod	SYSMOD ID

SMP/E could not process the indicated SYSMOD because a SYSMOD in another zone specified SYSMOD *sysmod* as a cross-zone requisite. Message GIM35923I follows this message and names the causer SYSMOD that specifies SYSMOD *sysmod* as a cross-zone requisite.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Restore the causers in the zones indicated in message GIM35923I and then restore failing SYSMOD *sysmod*.
- Restore SYSMOD *sysmod* using BYPASS(XZIFREQ) and then restore the causers in the zones indicated in message GIM35923I.

GIM30213E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. SYSMOD** *sysmod2***, WHICH IS SUPERSEDED BY SYSMOD** *sysmod1***, MUST ALSO BE SPECIFIED ON THE RESTORE COMMAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	superseded SYSMOD ID

SMP/E did not process SYSMOD *sysmod1* because it supersedes SYSMOD *sysmod2* and SYSMOD *sysmod2* was not specified on the RESTORE command.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Include SYSMOD *sysmod2* on the RESTORE command. Then rerun the job.

GIM30214I *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. AN ERROR OCCURRED WHEN SMP/E TRIED TO UPDATE THE SMPCSI DATA SET.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E processed all the elements in the SYSMOD, but an error occurred when SMP/E tried to add information about superseding SYSMODs and MODIDs. The message before this one shows the cause of the error.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages that describe the error. Then fix the error and rerun the job.

GIM30215I *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. A SYSTEM ABEND OCCURRED.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E did not process the indicated SYSMOD because a system abend occurred. Message GIM43201T indicates the failing program and the abend code.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine message GIM43201T, which describes the error. Then fix the error and rerun the job.

GIM30216I *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. SYSTEM UTILITY PROCESSING FAILED FOR AN ELEMENT IN** *sysmod***.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E did not process the indicated SYSMOD because one of the elements was not processed. The utility that was processing the element issued a return code that exceeded the specified or default maximum return code.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages that de cribe the error. Then fix the error and rerun the job.

GIM30217I *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. RETRANSFORMATION PROCESSING FAILED FOR AN ELEMENT IN** *sysmod***.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E did not process the indicated SYSMOD, which contained data that was transformed by GIMDTS, because processing failed for one of the elements. When SMP/E was processing the element, it issued a return code that exceeded the maximum allowable return code.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages that describe the error. Then fix the error and rerun the job.

GIM30218E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING IS INCOMPLETE FOR SYSMOD** *sysmod2***, WHICH WAS SPECIFIED ON THE ++VER PRE OPERAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing is incomplete for prerequisite SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30219E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING FAILED FOR SYSMOD** *sysmod2***, WHICH WAS SPECIFIED ON THE ++VER PRE OPERAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing failed for prerequisite SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30220E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING IS INCOMPLETE FOR SYSMOD** *sysmod2***, WHICH WAS SPECIFIED ON THE ++VER REQ OPERAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing is incomplete for requisite SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30221E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING FAILED FOR SYSMOD** *sysmod2***, WHICH WAS SPECIFIED ON THE ++VER REQ OPERAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing failed for requisite SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30222E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING IS INCOMPLETE FOR SYSMOD** *sysmod2***, WHICH WAS SPECIFIED ON THE ++IF REQ OPERAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing is incomplete for conditional requisite SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30223E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING FAILED FOR SYSMOD** *sysmod2***, WHICH WAS SPECIFIED ON THE ++IF REQ OPERAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing failed for conditional requisite SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30224E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING IS INCOMPLETE FOR SYSMOD** *sysmod2***, WHICH WAS SPECIFIED ON THE ++VER FMID OPERAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing is incomplete for function SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30225E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING FAILED FOR SYSMOD** *sysmod2***, WHICH WAS SPECIFIED ON THE ++VER FMID OPERAND.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing is incomplete for function SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30226E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1* **BECAUSE PROCESSING IS INCOMPLETE FOR SYSMOD** *sysmod2***. SYSMOD** *sysmod2* **CONTAINS HIGHER LEVELS OF COMMON ELEMENTS.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing is incomplete for SYSMOD *sysmod2*, where *sysmod2* contains higher levels of elements that also exist in SYSMOD *sysmod1*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30227E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1* **BECAUSE PROCESSING FAILED FOR SYSMOD** *sysmod2***. SYSMOD** *sysmod2* **CONTAINS HIGHER LEVELS OF COMMON ELEMENTS.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing failed for SYSMOD *sysmod2*, where *sysmod2* contains higher levels of elements that also exist in SYSMOD *sysmod1*.

Here is an example. Suppose PTF X and PTF Y both contain MOD1. PTF Y specifies PTF X as a prerequisite, and contains MOD2 in addition to MOD1. Because PTF Y specifies PTF X as a prerequisite, PTF Y is said to supply a higher service level of common element MOD1.

Now suppose that PTF Y fails because of a link-edit utility error for MOD2. This means that processing stops for PTF Y, and all elements supplied by PTF Y also fail. Because common element MOD1 fails from PTF Y, MOD1 is not truly installed correctly. Therefore, PTF X fails with message GIM30227 because a higher level of one of its elements (MOD1) failed and was not installed.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Also check the Causer SYSMOD Summary report to see why SYSMOD *sysmod2* failed, and check the Element Summary report to see which common elements were affected. Fix the error and rerun the job.

GIM30228E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1* **BECAUSE PROCESSING IS INCOMPLETE FOR SYSMOD** *sysmod2***. THESE SYSMODS CONTAIN COMMON ELEMENTS.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing is incomplete for SYSMOD *sysmod2*, where *sysmod2* contains elements that also exist in SYSMOD *sysmod1*.

System Action: SMP/E processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Also check the Causer SYSMOD Summary report to see why SYSMOD *sysmod2* failed, and check the Element Summary report to see which common elements were affected. Fix the error and rerun the job.

GIM30229E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1* **BECAUSE PROCESSING FAILED FOR SYSMOD** *sysmod2***. THESE SYSMODS CONTAIN COMMON ELEMENTS.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID

SMP/E could not process SYSMOD *sysmod1* because processing failed for SYSMOD *sysmod2*, where *sysmod2* contains elements that also exist in SYSMOD *sysmod1*.

System Action: SMP/E processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Also check the Causer SYSMOD Summary report to see why SYSMOD *sysmod2* failed, and check the Element Summary report to see which common elements were affected. Fix the error and rerun the job.

GIM30230E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING FAILED FOR CROSS-ZONE REQUISITE SYSMOD** *sysmod2***, WHICH WAS CAUSED BY SYSMOD** *sysmod3* **IN ZONE** *zonename***.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID
sysmod2	related SYSMOD ID
sysmod3	causer SYSMOD ID
zonename	zone name

SMP/E could not process SYSMOD *sysmod1*, because processing failed for cross-zone requisite SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30231E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. PROCESSING IS INCOMPLETE FOR CROSS-ZONE REQUISITE SYSMOD** *sysmod2***, WHICH WAS CAUSED BY SYSMOD** *sysmod3* **IN ZONE** *zonename***.**

Explanation:

command	SMP/E command
sysmod1	SYSMOD ID

| *sysmod2* requisite SYSMOD ID
| *sysmod3* causer SYSMOD ID
| *zonename* zone name

| SMP/E could not process SYSMOD *sysmod1*, because
| processing is incomplete for cross-zone requisite
| SYSMOD *sysmod2*.

| **System Action:** SYSMOD processing stops. SMP/E
| does not attempt to process any other elements in the
| SYSMOD.

| **Programmer Response:** Examine SMPOUT for
| messages relating to SYSMOD *sysmod2*. Fix the error
| and rerun the job.

GIM30232E **RESTORE PROCESSING FAILED FOR SYSMOD** *sysmod1* **BECAUSE SYSMOD** *sysmod2* **REQUIRES SYSMOD** *sysmod1***.** *sysmod1* **COULD NOT BE RESTORED BECAUSE** *sysmod2***. FAILED.**

Explanation:
sysmod1 SYSMOD ID
sysmod2 SYSMOD ID

SMP/E could not process SYSMOD *sysmod1*, because processing failed for SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30233E **RESTORE PROCESSING FAILED FOR SYSMOD** *sysmod1* **BECAUSE SYSMOD** *sysmod2* **REQUIRES SYSMOD** *sysmod1***.** *sysmod1* **COULD NOT BE RESTORED BECAUSE** *sysmod2* **IS INCOMPLETE.**

Explanation:
sysmod1 SYSMOD ID
sysmod2 SYSMOD ID

SMP/E could not process SYSMOD *sysmod1*, because processing is incomplete for SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

System Action: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30234E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. SYSMOD** *sysmod2* **CONTAINS A SHELL SCRIPT NEEDED TO PROCESS AN ELEMENT IN SYSMOD** *sysmod1***, BUT PROCESSING IS INCOMPLETE FOR SYSMOD** *sysmod2***.**

Explanation:
command SMP/E command
sysmod1 SYSMOD ID
sysmod2 related SYSMOD ID

An element in SYSMOD *sysmod1* specifies that a shell script in SYSMOD *sysmod2* is to be invoked to complete the element's installation. SMP/E cannot process SYSMOD *sysmod1*, however, because processing was incomplete for SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30235E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod1***. SYSMOD** *sysmod2* **CONTAINS A SHELL SCRIPT NEEDED TO PROCESS AN ELEMENT IN SYSMOD** *sysmod1***, BUT PROCESSING FAILED FOR SYSMOD** *sysmod2***.**

Explanation:
command SMP/E command
sysmod1 SYSMOD ID
sysmod2 related SYSMOD ID

An element in SYSMOD *sysmod1* specifies that a shell script in SYSMOD *sysmod2* is to be invoked to complete the element's installation. SMP/E cannot process SYSMOD *sysmod1*, however, because processing failed for SYSMOD *sysmod2*.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine SMPOUT for messages relating to SYSMOD *sysmod2*. Fix the error and rerun the job.

GIM30236I *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. SHELL SCRIPT PROCESSING FAILED FOR AN ELEMENT IN SYSMOD** *sysmod***.**

Explanation:
command SMP/E command
sysmod SYSMOD ID

SMP/E did not process the indicated SYSMOD because shell script processing failed for one of the SYSMOD's elements.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Examine the messages in SMPOUT to determine why shell script processing failed for the element. Fix the error and rerun the job.

GIM30237S THE RETRIEVAL OF DATA FROM HIERARCHICAL FILE SYSTEM FILE *filename* FAILED. THE DATA IN THE FILE APPEARS TO BE CORRUPTED OR INCOMPLETE.

Explanation:

filename	name of file

SMP/E was attempting to retrieve the data from the indicated hierarchical file system file and place it into an MVS data set. Checking of the data determined that it was corrupted or incomplete.

System Action: SMP/E stops processing the named file. SMP/E also terminates the process or command that initiated the file retrieval.

Programmer Response: Ensure that the MVS data set being used for retrieval is compatible with the data being retrieved. In other words, ensure that the record format, logical record length, and block size are correct. If no incompatibility problem is found, contact the IBM Support Center.

GIM30301T COMPRESS PROCESSING FAILED FOR THE *library* LIBRARY. THE RETURN CODE WAS *rc*.

Explanation:

library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing
rc	return code

SMP/E called the copy utility to compress the indicated library and received an error return code.

System Action: Command processing fails because the return code is nonzero and greater than one of the following:

- The maximum return code specified in the UTILITY entry
- The default return code

Programmer Response: Check the copy output to find the error. Fix the error and rerun the job.

GIM30302I COMPRESS PROCESSING WAS SUCCESSFUL FOR THE *library* LIBRARY. THE RETURN CODE WAS *rc*.

Explanation:

library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing
rc	return code

SMP/E called the copy utility to compress the indicated library and received a successful return code.

System Action: Processing continues because the return code is zero.

Programmer Response: None.

GIM30302W COMPRESS PROCESSING WAS SUCCESSFUL FOR THE *library* LIBRARY. THE RETURN CODE WAS *rc*.

Explanation:

library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing
rc	return code

SMP/E called the copy utility to compress the indicated library and received a successful return code.

System Action: Processing continues because the return code is not greater than **one** of the following:

- The maximum return code specified in the UTILITY entry
- The default return code

Programmer Response: None.

GIM30400S THE *ddname* DDNAME IS IN USE BY DATA SET *dataset*. THE CSI DATA SET SPECIFIED COULD NOT BE ALLOCATED.

Explanation:

ddname	the DDNAME (SMPCSI for a GLOBAL zone, or the name of the qualified zone contained in a data set other than the SMPCSI)
dataset	data set name allocated to the *ddname* DDNAME

The specified ddname is already in use, so the CSI data set provided by the user could not be allocated.

System Action: Command processing stops.

Programmer Response: Before calling GIMAPI, make sure the DDNAME is free.

GIM30600E THERE IS A SYNTAX ERROR IN THE *parameter* PARAMETER IN CHARACTER POSITION *position*.

Explanation:

parameter	command parameter name
position	character position where error was detected

SMP/E found a syntax error at the specified character position for the parameter specified.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the problem in the parameter and rerun the calling program.

GIM30700S GIMAPI COULD NOT OBTAIN ENOUGH STORAGE TO PROCESS THE *command* COMMAND.

Explanation:

command	command name

While processing the specified command, an attempt to allocate storage could not be satisfied.

Note: Because storage cannot be allocated, this message may not appear in the API message buffer.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:

- Increase the REGION parameter on the EXEC statement for the calling program.
- Increase the REGION parameter on your LOGON panel.

GIM30800E NO VALUE WAS SPECIFIED FOR THE *parameter* PARAMETER OF THE *command* COMMAND.

Explanation:

parameter	command parameter name
command	command name

The indicated command requires the specified parameter to be provided by the calling program. The parameter had no value.

System Action: Command processing stops.

Programmer Response: Be sure an appropriate value is specified as a parameter to the GIMAPI call to invoke the command, and rerun the calling program.

GIM30900E AN ASTERISK CANNOT BE USED WITH ANY OTHER VALUES ON THE *parameter* PARAMETER.

Explanation:

parameter	command parameter name

An asterisk was included on the specified parameter of the API QUERY command along with other values. The asterisk must be used alone.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Change the parameter to use either the asterisk only or the specific values, then rerun the calling program.

GIM31000E AN UNDEFINED ENTRY TYPE OF *enttype* WAS SPECIFIED ON THE ENTRY PARAMETER.

Explanation:

enttype	entry type

The entry specified is not a valid CSI entry. Refer to *SMP/E User's Guide* to see the entries supported by SMP/E.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the entry type and rerun the calling program.

GIM31100W THE VALUE *value* HAS BEEN SPECIFIED MORE THAN ONCE ON THE *parameter* PARAMETER. ALL BUT THE FIRST OCCURRENCE IS IGNORED.

Explanation:

value	value being repeated in the parameter
parameter	command parameter name

The indicated value was found more than once on either the *zone*, *entry*, or *subentry* parameter of the GIMAPI QUERY command. The duplicate value is ignored during command processing.

System Action: Command processing continues.

Programmer Response: Remove the duplicate value from the parameter prior to running the calling program again, or ignore the message.

GIM31200E AN UNDEFINED SUBENTRY TYPE OF *subtype* WAS SPECIFIED ON THE *parameter* PARAMETER.

Explanation:

subtype	subentry type
parameter	command parameter name

A subentry specified on either the *subentry* or *filter* parameter of the QUERY command is not defined for any entry.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the subentry type and rerun the calling program.

GIM31300E COMPARISON OPERATOR *oper* CANNOT BE USED WITH SUBENTRY TYPE *subtype*. ONLY TIME, DATE, AND RETURN CODE SUBENTRIES CAN BE COMPARED WITH THE >, >=, < OR <= OPERATORS.

Explanation:

oper	comparison operator
subtype	subentry type

The >, >=, <, and <= operators can only be used in comparisons with subentry types that are times, dates, or return codes.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the filter and rerun the calling program.

GIM31400E SUBENTRY TYPE *subtype* CANNOT BE USED ON THE FILTER PARAMETER.

Explanation:

subtype	subentry type

The following subentry types can be used on the *filter* parameter only when a null value (that is, two quotes with no blanks between them) is specified for the subentry type (for example, HOLDDATA!=''):

- ASMIN
- HRFPARM
- HOLDDATA
- LECNTL
- LEPARM
- LMODALIAS
- UTILPARM
- ZDESC

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the *filter* parameter and rerun the calling program.

GIM31500E THE DATE VALUE IN COLUMN *colnum* DID NOT USE THE FORMAT YYDDD.

Explanation:

colnum	column where incorrect date format was specified

A fixed value for a comparison against a date subentry was not in the proper format of YYDDD. There were non-numeric characters in the value.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the format of the date in the *filter* parameter an rerun the calling program.

GIM31600E THE TIME VALUE IN COLUMN *colnum* DID NOT USE THE FORMAT 'HH:MM:SS'.

Explanation:

colnum	column where incorrect date format was specified

A fixed value for a comparison against a time subentry was not in the proper format of 'HH:MM:SS' where:

HH	Numeric hours from 00 to 23
MM	Numeric minutes from 00 to 59
SS	Numeric seconds from 00 to 59

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the format of the date in the *filter* parameter and rerun the calling program.

GIM31700E A CLOSING QUOTE IS MISSING FROM A FILTER VALUE.

Explanation: Single quotes are required around fixed values in the *filter* parameter of the API QUERY command. The end of the *filter* parameter was encountered before a closing quote was found.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the *filter* parameter then rerun the calling program.

GIM31800E A LEFT PARENTHESIS WAS NOT CLOSED WITH A CORRESPONDING RIGHT PARENTHESIS.

Explanation: The end of the *filter* parameter was encountered before a right parenthesis was found to match an existing left parenthesis.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the *filter* parameter then rerun the calling program.

GIM31900E THE RETURN CODE VALUE IN COLUMN *colnum* IS NOT A NUMERIC VALUE OF NO MORE THAN TWO CHARACTERS IN LENGTH.

Explanation:

colnum column where the incorrect return code format was specified

A fixed value for a comparison against a RETURN CODE subentry was specified incorrectly. The value must be no more than two characters in length, and must contain only numerics (0-9).

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the RETURN CODE subentry value in the filter parameter and rerun the calling program.

GIM31901I SYSMOD *sysmod1* **DOES NOT SPECIFY** *sysmod2* **ON THE PRE OR SUP OPERAND.** *sysmod2* **IS THE RMID FOR** *elmtype elmname* **THAT IS CURRENTLY INSTALLED.**

Explanation:

sysmod1 ID of the SYSMOD being installed
sysmod2 ID of the SYSMOD specified as an RMID
elmtype element type
elmname element name

This message describes the ID check reported by message GIM38201W, GIM38201E, GIM38202W, or GIM38202E. SMP/E issues GIM31901I for every element in a service SYSMOD (or in a function SYSMOD that is being reinstalled) that does not name, in the PRE or SUP operands, the RMID of the previously processed version of the named element. When this happens, SMP/E cannot determine the relationship between the element in SYSMOD *sysmod1* and the previously processed version of the element. You may also get this message if you are trying to reinstall a SYSMOD after you have already installed additional service for it.

System Action: This message is informational. See the preceding GIM38201W, GIM38201E, GIM38202W, or GIM38202E message to determine the system action.

Programmer Response: Determine the relationship between the elements in the SYSMODs involved. If the relationship is not correctly defined, do **one** of the following:

- List SYSMOD *sysmod1* to see if it was already installed.
- Reject SYSMOD *sysmod1* and change the PRE and SUP operands that were specified.
- Use UCLIN to modify the RMID subentry for the affected element in the target zone or distribution zone.
- If necessary, install related SYSMODs before processing SYSMOD *sysmod1* to establish the correct relationship.

GIM31902I SYSMOD *sysmod1* **DOES NOT SPECIFY** *sysmod2* **ON THE PRE OR SUP OPERAND.** *sysmod2* **IS A UMID FOR** *elmtype elmname* **THAT IS CURRENTLY INSTALLED.**

Explanation:

sysmod1 ID of the SYSMOD being installed
sysmod2 ID of the SYSMOD specified as a UMID
elmtype element type
elmname element name

This message describes the ID check reported by message GIM38201W, GIM38201E, GIM38202W, or GIM38202E. SMP/E issues GIM31902I for every element in a service SYSMOD or USERMOD (or in a function SYSMOD that is being reinstalled) that does not name, in the PRE or SUP operands, a UMID of the previously processed version of the named element. When this happens, SMP/E cannot determine the relationship between the element in SYSMOD *sysmod1* and the previously processed version of the element. You may also get this message if you are trying to reinstall a SYSMOD after you have already installed additional service for it.

System Action: This message is informational. See the preceding GIM38201W, GIM38201E, GIM38202W, or GIM38202E message to determine the system action.

Programmer Response: If the SYSMOD being installed provides an update (not a replacement) for the element in question, it is not necessary to specify a relationship with any of the UMIDs for the element being updated. (For more information on how this applies to USERMODs, see the *SMP/E User's Guide*. For more information on the checking SMP/E does, see the MODID verification sections in the APPLY and ACCEPT chapters in *SMP/E Commands*.) Even though SMP/E has installed the update, you should determine the relationship between the elements in the SYSMODs involved.

- If the SYSMOD containing the update should not have been installed, follow these steps:
  1. Run the RESTORE command to remove the SYSMOD (and any associated SYSMODs that must also be restored).
  2. Run the REJECT command to remove the SYSMOD from the global zone.
  3. Update the ++VER MCS in the SYSMOD to include the appropriate UMIDs on the PRE or SUP operand.
  4. Run the RECEIVE command again to receive the corrected SYSMOD.
  5. Run the APPLY command again to install the corrected SYSMOD.

- If the SYSMOD can remain installed, but the relationship between the elements in the SYSMODs involved needs to be defined, follow the instructions below.

If the SYSMOD being installed provides a replacement (not an update) for the element in question, determine the relationship between the elements in the SYSMODs involved. If the relationship is not correctly defined, do **one** of the following:

- List SYSMOD *sysmod1* to see if it was already installed.
- Reject SYSMOD *sysmod1* and change the PRE and SUP operands that were specified. Then receive the SYSMOD again and rerun the job to install it.

  Note: Because SYSMOD *sysmod1* replaces the indicated element, it must specify a relationship with all the UMIDs for the element being replaced.
- Use UCLIN to modify the UMID subentries for the affected element in the target zone or distribution zone. Then rerun the job to install SYSMOD *sysmod1*.
- If necessary, install related SYSMODs before processing SYSMOD *sysmod1* to establish the correct relationship. Then rerun the job to install SYSMOD *sysmod1*.

GIM31903I **SYSMOD** *sysmod1* **DOES NOT SPECIFY** *sysmod2* **ON THE PRE OR SUP OPERAND.** *sysmod2* **IS THE RMID FOR** *elmtype elmname* **THAT IS BEING INSTALLED.**

Explanation:

sysmod1	ID of the SYSMOD being installed
sysmod2	ID of the SYSMOD specified as an RMID
elmtype	element type
elmname	element name

This message describes the ID check reported by message GIM38201W, GIM38201E, GIM38202W, or GIM38202E. SMP/E issues GIM31903I for every element in a service SYSMOD (or in a function SYSMOD that is being reinstalled) that does not name, in the PRE or SUP operands, the RMID of the previously processed version of the named element. When this happens, SMP/E cannot determine the relationship between the element in SYSMOD *sysmod1* and the previously processed version of the element. You may also get this message if you are trying to reinstall a SYSMOD after you have already installed additional service for it.

System Action: This message is informational. See the preceding GIM38201W, GIM38201E, GIM38202W, or GIM38202E message to determine the system action.

Programmer Response: Determine the relationship between the elements in the SYSMODs involved. If the relationship is not correctly defined, do **one** of the following:

- List SYSMOD *sysmod1* to see if it was already installed.
- Reject SYSMOD *sysmod1* and change the PRE and SUP operands that were specified.
- Use UCLIN to modify the RMID subentry for the affected element in the target zone or distribution zone.
- If necessary, install related SYSMODs before processing SYSMOD *sysmod1* to establish the correct relationship.

GIM31904I **SYSMOD** *sysmod1* **DOES NOT SPECIFY** *sysmod2* **ON THE PRE OR SUP OPERAND.** *sysmod2* **IS A UMID FOR** *elmtype elmname* **THAT IS BEING INSTALLED.**

Explanation:

sysmod1	ID of the SYSMOD being installed
sysmod2	ID of the SYSMOD specified as a UMID
elmtype	element type
elmname	element name

This message describes the ID check reported by message GIM38201W, GIM38201E, GIM38202W, or GIM38202E. SMP/E issues GIM31904I for every element in a service SYSMOD (or in a function SYSMOD that is being reinstalled) that does not name, in the PRE or SUP operands, a UMID of the previously processed version of the named element. When this happens, SMP/E cannot determine the relationship between the element in SYSMOD *sysmod1* and the previously processed version of the element. You may also get this message if you are trying to reinstall a SYSMOD after you have already installed additional service for it.

System Action: This message is informational. See the preceding GIM38201W, GIM38201E, GIM38202W, or GIM38202E message to determine the system action.

Programmer Response: Determine the relationship between the elements in the SYSMODs involved. If the relationship is not correctly defined, do **one** of the following:

- List SYSMOD *sysmod1* to see if it was already installed.
- Reject SYSMOD *sysmod1* and change the PRE and SUP operands that were specified. This is only necessary if SYSMOD *sysmod1* replaces the indicated element. In that case, it must specify a relationship with all the UMIDs for the element being replaced. If SYSMOD *sysmod1* is updating the

element, it is not necessary to specify a relationship with any of the UMIDs for the element being updated.

- Use UCLIN to modify the UMID subentries for the affected element in the target zone or distribution zone.
- If necessary, install related SYSMODs before processing SYSMOD *sysmod1* to establish the correct relationship.

GIM32000W NO ENTRIES MATCHING THE SPECIFIED CRITERIA WERE FOUND.

Explanation: There were no matches in the CSI for the query requested. No data is returned.

System Action: None.

Programmer Response: None.

GIM32100S *modtype* **MODULE** *modname* **COULD NOT BE LOADED. SMP/E API PROCESSING IS STOPPED. ABEND CODE:** *code* **REASON CODE:** *rc*

Explanation:

modtype	″MESSAGE″ or ″CONSTANT″
modname	name of the SMP/E module
code	system abend code
rc	system reason code

The SMP/E message or constant load module could not be loaded by the API program. Possible reasons include:

- The language parameter supplied to GIMAPI was specified incorrectly, or specified an unsupported language. Therefore, no corresponding message module could be found.
- The indicated load module was not found in the link list library.

System Action: Command processing stops.

Programmer Response: Do the following:

1. Verify that the language value supplied to GIMAPI correctly specifies a language supported by SMP/E.
2. Check that the load module was actually installed in the target library when SMP/E was installed.
3. See the *z/OS MVS System Codes* manual for an explanation of the system abend and reason codes.

GIM32401W SYSMOD *sysmod1* **REGRESSES** *elmtype elmname***, WHICH WAS PREVIOUSLY REPLACED BY SYSMOD** *sysmod2***.**

Explanation:

sysmod1	ID of the SYSMOD being processed
elmtype	element type
elmname	element name
sysmod2	ID of the SYSMOD that was previously processed

SYSMOD *sysmod1* should have superseded SYSMOD *sysmod2* or specified it as a prerequisite because SYSMOD *sysmod2* has already been processed. **One** of the following is true:

- SYSMOD *sysmod2* has already been applied.
- SYSMOD *sysmod2* has already been accepted.
- SYSMOD *sysmod2* is being installed concurrently with SYSMOD *sysmod1*.

Because the relationship between the SYSMODs is not defined, SMP/E considers SYSMOD *sysmod2* to be regressed by SYSMOD *sysmod1*.

System Action: SYSMOD processing continues.

Programmer Response: None.

GIM32402W SYSMOD *sysmod1* **REGRESSES** *elmtype elmname***, WHICH WAS PREVIOUSLY UPDATED BY SYSMOD** *sysmod2***.**

Explanation:

sysmod1	ID of the SYSMOD being processed
elmtype	element type
elmname	element name
sysmod2	ID of the SYSMOD that was previously processed

SYSMOD *sysmod1* should have superseded SYSMOD *sysmod2* or specified it as a prerequisite because SYSMOD *sysmod2* had already been processed. **One** of the following is true:

- SYSMOD *sysmod2* has already been applied.
- SYSMOD *sysmod2* has already been accepted.
- SYSMOD *sysmod2* is being installed concurrently with SYSMOD *sysmod1*.

Because the relationship between the SYSMODs is not defined, SMP/E considers SYSMOD *sysmod2* to be regressed by SYSMOD *sysmod1*.

System Action: SYSMOD processing continues.

Programmer Response: None.

GIM32501E *command* **PROCESSING HAS FAILED FOR SYSMOD** *sysmod1***. SYSMOD** *sysmod1* **SUPERSEDES SYSMOD** *sysmod2* **BUT DOES NOT CONTAIN** *elmtype elmname***.**

Explanation:

command	ACCEPT or APPLY
sysmod1	ID of the superseding SYSMOD
sysmod2	ID of the superseded SYSMOD
elmtype	element type
elmname	element name

The superseding SYSMOD and its requisite set, taken together, do not contain all of the elements contained in the superseded SYSMOD.

System Action: Processing stops for this SYSMOD.

Programmer Response: Correct the superseding SYSMOD, one or more of the SYSMODs in its requisite set, or both, so that the superseding SYSMOD and its requisite set, taken together, will contain all elements contained in all SYSMODs being superseded.

GIM32501W SYSMOD *sysmod1* SUPERSEDES SYSMOD *sysmod2* BUT DOES NOT CONTAIN ELEMENT *elmname*.

Explanation:

sysmod1	ID of the superseding SYSMOD
sysmod2	ID of the superseded SYSMOD
elmname	element name

The superseding SYSMOD does not have all the elements that the superseded SYSMOD had.

System Action: Processing continues.

Programmer Response: Check the SYSMODs and make any necessary corrections to the element shown in the message.

GIM32600E DDDEF ENTRY *dddef* IS NEEDED FOR PROCESSING FMID *fmid* BUT IS NOT IN ZONE *zonename*. THEREFORE DDNAME *dddef* IS USED AS THE DATA SET NAME ON THE FROMDS OPERAND.

Explanation:

dddef	DDDEF entry name
fmid	FMID
zonename	zone name

During BUILDMCS command processing, MCS for elements in distribution library *dddef* are created. The data set name to be used on the FROMDS operand of the MCS is retrieved from the distribution library's DDDEF entry. However, this DDDEF entry is not defined in the zone. Therefore, the distribution library's ddname is used as the data set name instead.

System Action: Processing continues.

Programmer Response: Do one of the following:

- Add the DDDEF entry to the zone and rerun the BUILDMCS command, or
- Edit the FROMDS operand on all affected element MCS to contain the correct data set name.

GIM32701E INLINE DATA IS NOT ALLOWED AFTER AN MCS IF THE DELETE, FROMDS, LKLIB, RELFILE, OR TXLIB OPERAND IS SPECIFIED.

Explanation: Input text appeared inline when it should not have. Inline text is not allowed after an element MCS or a ++JCLIN MCS in these cases:

- The element MCS specifies one of the following operands:
  - DELETE
  - FROMDS
  - LKLIB
  - RELFILE
  - TXLIB
- The ++JCLIN MCS specifies one of the following operands:
  - FROMDS
  - RELFILE
  - TXLIB

System Action: Processing stops for this SYSMOD but continues for any remaining SYSMODs.

Programmer Response: The SYSMOD is built incorrectly. Check the SYSMOD for any of the following:

- A missing element modification control statement
- Incorrectly coded "++" characters
- A conflict between the placement of the input text and the specification of the DELETE, FROMDS, LKLIB, RELFILE, or TXLIB operands.

Fix the error and rerun the job.

GIM32800W SIDE DECK *oldname* WAS RENAMED TO *newname* IN THE *library* LIBRARY BY SYSMOD *sysmod*. THE CONTENTS OF THIS SIDE DECK MAY NOT BE ACCURATE BECAUSE THE ASSOCIATED LOAD MODULE HAS ALSO BEEN RENAMED.

Explanation:

oldname	old name
newname	new name
library	ddname of the library
sysmod	SYSMOD ID

SMP/E successfully renamed the definition side deck member in the indicated library. However, the contents of the definition side deck will not be correct. A definition side deck contains binder IMPORT control statements, which specify the associated load module's name. Because the IMPORT statements specify the old load module name, the statements will be incorrect now that the load module has been renamed.

Note: The rename operation is the result of a ++RENAME statement to rename a load module.

System Action: Processing continues.

Programmer Response:

- If the renamed load module is link-edited by the binder later in SMP/E processing, the definition side deck will be rebuilt by the binder and the IMPORT control statements will specify the correct load module name. Therefore, no action is required.

- If the renamed load module is not link-edited by the binder later in SMP/E processing, decide whether to update the IMPORT statements.

  You can manually change the load module name specified on the IMPORT statements, or you can link-edit the load module outside of SMP/E by specifying the DYNAM(DLL) option and the SYSDEFSD DD statement to rebuild the definition side deck. Because the SYSDEFSD DD statement identifies the library to contain the definition side deck, you should specify the library identified by the load module's side deck library subentry value.

GIM32900W NO ZONES MATCHING THE SPECIFIED CRITERIA WERE FOUND.

Explanation: No zones in the zoneindex were found eligible for processing by GIMAPI based on the parameter specified for ZONE.

System Action: No data is returned

Programmer Response: None.

GIM33000W SIDE DECK LIBRARY *oldsdlib* **FOR LOAD MODULE** *loadmod* **HAS BEEN REPLACED BY LIBRARY** *newsdlib* **DURING JCLIN PROCESSING. THE SIDE DECK MEMBER IN LIBRARY** *oldsdlib* **WILL NOT BE DELETED.**

Explanation:

oldsdlib	ddname of the old side deck library
loadmod	load module name
newsdlib	ddname of the new side deck library

During JCLIN processing for the specified load module, SMP/E replaced the existing side deck library subentry value with a new value. The existing definition side deck member that resides in the old library will not be deleted by SMP/E and will subsequently not be maintained by SMP/E.

System Action: Processing continues.

Programmer Response: If the original definition side deck member is not needed, delete it outside of SMP/E.

GIM33001W SYSTEM LIBRARY *oldsdlib* **FOR LOAD MODULE** *loadmod* **HAS BEEN REPLACED BY LIBRARY** *newsdlib* **DURING JCLIN PROCESSING. THE LOAD MODULE MEMBER IN LIBRARY** *oldsdlib* **WILL NOT BE DELETED.**

Explanation:

oldsdlib	ddname of the old system library
loadmod	load module name
newsdlib	ddname of the new system library

During JCLIN processing for the specified load module, SMP/E replaced the existing system library subentry value with a new value. The existing load module member that resides in the old library will not be deleted by SMP/E and will subsequently not be maintained by SMP/E.

System Action: Processing continues.

Programmer Response: If the original load module member is not needed, delete it outside of SMP/E.

GIM33100E AN EXTRANEOUS RIGHT PARENTHESIS WAS ENCOUNTERED ON THE FILTER PARAMETER.

Explanation: A right parenthesis was found on the API QUERY *filter* parameter that could not be matched with a preceding left parenthesis.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the *filter* parameter, then rerun the calling program.

GIM33200E *elmtype elmname* **COULD NOT BE RETRANSFORMED BECAUSE THE** *attribute* **OF** *value* **FOR THE DATA IS NOT SUPPORTED BY THIS LEVEL OF SMP/E.**

Explanation:

elmtype	element type
elmname	element name
attribute	attribute type (RECFM)
value	attribute value for the transformed element

When SMP/E was installing the indicated element, it compared the attributes of the element, which are indicated in the records included with the transformed element, with the attributes supported by the current level of SMP/E. The attributes of the element are not supported by the level of SMP/E being used to install the element.

System Action: SYSMOD processing stops.

Programmer Response: Check the program directory for the product that supplies the element to find out the minimum required level of SMP/E needed to install the element.

GIM33201E *elmtype elmname* **COULD NOT BE RETRANSFORMED BECAUSE THE RECFM FOR LIBRARY** *library* **IS NOT SUPPORTED.**

Explanation:

elmtype	element type
elmname	element name
library	library ddname

When SMP/E was installing the indicated element, it determined that the record format of the library into which the element is being installed is not supported. The supported record formats are F, FB, FA, FBA, FM, FBM, FS, FBS, V, VB, VA, VBA, VM, VBM, VS, and VBS.

System Action: SYSMOD processing stops.

Programmer Response: Check the record format of the indicated library to be sure it is one supported by SMP/E. Correct the record format and rerun the job.

GIM33300E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. THE SYSMOD CONTAINS COMPACTED DATA BUT THE CURRENT LEVEL OF THE OPERATING SYSTEM DOES NOT SUPPORT COMPRESSION OR EXPANSION SERVICES. MVS/ESA SP 4.3 OR HIGHER IS REQUIRED.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E could not process SYSMOD *sysmod* because the MCS entry in the SMPPTS data set contains compacted data, but the operating system being used does not support compression or expansion services.

System Action: SYSMOD processing stops.

Programmer Response: Use a driving system that supports the required services and rerun the job.

GIM33301E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***.** *elmtype elmname* **WITHIN SYSMOD** *sysmod* **COULD NOT BE EXPANDED BECAUSE THE COMPACTED DATA IS CORRUPTED OR INCOMPLETE.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID
elmtype	element type
elmname	element name

SMP/E could not process SYSMOD *sysmod1* because the indicated element within the SYSMOD has been compacted, but data has been corrupted or lost.

Compacted data can be corrupted if the SYSMOD is moved or copied with a process that can convert or transform the data being moved or copied. For example, if the SYSMOD was moved by an upload, download, or FTP process, and a conversion was performed because of an incorrect profile or incorrectly specified option, then the compacted data could be corrupted.

System Action: SYSMOD processing stops.

Programmer Response: Report the error to your support group.

If the SYSMOD was moved or copied using an upload, download, FTP, or similar process, ensure that no conversion of data occurred during the operation. Compacted data should be treated as binary data and moved accordingly.

GIM33302E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***.** *elmtype elmname* **WITHIN SYSMOD** *sysmod* **COULD NOT BE EXPANDED BECAUSE THE** *dictionary* **DICTIONARY USED TO COMPACT THE DATA IS NOT SUPPORTED.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID
elmtype	element type
elmname	element name
dictionary	compaction dictionary identifier or blank

SMP/E could not process SYSMOD *sysmod* because the indicated element within the SYSMOD has been compacted, but the current level of SMP/E does not recognize the dictionary used during the compaction operation. A service level of SMP/E that supports and recognizes the compaction dictionary is required to expand the compacted element data within the SYSMOD.

System Action: SYSMOD processing stops.

Programmer Response: Using a level of SMP/E that supports the identified compaction dictionary, rerun the job.

GIM33400S GIMCPTS PROCESSING FAILED BECAUSE COMPRESSION AND EXPANSION SERVICES ARE NOT SUPPORTED BY THE CURRENT LEVEL OF THE OPERATING SYSTEM. MVS/ESA SP 4.3 OR HIGHER IS REQUIRED.

Explanation: The GIMCPTS service routine could not operate because the current level of the operating system does not support compression or expansion services. MVS/ESA SP 4.3 or higher is required to compact or expand SMP/E data.

System Action: GIMCPTS processing stops.

Programmer Response: Using a driving system of MVS/ESA SP 4.3 or higher, rerun the job.

GIM33500S GIMCPTS PROCESSING FAILED BECAUSE THE DATA SET ORGANIZATION OF SYSUT2 DOES NOT MATCH THE *dsorg* ORGANIZATION OF SYSUT1.

Explanation:

dsorg	data set organization

The GIMCPTS service routine compacts or expands inline element data within SYSMODs found in the SYSUT1 data set and writes those SYSMODs to the SYSUT2 data set. GIMCPTS processing has failed because the data set organization of the SYSUT2 data set does not match the data set organization of the SYSUT1 data set. If SYSUT1 is sequential, then SYSUT2 must also be sequential, and likewise, if SYSUT1 is partitioned, then SYSUT2 must also be partitioned.

System Action: GIMCPTS processing stops.

Programmer Response: Use a data set for SYSUT2 that has the same data set organization as SYSUT1, and then rerun the job.

GIM33501S GIMCPTS PROCESSING FAILED BECAUSE THE SYSUT2 DATA SET MUST BE DIFFERENT THAN THE SYSUT1 DATA SET.

Explanation: The GIMCPTS service routine compacts or expands inline element data within SYSMODs found in the SYSUT1 data set and writes those SYSMODs to the SYSUT2 data set. GIMCPTS processing has failed because the SYSUT2 data set is the same as the SYSUT1 data set. When the SYSUT1 and SYSUT2 data sets are both sequential, the SYSUT2 data set must be different than the SYSUT1 data set.

System Action: GIMCPTS processing stops.

Programmer Response: Use a data set for SYSUT2 that is different than the SYSUT1 data set.

GIM33502S GIMCPTS PROCESSING FAILED. THE *dataset* DATA SET COULD NOT BE USED BECAUSE THE *attribute* IS NOT *value*.

Explanation:

dataset	ddname of the unusable data set (SYSUT1 or SYSUT2)
attribute	data set attribute; RECFM (record format) or LRECL (logical record length)
value	data set attribute value

The GIMCPTS service routine compacts or expands inline element data within SYSMODs found in the SYSUT1 data set and writes those SYSMODs to the SYSUT2 data set. GIMCPTS processing has failed because the SYSUT1 or SYSUT2 data set does not have the correct attributes. These data sets must have a record format (RECFM) of fixed-block (FB), and a logical record length (LRECL) of 80.

System Action: GIMCPTS processing stops.

Programmer Response: Ensure the SYSUT1 and SYSUT2 data sets have a RECFM of FB and an LRECL of 80, and rerun the job.

GIM33503S *progname* PROCESSING FAILED. THE *dataset* DATA SET COULD NOT BE USED BECAUSE THE DATA SET ORGANIZATION IS NOT SEQUENTIAL OR PARTITIONED.

Explanation:

progname	name of SMP/E routine that failed
dataset	ddname associated with the data set name

Processing has failed for the SMP/E routine identified because the indicated data set does not have an acceptable data set organization. The identified data set must be sequential or partitioned.

System Action: Processing stops for the indicated SMP/E routine.

Programmer Response: Ensure that the identified data set has an organization of sequential (PS) or partitioned (PO).

GIM33600S *progname* PROCESSING FAILED BECAUSE THE *dataset* DATA SET IS MISSING.

Explanation:

progname	program name
dataset	ddname of the missing data set

The indicated data set is required for processing. However, the JCL contained no DD statement for the data set.

System Action: Processing stops.

Programmer Response: Add a DD statement for the required data set and rerun the job.

GIM33700I *zonetype* ZONE *zonename* WILL BE USED FOR RECEIVE APPLYCHECK OR ACCEPTCHECK COMMAND PROCESSING.

Explanation:

zonetype	DISTRIBUTION or TARGET
zonename	zone name

The specified zone will be used for APPLYCHECK and ACCEPTCHECK processing in the SYSMOD selection phase of RECEIVE command processing.

System Action: Processing continues.

Programmer Response: None.

GIM33801E THE ++*mcstype* MCS IS NOT ALLOWED IN FUNCTION SYSMODS.

Explanation:
mcstype an element update MCS

A function SYSMOD cannot update elements. It can only replace them.

System Action: SYSMOD processing stops.

Programmer Response: Repackage the function SYSMOD so that it replaces elements instead of updating them.

GIM33901E THE *operand* OPERAND IS ONLY ALLOWED IN FUNCTION SYSMODS.

Explanation:
operand RMID or UMID

An element replacement MCS can only specify the RMID or UMID operand if that MCS is in a function SYSMOD.

System Action: SYSMOD processing stops.

Programmer Response: Repackage the SYSMOD so that it defines the SYSMOD relationships using the PRE or SUP operands on the ++VER MCS instead of the RMID or UMID operands on the element MCS.

GIM34001E THE RELFILE OPERAND IS NOT ALLOWED BECAUSE THE FILES OPERAND WAS NOT SPECIFIED ON THE HEADER MCS.

Explanation: The RELFILE operand is only allowed on an element or ++JCLIN MCS if the FILES operand is specified on the header MCS. The header MCS indicates the SYSMOD type:

- ++APAR
- ++FUNCTION
- ++PTF
- ++USERMOD

System Action: SYSMOD processing stops.

Programmer Response: Specify the FILES operand on the header MCS and receive this SYSMOD again.

GIM34100W THE GLOBAL ZONE WAS SPECIFIED FOR RECEIVE APPLYCHECK OR ACCEPTCHECK PROCESSING. IT IS NOT VALID AND IS IGNORED.

Explanation: One of the zones specified was the global zone. This zone is not valid for APPLYCHECK and ACCEPTCHECK processing during the SYSMOD selection phase of RECEIVE command processing.

System Action: Processing continues.

Programmer Response: None.

GIM34201E MORE THAN ONE ++JCLIN STATEMENT WAS SPECIFIED IN A SYSMOD. ONLY ONE IS ALLOWED.

Explanation: A SYSMOD being processed is built incorrectly because it contains more than one ++JCLIN MCS.

System Action: SYSMOD processing stops.

Programmer Response: Remove the extra ++JCLIN MCS and rerun the job.

GIM34202E MORE THAN ONE *operand* OPERAND WAS SPECIFIED ON AN MCS. ONLY ONE IS ALLOWED.

Explanation:
operand MCS operand

An MCS being processed is coded incorrectly because the indicated operand is specified more than once.

System Action: SYSMOD processing stops.

Programmer Response: Remove the extra operand and rerun the job.

GIM34203W THE SYMPATH OPERAND SPECIFIES *number* MORE VALUES THAN THE SYMLINK OPERAND ON THE ++*mcstype* MCS FOR *elmname* IN SYSMOD *sysmod*.

Explanation:
number a decimal number
mcstype MCS type
elmname element name
sysmod SYSMOD ID

The indicated MCS for the indicated element specified the **SYMLINK** and **SYMPATH** operands. The **SYMPATH** operand had more values than the **SYMLINK** operand. When this element is installed during APPLY processing, the additional **SYMPATH** values are going to be ignored by the hierarchical file system copy utility.

One reason this can occur is that duplicate SYMLINK values are specified in the SYMLINK operand list of the indicated element MCS. These duplicate values are ignored.

System Action: SYSMOD processing continues.

Programmer Response: If the **SYMPATH** should not have more values than the **SYMLINK** operand, correct the error in the MCS and reject and re-receive the SYSMOD.

GIM34301T SMPPTFIN MUST BE A STANDARD LABEL TAPE BECAUSE THE SYSMOD BEING RECEIVED CONTAINS RELFILES.

Explanation: SMP/E could not receive a SYSMOD built with relative files because the SMPPTFIN data set that contains the SYSMOD is not a standard labeled tape.

System Action: SMP/E processing stops.

Programmer Response: You may have mounted the wrong tape. Rerun the job using the correct standard labeled tape.

GIM34400S THE *operand* OPERAND WAS FOUND ON THE *command* COMMAND. THIS RELEASE OF SMP/E DOES NOT SUPPORT THE *operand* OPERAND. OS/390 RELEASE 5 OR HIGHER IS NEEDED.

Explanation:

command	RECEIVE
operand	ZONEGROUP

SMP/E found the indicated operand on the indicated command. The operand is not supported by the release of SMP/E that is executing.

System Action: Command processing stops.

Programmer Response: Use a level of SMP/E that supports the operand.

GIM34500W THE ACTIVE OPTIONS ENTRY *entname* CONTAINS A *subname* SUBENTRY. THIS RELEASE OF SMP/E DOES NOT SUPPORT APPLYCHECK OR ACCEPTCHECK PROCESSING DURING THE RECEIVE COMMAND. OS/390 RELEASE 5 OR HIGHER IS NEEDED.

Explanation:

entname	OPTIONS entry name
subname	RECZGRP or RECEXZGRP

SMP/E found the user had established an OPTIONS entry with subentry information for APPLYCHECK or ACCEPTCHECK processing. APPLYCHECK and ACCEPTCHECK processing are not available in the release of SMP/E that is executing.

System Action: Command processing continues.

Programmer Response: If you want to use APPLYCHECK or ACCEPTCHECK processing, use the appropriate level of SMP/E.

GIM34601E THE ONLY UPDATE CONTROL STATEMENTS ALLOWED ARE ./ CHANGE AND ./ ENDUP.

Explanation: During RECEIVE processing, SMP/E found an update control statement other than ./ CHANGE or ./ ENDUP following an element update MCS.

System Action: SYSMOD processing stops.

Programmer Response: Fix the SYSMOD and run RECEIVE again.

GIM34701E SMP/E EXPECTED TO FIND AN MCS. IT FOUND INLINE DATA INSTEAD.

Explanation: During RECEIVE processing, SMP/E expected a MCS, but instead found input in SMPPTFIN that was not an SMP/E control statement. This can happen when input text follows an MCS that has a syntax error.

System Action: SYSMOD processing stops. SMP/E ignores all subsequent data that is not an SMP/E control statement.

Programmer Response: Fix the problem and run RECEIVE again.

GIM34801E RECEIVE PROCESSING FAILED FOR THE SYSMOD BECAUSE IT CONTAINS MORE THAN ONE ++VER MCS FOR THE SAME SREL-FMID PAIR. ONLY ONE IS ALLOWED.

Explanation: During RECEIVE processing, SMP/E found a SYSMOD with more than one ++VER MCS naming the same SREL and FMID. (The ++VER MCS that caused the error immediately precedes this message.) A SYSMOD like this creates an ambiguous situation when you want to run the APPLY or ACCEPT commands.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and run RECEIVE again.

GIM34901E THE ++IF MCS MUST IMMEDIATELY FOLLOW A ++VER MCS.

Explanation: The ++IF MCS statements must follow a ++VER MCS. This allows SMP/E to associate the ++IF MCS statements with the ++VER MCS statements chosen at APPLY or ACCEPT time.

In this case, SMP/E found an ++IF MCS that did not follow a ++VER MCS. (The ++IF MCS that caused the error immediately precedes this message).

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and run RECEIVE again.

GIM35001S THE RELFILE VALUE SPECIFIED ON AN MCS STATEMENT IS GREATER THAN THE NUMBER OF FILES IN THE SYSMOD. THIS IS NOT ALLOWED.

Explanation: The RELFILE value specified on a ++JCLIN or element MCS is greater than the number of files specified on the FILES operand of the header MCS. The header MCS identifies the SYSMOD type: ++APAR, ++FUNCTION, ++PTF, and ++USERMOD. (The JCLIN or element MCS that caused the error immediately precedes this message.)

System Action: RECEIVE processing stops.

Programmer Response: Fix the problem and run RECEIVE again.

GIM35101E SYSMOD *sysmod* FAILED BECAUSE OF AN ERROR THAT OCCURRED WHILE LOADING SMPTLIB DATA SETS. THE RETURN CODE FROM THE COPY UTILITY WAS *rc*.

Explanation:

sysmod	SYSMOD ID
rc	return code from copy processing

An error occurred while the copy utility was loading the SMPTLIBs for the indicated SYSMOD.

System Action: SYSMOD processing stops.

Programmer Response: Look at the SYSPRINT output for the copy utility to determine the problem.

Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:

- The PRINT subentry specifies a DDDEF of DUMMY.
- The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

GIM35201I SMPTLIB *smptlib* WAS ALLOCATED AND CATALOGED ON VOLUME *volser*.

Explanation:

smptlib	SMPTLIB data set name
volser	volume serial number

During RECEIVE processing, SMP/E allocated and cataloged the SMPTLIB data set on the indicated volume.

System Action: SMP/E uses the allocated data set to load the relative files.

Programmer Response: None.

GIM35202I SMP/E USED SMPTLIB *smptlib*, WHICH ALREADY EXISTS ON VOLUME *volser*.

Explanation:

smptlib	SMPTLIB data set name
volser	volume serial number

During RECEIVE processing, SMP/E found a preallocated and cataloged SMPTLIB data set on the indicated volume.

System Action: SMP/E uses the preallocated data set to load the relative files.

Programmer Response: None.

GIM35301E SMPTLIB *smptlib* WAS NOT ALLOCATED ON VOLUME *volser* BECAUSE IT IS NOT A PARTITIONED DATA SET.

Explanation:

smptlib	data set name
volser	volume serial number

During RECEIVE processing, SMP/E found an existing SMPTLIB data set on the indicated volume. However, the data set was not a partitioned data set, so SMP/E could not use it to load relative files.

System Action: SYSMOD processing stops.

Programmer Response: Define a new SMPTLIB data set that is a partitioned data set, or define a DDDEF entry to have SMP/E dynamically allocate the SMPTLIB data set.

GIM35302E SMPTLIB *smptlib* WAS NOT ALLOCATED ON VOLUME *volser* BECAUSE ERROR CODE *code* WAS ISSUED BY DYNAMIC ALLOCATION.

Explanation:

smptlib	data set name
volser	volume serial number
code	error code issued by dynamic allocation

During RECEIVE processing, SMP/E tried to allocate the SMPTLIB data set on the indicated volume to load relative files for a SYSMOD. However, an error occurred during dynamic allocation.

Error codes from dynamic allocation fall into the following ranges:

Codes	Meaning
0000–00FC	Information reason codes
0021–0029	CATALOG or UNCATALOG unsuccessful
0031–0038	Delete unsuccessful
0100–01FC	I/O error
0200–02FC	System resources not available

0300–03FC	Invalid parameter list
0400–04FC	Environment error
1704–17FF	LOCATE error
4704–4780	DADSM error
47A8–47AC	RACF error
47B0	DADSM error
5704–571C	CATALOG error
6704–6714	OBTAIN error
7700	Subsystem error
7704–770C	SCRATCH error
8700–9728	System error

System Action: SYSMOD processing stops.

Programmer Response: See *z/OS MVS Programming: Authorized Assembler Services Guide*, SA22-7608, for more information about the error code. Message GIM54701E accompanies this message and contains information about the allocation failure.

GIM35305S SMPTLIB *smptlib* WAS NOT ALLOCATED BECAUSE IT IS DEFINED AS A DUMMY DATA SET.

Explanation:

smptlib	data set name

During RECEIVE processing, SMP/E tried to allocate the SMPTLIB data set on the indicated volume to load relative files for a SYSMOD. However, the SMPTLIB DD statement specified either the DUMMY operand or **DSN=NULLFILE**.

System Action: Command processing stops.

Programmer Response: Change the SMPTLIB DD statement so that it does not define a dummy data set.

GIM35306E SMPTLIB *smptlib* WAS NOT ALLOCATED BECAUSE ERROR CODE *code* WAS ISSUED BY DYNAMIC ALLOCATION.

Explanation:

smptlib	data set name
code	error code issued by dynamic allocation

An error occurred during dynamic allocation, when SMP/E tried to allocate the SMPTLIB data set to load relative files for a SYSMOD. If the error code is 0220, the relfile tape data set has the same name as the RFDSNPFX data set. This is because the RFDSNPFX on the header MCS statement has the same value as the DSPREFIX for the SMPTLIB data set.

Error codes from dynamic allocation fall into the following ranges:

Codes	**Meaning**
0000–00FC	Information reason codes
0021–0029	CATALOG or UNCATALOG unsuccessful
0031–0038	Delete unsuccessful
0100–01FC	I/O error
0200–02FC	System resources not available
0300–03FC	Invalid parameter list
0400–04FC	Environment error
1704–17FF	LOCATE error
4704–4780	DADSM error
47A8–47AC	RACF error
47B0	DADSM error
5704–571C	CATALOG error
6704–6714	OBTAIN error
7700	Subsystem error
7704–770C	SCRATCH error
8700–9728	System error

System Action: SYSMOD processing stops.

Programmer Response: If the error code is 0220, change the DSPREFIX value and rerun the job. For more information about other error codes, see *z/OS MVS Programming: Authorized Assembler Services Guide*, SA22-7608. Message GIM54701E accompanies this message and contains information about the allocation failure.

GIM35307E SMPTLIB *smptlib* FOR *sysmod* WAS NOT ALLOCATED TO *ddname* BECAUSE THE DATA SET WAS NOT FOUND.

Explanation:

smptlib	data set name
sysmod	SYSMOD ID
ddname	ddname

Allocation failed for SMPTLIB data set *smptlib* because the data set was not found.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Rename the SMPTLIB data set name to the same name it was at RECEIVE processing time.
- Check that the SMPTLIB DDDEF specifies the volume where the SMPTLIB data set was allocated at RECEIVE time. If the SMPTLIB data set has been deleted, receive the SYSMOD again.

GIM35401I SMPTLIB *smptlib* WAS DELETED AND UNCATALOGED FROM VOLUME *volser*.

Explanation:

smptlib	data set name
volser	volume serial number

SMP/E deleted and uncataloged the SMPTLIB data set from the indicated volume.

System Action: SYSMOD processing continues.

Programmer Response: None.

GIM35501E AN ERROR OCCURRED DURING PROCESSING OF THE *enttype* ENTRY FOR SYSMOD *sysmod* IN THE *library* LIBRARY.

Explanation:

enttype	entry type
sysmod	SYSMOD ID
library	ddname of the library

An error occurred while SMP/E was processing the indicated SYSMOD.

- If the entry type is MCS, SMP/E found an error when it was checking the MCS entry on the SMPPTS data set. The error could be one of the following:
  - An I/O error
  - A mismatch between the MCS and SYSMOD entries on the SMPPTS.
- If the entry type is not MCS, look at the preceding SMP/E output to find the cause of the error for the indicated entry type.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:
- If the entry type is MCS, reject the SYSMOD and receive it again.
- If any other case, follow the recommended action in the preceding error messages.

GIM35601E *subtype* SUBENTRY *subname* WAS NOT ADDED BECAUSE IT ALREADY EXISTS.

Explanation:

subtype	subentry type
subname	subentry name or *prodid vv.rr.mm* for PRODSUP subentries (optional)

A UCL statement tried to add the indicated subentry to an entry, but the subentry was already there. Either the entry or the subentry is incorrect. Note that if the subentry name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM35602E *subtype* SUBENTRY *subname1* WAS NOT ADDED BECAUSE *subname2* ALREADY EXISTS.

Explanation:

subtype	subentry type
subname1	subentry name
subname2	conflicting subentry name

A UCL statement tried to add the indicated subentry to an entry, but a mutually exclusive subentry was already there.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM35603E THE INITIAL DISPOSITION SUBENTRY WAS NOT ADDED BECAUSE IT ALREADY EXISTS.

Explanation: A UCL statement tried to add an initial disposition subentry (MOD, NEW, SHR, or OLD) to an entry, but the subentry was already there. Either the entry or the subentry is incorrect.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM35604E THE FINAL DISPOSITION SUBENTRY WAS NOT ADDED BECAUSE IT ALREADY EXISTS.

Explanation: A UCL statement tried to add a final disposition subentry (KEEP, DELETE, or CATALOG) to an entry, but the subentry was already there. Either the entry or the subentry is incorrect.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM35701E *subtype* SUBENTRY *subname* WAS NOT DELETED BECAUSE IT DOES NOT EXIST.

Explanation:

subtype	subentry type
subname	subentry name or *prodid vv.rr.mm* for PRODSUP subentries (optional)

A UCL statement tried to delete the indicated subentry from an entry, but the subentry was not there. Note that if the subentry name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Rerun the UCL statement without the specified subentry.

GIM35702E *subtype* **SUBENTRY** *subname* **WAS NOT ADDED BECAUSE IT IS THE SAME AS THE** *enttype* **ENTRY THAT WOULD CONTAIN IT.**

Explanation:

subtype	subentry type (PRODSUP)
subname	subentry name (*prodid vv.rr.mm* for PRODSUP subentries)
enttype	entry type (PRODUCT)

The specified subentry may not contain a value that is the same as the entry name that will contain it. That is, a PRODUCT entry may not supersede itself.

System Action: SMP/E does not process any of the changes in this UCL statement.

Programmer Response: Rerun the UCL statement without the specified subentry.

GIM35703E THE INITIAL DISPOSITION SUBENTRY *value* **WAS NOT DELETED BECAUSE IT DOES NOT EXIST.**

Explanation:

value	subentry value

A UCL statement tried to delete the indicated initial disposition value from an entry, but it did not exist. Either the subentry did not exist or the indicated subentry value is not the existing one.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM35704E THE FINAL DISPOSITION SUBENTRY *value* **WAS NOT DELETED BECAUSE IT DOES NOT EXIST.**

Explanation:

value	subentry value

A UCL statement tried to delete the indicated final disposition value from an entry, but it did not exist. Either the subentry did not exist or the indicated subentry value is not the existing one.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM35705E DELETING THE SPECIFIED SUBENTRIES FROM THE *listname* **SUBENTRY LIST WOULD STILL LEAVE SUBENTRIES IN THE LIST. THIS IS NOT ALLOWED.**

Explanation:

listname	SYMLINK or SYMPATH

You have requested that specific subentries be removed from the indicated list. However, if this update is made, there will still be subentries in the list. Because this list is ordered and its subentries have an order-dependent correspondence with another list in the hierarchical file system element entry, you are not allowed to manipulate individual subentries.

System Action: SMP/E does not process any of the changes in this UCL statement.

Programmer Response: You can replace the entire list or possibly delete the entire list.

GIM35801W *subtype* **SUBENTRY** *subname* **DOES NOT EXIST. THE REPLACE OPERATION HAS BEEN CHANGED TO AN ADD.**

Explanation:

subtype	subentry type
subname	subentry name (optional)

A UCL statement tried to replace the indicated subentry, but it does not exist. SMP/E assumed that you meant to add the subentry.

System Action: SMP/E processes the UCL statement as if ADD were specified.

Programmer Response: Do **one** of the following:

- If you meant to add the subentry, no response is required.
- If SMP/E should not have added the subentry, do the following:
  1. Use the UCL DEL statement to delete the subentry.
  2. Specify the correct entry and subentry.
  3. Rerun the UCL REP statement.

GIM35901I ERROR HOLD *reason* **WAS NOT RESOLVED.**

Explanation:

reason	error hold reason ID

SMP/E could not process the SYSMOD named in GIM30206E because it was held for the indicated error reason ID (APAR), which was not resolved.

System Action: See message GIM30206E.

Programmer Response: Do **one** of the following:

- Install a SYSMOD that matches or supersedes the indicated APAR. This will automatically resolve the error reason ID. You can install the SYSMOD named in GIM30206E along with or after the resolving SYSMOD.
- Rerun the ACCEPT or APPLY command to install the SYSMOD named in GIM30206E, but specify **BYPASS(HOLDERR(***reason***))**.

GIM35902I SYSTEM HOLD *reason* **WAS NOT RESOLVED.**

Explanation:
reason system hold reason ID

One of the following occurred:

- GIM30206E was also issued

  SMP/E could not process the SYSMOD named in GIM30206E because it was held for the indicated system reason ID, which was not resolved.
- A GIM302*xx* message other than GIM30206E was also issued.

  The previous GIM302*xx* (for example, GIM30209E) indicates why APPLY or ACCEPT processing failed for the indicated SYSMOD. Message GIM35902 is issued along with other GIM359*xx* messages, even if you specified **BYPASS(HOLDSYSTEM)**.

System Action: See the description of the associated GIM302*xx* message that was issued.

Programmer Response:

- GIM30206E was also issued.

  Do any processing needed to resolve the system reason ID. Then rerun the ACCEPT or APPLY command to install the SYSMOD named in GIM30206E, but specify **BYPASS(HOLDSYS(***reason***))**.
- A GIM302*xx* message other than GIM30206E was also issued.

  Do any processing needed to resolve the other GIM359*xx* messages that were issued. Then rerun the ACCEPT or APPLY command to install the SYSMOD named in the GIM302*xx* message, keeping **BYPASS(HOLDSYS(***reason***))**.

GIM35903I USER HOLD *reason* **WAS NOT RESOLVED.**

Explanation:
reason user hold reason ID

SMP/E could not process the SYSMOD named in GIM30206E because it was held for the indicated user reason ID, which was not resolved.

System Action: See message GIM30206E.

Programmer Response: Do any processing needed to resolve the user reason ID. Then rerun the ACCEPT or APPLY command to install the SYSMOD named in GIM30206E, but specify **BYPASS(HOLDUSER(***reason***))**.

GIM35904I PREREQUISITE SYSMOD *sysmod* **WAS MISSING.**

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a prerequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* either is not available (it is not in the global zone), or is available but was not included on the SELECT operand that was specified. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was missing.)

Note: For RESTORE processing, this message is only issued for a prerequisite if the SYSMOD named in GIM302*xx* failed. It is not issued for all the prerequisites in the chain.

If this is ACCEPT processing, the SYSMOD may not be available because it has not been applied in the target zone.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- If SYSMOD *sysmod* is no longer in the global zone, receive it again so that it is available to be installed. Then rerun the job.
- If SYSMOD *sysmod* is available but was not specified on the SELECT operand, include it on the SELECT operand and rerun the job.
- If this is ACCEPT processing and this SYSMOD is not available because it has not been applied in the target zone, APPLY the SYSMOD and then re-run the ACCEPT command.

GIM35905I PREREQUISITE SYSMOD *sysmod* **WAS EXCLUDED.**

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a prerequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* was specified on the EXCLUDE operand. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was excluded.)

Note: For RESTORE processing, this message is only issued for a prerequisite if the SYSMOD named in GIM302*xx* failed. It is not issued for all the prerequisites in the chain.

System Action: See message GIM302*xx*.

Programmer Response: Remove SYSMOD *sysmod* from the EXCLUDE operand and rerun the job.

GIM35906I PREREQUISITE SYSMOD *sysmod* WAS HELD.

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a prerequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because there is a ++HOLD MCS for SYSMOD *sysmod* that was not resolved. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was held.)

Note: For RESTORE processing, this message is only issued for a prerequisite if the SYSMOD named in GIM302*xx* failed. It is not issued for all the prerequisites in the chain.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- For an ERROR hold, resolve it by installing a resolving SYSMOD: either a SYSMOD whose ID matches the ERROR hold reason ID, or a SYSMOD that supersedes that reason ID. Do this either before rerunning the job or as part of rerunning the job.
- For a SYSTEM or USER hold, specify the appropriate value on the BYPASS operand and rerun the job.

GIM35907I PREREQUISITE SYSMOD *sysmod* FAILED.

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a prerequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* failed. The reason is indicated by previous messages. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* failed.)

Note: For RESTORE processing, this message is only issued for a prerequisite if the SYSMOD named in GIM302*xx* failed. It is not issued for all the prerequisites in the chain.

System Action: See message GIM302*xx*.

Programmer Response: Look at the preceding messages in SMPOUT to determine why the SYSMOD failed. Fix the error and rerun the job.

GIM35908I COREQUISITE SYSMOD *sysmod* WAS MISSING.

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a corequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* either is not available (it is not in the global zone), or is available but was not included on the SELECT operand that was specified. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was missing.)

If this is ACCEPT processing, the SYSMOD may not be available because it has not been applied in the target zone.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- If SYSMOD *sysmod* is no longer in the global zone, receive it again so that it is available to be installed. Then rerun the job.
- If SYSMOD *sysmod* is available but was not specified on the SELECT operand, include it on the SELECT operand and rerun the job.
- If this is ACCEPT processing and this SYSMOD is not available because it has not been applied in the target zone, APPLY the SYSMOD and then re-run the ACCEPT command.

GIM35909I COREQUISITE SYSMOD *sysmod* WAS EXCLUDED.

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a corequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* was specified on the EXCLUDE operand. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was excluded.)

System Action: See message GIM302*xx*.

Programmer Response: Remove SYSMOD *sysmod* from the EXCLUDE operand and rerun the job.

GIM35910I COREQUISITE SYSMOD *sysmod* WAS HELD.

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a corequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because

there is a ++HOLD MCS for SYSMOD *sysmod* that was not resolved. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was held.)

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- For an ERROR hold, resolve it by installing a resolving SYSMOD: either a SYSMOD whose ID matches the ERROR hold reason ID, or a SYSMOD that supersedes that reason ID. Do this either before rerunning the job or as part of rerunning the job.
- For a SYSTEM or USER hold, specify the appropriate value on the BYPASS operand and rerun the job.

GIM35911I COREQUISITE SYSMOD *sysmod* **FAILED.**

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a corequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* failed. The reason is indicated by previous messages. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* failed.)

System Action: See message GIM302*xx*.

Programmer Response: Look at the preceding messages in SMPOUT to determine why the SYSMOD failed. Fix the error and rerun the job.

GIM35912I CONDITIONAL REQUISITE SYSMOD *sysmod* **WAS MISSING.**

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* either is not available (it is not in the global zone), or is available but was not included on the SELECT operand that was specified. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was missing.) The SYSMOD named in GIM302*xx* contains the ++IF MCS that defines the requisite.

If this is ACCEPT processing, the SYSMOD may not be available because it has not been applied in the target zone.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- If SYSMOD *sysmod* is no longer in the global zone, receive it again so that it is available to be installed. Then rerun the job.
- If SYSMOD *sysmod* is available but was not specified on the SELECT operand, include it on the SELECT operand and rerun the job.
- If this is ACCEPT processing and this SYSMOD is not available because it has not been applied in the target zone, APPLY the SYSMOD and then re-run the ACCEPT command.

GIM35913I CONDITIONAL REQUISITE SYSMOD *sysmod* **WAS EXCLUDED.**

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* was specified on the EXCLUDE operand. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was excluded.) The SYSMOD named in GIM302*xx* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Remove SYSMOD *sysmod* from the EXCLUDE operand and rerun the job.

GIM35914I CONDITIONAL REQUISITE SYSMOD *sysmod* **WAS HELD.**

Explanation:
sysmod SYSMOD ID

SYSMOD *sysmod* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because there is a ++HOLD MCS for SYSMOD *sysmod* that was not resolved. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* was held.) The SYSMOD named in GIM302*xx* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- For an ERROR hold, resolve it by installing a resolving SYSMOD: either a SYSMOD whose ID matches the ERROR hold reason ID, or a SYSMOD that supersedes that reason ID. Do this either before rerunning the job or as part of rerunning the job.
- For a SYSTEM or USER hold, specify the appropriate value on the BYPASS operand and rerun the job.

GIM35915I CONDITIONAL REQUISITE SYSMOD *sysmod* FAILED.

Explanation:

sysmod SYSMOD ID

SYSMOD *sysmod* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* failed. The reason is indicated by previous messages. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod* failed.) The SYSMOD named in GIM302*xx* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Look at the preceding messages in SMPOUT to determine why the SYSMOD failed. Fix the error and rerun the job.

GIM35916I CONDITIONAL REQUISITE SYSMOD *sysmod1* WAS MISSING. SYSMOD *sysmod2* CONTAINED THE ++IF STATEMENT.

Explanation:

sysmod1 SYSMOD ID of the requisite
sysmod2 ID of the SYSMOD that contained the ++IF MCS

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod1* either is not available (it is not in the global zone), or is available but was not included on the SELECT operand that was specified. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod1* was missing.) SYSMOD *sysmod2* contains the ++IF MCS that defines the requisite.

If this is ACCEPT processing, the SYSMOD may not be available because it has not been applied in the target zone.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- If SYSMOD *sysmod1* is no longer in the global zone, receive it again so that it is available to be installed. Then rerun the job.
- If SYSMOD *sysmod1* is available but was not specified on the SELECT operand, include it on the SELECT operand and rerun the job.
- If this is ACCEPT processing and this SYSMOD is not available because it has not been applied in the target zone, APPLY the SYSMOD and then re-run the ACCEPT command.

GIM35917I CONDITIONAL REQUISITE SYSMOD *sysmod1* WAS EXCLUDED. SYSMOD *sysmod2* CONTAINED THE ++IF STATEMENT.

Explanation:

sysmod1 SYSMOD ID of the requisite
sysmod2 ID of the SYSMOD that contained the ++IF MCS

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod1* was specified on the EXCLUDE operand. SYSMOD *sysmod2* contains the ++IF MCS that defines the requisite. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod1* was excluded.)

System Action: See message GIM302*xx*.

Programmer Response: Remove SYSMOD *sysmod1* from the EXCLUDE operand and rerun the job.

GIM35918I CONDITIONAL REQUISITE SYSMOD *sysmod1* WAS HELD. SYSMOD *sysmod2* CONTAINED THE ++IF STATEMENT.

Explanation:

sysmod1 SYSMOD ID of the requisite
sysmod2 ID of the SYSMOD that contained the ++IF MCS

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because there is a ++HOLD MCS for SYSMOD *sysmod1* that was not resolved. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod1* was held.) SYSMOD *sysmod2* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- For an ERROR hold, resolve it by installing a resolving SYSMOD: either a SYSMOD whose ID matches the ERROR hold reason ID, or a SYSMOD that supersedes that reason ID. Do this either before rerunning the job or as part of rerunning the job.
- For a SYSTEM or USER hold, specify the appropriate value on the BYPASS operand and rerun the job.

GIM35919I CONDITIONAL REQUISITE SYSMOD *sysmod1* FAILED. SYSMOD *sysmod2* CONTAINED THE ++IF STATEMENT.

Explanation:

sysmod1 SYSMOD ID of the requisite

sysmod2 ID of the SYSMOD that contained the ++IF MCS

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod1* failed. The reason is indicated by previous messages. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *sysmod1* failed.) SYSMOD *sysmod2* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Look at the preceding messages in SMPOUT to determine why the SYSMOD failed. Fix the error and rerun the job.

GIM35920I SYSMOD *sysmod*, WHICH SUPERSEDES THE INDICATED SYSMOD, FAILED.

Explanation:
sysmod SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* supersedes the SYSMOD in GIM30209E. Therefore, SYSMOD *sysmod* must be restored along with the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.

GIM35921I CROSS-ZONE REQUISITE SYSMOD *sysmod* REQUIRED IN ZONE *zonename* WAS BYPASSED.

Explanation:
sysmod SYSMOD ID
zonename zone name that needs the cross-zone requisite

The SYSMOD named in GIM42001W specified SYSMOD *sysmod* as a cross-zone requisite. SMP/E found that the cross-zone requisite was needed in zone *zonename*. BYPASS(XZIFREQ) was specified, however, so SMP/E allowed processing to continue for the SYSMOD named in GIM42001W. The SYSMOD named in GIM42001W contains a ++IF MCS that defines the cross-zone requisite.

System Action: Processing continues.

Programmer Response: To ensure that the system will function properly, install the named cross-zone requisite in the indicated zone.

GIM35922I SYSMOD *sysmod*, WHICH SPECIFIES THE INDICATED SYSMOD AS A PREREQUISITE, FAILED.

Explanation:
sysmod SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* specifies the SYSMOD in GIM30209E as a prerequisite. Therefore, SYSMOD *sysmod* must be restored along with the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.

GIM35923I SYSMOD *sysmod1* IN ZONE *zonename* REQUIRES SYSMOD *sysmod2*.

Explanation:
sysmod1 SYSMOD ID
zonename zone name that needs the cross-zone requisite
sysmod2 SYSMOD ID

SMP/E found that SYSMOD *sysmod2* is a cross-zone requisite for SYSMOD *sysmod1* in zone *zonename*. SYSMOD *sysmod1* contains a ++IF MCS that defines the cross-zone requisite.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Restore the causers in the zones indicated in message GIM35923I and then restore failing SYSMOD *sysmod1*.
- Restore SYSMOD *sysmod1* using BYPASS(XZIFREQ) and then Restore the causers in the zones indicated in message GIM35923I.

GIM35924I SYSMOD *sysmod*, WHICH SPECIFIES THE INDICATED SYSMOD AS A COREQUISITE, FAILED.

Explanation:
sysmod SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* specifies the SYSMOD in GIM30209E as a corequisite. Therefore, SYSMOD *sysmod* must be restored along with or before the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.
- Restore SYSMOD *sysmod*, then restore the SYSMOD in GIM30209E.

GIM35925I CROSS-ZONE REQUISITE SYSMOD *sysmod* REQUIRED IN ZONE *zonename* WAS MISSING.

Explanation:

sysmod	SYSMOD ID
zonename	zone name that needs the cross-zone requisite

The SYSMOD named in GIM30209E specified SYSMOD *sysmod* as a cross-zone requisite. SMP/E found that the cross-zone requisite was needed in zone *zonename*, but was missing. The SYSMOD named in GIM30209E contains a ++IF MCS that defines the cross-zone requisite.

System Action: See message GIM30209E.

Programmer Response: See message GIM30209E.

GIM35926I SYSMOD *sysmod*, WHICH SPECIFIES THE INDICATED SYSMOD AS A CONDITIONAL REQUISITE, FAILED.

Explanation:

sysmod	SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* specifies the SYSMOD in GIM30209E as a conditional requisite. Therefore, SYSMOD *sysmod* must be restored along with or before the SYSMOD in GIM30209E. SYSMOD *sysmod* contains the ++IF MCS that defines the requisite.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.
- Restore SYSMOD *sysmod*, then restore the SYSMOD in GIM30209E.

GIM35927I SYSMOD *sysmod*, WHICH SPECIFIES THE INDICATED SYSMOD AS AN FMID, FAILED.

Explanation:

sysmod	SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* specifies the SYSMOD in GIM30209E as an FMID. Therefore, SYSMOD *sysmod* must be restored along with the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.

GIM35928I SYSMOD *sysmod*, WHICH IS AN FMID FOR THE INDICATED SYSMOD, FAILED.

Explanation:

sysmod	SYSMOD ID

SYSMOD *sysmod* is the FMID that owns the SYSMOD indicated in a previous SMP/E message.

- If this message was issued for APPLY or ACCEPT processing, it indicates that SYSMOD *sysmod* must be applied or accepted along with or before the SYSMOD indicated in the previous SMP/E message.
- If this message was issued for RESTORE processing, SMP/E tried to restore the SYSMOD named in GIM30209E. Therefore, SYSMOD *sysmod* must either be accepted, or it must be restored along with the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response:

If this message was issued for APPLY or ACCEPT processing, check the following:

- Did SMP/E successfully install SYSMOD *sysmod* in the target libraries?
- Did SMP/E select SYSMOD *sysmod* for installation during the APPLY or ACCEPT step for the SYSMOD named in message GIM30209E?

  If SMP/E is installing the requisite SYSMOD during the current APPLY or ACCEPT step, look at the preceding messages in SMPOUT to find why the requisite SYSMOD failed.

Note: If a SYSMOD that supersedes SYSMOD *sysmod* fails, the requisite SYSMOD might also fail.

If this message was issued for RESTORE processing, do **one** of the following:

- Accept SYSMOD *sysmod* if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209E.
- Restore SYSMOD *sysmod* along with the SYSMOD named in GIM30209E. There are two ways to do this:

– Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
– Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.

After this, you can reapply SYSMOD *sysmod* if you want those changes in the target libraries.

GIM35929I CROSS-ZONE REQUISITE SYSMOD *sysmod1* CAUSED BY SYSMOD *sysmod2* IN ZONE *zonename* WAS BYPASSED.

Explanation:

sysmod1	SYSMOD ID
sysmod2	SYSMOD ID
zonename	zone name

The SYSMOD named in GIM42001W had SYSMOD *sysmod1* as a cross-zone requisite from zone *zonename*. However, the XZIFREQ bypass option was specified for SYSMOD *sysmod1*, so SMP/E allowed the processing of the SYSMOD named in message GIM42001W to continue. The cross-zone requisite in zone *zonename* was caused by SYSMOD *sysmod2*.

System Action: Processing continues.

Programmer Response: None.

GIM35930I PREREQUISITE SYSMOD *sysmod* WAS BYPASSED.

Explanation:

sysmod	SYSMOD ID

The SYSMOD named in GIM42001W named SYSMOD *sysmod* as a prerequisite. However, **BYPASS(PRE)** was specified, so SMP/E did not make sure the requisite was satisfied.

System Action: See message GIM42001W.

Programmer Response: None.

GIM35931I COREQUISITE SYSMOD *sysmod* WAS BYPASSED.

Explanation:

sysmod	SYSMOD ID

The SYSMOD named in GIM42001W named SYSMOD *sysmod* as a corequisite. However, **BYPASS(REQ)** was specified, so SMP/E did not make sure the requisite was satisfied.

System Action: See message GIM42001W.

Programmer Response: None.

GIM35932I CONDITIONAL REQUISITE SYSMOD *sysmod* WAS BYPASSED.

Explanation:

sysmod	SYSMOD ID

The SYSMOD named in GIM42001W named SYSMOD *sysmod* as a conditional requisite. However, **BYPASS(IFREQ)** was specified, so SMP/E did not make sure the requisite was satisfied. The SYSMOD named in GIM42001W contains the ++IF MCS that defines the requisite.

System Action: See message GIM42001W.

Programmer Response: None.

GIM35933I CONDITIONAL REQUISITE SYSMOD *sysmod1* WAS BYPASSED. SYSMOD *sysmod2* CONTAINED THE ++IF STATEMENT.

Explanation:

sysmod1	SYSMOD ID
sysmod2	ID of the SYSMOD that contained the ++IF MCS

The SYSMOD named in GIM42001W named SYSMOD *sysmod1* as a conditional requisite. However, **BYPASS(IFREQ)** was specified, so SMP/E did not make sure the requisite was satisfied. SYSMOD *sysmod2* contains the ++IF MCS that defines the requisite.

System Action: See message GIM42001W.

Programmer Response: None.

GIM35934I ERROR HOLD *reason* WAS BYPASSED.

Explanation:

reason	error reason ID

The SYSMOD named in GIM42001W was held for the indicated error reason ID. However, **BYPASS(HOLDERR(*reason*))** was specified, so SMP/E assumed that the reason ID was resolved. (The user is responsible for resolving the reason ID before specifying the BYPASS operand.)

System Action: See message GIM42001W.

Programmer Response: None.

GIM35935I SYSTEM HOLD *reason* WAS BYPASSED.

Explanation:

reason	system reason ID

The SYSMOD named in GIM42001W was held for the indicated system reason ID. However, **BYPASS(HOLDSYS(*reason*))** was specified, so SMP/E

assumed that the reason ID was resolved. (The user is responsible for resolving the reason ID before specifying the BYPASS operand.)

System Action: See message GIM42001W.

Programmer Response: None.

GIM35936I USER HOLD *reason* WAS BYPASSED.

Explanation:

reason	user reason ID

The SYSMOD named in GIM42001W was held for the indicated user reason ID. However, **BYPASS(HOLDUSER(*reason*))** was specified, so SMP/E assumed that the reason ID was resolved. (The user is responsible for resolving the reason ID before specifying the BYPASS operand.)

System Action: See message GIM42001W.

Programmer Response: None.

GIM35937I ERROR HOLD *reason* WAS BYPASSED BECAUSE CLASS *class* WAS BYPASSED.

Explanation:

reason	error reason ID
class	hold class

The SYSMOD named in GIM42001W was held for the indicated error reason ID and the associated class. However, **BYPASS(HOLDCLASS(*reason*))** was specified, so SMP/E assumed that the reason ID was resolved. (The user is responsible for resolving the reason ID before specifying the BYPASS operand.)

System Action: See message GIM42001W.

Programmer Response: None.

GIM35938I SYSTEM HOLD *reason* WAS BYPASSED BECAUSE CLASS *class* WAS BYPASSED.

Explanation:

reason	system reason ID
class	hold class

The SYSMOD named in GIM42001W was held for the indicated system reason ID and the associated class. However, **BYPASS(HOLDCLASS(*reason*))** was specified, so SMP/E assumed that the reason ID was resolved. (The user is responsible for resolving the reason ID before specifying the BYPASS operand.)

System Action: See message GIM42001W.

Programmer Response: None.

GIM35939I USER HOLD *reason* WAS BYPASSED BECAUSE CLASS *class* WAS BYPASSED.

Explanation:

reason	user reason ID
class	hold class

The SYSMOD named in GIM42001W was held for the indicated user reason ID and the associated class. However, **BYPASS(HOLDCLASS(*reason*))** was specified, so SMP/E assumed that the reason ID was resolved. (The user is responsible for resolving the reason ID before specifying the BYPASS operand.)

System Action: See message GIM42001W.

Programmer Response: None.

GIM35940I SYSMOD *sysmod1*, WHICH WOULD HAVE RESOLVED HOLD *holdid* FOR SYSMOD *sysmod2*, FAILED.

Explanation:

sysmod1	superseding SYSMOD ID
holdid	HOLD ID
sysmod2	failing SYSMOD ID

During GROUPEXTEND processing, the SYSMOD that supersedes a hold has failed. This message follows GIM30206E.

System Action: SYSMOD processing stops.

Programmer Response: Fix the error and rerun the job.

GIM35941I SYSMOD *sysmod1*, WHICH WOULD HAVE DELETED SYSMOD *sysmod2*, FAILED.

Explanation:

sysmod1	deleting SYSMOD ID
sysmod2	SYSMOD ID

This message follows GIM30201E and names the SYSMODs that delete the SYSMOD in GIM30201E. SMP/E could not process SYSMOD *sysmod2* because it was supposed to be deleted, but all the SYSMODs that were supposed to delete it failed.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Fix the error and rerun the job.

GIM35942I SYSMOD *sysmod1*, WHICH WOULD HAVE SUPERSEDED SYSMOD *sysmod2*, FAILED.

Explanation:

sysmod1	superseding SYSMOD ID
sysmod2	SYSMOD ID

This message follows GIM30202E and names the superseding SYSMODs. SMP/E could not process SYSMOD *sysmod2* because it was supposed to be superseded, but all the SYSMODs that were supposed to supersede it failed.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response: Fix the error and rerun the job.

GIM35943I SYSMOD *sysmod*, WHICH SUPERSEDES THE INDICATED SYSMOD, WAS MISSING.

Explanation:

sysmod	SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* supersedes the SYSMOD in GIM30209E. Therefore, SYSMOD *sysmod* must be restored along with the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.

GIM35944I SYSMOD *sysmod*, WHICH SPECIFIES THE INDICATED SYSMOD AS A PREREQUISITE, WAS MISSING.

Explanation:

sysmod	SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* specifies the SYSMOD in GIM30209E as a prerequisite. Therefore, SYSMOD *sysmod* must be restored along with the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.

GIM35945I SYSMOD *sysmod*, WHICH SPECIFIES THE INDICATED SYSMOD AS A COREQUISITE, WAS MISSING.

Explanation:

sysmod	SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* specifies the SYSMOD in GIM30209E as a corequisite. Therefore, SYSMOD *sysmod* must be restored along with or before the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.
- Restore SYSMOD *sysmod*, then restore the SYSMOD in GIM30209E.

GIM35946I SYSMOD *sysmod*, WHICH SPECIFIES THE INDICATED SYSMOD AS A CONDITIONAL REQUISITE, WAS MISSING.

Explanation:

sysmod	SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* specifies the SYSMOD in GIM30209E as a conditional requisite. Therefore, SYSMOD *sysmod* must be restored along with or before the SYSMOD in GIM30209E. SYSMOD *sysmod* contains the ++IF MCS that defines the requisite.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.
- Restore SYSMOD *sysmod*, then restore the SYSMOD in GIM30209E.

GIM35947I SYSMOD *sysmod*, WHICH SPECIFIES THE INDICATED SYSMOD AS AN FMID, WAS MISSING.

Explanation:

sysmod	SYSMOD ID

SMP/E tried to restore the SYSMOD named in GIM30209E, but SYSMOD *sysmod* specifies the SYSMOD in GIM30209E as an FMID. Therefore, SYSMOD *sysmod* must be restored along with the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.

GIM35948I SYSMOD *sysmod*, WHICH IS AN FMID FOR THE INDICATED SYSMOD, WAS MISSING.

Explanation:

sysmod	SYSMOD ID

SYSMOD *sysmod* is the FMID that owns the SYSMOD indicated in a previous SMP/E message.

- If this message was issued for APPLY or ACCEPT processing, it indicates that SYSMOD *sysmod* must be applied or accepted along with or before the SYSMOD indicated in the previous SMP/E message.
- If this message was issued for RESTORE processing, SMP/E tried to restore the SYSMOD named in GIM30209E. Therefore, SYSMOD *sysmod* must either be accepted, or it must be restored along with the SYSMOD in GIM30209E.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- For APPLY or ACCEPT processing, check the following:
  - Did SMP/E successfully install SYSMOD *sysmod* in the target libraries?
  - Did SMP/E select SYSMOD *sysmod* for installation during the APPLY or ACCEPT step for the SYSMOD named in message GIM30209E?

    If SMP/E is installing the requisite SYSMOD during the APPLY or ACCEPT step, look at the preceding messages in SMPOUT to find why the requisite SYSMOD failed.

  Note: If a SYSMOD that supersedes SYSMOD *sysmod* fails, the requisite SYSMOD might also fail.
- For RESTORE processing, do **one** of the following:
  - Accept SYSMOD *sysmod*, if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209E.
  - Restore SYSMOD *sysmod* along with the SYSMOD named in GIM30209E. There are two ways to do this:
    1. Specify SYSMOD *sysmod* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209E.
    2. Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209E.

    After this, you can reapply SYSMOD *sysmod* if you want those changes in the target libraries.

GIM35949I ERROR HOLD *reason* WAS NOT RESOLVED. SYSMOD *sysmod* WOULD HAVE SUPERSEDED *reason*, BUT HAS FAILED.

Explanation:

reason	error hold reason ID
sysmod	ID of a SYSMOD that supersedes the reason ID

SMP/E could not process the SYSMOD named in GIM30206E, because that SYSMOD was held for the indicated error reason ID (APAR), which was not resolved. SYSMOD *sysmod* would have resolved the error reason ID, but has failed processing.

System Action: See message GIM30206E.

Programmer Response: Do **one** of the following:

- Resolve the errors that caused SYSMOD *sysmod* to fail, and rerun the APPLY or ACCEPT command.
- Rerun the APPLY or ACCEPT command to install the SYSMOD named in GIM30206E, but specify **BYPASS(HOLDERR(*reason*))**.

GIM35950I ERROR HOLD *reason* WAS NOT RESOLVED. SYSMOD *sysmod* WOULD HAVE SUPERSEDED *reason*, BUT WAS EXCLUDED.

Explanation:

reason	error hold reason ID
sysmod	ID of a SYSMOD that supersedes the reason ID

SMP/E could not process the SYSMOD named in GIM30206E, because that SYSMOD was held for the indicated error reason ID (APAR), which was not resolved. SYSMOD *sysmod* would have resolved the error reason ID, but was excluded from processing.

System Action: See message GIM30206E.

Programmer Response: Do **one** of the following:

- Remove SYSMOD *sysmod* from the EXCLUDE operand, and rerun the command.
- Rerun the APPLY or ACCEPT command to install the SYSMOD named in GIM30206E, but specify **BYPASS(HOLDERR(*reason*))**.

GIM35951I PREREQUISITE SYSMOD *sysmod1* **WAS MISSING. SYSMOD** *sysmod2* **WOULD HAVE SUPERSEDED** *sysmod1***, BUT HAS FAILED.**

Explanation:

sysmod1 SYSMOD ID
sysmod2 ID of a SYSMOD that supersedes *sysmod1*

SYSMOD *sysmod1* is a prerequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx*, because SYSMOD *sysmod1* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *sysmod2* would have superseded SYSMOD *sysmod1*, but has failed processing.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- Resolve the errors that caused SYSMOD *sysmod2* to fail, and rerun the APPLY or ACCEPT command.
- Receive SYSMOD *sysmod1*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35952I PREREQUISITE SYSMOD *sysmod1* **WAS MISSING. SYSMOD** *sysmod2* **WOULD HAVE SUPERSEDED** *sysmod1***, BUT WAS EXCLUDED.**

Explanation:

sysmod1 SYSMOD ID
sysmod2 ID of a SYSMOD that supersedes *sysmod1*

SYSMOD *sysmod1* is a prerequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx*, because SYSMOD *sysmod1* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *sysmod2* would have superseded SYSMOD *sysmod1*, but was excluded from processing.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- Remove SYSMOD *sysmod2* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *sysmod1*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35953I COREQUISITE SYSMOD *sysmod1* **WAS MISSING. SYSMOD** *sysmod2* **WOULD HAVE SUPERSEDED** *sysmod1***, BUT HAS FAILED.**

Explanation:

sysmod1 SYSMOD ID
sysmod2 ID of a SYSMOD that supersedes *sysmod1*

SYSMOD *sysmod1* is a corequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx*, because SYSMOD *sysmod1* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *sysmod2* would have superseded SYSMOD *sysmod1*, but has failed processing.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- Resolve the errors that caused SYSMOD *sysmod2* to fail, and rerun the APPLY or ACCEPT command.
- Receive SYSMOD *sysmod1*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35954I COREQUISITE SYSMOD *sysmod1* **WAS MISSING. SYSMOD** *sysmod2* **WOULD HAVE SUPERSEDED** *sysmod1***, BUT WAS EXCLUDED.**

Explanation:

sysmod1 SYSMOD ID
sysmod2 ID of a SYSMOD that supersedes *sysmod1*

SYSMOD *sysmod1* is a corequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx*, because SYSMOD *sysmod1* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *sysmod2* would have superseded SYSMOD *sysmod1*, but was excluded from processing.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- Remove SYSMOD *sysmod2* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *sysmod1*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35955I CONDITIONAL REQUISITE SYSMOD *sysmod1* **WAS MISSING. SYSMOD** *sysmod2* **WOULD HAVE SUPERSEDED** *sysmod1***, BUT HAS FAILED.**

Explanation:

sysmod1 SYSMOD ID
sysmod2 ID of a SYSMOD that supersedes *sysmod1*

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx*, because SYSMOD *sysmod1* either is not available (it is

not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *sysmod2* would have superseded SYSMOD *sysmod1*, but has failed processing.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- Resolve the errors that caused SYSMOD *sysmod2* to fail, and rerun the APPLY or ACCEPT command.
- Receive SYSMOD *sysmod1*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35956I CONDITIONAL REQUISITE SYSMOD *sysmod1* WAS MISSING. SYSMOD *sysmod2* WOULD HAVE SUPERSEDED *sysmod1*, BUT WAS EXCLUDED.

Explanation:

sysmod1	SYSMOD ID
sysmod2	ID of a SYSMOD that supersedes *sysmod1*

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx*, because SYSMOD *sysmod1* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *sysmod2* would have superseded SYSMOD *sysmod1*, but was excluded from processing.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- Remove SYSMOD *sysmod2* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *sysmod1*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35957I CONDITIONAL REQUISITE SYSMOD *sysmod1* WAS MISSING. SYSMOD *sysmod2* WOULD HAVE SUPERSEDED *sysmod1*, BUT HAS FAILED. SYSMOD *sysmod3* CONTAINED THE ++IF STATEMENT.

Explanation:

sysmod1	SYSMOD ID
sysmod2	ID of a SYSMOD that supersedes *sysmod1*
sysmod3	ID of the SYSMOD that contained the ++IF MCS

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx*, because SYSMOD *sysmod1* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *sysmod2* would have superseded SYSMOD *sysmod1*, but has failed processing. SYSMOD *sysmod3* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- Resolve the errors that caused SYSMOD *sysmod2* to fail, and rerun the APPLY or ACCEPT command.
- Receive SYSMOD *sysmod1*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35958I CONDITIONAL REQUISITE SYSMOD *sysmod1* WAS MISSING. SYSMOD *sysmod2* WOULD HAVE SUPERSEDED *sysmod1*, BUT WAS EXCLUDED. SYSMOD *sysmod3* CONTAINED THE ++IF STATEMENT.

Explanation:

sysmod1	SYSMOD ID
sysmod2	ID of a SYSMOD that supersedes *sysmod1*
sysmod3	ID of the SYSMOD that contained the ++IF MCS

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx*, because SYSMOD *sysmod1* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *sysmod2* would have superseded SYSMOD *sysmod1*, but was excluded from processing. SYSMOD *sysmod3* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Do **one** of the following:

- Remove SYSMOD *sysmod2* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *sysmod1*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35959I CROSS-ZONE REQUISITE SYSMOD *sysmod1* CAUSED BY SYSMOD *sysmod2* IN ZONE *zonename* WAS MISSING.

Explanation:

sysmod1	SYSMOD ID
sysmod2	SYSMOD ID
zonename	zone name

The SYSMOD named in GIM30209E had SYSMOD *sysmod1* as a cross-zone requisite. The cross-zone requisite was caused by SYSMOD *sysmod2* in zone *zonename*. SMP/E could not process the SYSMOD

named in GIM30209E, because SYSMOD *sysmod1* is not available (it is not in the global zone), or is available but was not selected for processing.

System Action: See message GIM30209E.

Programmer Response: Receive the missing SYSMOD into the global zone and make sure that is selectable for processing. Then rerun the job.

GIM35960I CROSS-ZONE REQUISITE SYSMOD *sysmod1* CAUSED BY SYSMOD *sysmod2* IN ZONE *zonename* WAS EXCLUDED.

Explanation:

sysmod1	SYSMOD ID
sysmod2	SYSMOD ID
zonename	zone name

The SYSMOD named in GIM30209E had SYSMOD *sysmod1* as a cross-zone requisite. The cross-zone requisite was caused by SYSMOD *sysmod2* in zone *zonename*. SMP/E could not process the SYSMOD named in GIM30209E, because SYSMOD *sysmod1* was specified on the EXCLUDE operand.

System Action: See message GIM30209E.

Programmer Response: Remove SYSMOD *sysmod1* from the EXCLUDE operand and rerun the job.

GIM35961I CROSS-ZONE REQUISITE SYSMOD *sysmod1* CAUSED BY SYSMOD *sysmod2* IN ZONE *zonename* WAS HELD.

Explanation:

sysmod1	SYSMOD ID
sysmod2	SYSMOD ID
zonename	zone name

The SYSMOD named in GIM30209E had SYSMOD *sysmod1* as a cross-zone requisite. The cross-zone requisite was caused by SYSMOD *sysmod2* in zone *zonename*. SMP/E could not process the SYSMOD named in GIM30209E, because a ++HOLD MCS for SYSMOD *sysmod1* was not resolved.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- For an ERROR hold, resolve it by installing a resolving SYSMOD: either a SYSMOD whose ID matches the ERROR hold reason ID, or a SYSMOD that supersedes that reason ID. Do this either before rerunning the job or as part of rerunning the job.
- For a SYSTEM or USER hold, specify the appropriate value on the BYPASS operand and rerun the job.

GIM35962I CROSS-ZONE REQUISITE SYSMOD *sysmod1* CAUSED BY SYSMOD *sysmod2* IN ZONE *zonename* FAILED.

Explanation:

sysmod1	SYSMOD ID
sysmod2	SYSMOD ID
zonename	zone name

The SYSMOD named in GIM30209E had SYSMOD *sysmod1* as a cross-zone requisite. The cross-zone requisite was caused by SYSMOD *sysmod2* in zone *zonename*. SMP/E could not process the SYSMOD named in GIM30209E, because SYSMOD *sysmod1* failed. The reason for the failure is indicated by a previous message.

System Action: See message GIM30209E.

Programmer Response: Look at the preceding messages in SMPOUT to determine why the SYSMOD failed. Fix the error and rerun the job.

GIM35963I CROSS-ZONE REQUISITE SYSMOD *sysmod1* CAUSED BY SYSMOD *sysmod2* IN ZONE *zonename* WAS MISSING. SYSMOD *sysmod3* WOULD HAVE SUPERSEDED *sysmod1*, BUT HAS FAILED.

Explanation:

sysmod1	SYSMOD ID
sysmod2	causing SYSMOD ID
zonename	zone name for causing SYSMOD
sysmod3	ID of a SYSMOD that supersedes *sysmod1*

SYSMOD *sysmod1* is a cross-zone requisite for SYSMOD *sysmod2* in zone *zonename*. However, SMP/E could not process SYSMOD *sysmod2* because SYSMOD *sysmod1* either is not available (it is not in the global zone) or is available, but was not selected for processing. SYSMOD *sysmod3* would have superseded *sysmod1*, but has failed processing. but has failed processing.

System Action: See message GIM30209E.

Programmer Response: Do **one** of the following:

- Remove SYSMOD *sysmod3* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *sysmod1* into the global zone, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35964I CROSS-ZONE REQUISITE SYSMOD *sysmod1* CAUSED BY SYSMOD *sysmod2* IN ZONE *zonename* WAS MISSING. SYSMOD *sysmod3* WOULD HAVE SUPERSEDED *sysmod1*, BUT WAS EXCLUDED.

Explanation:

sysmod1	SYSMOD ID
sysmod2	causing SYSMOD ID
zonename	zone name for causing SYSMOD
sysmod3	ID of a SYSMOD that supersedes *sysmod1*

SYSMOD *sysmod1* is a cross-zone requisite for SYSMOD *sysmod2* in zone *zonename*. However, SMP/E could not process SYSMOD *sysmod2*, because SYSMOD *sysmod1* either is not available (it is not in the global zone) or is available, but was not selected for processing. SYSMOD *sysmod3* would have superseded SYSMOD *sysmod1*, but was excluded from processing.

System Action: See message GIM30206E.

Programmer Response: Do **one** of the following:

- Remove SYSMOD *sysmod3* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *sysmod1* into the global zone, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

GIM35965I SYSTEM HOLD *reason* ORIGINATED BY SYSMOD *sysmod* WAS NOT RESOLVED.

Explanation:

reason	system hold reason ID
sysmod	SYSMOD ID

One of the following occurred:

- GIM30206E was also issued.

  SMP/E could not process the SYSMOD named in GIM30206E because it was held for the indicated system reason ID as originated by the indicated SYSMOD ID. The hold was not resolved.
- A GIM302xx message other than GIM30206E was also issued.

  The previous GIM302xx (for example, GIM30209E) indicates why APPLY or ACCEPT processing failed for the indicated SYSMOD. Message GIM35965 is issued along with other GIM359xx messages, even if you specified BYPASS(HOLDSYSTEM) or some other variation of the BYPASS operand that would have bypassed the reported condition.

System Action: See the description of the associated GIM302xx message that was issued.

Programmer Response:

- If GIM30206E was also issued, do any processing needed to resolve the system reason ID as originated by the indicated SYSMOD ID. Then rerun the ACCEPT or APPLY command to install the SYSMOD named in GIM30206E. You may need to specify BYPASS(HOLDSYS(*reason*)) or some other variation of BYPASS.
- If a GIM302xx message other than GIM30206E was also issued, do any processing needed to resolve the other GIM359xx messages that were issued. Then rerun the ACCEPT or APPLY command to install the SYSMOD named in the GIM302xx message. You may need to specify BYPASS(HOLDSYS(*reason*)) or some other variation of BYPASS.

GIM35966I SYSTEM HOLD *reason* ORIGINATED BY SYSMOD *sysmod* WAS BYPASSED.

Explanation:

reason	system hold reason ID
sysmod	SYSMOD ID

The SYSMOD named in GIM42001W was held for the indicated system reason ID as originated by the indicated SYSMOD ID. However, some form of BYPASS was specified that bypassed the indicated system reason ID, so SMP/E assumed that the reason ID was resolved. (The user is responsible for resolving the reason ID before specifying the BYPASS operand.)

System Action: See message GIM42001W.

Programmer Response: None.

GIM35967I PREREQUISITE SYSMOD *sysmod* IS BEING DELETED.

Explanation:

sysmod	SYSMOD ID

SYSMOD *sysmod* is a prerequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* is being deleted.

System Action: See message GIM302*xx*.

Programmer Response: Install a SYSMOD which supersedes *sysmod* along with the current group of SYSMODs being installed.

GIM35968I COREQUISITE SYSMOD *sysmod* IS BEING DELETED.

Explanation:

sysmod	SYSMOD ID

SYSMOD *sysmod* is a corequisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* is being deleted.

System Action: See message GIM302*xx*.

Programmer Response: Install a SYSMOD which supersedes *sysmod* along with the current group of SYSMODs being installed.

GIM35969I CONDITIONAL REQUISITE SYSMOD *sysmod* IS BEING DELETED.

Explanation:

sysmod	SYSMOD ID

SYSMOD *sysmod* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod* is being deleted. The SYSMOD named in GIM302*xx* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Install a SYSMOD which supersedes *sysmod* along with the current group of SYSMODs being installed.

GIM35970I CONDITIONAL REQUISITE SYSMOD *sysmod1* IS BEING DELETED. SYSMOD *sysmod2* CONTAINED THE ++IF STATEMENT.

Explanation:

sysmod1	SYSMOD ID
sysmod2	ID of the SYSMOD that contained the ++IF MCS

SYSMOD *sysmod1* is a conditional requisite for the SYSMOD named in GIM302*xx*. However, SMP/E could not process the SYSMOD named in GIM302*xx* because SYSMOD *sysmod1* is being deleted. SYSMOD *sysmod2* contains the ++IF MCS that defines the requisite.

System Action: See message GIM302*xx*.

Programmer Response: Install a SYSMOD which supersedes *sysmod1* along with the current group of SYSMODs being installed.

GIM35971I CROSS-ZONE REQUISITE SYSMOD *sysmod1* CAUSED BY SYSMOD *sysmod2* IN ZONE *zonename* IS BEING DELETED.

Explanation:

sysmod1	SYSMOD ID
sysmod2	SYSMOD ID
zonename	zone name

The SYSMOD named in GIM30209E had SYSMOD *sysmod1* as a cross-zone requisite. The cross-zone requisite was caused by SYSMOD *sysmod2* in zone *zonename*. SMP/E could not process the SYSMOD named in GIM30209E because SYSMOD *sysmod1* is being deleted.

System Action: See message GIM30209E

Programmer Response: Install a SYSMOD which supersedes *sysmod1* along with the current group of SYSMODs being installed.

GIM36000S THE *suboper* SUBOPERAND ON THE BYPASS OPERAND WAS FOUND ON THE RECEIVE COMMAND. THIS RELEASE OF SMP/E DOES NOT SUPPORT THE *suboper* SUBOPERAND. OS/390 RELEASE 5 OR HIGHER IS NEEDED.

Explanation:

suboper	APPLYCHECK or ACCEPTCHECK

SMP/E found the indicated BYPASS suboperand on the indicated command. The suboperand is not supported by the release of SMP/E that is executing.

System Action: Command processing stops.

Programmer Response: Use the appropriate level of SMP/E that supports the BYPASS suboperand (the SMP/E element of OS/390 Release 5 or higher).

GIM36100I DATA SET *sysut1* WAS *action* INTO DATA SET *sysut2*.

Explanation:

sysut1	data set name of the SYSUT1 data set
action	COMPACTED or EXPANDED
sysut2	data set name of the SYSUT2 data set

GIMCPTS successfully processed the records from the SYSUT1 data set. It either compacted or expanded the input from the SYSUT1 data set into the SYSUT2 data set.

System Action: GIMCPTS processing is complete.

Programmer Response: None.

GIM36201E THE ./ CHANGE STATEMENT WAS MISSING FROM THE INPUT FOR THE UPDATE UTILITY.

Explanation: The text following an element update MCS did not contain a ./ CHANGE statement.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and run RECEIVE again.

GIM36202E THERE WAS MORE THAN ONE ./ CHANGE STATEMENT IN THE INPUT FOR THE UPDATE UTILITY.

Explanation: The text following an element update MCS contained more than one ./ CHANGE statement.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and run RECEIVE again.

GIM36203E AN UPDATE CONTROL STATEMENT FOLLOWS A ./ ENDUP STATEMENT. THIS IS NOT ALLOWED.

Explanation: The ./ ENDUP statement is used to indicate the end of input for the update utility. However, SMP/E found an update control statement after a ./ ENDUP statement. This is not allowed.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and run RECEIVE again.

GIM36204E THE SEQUENCE NUMBERS ARE OUT OF ORDER IN THE INPUT FOR THE UPDATE UTILITY.

Explanation: In the text following an element update MCS, the sequence numbers in columns 73—80 of one update input record are lower than those of the previous input record. For both SMP/E and update utility processing, sequence numbers must be in increasing order.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and run RECEIVE again.

GIM36301E *value* **IS NOT ALLOWED AS** *operand1* **AND** *operand2***.**

Explanation:

value	operand value
operand1	first operand
operand2	second operand

The ++PRODUCT MCS or SYSMOD is not built correctly because the same value was specified for the indicated operands.

System Action: ++PRODUCT MCS or SYSMOD processing stops.

Programmer Response: Fix the problem and rerun the job.

GIM36302E THE ++IF FMID VALUE MATCHES THE ++VER FMID VALUE. THIS IS NOT ALLOWED.

Explanation: The SYSMOD is not built correctly because the FMID value on the ++IF MCS is the same as the FMID value on the ++VER MCS.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and rerun the job.

GIM36303E ONE ++VER MCS IN THE SYSMOD SPECIFIES AN FMID VALUE, BUT ANOTHER DOES NOT. THIS IS NOT ALLOWED.

Explanation: The SYSMOD is not built correctly because the ++VER MCS statements are inconsistent. One specified an FMID value, which indicates that the SYSMOD **is not** a base function. However, another did not specify an FMID value, which indicates that the SYSMOD **is** a base function. A SYSMOD must be one or the other; it cannot be both.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and rerun the job.

GIM36304E THE ++VER *operand* **OPERAND IS NOT ALLOWED IN A SERVICE SYSMOD.**

Explanation:

operand	DELETE or NPRE

The SYSMOD is not built correctly because the indicated operand was specified on the ++VER MCS, but the SYSMOD is not a function. Only functions may specify these operands.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and rerun the job.

GIM36305E THE ++IF FMID VALUE MATCHES THE ID VALUE ON THE SYSMOD. THIS IS NOT ALLOWED.

Explanation: The SYSMOD is not built correctly because it specifies its own ID as the FMID on the ++IF MCS. Any unconditional requisites for this SYSMOD should be specified on its ++VER MCS. If there are conditional requisites for the SYSMOD that depend on the presence of another function, that other function's FMID should be specified on the ++IF MCS.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and rerun the job.

GIM36306E THE ++ASSIGN MCS IS OUT OF SEQUENCE.

Explanation: SMP/E expects the next statement after ++ASSIGN to be one of the following:

- ++FUNCTION, ++PTF, ++APAR, or ++USERMOD
- Another ++ASSIGN
- An end-of-file

None of these followed the ++ASSIGN MCS.

System Action: None of the SYSMODs between this ++ASSIGN MCS and the next valid ++ASSIGN MCS, or

an end-of-file, will be received. Likewise, no ++ASSIGN SOURCEID TO statements up to the next valid ++ASSIGN MCS or end-of-file will be processed.

Programmer Response: Fix the problem and rerun the job.

GIM36307E THE ++HOLD MCS IS OUT OF SEQUENCE.

Explanation: The SYSMOD is not built correctly. Any ++HOLD statements must be placed after all ++VER and ++IF statements and before the first ++JCLIN or element statement.

System Action: SYSMOD processing stops.

Programmer Response: Fix the problem and rerun the job.

GIM36308E THE ++*mcsname* FOR *entname* MCS IS OUT OF SEQUENCE.

Explanation:

mcsname	FEATURE or PRODUCT
entname	entry name

SMP/E expects the next statement after the ++FEATURE or ++PRODUCT to be one of the following:

- ++FUNCTION, ++PTF, ++APAR, or ++USERMOD
- ++ASSIGN
- another ++FEATURE or ++PRODUCT
- An end-of-file.

None of these followed the ++FEATURE or ++PRODUCT MCS.

System Action: The FEATURE or PRODUCT is not received.

Programmer Response: Contact the IBM Support Center.

GIM36401E *command* PROCESSING FAILED FOR *sysmod1* AND *sysmod2* BECAUSE SYSMOD *sysmod1* DELETES SYSMOD *sysmod2*. ONLY ONE OF THESE SYSMODS CAN BE PROCESSED AT A TIME.

Explanation:

command	an SMP/E command (APPLY or ACCEPT)
sysmod1	ID of the deleting SYSMOD
sysmod2	ID of the deleted SYSMOD

The two indicated function SYSMODs are eligible for processing. However, SYSMOD *sysmod1* is supposed to delete SYSMOD *sysmod2*. SMP/E cannot determine which SYSMOD should be processed.

System Action: SYSMOD processing stops for both functions. Processing also stops for any related SYSMODs, as indicated by message GIM37001.

Programmer Response: Change the APPLY or ACCEPT command so that only one of the SYSMODs is processed. Then rerun the job.

GIM36500E ALLZONES IS NOT ALLOWED AS THE VALUE OF THE RECEXZGRP OPERAND.

Explanation: *ALLZONES* is not allowed as a zoneset name specified on the RECEXZGRP operand.

System Action: SMP/E processing for this statement stops. Processing continues with the next UCL statement.

Programmer Response: If ALLZONES is the name of a zoneset, rename the zoneset for use by the RECEXZGRP operand.

GIM36600S *svc99* FAILED FOR THE *dataset* DATA SET USING THE *ddname* DDNAME - *errortext*

Explanation:

svc99	SVC99 function that failed
dataset	data set name
ddname	ddname associated with the data set name
errortext	the text of the error message from the message processing facility of SVC99 that shows the reason for the failure.

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated ddname.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC99.

System Action: A subsequent message identifies the system action. GIMXTRX or GIMZIP processing stops.

Programmer Response: Fix the error and retry the operation.

GIM36700I GIMXTRX *function* PROCESSING HAS ENDED. THE HIGHEST RETURN CODE WAS *rc*.

Explanation:

1. *function* GIMXTRX function

   The valid function names are:

BMPTZN	Create BITMAP of CSI
LSTTZN	List target zone names

2. *rc* return code

This message is issued upon completion of the GIMXTRX function.

Note:

System Action: When the return code is zero, no action is taken. For non-zero return codes, a previous message will indicate what the problem is.

Programmer Response: When the return code is zero, no action is required. For non-zero return codes, take the appropriate actions indicated by the previous messages.

GIM36800E SHELL SCRIPT *shellscr* **IS NEEDED TO PROCESS** *elmtype elmname* **FOR SYSMOD** *sysmod1*, **BUT IS BEING DELETED BY SYSMOD** *sysmod2*.

Explanation:

shellscr	shell script name
elmtype	element type
elmname	element name
sysmod1	SYSMOD ID that contains the element
sysmod2	SYSMOD ID that deletes the shell script

The indicated element requires the shell script *shellscr* to complete its processing. However, the shell script has been identified for deletion.

System Action: SYSMOD processing stops.

Programmer Response: Determine whether the shell script should be deleted.

- If the shell script should be deleted, then:
  - If you want to replace it with another shell script of the same name, add the SYSMOD with the replacement to the command's selection list. Retry the command.
  - If you just want to delete the shell script, do not select the SYSMOD that contains the element that requires it. Retry the command.
- If the shell script should not be deleted, do not select the SYSMOD that deletes it. Retry the command.

GIM36901E SYSMOD *sysmod* **WAS SPECIFIED ON THE SELECT OPERAND BUT IS NOT IN THE SMPPTFIN DATA SET.**

Explanation:

sysmod	SYSMOD ID

SYSMOD *sysmod* was specified on the SELECT operand of the RECEIVE command, but it could not be processed because it was not in the SMPPTFIN data set. For example, it may not be on the tape that is currently defined by the SMPPTFIN DD statement.

System Action: SYSMOD processing fails. RECEIVE processing continues.

Programmer Response: Obtain SYSMOD *sysmod* and define a SMPPTFIN DD statement that points to the tape or data set that contains the SYSMOD. Then rerun the job.

GIM37001S *command* **PROCESSING STOPPED BECAUSE FUNCTION SYSMOD** *sysmod* **FAILED.**

Explanation:

command	an SMP/E command
sysmod	SYSMOD ID

When a function SYSMOD fails, the command being processed also fails.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:

- Fix the error that caused the SYSMOD to fail.
- Exclude the failing SYSMOD from processing.

Then rerun the job.

GIM37100S REQUIRED KEYWORD PARAMETER *keyword* **IS MISSING OR INVALID.**

Explanation:

keyword	missing required keyword

This message is issued when the executing program could not find a required input parameter or the parameters following the keyword are invalid.

System Action: Program or command processing stops.

Programmer Response: If the program executing is GIMXTRX, contact your IBM service representative. Otherwise, fix the error and retry the operation.

GIM37150S THE *keyword* **KEYWORD IS MISSING OR INVALID.**

Explanation:

keyword	missing or invalid keyword

A keyword required in the input data set is missing or a specified keyword is invalid.

System Action: Processing stops.

Programmer Response: Correct the error and rerun the job.

GIM37201E *command* **PROCESSING FAILED FOR** *sysmod1* **AND** *sysmod2* **BECAUSE** *sysmod1* **AND** *sysmod2* **ARE MUTUALLY EXCLUSIVE SYSMODS. ONLY ONE OF THESE SYSMODS CAN BE PROCESSED AT A TIME.**

Explanation:

command	an SMP/E command (APPLY or ACCEPT)

sysmod1 ID of one SYSMOD
sysmod2 ID of the other SYSMOD

The two indicated function SYSMODs are eligible for processing. However, they specify each other as ++VER NPRE values, which makes them mutually exclusive. SMP/E cannot determine which SYSMOD should be processed.

System Action: SYSMOD processing stops for both functions. Processing also stops for any related SYSMODs, as indicated by message GIM37001.

Programmer Response: Change the APPLY or ACCEPT command so that only one of the SYSMODs is processed. Then rerun the job.

GIM37301E SYSMOD *sysmod* **HAS MORE THAN ONE APPLICABLE ++VER MCS. ONLY ONE IS ALLOWED.**

Explanation:
sysmod SYSMOD ID

SMP/E tried to apply or accept the indicated SYSMOD. However, because that SYSMOD contained ++VER statements that specified more than one valid FMID-SREL pair, SMP/E could not determine which of the FMIDs the SYSMOD should apply to.

An FMID-SREL pair is considered valid when the function specified as the FMID has already been applied or accepted, or when that function is being applied or accepted concurrently with SYSMOD *sysmod*.

System Action: SYSMOD processing stops.

Programmer Response: Change the SYSMOD so that it only contains one applicable ++VER MCS. Then rerun the job.

GIM37401I PROCESSING FAILED FOR SYSMOD *sysmod1***.** *sysmod1* **REQUIRES SYSMOD** *sysmod2***. HOWEVER, SYSMOD** *sysmod2* **SPECIFIES** *sysmod1* **ON ITS ++VER** *operand* **OPERAND.**

Explanation:
sysmod1 SYSMOD ID
sysmod2 second SYSMOD ID
operand operand on ++VER (FMID or PRE)

Processing for SYSMOD *sysmod2* stopped because SMP/E could not determine which SYSMOD to process first: SYSMOD *sysmod2* or SYSMOD *sysmod1*. The following example shows the error:

```
++FUNCTION(F000001).
++VER(Z038) PRE(P000001).
++MOD(IEYMYMOD).

++PTF(P000001).
++VER(Z038) FMID(F000001).
```

In this example, the function SYSMOD names a PTF SYSMOD as a prerequisite. However, SMP/E cannot process this prerequisite SYSMOD until it processes the function. This situation might also occur when two SYSMODs name each other as prerequisites.

System Action: SYSMOD processing stops.

Programmer Response: Do the following:
1. Reject the SYSMODs in error.
2. Define the relationship between these SYSMODs correctly.
3. Receive and either apply or accept the SYSMODs.

GIM37500I GIMXTRX *function* **PROCESSING HAS ENDED. THE OUTPUT DATA SET NAME IS** *dataset***.**

Explanation:

1. *function* GIMXTRX function

   The valid function names are:
   BMPTZN Create BITMAP of CSI
   LSTTZN List target zone names
2. *dataset* name of the output data set from GIMXTRX

This message is issued when the GIMXTRX program completes successfully.

System Action: None.

Programmer Response: None.

GIM37600S *command* **PROCESSING FAILED FOR THE** *dataset* **DATA SET.**

Explanation:
command OPEN or CLOSE
dataset file name

This message is issued when the GIMXTRX program could not successfully open or close the indicated file name.

System Action: GIMXTRX processing ends.

Programmer Response: Contact your IBM service representative.

GIM37701S SMPTLIB *smptlib* **WAS NOT DELETED BECAUSE IT IS DEFINED AS A DUMMY DATA SET.**

Explanation:
smptlib data set name

During REJECT processing, the indicated SMPTLIB data set could not be deleted because the SMPTLIB DD statement specifies either the DUMMY operand or **DSN=NULLFILE**.

System Action: The SYSMOD is not rejected.

Programmer Response:
1. Remove the DUMMY operand or DSN=NULLFILE from the SMPTLIB DD statement.
2. Specify the VOLUME and UNIT operands on the DD statement or in a DDDEF entry.
3. Run the REJECT command again for the associated SYSMOD.

GIM37702I SMPTLIB *smptlib* WAS NOT DELETED FROM VOLUME *volser* BECAUSE THE DATA SET WAS NOT FOUND.

Explanation:

smptlib	data set name
volser	volume serial number

During REJECT processing, the indicated SMPTLIB data set could not be deleted because SMP/E could not find the data set through the catalog or on any volume specified on the SMPTLIB DD statement or DDDEF entry.

System Action: The SYSMOD is rejected.

Programmer Response: Delete the SMPTLIB data sets outside of SMP/E.

GIM37703W SMPTLIB *smptlib* WAS NOT DELETED FROM VOLUME *volser* BECAUSE ERROR CODE *errcode* WAS ISSUED BY DYNAMIC ALLOCATION.

Explanation:

smptlib	data set name
volser	volume serial number
errcode	error code

Dynamic deallocation failed for the indicated SMPTLIB data set.

Error codes from dynamic allocation fall into the following ranges:

Codes	Meaning
0000–00FC	Information reason codes
0021–0029	CATALOG or UNCATALOG unsuccessful
0031–0038	Delete unsuccessful
0100–01FC	I/O error
0200–02FC	System resources not available
0300–03FC	Invalid parameter list
0400–04FC	Environment error
1704–17FF	LOCATE error
4704–4780	DADSM error
47A8–47AC	RACF error
47B0	DADSM error
5704–571C	CATALOG error
6704–6714	OBTAIN error
7700	Subsystem error
7704–770C	SCRATCH error
8700–9728	System error

System Action: The SYSMOD was not rejected.

Programmer Response: See *z/OS MVS Programming: Authorized Assembler Services Guide*, SA22-7608, for more information about the error code. Message GIM54701W accompanies this message and contains information about the allocation failure.

GIM37704S SMPTLIB *smptlib* WAS NOT DELETED BECAUSE NO VOLUMES WERE SPECIFIED.

Explanation:

smptlib	data set name

During REJECT processing, the indicated SMPTLIB data set could not be deleted because no volumes were specified on the SMPTLIB DD statement or DDDEF entry.

System Action: The SYSMOD is rejected.

Programmer Response: Delete the SMPTLIB data sets outside of SMP/E.

GIM37705W SMPTLIB *smptlib* WAS NOT DELETED BECAUSE ERROR CODE *errcode* WAS ISSUED BY DYNAMIC ALLOCATION.

Explanation:

smptlib	data set name
errcode	error code

Dynamic deallocation failed for the indicated SMPTLIB data set because a resource was not available.

System Action: The SYSMOD was not rejected.

Programmer Response: See *z/OS MVS Programming: Authorized Assembler Services Guide*, SA22-7608, for more information about the error code. Message GIM54701W accompanies this message and contains information about the allocation failure.

GIM37801E ++*mcsname1* AND ++*mcsname2* FOR ELEMENT *elmname* ARE NOT ALLOWED IN THE SAME SYSMOD.

Explanation:

mcsname1	first element MCS
mcsname2	second element MCS
elmname	element name

The indicated combination of MCS statements is not allowed for the same element name in the same SYSMOD. The following table shows the combinations that are allowed.

MCS	MOD	ZAP	SRC	SRCUPD	MAC	MACUPD	Hierarchical File System Element	Data Element
MOD			Yes	Yes	Yes	Yes	Yes	Yes
ZAP					Yes	Yes	Yes	Yes
SRC	Yes				Yes	Yes	Yes	Yes

MCS	MOD	ZAP	SRC	SRCUPD	MAC	MACUPD	Hierarchical File System Element	Data Element
SRCUPD	Yes				Yes	Yes	Yes	Yes
MAC	Yes	Yes	Yes	Yes			Yes	Yes
MACUPD	Yes	Yes	Yes	Yes			Yes	Yes
Hierarchical File System Element	Yes	Yes	Yes	Yes	Yes	Yes		Yes
Data Element	Yes	Yes	Yes	Yes	Yes	Yes	Yes	See note
Note: MCS statements for data elements can specify the same element name only if the MCS statements are for different types of data elements. For example, ++CLIST and ++SKL can specify the same element name; so can ++MSGENU and ++MSGFRA. However, two ++CLIST statements or two ++MSG statements must specify different element names.								

System Action: The SYSMOD is not received.

Programmer Response: Fix the problem and rerun RECEIVE.

GIM37901E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE IT WAS PREVIOUSLY SUPERSEDED.**

Explanation:
command APPLY or ACCEPT
sysmod SYSMOD ID

SMP/E cannot apply or accept the indicated SYSMOD because it has already applied or accepted one or more SYSMODs that supersede SYSMOD *sysmod*. The system is at a higher level than the indicated SYSMOD, so SMP/E did not install it.

System Action: SYSMOD processing stops.

Programmer Response: None.

GIM37902E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE IT WAS PREVIOUSLY DELETED.**

Explanation:
command APPLY or ACCEPT
sysmod SYSMOD ID

SMP/E cannot apply or accept the indicated SYSMOD because it has already applied or accepted a SYSMOD that deletes SYSMOD *sysmod*. The system is at a higher level than the indicated SYSMOD, so SMP/E did not reinstall it.

System Action: SYSMOD processing stops.

Programmer Response: Decide which SYSMOD you want on your system: the deleting SYSMOD or SYSMOD *sysmod*.

- If you want the deleting SYSMOD on your system, do not try to reinstall SYSMOD *sysmod*.
- If you want SYSMOD *sysmod* back on your system, you must remove the deleting SYSMOD from the system, delete the SYSMOD entry for SYSMOD *sysmod*, then reinstall SYSMOD *sysmod*.

GIM37903E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE IT HAS NO APPLICABLE ++VER MCS.**

Explanation:
command APPLY or ACCEPT
sysmod SYSMOD ID

SMP/E cannot apply or accept the indicated SYSMOD because its ++VER MCS does not specify an FMID-SREL pair that is defined in the target or distribution zone being processed.

System Action: SYSMOD processing stops.

Programmer Response: Make sure that the SREL on the ++VER MCS matches the SREL in the zone definition entry, and that the FMID on the ++VER MCS matches a function SYSMOD entry in the zone being processed. These are some of the actions you may need to take:

- Make sure the SET command specifies the correct zone.
- If the FMID was installed in the zone being processed, check whether the ERROR indicator is on.
- Install the FMID in the zone being processed.
- Add the SREL to the zone definition entry.
- Reject the SYSMOD, change it to specify the correct FMID and SREL, and receive it again.

Then rerun the job.

GIM37904E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE OF AN ERROR DURING A PREVIOUS ATTEMPT TO RESTORE** *sysmod***.**

Explanation:
command APPLY or ACCEPT
sysmod SYSMOD ID

SMP/E cannot apply or accept the indicated SYSMOD because an error occurred during a previous attempt to restore that SYSMOD. It has been partially, but not totally, restored.

System Action: SYSMOD processing stops.

Programmer Response: Restore the SYSMOD again, then rerun the APPLY job.

GIM37905E **ACCEPT PROCESSING FAILED FOR SYSMOD** *sysmod1***. SYSMOD** *sysmod1* **IS SUPERSEDED BY SYSMOD** *sysmod2* **IN TARGET ZONE** *zonename***, AND THEREFORE MUST BE SUPERSEDED DURING ACCEPT PROCESSING. ALL SYSMODS WHICH WOULD HAVE SUPERSEDED** *sysmod1* **HAVE EITHER FAILED OR ARE MISSING.**

Explanation:

sysmod1	superseded SYSMOD ID
sysmod2	superseding SYSMOD ID
zonename	target zone containing superseded SYSMOD

The indicated SYSMOD was selected for processing. SMP/E cannot accept the indicated SYSMOD because it was never actually applied. It exists in the target zone as a superseded-only SYSMOD. If this SYSMOD were accepted without a superseding SYSMOD, succeeding SYSMODs could not be restored.

The superseding SYSMOD listed in the message is only the last SYSMOD to supersede the failing SYSMOD. There may be other superseding SYSMODs.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Accept at least one of the SYSMODs that supersede the failing SYSMOD.
- Specify BYPASS(APPLYCHECK) on the ACCEPT command. Note that if you specify BYPASS(APPLYCHECK), the superseded SYSMOD will be accepted, but you will be unable to restore back to it.

GIM38001E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE THE SYSMOD ENTRY FOR** *sysmod* **IN THE** *zonename* **ZONE WAS IN ERROR.**

Explanation:

command	APPLY or ACCEPT
sysmod	SYSMOD ID
zonename	zone name

SMP/E cannot apply or accept the indicated SYSMOD because the associated target or global zone SYSMOD entry is in error.

System Action: SYSMOD processing stops.

Programmer Response: Receive or apply the SYSMOD again to replace the SYSMOD entry that is in error. Then rerun the APPLY or ACCEPT job.

GIM38101E *command* **PROCESSING STOPPED. A ++***mcsname* **MCS IN SYSMOD** *sysmod* **APPLIES TO ELEMENT** *elmname***, WHICH WAS DELETED BY ANOTHER SYSMOD THAT IS BEING PROCESSED.**

Explanation:

command	APPLY or ACCEPT
mcsname	element update MCS
sysmod	SYSMOD ID
elmname	element name

The indicated MCS refers to an element that no longer exists.

System Action: SYSMOD processing stops.

Programmer Response: Reinstall the deleted element and rerun the job.

GIM38201W **THERE IS A MODID ERROR FOR** *elmtype* **ENTRY** *elmname* **IN SYSMOD** *sysmod***.**

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID

SMP/E discovered an error when it was checking the relationship between the following:

- An element in the indicated SYSMOD
- Elements in other SYSMODs that have already been installed or that are being installed concurrently.

Additional messages describe the specific error that was found (GIM319*xx* for APPLY or ACCEPT, or GIM430*xx* for RESTORE).

The severity of this message depends on the BYPASS options specified on the APPLY or ACCEPT command.

System Action: SYSMOD processing continues. At least **one** of the following conditions was met:

- The SYSMOD supplies an update element, and the PRE and SUP operands (1) specify the RMID of the previously processed version of the element but (2) do not specify all UMIDs of the previously processed version of the element.

  In this case, message GIM319*xx* or message GIM430*xx* follows this message and names the updates in the previously processed version of the element that the SYSMOD does not supersede or specify as a prerequisite. SMP/E processes the update supplied by the SYSMOD.
- **BYPASS(ID)** was not specified, and the SYSMOD is a function SYSMOD that was previously installed. SMP/E also does not stop processing if it can determine that the system is at a higher level than the SYSMOD it is reinstalling.

The system is considered to be at a higher level if **one** of the following is true:

– The RMID of the element from the SYSMOD differs from the RMID of the element on the system, and the RMID of the element from the SYSMOD is on the target system.

  Note: In this case, SMP/E does not issue GIM382*xx*. SMP/E does not process the element from the SYSMOD it is reinstalling, and the higher-level version of the element stays on the system.

– The RMID of the element from the SYSMOD is the same as the RMID of the element on the system, and UMIDs are associated with the system element.

  Note: SMP/E issues message GIM382*xx*, followed by GIM319*xx* or GIM430*xx*. These messages name the updates to the system element that the SYSMOD does not supersede or specify as prerequisites. SMP/E does not process the element from the SYSMOD that it is reinstalling, and the higher-level element stays on the target system.

If **BYPASS(ID)** was specified, SYSMOD processing continues for any ID checks reported by message GIM382*xx*. SMP/E selects the indicated element, and SYSMOD processing continues.

Programmer Response: Carefully check the messages that follow this message in SMPOUT. If SMP/E already installed the indicated element, it may regress IBM-supplied service or user-supplied modifications included in the SYSMODs named in subsequent GIM319*xx* or GIM430*xx* messages.

Specify **BYPASS(ID)** on the APPLY or ACCEPT command to prevent SMP/E from stopping SYSMOD processing. However, you may lose the modifications included in the SYSMODs named in subsequent GIM319*xx* or GIM430*xx* messages. Rework and reinstall these modifications.

GIM38201E THERE IS A MODID ERROR FOR *elmtype* **ENTRY** *elmname* **IN SYSMOD** *sysmod***.**

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID

SMP/E discovered an error when it was checking the relationship between the following:

- An element in the indicated SYSMOD
- Elements in other SYSMODs that have already been installed or that are being installed concurrently.

Additional messages describe the specific error that was found (GIM319*xx* for APPLY or ACCEPT, or GIM430*xx* for RESTORE).

The severity of this message depends on the BYPASS options specified on the APPLY or ACCEPT command.

System Action: SYSMOD processing stops. **All** of the following conditions were met:

- BYPASS(ID) was not specified.
- The SYSMOD was not previously installed.
- The SYSMOD supplied at least one of the following:
  – A replacement element: This occurs if the SYSMOD's PRE and SUP operands do not specify the RMID and all UMIDs of the previously processed version of the element.
  – An update element: This occurs if the SYSMOD's PRE and SUP operands do not specify the RMID of the previously processed version of the element.

Programmer Response: Carefully check the messages that follow this message in SMPOUT. If SMP/E already installed the indicated element, it may regress IBM-supplied service or user-supplied modifications included in the SYSMODs named in subsequent GIM319*xx* or GIM430*xx* messages.

Specify **BYPASS(ID)** on the APPLY or ACCEPT command to prevent SMP/E from stopping SYSMOD processing. However, you may lose the modifications included in the SYSMODs named in subsequent GIM319*xx* or GIM430*xx* messages. Rework and reinstall these modifications.

GIM38202W THERE IS A MODID ERROR FOR ASSEMBLY *asmname* **FOR** *elmtype* **ENTRY** *elmname* **IN SYSMOD** *sysmod***.**

Explanation:

asmname	assembly name
elmtype	element type
elmname	element name
sysmod	ID of the SYSMOD that contains the assembly

SMP/E discovered an error when it was checking the relationship between the following:

- An assembly for source or a macro
- The installed version of the element that will be replaced by the object module from the assembly.

Additional messages describe the specific error that was found (GIM319*xx* for APPLY or ACCEPT).

The severity of this message depends on the BYPASS options specified on the APPLY or ACCEPT command.

System Action: SYSMOD processing continues. **One** of the following conditions was met:

- The SYSMOD supplies an update element, and the PRE and SUP operands (1) specify the RMID of the previously processed version of the element but (2) do not specify all UMIDs of the previously processed version of the element.

  In this case, message GIM319*xx* follows this message and names the updates in the previously processed version of the element that the SYSMOD does not supersede or specify as a prerequisite. SMP/E processes the update supplied by the SYSMOD.
- **BYPASS(ID)** was not specified, and the SYSMOD is a function SYSMOD that was previously installed. SMP/E also does not stop processing if it can determine that the system is at a higher level than the SYSMOD it is reinstalling.

  The system is considered to be at a higher level if **one** of the following is true:
  - The RMID of the element from the SYSMOD differs from the RMID of the element on the system, and the RMID of the element from the SYSMOD is on the target system.

    Note: In this case, SMP/E does not issue GIM382*xx*. SMP/E does not process the element from the SYSMOD it is reinstalling, and the higher-level version of the element stays on the system.
  - The RMID of the element from the SYSMOD is the same as the RMID of the element on the system, and UMIDs are associated with the system element.

    Note: SMP/E issues message GIM382*xx*, followed by GIM319*xx*. These messages name the updates to the system element that the SYSMOD does not supersede or specify as prerequisites. SMP/E does not process the element from the SYSMOD that it is reinstalling, and the higher-level element stays on the target system.

If **BYPASS(ID)** was specified, SYSMOD processing continues for any ID checks reported by message GIM382*xx*. SMP/E selects the indicated element, and SYSMOD processing continues.

Programmer Response: Carefully check the messages that follow this message in SMPOUT. If SMP/E already installed the indicated element, it may regress IBM-supplied service or user-supplied modifications included in the SYSMODs named in subsequent GIM319*xx* messages.

Specify **BYPASS(ID)** on the APPLY or ACCEPT command to prevent SMP/E from stopping SYSMOD processing. However, you may lose the modifications included in the SYSMODs named in subsequent GIM319*xx* messages. Rework and reinstall these modifications.

GIM38202E THERE IS A MODID ERROR FOR ASSEMBLY *assembly* **FOR** *elmtype* **ENTRY** *elmname* **IN SYSMOD** *sysmod***.**

Explanation:

assembly	assembly name
elmtype	element type
elmname	element name
sysmod	ID of the SYSMOD that contains the assembly

SMP/E discovered an error when it was checking the relationship between the following:

- An assembly for source or a macro
- The installed version of the element that will be replaced by the object module from the assembly.

Additional messages describe the specific error that was found (GIM319*xx* for APPLY or ACCEPT).

The severity of this message depends on the BYPASS options specified on the APPLY or ACCEPT command.

System Action: SYSMOD processing stops. **All** of the following conditions were met:

- **BYPASS(ID)** was not specified.
- The SYSMOD was not previously installed.
- The SYSMOD supplies any of the following:
  - A replacement element: This occurs if the SYSMOD's PRE and SUP operands do not specify the RMID and all UMIDs of the previously processed version of the element.
  - An update element: This occurs if the SYSMOD's PRE and SUP operands do not specify the RMID of the previously processed version of the element.

Programmer Response: Carefully check the messages that follow this message in SMPOUT. If SMP/E already installed the indicated element, it may regress IBM-supplied service or user-supplied modifications included in the SYSMODs named in subsequent GIM319*xx* messages.

Specify **BYPASS(ID)** on the APPLY or ACCEPT command to prevent SMP/E from stopping SYSMOD processing. However, you may lose the modifications included in the SYSMODs named in subsequent GIM319*xx* messages. Rework and reinstall these modifications.

GIM38300S AN I/O ERROR OCCURRED DURING GIMXTRX *command* **PROCESSING FOR THE** *dataset* **DATA SET.**

Explanation:

command	READ or WRITE
dataset	file name

This message is issued when the GIMXTRX program was processing the indicated data set when it encountered an I/O error.

System Action: GIMXTRX processing ends.

Programmer Response: Contact your IBM service representative.

GIM38401E *command* **PROCESSING FAILED FOR** *sysmod1* **AND** *sysmod2*. **SYSMOD** *sysmod1* **DELETES OTHER SYSMODS, BUT IT IS SUPERSEDED BY SYSMOD** *sysmod2*. **ONLY ONE OF THESE SYSMODS CAN BE PROCESSED AT A TIME.**

Explanation:

command	an SMP/E command
sysmod1	ID of the deleting SYSMOD
sysmod2	ID of the superseding SYSMOD

SMP/E is concurrently processing two function SYSMODs. SYSMOD *sysmod1* deletes other SYSMODs, but it is superseded by SYSMOD *sysmod2*. SMP/E cannot determine which function to process.

System Action: Message GIM370*xx* follows this message, and command processing stops.

Programmer Response: Decide which SYSMOD should not be processed. Delete it from the command and rerun the job.

GIM38501I **SYSMOD** *sysmod* **WAS NOT RECEIVED BECAUSE THE RETURN CODE FROM THE EXIT ROUTINE IS 8 OR HIGHER.**

Explanation:

sysmod	SYSMOD ID

During RECEIVE processing, the RECEIVE installation exit routine was given control. After processing the SYSMOD, it set a return code of 8 or higher.

System Action: SMP/E does not receive the SYSMOD.

Programmer Response: None.

GIM38601I **SYSMOD** *sysmod* **WAS NOT RECEIVED BECAUSE IT HAS ALREADY BEEN RECEIVED.**

Explanation:

sysmod	SYSMOD ID

SMP/E has already successfully received SYSMOD *sysmod* into the SMPPTS data set.

System Action: SMP/E does not receive the SYSMOD again.

Programmer Response: To receive the new version of the SYSMOD, do **one** of the following:

- Reject and re-receive the SYSMOD.
  1. Use the REJECT command to remove the SYSMOD from the SMPPTS.
  2. Rerun the RECEIVE command to process the new version of the SYSMOD.
- Rework and re-receive the SYSMOD.
  1. Add the REWORK operand to the version of the SYSMOD in the SMPPTFIN data set.
  2. Rerun the RECEIVE command to process the new version of the SYSMOD.

GIM38601E **SYSMOD** *sysmod* **WAS NOT RECEIVED BECAUSE IT HAS ALREADY BEEN RECEIVED.**

Explanation:

sysmod	SYSMOD ID

SMP/E has already successfully received SYSMOD *sysmod* into the SMPPTS data set.

System Action: SMP/E does not receive the SYSMOD again.

Programmer Response: To receive the new version of the SYSMOD, do **one** of the following:

- Reject and re-receive the SYSMOD.
  1. Use the REJECT command to remove the SYSMOD from the SMPPTS.
  2. Rerun the RECEIVE command to process the new version of the SYSMOD.
- Rework and re-receive the SYSMOD.
  1. Add the REWORK operand to the version of the SYSMOD in the SMPPTFIN data set.
  2. Rerun the RECEIVE command to process the new version of the SYSMOD.

GIM38701I **SYSMOD** *sysmod* **WAS NOT RECEIVED BECAUSE IT WAS NOT SPECIFIED ON THE SELECT OPERAND.**

Explanation:

sysmod	SYSMOD ID

SMP/E found the indicated SYSMOD in the SMPPTFIN data set. Because the SELECT operand was specified, only SYSMODs included on that operand could be received. However, the indicated SYSMOD was not included on the SELECT operand.

System Action: SMP/E does not receive the SYSMOD.

Programmer Response: None.

GIM38801I SYSMOD *sysmod* WAS NOT RECEIVED BECAUSE IT WAS SPECIFIED ON THE EXCLUDE OPERAND.

Explanation:
sysmod SYSMOD ID

SMP/E found the indicated SYSMOD in the SMPPTFIN data set. However, the indicated SYSMOD was specified on the EXCLUDE operand, so it was not processed.

System Action: SMP/E does not receive the SYSMOD.

Programmer Response: None.

GIM38901I *command* PROCESSING FAILED FOR SYSMOD *sysmod* BECAUSE IT HAS NO APPLICABLE ++VER MCS.

Explanation:
command an SMP/E command
sysmod SYSMOD ID

The indicated SYSMOD could not be processed because it does not have a ++VER MCS that names an SREL or an FMID defined in the GLOBALZONE entry.

System Action: SMP/E does not receive the SYSMOD.

Programmer Response: If the SYSMOD is a PTF, APAR, or USERMOD, make sure the SYSMOD has at least one ++VER MCS that specifies an FMID. Then do **one** of the following:
- Specify **BYPASS(FMID)** on the RECEIVE command.
- Use a UCL statement to add the required SREL or FMID to the GLOBALZONE entry.

Then rerun the job.

GIM39001E SYSMOD *sysmod* WAS NOT RECEIVED EVEN THOUGH IT WAS SPECIFIED ON THE SELECT OPERAND.

Explanation:
sysmod SYSMOD ID

The indicated SYSMOD was specified on the SELECT operand, but it was not successfully received.

System Action: RECEIVE processing continues.

Programmer Response: See preceding messages in SMPOUT to find why SMP/E did not receive the indicated SYSMOD.

GIM39101E SYSMOD *sysmod* WAS NOT RECEIVED BECAUSE AN ERROR OCCURRED DURING SMPTLIB DATA SET PROCESSING.

Explanation:
sysmod SYSMOD ID

An error occurred when the SMPTLIB data sets were being loaded for the indicated SYSMOD. These are some of the errors that might have occurred:

- SMP/E could not position the SMPPTFIN data set because of one of the following:
  - An I/O error.
  - An incorrect data set name for one of the relative files.

    SMP/E expects the last two parts of the data set name for a relative file to be *sysmod-id*.F*nnnn*. If the data set name contains high-level qualifiers, SMP/E expects the data set name to be *rfprefix.rfdsnpfx.sysmod-id*.F*nnnn*.

    rfprefix
    is the prefix specified by the RFPREFIX operand on the RECEIVE command. This part of the data set name is optional. (If no RFPREFIX value was specified, this prefix is not included in the data set name allocated by SMP/E.)

    rfdsnpfx
    is the prefix specified by the RFDSNPFX operand on the header MCS statement that is being processed. This part of the data set name is optional. (If no RFDSNPFX value was specified, this prefix is not included in the data set name allocated by SMP/E.)

    sysmod-id
    is the SYSMOD ID of the SYSMOD to which that file belongs. This part of the data set name is required.

    F*nnnn*
    is the letter **F** followed by the number specified on the RELFILE operand of the corresponding MCS statement in the SYSMOD. This part of the data set name is required.

    This is similar to the name of the SMPTLIB data set into which the relative file will be received. The data set name for an SMPTLIB data set must be in the form *dsprefix.sysmod-id*.F*nnnn*, where *dsprefix* is an optional data set prefix.
- SMP/E could not allocate an SMPTLIB data set.
- SMP/E could not allocate a FROMDS data set.
- The return code from the copy utility was not 0. (See the copy SYSPRINT output to find the cause.)

Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:
  - The PRINT subentry specifies a DDDEF of DUMMY.
  - The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

- The return code received from allocation or unallocation of the RELFILE data set on DASD was not zero.

Additional messages are issued if allocation failed for the SMPTLIB data sets or if the return code from the copy utility was not 0.

System Action: SMP/E does not receive the SYSMOD, and RECEIVE processing stops. SMP/E deletes the SMPTLIB data sets and deletes the SYSMOD from the SMPPTS.

Programmer Response: If additional error messages were issued for SMPTLIB allocation or for a nonzero return code from the copy utility, fix the errors reported by those messages.

Make sure that the data set names for the SMPTLIB data sets and the relative files are correct. If the data set names of the relative files are in the wrong format, rebuild them using the correct data set names.

Then receive the SYSMOD again.

GIM39200S *modtype* **MODULE** *modname* **COULD NOT BE LOADED. SMP/E SERVICE UPDATE FACILITY PROCESSING IS STOPPED. ABEND CODE:** *code* **REASON CODE:** *rc*

Explanation:

modtype	API, CONSTANT, or MESSAGE
modname	name of the SMP/E load module
code	system abend code
rc	system reason code

The SMP/E API, message or constant load module could not be loaded by the Service Update Facility. Possible reasons include:

- The indicated load module was not found in the link list library.

System Action: Command processing stops.

Programmer Response: Check that the library containing the indicated load module was correctly specified in the system link list and that the indicated library does contain the requested load module.

See *z/OS MVS System Codes*, SA22-7626, for an explanation of the system abend and reason codes.

GIM39201S *modtype* **MODULE** *modname* **COULD NOT BE LOADED. SMP/E COMPACTION SERVICE PROCESSING IS STOPPED. ABEND CODE:** *code* **REASON CODE:** *rc*.

Explanation:

modtype	CONSTANT, DICTIONARY, or MESSAGE
modname	name of the SMP/E load module
code	system abend code
rc	system reason code

The message, message constant, or dictionary load module could not be loaded, possibly because the indicated load module was not found in the link list library.

System Action: Command processing stops.

Programmer Response: Ensure the library containing the indicated load module was correctly specified in the system link list and that indicated library does contain the requested load module.

See *z/OS MVS System Codes*, SA22-7626, for an explanation of the system abend and reason codes.

GIM39202S *modtype* **MODULE** *modname* **COULD NOT BE LOADED.** *progname* **PROCESSING IS STOPPED. ABEND CODE:** *code* **REASON CODE:** *rc*.

Explanation:

modtype	CONSTANT or MESSAGE
modname	name of the SMP/E load module
progname	name of the executing program
code	system abend code
rc	system reason code

The message or message constant load module could not be loaded, possibly because the indicated load module was not found in the link list.

System Action: Processing stops for the indicated program.

Programmer Response: Ensure the library containing the indicated load module was correctly spelled in the system link list and that indicated library does contain the requested load module.

See *z/OS MVS System Codes*, SA22-7626, for an explanation of the system abend and reason codes.

GIM39203S **MODULE** *loadmod* **COULD NOT BE LOADED.** *command* **PROCESSING IS STOPPED. ABEND CODE** *code* **REASON CODE** *rc*.

Explanation:

loadmod	name of the load module
command	an SMP/E command

code system abend code
rc system reason code

The load module could not be loaded, possibly because the indicated load module was not found in the link list library.

System Action: Command processing stops.

Programmer Response: Ensure the library containing the indicated load module was correctly specified in the system link list and that indicated library does contain the requested load module.

See *z/OS MVS System Codes*, SA22-7626, for an explanation of the system abend and reason codes.

GIM39301I SHELL SCRIPT *shellscr* **PROCESSING FOR** *elmtype elmname* **IN THE** *library* **LIBRARY WAS SUCCESSFUL. SEQUENCE NUMBER** *seqnum***.**

Explanation:
shellscr shell script name
elmtype element type
elmname element name
library library ddname
seqnum utility sequence number

Shell script processing was successful for the indicated element. The utility sequence number matches the sequence number of the shell script output in the PRINT data set.

Note: SYSPRINT is SMP/E's default print data set, and is used if no PRINT subentry was specified in the active UTILITY entry for the HFSCOPY utility.

System Action: None.

Programmer Response: None.

GIM39302E SHELL SCRIPT *shellscr* **PROCESSING FOR** *elmtype elmname* **IN THE** *library* **LIBRARY FAILED FOR SYSMOD** *sysmod***. SEQUENCE NUMBER** *seqnum***.**

Explanation:
shellscr shell script name
elmtype element type
elmname element name
library library ddname
sysmod SYSMOD ID
seqnum utility sequence number

Shell script processing failed for the indicated element. The utility sequence number matches the sequence number of the shell script output in the PRINT data set.

Note: SYSPRINT is SMP/E's default print data set, and is used if no PRINT subentry was specified in the active UTILITY entry for the HFSCOPY utility.

System Action: SYSMOD processing stops.

Programmer Response: Check the shell script output in the PRINT data set for the cause of the error. You can use the utility sequence number as an index into data set to find the output.

If an ABEND 806 occurred during job execution, or there is no shell script output, then ensure the SCEERUN library is either in the link list or in the job's STEPLIB or JOBLIB.

Fix the error and rerun the job.

GIM39303E THE PRE VALUE IS NOT ALLOWED ON THE SHSCRIPT OPERAND OF THE ++SHELLSCR MCS FOR *elmname***.**

Explanation:
elmname element name

The SHSCRIPT operand specifies a PRE value, which is not allowed. Shell scripts for SHELLSCR elements can run only after these elements are copied to the hierarchical file system.

System Action: RECEIVE processing stops for the SYSMOD.

Programmer Response: Remove the PRE specification from the SHSCRIPT operand and rerun the job.

GIM39304E THE SHELL SCRIPT NAME ON THE SHSCRIPT OPERAND, *shellscr***, DOES NOT MATCH THE ++SHELLSCR NAME** *elmname***. THIS IS NOT ALLOWED.**

Explanation:
shellscr shell script name
elmname element name

The SHSCRIPT operand specifies a shell script name that does not match the name on the corresponding element MCS. This is not allowed. SHELLSCR elements must specify their own name as the shell script to be run when the element is copied to the hierarchical file system.

System Action: RECEIVE processing stops for the SYSMOD.

Programmer Response: Correct the shell script name on the SHSCRIPT operand and rerun the job.

GIM39305E THE SHELL SCRIPT NAME FOR THE SHSCRIPT SUBENTRY MUST MATCH THE NAME OF THE SHELLSCR ENTRY.

Explanation: The SHSCRIPT subentry specifies a shell script name that does not match the name of the element entry. This is not allowed. SHELLSCR elements must specify their own name as the shell script to be

run when the element is copied to the hierarchical file system.

System Action: SMP/E does not process the changes for this UCL statement.

Programmer Response: Correct the shell script name for the SHSCRIPT subentry and rerun the job.

GIM39306E THE PRE VALUE IS NOT ALLOWED IN THE SHSCRIPT SUBENTRY FOR SHELLSCR ENTRIES.

Explanation: The SHSCRIPT subentry specifies the PRE value, which is not allowed. Shell scripts for SHELLSCR elements can run only after these elements are copied to the hierarchical file system.

System Action: SMP/E does not process the changes for this UCL statement.

Programmer Response: Remove the PRE specification from the SHSCRIPT subentry and rerun the job.

GIM39307I ALLOCATION FAILED FOR *library*, **WHICH IS REQUIRED IN ORDER TO INVOKE SHELL SCRIPT** *shellscr* **FOR** *elmtype elmname* **IN SYSMOD** *sysmod*.

Explanation:

library	ddname of the library that contains the shell script
shellscr	shell script name
elmtype	element type
elmname	element name
sysmod	SYSMOD ID

SMP/E could not allocate the library that contains shell script *shellscr*. As a result, SMP/E did not invoke the shell script to complete the installation of the indicated element.

System Action: Command processing for the SYSMOD stops.

Programmer Response: Examine the messages in SMPOUT to determine why the library could not be allocated. Fix the error and rerun the job.

GIM39308E SHELL SCRIPT *shellscr* **CANNOT PROCESS** *elmtype elmname* **IN SYSMOD** *sysmod* **BECAUSE SYSLIB** *library* **FOR SHELL SCRIPT** *shellscr* **IS NOT ALLOCATED TO A PATH.**

Explanation:

shellscr	shell script name
elmtype	element type
elmname	element name
sysmod	SYSMOD ID
library	ddname of the library that contains the shell script

Shell script *shellscr*, which is needed to complete the installation of the indicated element, could not be found in the hierarchical file system.

System Action: Command processing for the SYSMOD stops.

Programmer Response: Ensure that the DDDEF entry or DD statement for the indicated library identifies a path in the hierarchical file system. Rerun the job.

GIM39309E SHELL SCRIPT *shellscr* **PROCESSING IN THE** *elmtype* **LIBRARY FOR** *elmname sysmod* **IN SYSMOD** *library* **FAILED. THE RETURN CODE FROM THE** *service* **SERVICE WAS** *rc* **AND THE REASON CODE WAS** *reason*.

Explanation:

shellscr	shell script name
elmtype	element type
elmname	element name
sysmod	SYSMOD ID
library	ddname of the library that contains the shell script
service	callable service name
rc	return code, in hexadecimal
reason	reason code, in hexadecimal

Shell script processing failed for the indicated element. SMP/E called the indicated service, but the service was not successful.

System Action: SYSMOD processing stops.

Programmer Response: See *z/OS UNIX System Services Programming: Assembler Callable Services Reference* for the cause of the error. Correct the error and rerun the job step.

GIM39310I SHELL SCRIPT *shellscr* **PROCESSING TO DELETE** *elmtype elmname* **IN THE** *library* **LIBRARY WAS SUCCESSFUL. SEQUENCE NUMBER** *seqnum*.

Explanation:

shellscr	shell script name
elmtype	element type
elmname	element name
library	ddname of the library that contained the deleted element
seqnum	utility sequence number

Shell script processing to delete the indicated element was successful. The utility sequence number matches the sequence number of the shell script output in the PRINT data set.

Note: SYSPRINT is SMP/E's default print data set, and is used if no PRINT subentry was specified in the active UTILITY entry for the HFSCOPY utility.

System Action: None.

Programmer Response: None.

GIM39311E **SHELL SCRIPT** *shellscr* **PROCESSING TO DELETE** *elmtype elmname* **IN THE** *library* **LIBRARY FAILED FOR SYSMOD** *sysmod*. **SEQUENCE NUMBER** *seqnum*.

Explanation:

shellscr	shell script name
elmtype	element type
elmname	element name
library	ddname of the library that contains the element
sysmod	SYSMOD ID
seqnum	utility sequence number

Shell script processing to delete the indicated element has failed. The utility sequence number matches the sequence number of the shell script output in the PRINT data set.

Note: SYSPRINT is SMP/E's default print data set, and is used if no PRINT subentry was specified in the active UTILITY entry for the HFSCOPY utility.

System Action: SYSMOD processing stops.

Programmer Response: Check the shell script output in the PRINT data set for the cause of the error. You can use the utility sequence number as an index into SYSPRINT to find the output.

Fix the error and rerun the job.

GIM39312I **SHELL SCRIPT** *shellscr* **PROCESSING FOR** *elmtype elmname* **IN THE** *library* **LIBRARY FAILED FOR SYSMOD** *sysmod*.

Explanation:

shellscr	shell script name
elmtype	element type
elmname	element name
library	ddname of the library that contains the element
sysmod	SYSMOD ID

Shell script *shellscr* must be invoked to complete the installation of the indicated element, but an error was detected when SMP/E tried to invoke the shell script.

Note: SYSPRINT is SMP/E's default print data set, and is used if no PRINT subentry was specified in the active UTILITY entry for the HFSCOPY utility.

System Action: SYSMOD processing stops.

Programmer Response: Check previous SMP/E messages to determine the cause of the error. Fix the error and rerun the job.

GIM39313E *elmtype elmname* **CANNOT BE PROCESSED IN LIBRARY** *library* **FOR SYSMOD** *sysmod* **BECAUSE** *library* **IS NOT ALLOCATED TO A PATH.**

Explanation:

elmtype	element type
elmname	element name
library	library ddname
sysmod	SYSMOD ID

SMP/E attempted to process the indicated element, but the library is allocated to a data set. The indicated element type cannot reside in a data set. Therefore, the library must be allocated to a path in the hierarchical file system.

System Action: SYSMOD processing stops.

Programmer Response: Ensure that the DDDEF entry or DD statement for the specified library identifies a path in the hierarchical file system. Rerun the job.

GIM39317I **ALLOCATION FAILED FOR** *library*, **WHICH IS REQUIRED IN ORDER TO INVOKE SHELL SCRIPT** *shellscr* **TO DELETE** *elmtype elmname* **FOR SYSMOD** *sysmod*.

Explanation:

library	ddname of the library that contains the shell script
shellscr	shell script name
elmtype	element type
elmname	element name
sysmod	SYSMOD ID

SMP/E could not allocate the library that contains shell script *shellscr*. As a result, SMP/E did not invoke the shell script to complete the installation of the indicated element.

System Action: Command processing for the SYSMOD stops.

Programmer Response: Examine the messages in SMPOUT to determine why the library could not be allocated. Fix the error and rerun the job.

GIM39318E **SHELL SCRIPT** *shellscr* **CANNOT DELETE** *elmtype elmname* **FOR SYSMOD** *sysmod* **BECAUSE SYSLIB** *library* **FOR SHELL SCRIPT** *shellscr* **IS NOT ALLOCATED TO A PATH.**

Explanation:

shellscr	shell script name
elmtype	element type
elmname	element name
sysmod	SYSMOD ID
library	ddname of the library that contains the shell script

Shell script *shellscr* does not reside in the hierarchical file system. As a result, SMP/E cannot invoke the shell script to delete the indicated element.

System Action: Command processing for the SYSMOD stops.

Programmer Response: Ensure that the DDDEF entry or DD statement for the indicated library identifies a path in the hierarchical file system. Rerun the job.

GIM39401I SMPTLIB DATA SETS WERE LOADED FOR SYSMOD *sysmod*.

Explanation:
sysmod SYSMOD ID

The copy utility successfully loaded the SMPTLIB data sets for the indicated SYSMOD.

System Action: SYSMOD processing continues. SMP/E indicates that the indicated SYSMOD has been received successfully.

Programmer Response: None.

GIM39402I SMPTLIB DATA SETS WERE LOADED FOR SYSMOD *sysmod*. THE RETURN CODE FROM THE COPY UTILITY WAS *rtncode*.

Explanation:
sysmod SYSMOD ID
rtncode maximum return code from the copy utility

The copy utility successfully loaded the SMPTLIB data sets for the indicated SYSMOD. The copy utility issued a nonzero return code less than or equal to the acceptable return code that was specified in the associated UTILITY entry.

System Action: SYSMOD processing continues. SMP/E indicates that the indicated SYSMOD has been received successfully.

Programmer Response: Check the copy output in SYSPRINT to find the cause of the nonzero return code, and determine whether this would adversely affect further processing.

Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:

- The PRINT subentry specifies a DDDEF of DUMMY.
- The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

GIM39601E SYSMOD *sysmod* WAS NOT RECEIVED BECAUSE IT HAS NO ++VER MCS.

Explanation:
sysmod SYSMOD ID

SMP/E did not find a ++VER MCS for the indicated SYSMOD.

System Action: SMP/E does not receive the SYSMOD.

Programmer Response: To receive the SYSMOD, include an applicable ++VER MCS and run RECEIVE again.

GIM39701W SYSMOD *sysmod* HAS NO ELEMENTS.

Explanation:
sysmod SYSMOD ID

The indicated SYSMOD contains no element, JCLIN, MOVE, RENAME, or DELETE MCS statements.

System Action: Other messages indicate whether SMP/E received the SYSMOD.

Programmer Response: If the SYSMOD was successfully received, no action is required. Otherwise, fix the error and rerun the job.

GIM39701E SYSMOD *sysmod* HAS NO ELEMENTS.

Explanation:
sysmod SYSMOD ID

The indicated SYSMOD contains no element MCS statements. It does, however, have an applicable ++VER MCS and may have inline JCLIN.

System Action: Other messages indicate whether SMP/E received the SYSMOD.

Programmer Response: If the SYSMOD was successfully received, no action is required. Otherwise, fix the error and rerun the job.

GIM39801I SYSMOD *sysmod* WAS NOT RECEIVED BECAUSE IT HAS A SYNTAX OR CONSTRUCTION ERROR.

Explanation:
sysmod SYSMOD ID

The indicated SYSMOD contains a syntax error or a construction error. Other messages describe the error.

System Action: SMP/E does not receive the SYSMOD.

Programmer Response: Look at SMPOUT for other messages that contain more specific information about the syntax or construction error.

GIM399I ENTER THE JULIAN DATE (yyddd) OR "U".

Explanation: SMP/E is requesting the date it needs to record this SMP/E job.

System Action: None.

Programmer Response: Do **one** of the following:
- Enter the date as *yyddd* (*yy*=year, *ddd*=day).
- Enter **U** to use the system IPL date.

GIM40001E SYSMOD *sysmod* HAS ALREADY BEEN FOUND IN THE SMPPTFIN DATA SET THAT IS CURRENTLY BEING PROCESSED.

Explanation:

sysmod	SYSMOD ID

The indicated SYSMOD appeared more than once in the same SMPPTFIN data set. SMP/E may have successfully received this previous copy of the SYSMOD.

This can occur when you are processing PTFs for an FMID that is applicable to several licensed products. If you are licensed for those products, PTFs associated with that FMID are loaded on the expanded service options (ESO) tape for each product. Consequently, multiple copies of the same PTF may be found on that ESO. During RECEIVE processing, SMP/E finds the duplicated copies of the PTF and issues this message.

System Action: SMP/E does not receive the second version of the SYSMOD. The previous copy of the SYSMOD is not changed.

Programmer Response: Do **one** of the following:
- If you received an ESO with multiple versions of the same PTF, take no action.
- To keep the earlier copy of the SYSMOD, take no action.
- To keep this copy of the SYSMOD, do the following:
  1. Reject the SYSMOD.
  2. Correct the SMPPTFIN input stream so that only the SYSMOD you want appears.

GIM40002E ANOTHER ++*mcstype* MCS FOR *mcsname* HAS ALREADY BEEN FOUND IN THE SMPPTFIN DATA SET THAT IS CURRENTLY BEING PROCESSED.

Explanation:

mcstype	type of MCS (PRODUCT or FEATURE)
mcsname	name for the MCS (product-id and vrm, or feature name)

The indicated ++FEATURE or ++PRODUCT MCS appeared more than once in the same SMPPTFIN data set. SMP/E may have successfully received the first copy of the ++FEATURE or ++PRODUCT MCS. During RECEIVE processing, SMP/E finds the duplicated copies of the ++FEATURE or ++PRODUCT MCS and issues this message.

System Action: SMP/E does not receive the second instance of the ++FEATURE or ++PRODUCT MCS. The first instance is not affected.

Programmer Response: Do one of the following:
- To keep the first instance of the ++FEATURE or ++PRODUCT, take no action.
- To keep this copy of the ++FEATURE or ++PRODUCT, do the following:
  1. Reject the first ++FEATURE or ++PRODUCT, or delete it with UCLIN.
  2. Correct the SMPPTFIN input stream so that only the desired ++FEATURE or ++PRODUCT MCS appears.

GIM40101E *command* PROCESSING FAILED FOR SYSMOD *sysmod* EVEN THOUGH IT WAS SPECIFIED ON THE SELECT OPERAND. *sysmod* IS NOT IN THE *zonetype* ZONE.

Explanation:

command	an SMP/E command
sysmod	SYSMOD ID
zonetype	zone type

The indicated SYSMOD was specified on the SELECT operand but the SYSMOD is not available for SMP/E to process.

System Action: SYSMOD processing stops.

Programmer Response: Make sure you specified the correct SYSMOD ID on the command. Then rerun the job.

GIM40102E *command* PROCESSING FAILED FOR SYSMOD *sysmod*. *sysmod1* IS A CROSS-ZONE REQUISITE FROM ZONE *zonename* FOR FMID *sysmod2*, BUT *sysmod1* IS NOT IN THE GLOBAL ZONE.

Explanation:

command	APPLY or ACCEPT
sysmod1	SYSMOD ID
zonename	zone name
sysmod2	function SYSMOD ID

The XZREQ operand was used for the specified command. SYSMOD *sysmod1* was needed for function *sysmod2* in the set-to zone because of CIFREQ data in another zone. SYSMOD *sysmod1* was not received and therefore could not be processed by the indicated command.

System Action: Processing continues.

Programmer Response: If you want the specified SYSMOD installed, obtain it and receive it into the GLOBAL zone and the PTS data set. You may then install it by using the XZREQ operand or by specifically selecting it.

GIM40103E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod*. *sysmod1* **IS A CROSS-ZONE REQUISITE FROM ZONE** *zonename* **FOR FMID** *sysmod2* **BUT** *sysmod1* **WAS EXCLUDED.**

Explanation:

command	APPLY or ACCEPT
sysmod1	SYSMOD ID
zonename	zone name
sysmod2	function SYSMOD ID

The XZREQ operand was used for the specified command. SYSMOD *sysmod1* was needed for function *sysmod2* in the set-to zone because of CIFREQ data in another zone. SYSMOD *sysmod1*, however, was excluded and therefore could not be processed by the indicated command.

System Action: Processing continues.

Programmer Response: If you want the specified SYSMOD installed, remove it from the EXCLUDE operand and rerun the command.

GIM40104E **ACCEPT PROCESSING FAILED FOR SYSMOD** *sysmod1*. *sysmod1* **IS A CROSS-ZONE REQUISITE FROM ZONE** *zonename* **FOR FMID** *sysmod2* **BUT** *sysmod1* **COULD NOT BE PROCESSED BECAUSE BYPASS(APPLYCHECK) WAS NOT SPECIFIED.**

Explanation:

sysmod1	SYSMOD ID
zonename	zone name
sysmod2	Function SYSMOD ID

The XZREQ operand was used for the specified command. SYSMOD *sysmod1* was needed for function *sysmod2* in the set-to zone because of CIFREQ data in another zone. SYSMOD *sysmod1*, however, could not be accepted because it was not successfully applied.

System Action: Processing continues.

Programmer Response: If you want the specified SYSMOD installed, specify BYPASS(APPLYCHECK) on the command and rerun the command.

GIM40201E **THE** *operand* **OPERAND IS NOT ALLOWED FOR SUPERSEDED ONLY SYSMODS.**

Explanation:

operand	UCL statement operand

The SYSMOD entry to be updated by the UCL statement is for a superseded-only SYSMOD. (The entry indicates that the SYSMOD has not been installed, but a SYSMOD that supersedes it has been.) However, the indicated operand is not allowed for superseded-only SYSMODs.

System Action: SMP/E does not do the requested UCLIN processing.

Programmer Response: Do **one** of the following:

- Specify additional operands on the UCL statement so that the SYSMOD is no longer superseded-only. (For example, add a module, RECEIVE and APPLY dates, and a SYSMOD, if these are applicable.)
- Delete the operand from the UCL statement.

Then rerun the job.

GIM40202E **THE** *operand* **OPERAND IS NOT ALLOWED FOR DELETED SYSMODS**

Explanation:

operand	UCL statement operand

The indicated operand is not allowed on a UCL statement for a deleted SYSMOD. (The SYSMOD entry for a deleted SYSMOD contains the DELBY subentry.)

System Action: SMP/E does not do the requested UCLIN processing.

Programmer Response: Delete the indicated operand from the UCL statement and rerun the job.

GIM40203E **STATUS OPERANDS ARE NOT ALLOWED FOR DELETED SYSMODS.**

Explanation: Status operands (such as APP, ACC, or ERR) are not allowed on a UCL statement for a deleted SYSMOD. (The SYSMOD entry for a deleted SYSMOD contains the DELBY subentry.)

System Action: SMP/E does not do the requested UCLIN processing.

Programmer Response: Delete the indicated operand from the UCL statement and rerun the job.

GIM40204E **STATUS OPERANDS ARE NOT ALLOWED FOR SUPERSEDED ONLY SYSMODS.**

Explanation: The SYSMOD entry to be updated by the UCL statement is for a superseded-only SYSMOD. (The entry indicates that the SYSMOD has not been installed, but a SYSMOD that supersedes it has been.) However, status operands are not allowed for a superseded-only SYSMOD.

System Action: SMP/E does not do the requested UCLIN processing.

Programmer Response: Do **one** of the following:

- Specify additional operands on the UCL statement so that the SYSMOD is no longer superseded-only. (For example, add a module, RECEIVE and APPLY dates, and a SYSMOD, if these are applicable.)
- Delete the status operand from the UCL statement.

Then rerun the job.

GIM40300E THE *operand* **OPERAND IS MUTUALLY EXCLUSIVE WITH ALL OTHER SYSMOD SUBENTRIES.**

Explanation:

operand	UCL statement operand or SYSMOD subentry

A UCL statement for a SYSMOD entry specified the indicated operand or an existing SYSMOD entry already contains the indicated operand as a subentry. This operand cannot be specified with any other UCL operands. If this subentry already exists in the SYSMOD entry, no other subentries can be added to the SYSMOD entry.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Specify the correct operands on the UCL statement and re-run the job. Or delete the indicated subentry from the SYSMOD entry, correct the UCL statement, and re-run the job.

GIM40401W SMP/E DID NOT RESTORE *enttype entname* **FOR SYSMOD** *sysmod* **BECAUSE** *enttype entname* **IS NOT IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
sysmod	SYSMOD ID
library	ddname of the library (either SMPSCDS or the target zone name)

During RESTORE processing for SYSMOD *sysmod*, SMP/E tried to copy the indicated entry from the SMPSCDS to the target zone. However, SMP/E could not find the entry in the indicated data set.

System Action: **One** of the following occurs:

- If SMP/E could not find the entry on the target zone, it copies the BACKUP entry from the SMPSCDS to the target zone.
- If SMP/E could not find the BACKUP entry in the SMPSCDS, the indicated entry is not restored.

In both cases, SYSMOD processing continues.

Programmer Response: Do the following:

1. Look at the SMPLOG data set to see if the entry in the target zone was updated by UCLIN.
2. List the BACKUP entry to check that SMP/E copied the correct version of the entry into the SMPSCDS.
3. If the entry in the target zone is incorrect, use UCLIN to make the appropriate updates.

Then rerun the job.

GIM40501E THE DISTLIB VALUE *distlib1* **SPECIFIED FOR** *enttype entname* **IN SYSMOD** *sysmod* **DOES NOT MATCH THE DISTLIB VALUE** *distlib2* **IN THE** *enttype* **ENTRY FOR** *entname***.**

Explanation:

distlib1	DISTLIB name in MCS
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
distlib2	DISTLIB name in SYSMOD

The DISTLIB value specified on an element MCS in the indicated SYSMOD is different from the DISTLIB value in the element entry in the target or distribution zone being processed.

Here is an example of a situation that would cause this message to be issued:

```
In SYSMOD UZ12345:
   ++PTF(UZ12345).
   ...
   ++MOD(A) ... DISTLIB(DLIB2).

In the target zone MOD entry:
   MOD(A)
   ...
   DISTLIB(DLIB1)
```

System Action: APPLY or ACCEPT processing stops for the indicated SYSMOD.

Programmer Response: Correct the DISTLIB value on the element MCS or in the element entry. Then rerun the job.

GIM40600I SMP/E PROCESSING TO THE *library* **LIBRARY WAS SUCCESSFUL FOR** *enttype entname* **IN SYSMOD** *sysmod***. BLANK DATA HAS BEEN TRUNCATED. DATE** *yy.ddd* **- TIME** *hh:mm:ss*

Explanation:

library	ddname of the library to which the element was to be installed.
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

The indicated element was successfully processed. However, blank data has been truncated during the installation of the element. You should check the output to verify that the results are acceptable.

System Action: Processing continues.

Programmer Response: None.

GIM40700E **SMP/E PROCESSING TO THE** *library* **LIBRARY FAILED FOR** *enttype entname* **IN SYSMOD** *sysmod***. NON-BLANK DATA WOULD HAVE BEEN TRUNCATED. DATE** *yy.ddd* **– TIME** *hh:mm:ss*

Explanation:

library	ddname of the library to which the element was to be installed
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds

The indicated element was not processed because non-blank data would have been truncated during the installation of the element.

System Action: Processing stops for the SYSMOD that contains the element.

Programmer Response: Increase the record length of the output data set so that it matches the record length of the input data set. Then, rerun the job.

GIM40800I *command* **PROCESSING FAILED FOR SYSMOD** *sysmod***. SMP/E PROCESSING FAILED FOR AN ELEMENT IN** *sysmod***.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID

SMP/E did not process the indicated SYSMOD because one of the elements was not processed. SMP/E encountered an error while processing the element.

System Action: SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

Programmer Response:

- Examine SMPOUT for messages relating to elements of the SYSMOD
- Use the APPLY/RESTORE/ACCEPT Element Summary Report to find the associated SYSMOD that caused the failure.
- Include the superseded SYSMODs on the RESTORE command if you have already applied the indicated SYSMOD.

Fix the error and rerun the job.

GIM40901E **COPY PROCESSING TO THE** *library* **LIBRARY FAILED FOR** *elmtype elmname* **IN SYSMOD** *sysmod***. THE RETURN CODE (***rc***) EXCEEDED THE ALLOWABLE VALUE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqnum***.**

Explanation:

library	ddname of the library to which the element was to be copied
elmtype	element type
elmname	element name
sysmod	SYSMOD ID
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqnum	utility sequence number

Copy processing failed for the indicated element. Note that if the element name is greater than 300 characters, only the first 300 characters are placed in the message.

Multiple elements may be copied from one input data set to one output data set during a single invocation of the copy utility. If an error occurs during a batched copy invocation, the updates are debatched. If an error occurs during an invocation for a debatched member, processing fails for the associated SYSMOD (and possibly for related SYSMODs).

System Action: SYSMOD processing stops because the return code was not 0, and **one** of the following is true:

- The return code is greater than the maximum return code specified in the UTILITY entry.
- The return code is greater than the default return code.

Programmer Response: Check the copy output in SYSPRINT to find the cause of the error. The sequence number can be used as an index into the SYSPRINT output. This number is incremented for each utility call in an SMP/E run.

If copy output indicates that an unloaded PDS cannot be reloaded to a PDSE or that an unloaded PDSE cannot be reloaded to a PDS, you must define an SMPTLOAD DDDEF in the target or distribution zone. See the SMPTLOAD data set description in *SMP/E Reference* for information on how to do this.

Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:

- The PRINT subentry specifies a DDDEF of DUMMY.
- The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

GIM40902I **COPY PROCESSING TO THE** *library* **LIBRARY WAS SUCCESSFUL FOR** *elmtype elmname* **IN SYSMOD** *sysmod*. **THE RETURN CODE WAS** *rc*. **DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqnum*.

Explanation:

library	ddname of the library to which the element was to be copied
elmtype	element type
elmname	element name
sysmod	SYSMOD ID
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqnum	utility sequence number

The indicated element was successfully copied. Note that if the element name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: Processing continues.

Programmer Response: None.

GIM40902W **COPY PROCESSING TO THE** *library* **LIBRARY WAS SUCCESSFUL FOR** *elmtype elmname* **IN SYSMOD** *sysmod*. **THE RETURN CODE WAS** *rc*. **DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqnum*.

Explanation:

library	ddname of the library to which the element was to be copied
elmtype	element type
elmname	element name
sysmod	SYSMOD ID
rc	return code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqnum	utility sequence number

The indicated element was successfully copied. Note that if the element name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: Processing continues.

Programmer Response: No action is required. However, because the return code indicates that an error may have occurred, you should check the utility output to verify that the results are acceptable.

GIM40903E **COPY PROCESSING TO THE** *library* **LIBRARY FAILED FOR** *elmtype elmname* **IN SYSMOD** *sysmod*. **ABEND** *abncode* **OCCURRED WHILE PROCESSING** *library*. **THE** *library* **LIBRARY RAN OUT OF SPACE. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqnum*.

Explanation:

library	ddname of the library
elmtype	element type
elmname	element name
sysmod	SYSMOD ID
abncode	abend code
yy	year
ddd	Julian day
hh	military hour
mm	minutes
ss	seconds
seqnum	utility sequence number

Copy processing failed for the indicated element. Note that if the element name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: SYSMOD processing stops for the indicated SYSMOD.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM40904E **STOW PROCESSING TO THE** *library* **LIBRARY FAILED FOR** *alias* **(ALIAS of** *elmtype elmname***) IN SYSMOD** *sysmod*.

Explanation:

library	ddname of the library into which the alias was to be stowed
alias	alias name
elmtype	data element type
elmname	data element name
sysmod	name of sysmod supplying the data element

STOW processing failed for the indicated alias, leaving the installation of the data element incomplete.

System Action: SYSMOD processing stops.

Programmer Response: Verify that there are no I/O errors on the indicated library, check the size of the

directory for the indicated library, and try to reinstall the element.

GIM40905I COPY PROCESSING TO THE *library* LIBRARY WAS SUCCESSFUL FOR *elmtype elmname*.

Explanation:

library	library name
elmtype	data element type
elmname	data element name

The indicated data element was successfully copied.

System Action: Processing continues.

Programmer Response: None.

GIM40905W COPY PROCESSING TO THE *library* LIBRARY WAS SUCCESSFUL FOR *elmtype elmname*. THE RETURN CODE WAS *rc*.

Explanation:

library	library name
elmtype	data element type
elmname	data element name
rc	return code

The indicated data element was successfully copied.

System Action: Processing continues.

Programmer Response: No action is required. However, because the return code is non-zero, you should check the utility output to verify that the results are acceptable.

GIM40906E COPY PROCESSING TO THE *library* LIBRARY FAILED FOR *elmtype elmname*. THE RETURN CODE *rc* EXCEEDED THE ALLOWABLE VALUE OF *max*.

Explanation:

library	library name
elmtype	data element type
elmname	data element name
rc	return code
max	maximum allowable return code

Copy failed for the indicated data element.

System Action: Processing of the data element stops. The return code is not zero, and one of the following is true:

- The return code is greater than the maximum return code specified in the INVOKE control statement.
- The return code is greater than the default return code.

Programmer Response: Check the copy output to determine the cause of the error. Fix the error and rerun the job.

GIM41001E *command* PROCESSING FAILED FOR *sysmod1* BECAUSE BOTH *sysmod1* AND *sysmod2* CONTAIN ZAP UPDATES FOR MODULE *modname*. ONLY ONE OF THESE SYSMODS CAN BE PROCESSED AT A TIME.

Explanation:

command	APPLY or ACCEPT
sysmod1	first SYSMOD ID
sysmod2	second SYSMOD ID
modname	module name

SMP/E will not process more than one ZAP for the same module for a single APPLY or ACCEPT command.

System Action: SYSMOD processing stops.

Programmer Response: Install the SYSMODs with separate APPLY or ACCEPT commands.

GIM41101W SMP/E DID NOT RESTORE *enttype* ENTRY *entname* FOR SYSMOD *sysmod* BECAUSE *enttype entname* WAS CHANGED BY *changer*.

Explanation:

enttype	entry type
entname	entry name
sysmod	ID of the SYSMOD being restored
changer	JCLIN, UCLIN, or the ID of the SYSMOD that updated the element

When SMP/E was restoring SYSMOD *sysmod*, it tried to copy the indicated entry from the SYSMOD's BACKUP entries in the SMPSCDS data set to the target zone. However, the target zone entry indicated that changes were made to the element after the BACKUP entries were made for SYSMOD *sysmod*. Because these changes are not in the BACKUP entries, they would be lost if SYSMOD *sysmod* were restored.

System Action: Processing continues.

Programmer Response: Do the following:

1. List the specified target zone entry.
2. Check that the entry is valid, even after SMP/E restores the SYSMOD.
3. If the entry requires any changes, update it using UCLIN.

GIM41101E SMP/E DID NOT RESTORE *enttype* ENTRY *entname* FOR SYSMOD *sysmod* BECAUSE *enttype entname* WAS CHANGED BY *changer*.

Explanation:

enttype	entry type
entname	entry name

sysmod	ID of the SYSMOD being restored
changer	JCLIN, UCLIN, or the ID of the SYSMOD that updated the element

When SMP/E was restoring SYSMOD *sysmod*, it tried to copy the indicated entry from the SYSMOD's BACKUP entries in the SMPSCDS data set to the target zone. However, the target zone entry indicated that changes were made to the element after the BACKUP entries were made for SYSMOD *sysmod*. Because these changes are not in the BACKUP entries, they would be lost if SYSMOD *sysmod* were restored.

System Action: SYSMOD processing stops.

Programmer Response: Do the following:
1. List the specified target zone entry.
2. Check that the entry is valid, even after SMP/E restores the SYSMOD.
3. If the entry requires any changes, update it using UCLIN.

GIM41201S SMP/E CANNOT DETERMINE THE RELATIONSHIPS BETWEEN THE FOLLOWING SYSMODS:

Explanation: SMP/E could not determine the order in which to process the SYSMODs listed in message GIM41301I. SMP/E may have been processing the SYSMODs concurrently, or one of the SYSMODs may already have been installed. SMP/E determines the processing order using information on the ++VER MCS (FMID, VERSION, PRE, SUP).

System Action: Command processing stops.

Programmer Response: Do **one** of the following:
- Specify the correct relationship on the ++VER MCS for the specified SYSMODs.
- Use separate commands to process each SYSMOD in the required order.

GIM41301I SYSMOD=*sysmod* FMID=*fmid* PRE=*prereq*.

Explanation:

sysmod	SYSMOD ID
fmid	FMID associated with the SYSMOD
prereq	list of prerequisites for the SYSMOD

This message follows message GIM41201S and lists the SYSMODs for which SMP/E could not determine a processing order.

System Action: Command processing stops.

Programmer Response: See message GIM41201S.

GIM41400W THE EFFECTIVE UID COULD NOT BE CHANGED BACK TO *uidval*. THE RETURN CODE FROM THE *service* CALLABLE SERVICE WAS '*rtncode*'X AND THE REASON CODE WAS '*rsncode*'X.

Explanation:

uidval	original UID value before it was changed to 0
service	callable service module name (usually BPX1SEU)
rc	return code from callable service in hexadecimal
rsncode	reason code from callable service in hexadecimal

During SMP/E initialization, the effective UID was set to 0, so that any updates to the hierarchical file system would be done with a UID of 0. During command termination (normal or abnormal) SMP/E attempted to reset the effective UID to its original value, but the attempt failed.

System Action: Processing continues.

Programmer Response: Determine from the return code and reason code given by the callable service why the attempt to reset the effective UID failed. Correct the problem.

GIM41501E THE ++*mcsname* MCS FOR ELEMENT *elmname* IN SYSMOD *sysmod* WAS NOT PROCESSED BECAUSE *elmname* IS NOT IN THE *zonename* ZONE.

Explanation:

mcsname	element update MCS
elmname	element name
sysmod	SYSMOD ID
zonename	TARGET or DLIB

SMP/E cannot update the indicated element because there is no valid entry for that element in the zone being processed.

Note: This can happen if the SYSMOD that supplied the element failed. In this case, there may be an element entry that has no FMID and no RMID.

System Action: APPLY or ACCEPT processing stops for the SYSMOD.

Programmer Response: Do the following:
1. List the target zone or distribution zone to find out if there is an entry for the element.

   If you find an entry with no RMID, the entry represents an element that SMP/E does not consider to be in the target system.
2. Do **one** of the following:
   - Install a SYSMOD to supply the element.

- Use UCLIN to properly initialize the FMID and RMID subentries in the element entry.

3. Rerun the job.

GIM41601E DALIAS *dalias* **FOR MOD** *modname* **HAS BEEN FOUND AS A MOD ENTRY WITH THE SAME DISTLIB.**

Explanation:

dalias	DALIAS value on the ++MOD MCS
modname	element name on the ++MOD MCS

During APPLY or ACCEPT processing, SMP/E found an existing MOD entry in a target or DLIB zone with the same name as the DALIAS value specified on a ++MOD MCS in a SYSMOD being installed. In addition, the existing MOD entry specifies the same DISTLIB value as the ++MOD MCS being processed. However, an alias for a module cannot coexist in a distribution library that already has a MOD entry with the same name as the alias.

System Action: APPLY or ACCEPT processing stops for the SYSMOD, which is identified in the subsequent GIM22601I or GIM22601E message.

Programmer Response: Do **one** of the following:

- Remove the DALIAS operand or change the DALIAS value on the failing MCS.
- Change the DISTLIB value on the failing MCS.

  Note: Before you change the MCS, get the approval of the owner of the SYSMOD that contains the failing MCS. Otherwise, errors may occur when you try to install subsequent SYSMODs that apply to the affected element.
- If the data in the existing MOD entry is incorrect, use the UCLIN command to correct or delete the MOD entry.

GIM41700E SYSMOD *sysmod*, **SPECIFIED ON A JCLIN SMPE-IF CONTROL STATEMENT, IS IN ERROR. JCLIN PROCESSING HAS FAILED.**

Explanation:

sysmod	SYSMOD ID

During JCLIN processing, SMP/E found that the SYSMOD specified on a JCLIN **SMPE-IF** control statement was in ERROR or NOGO. Because the specified SYSMOD terminated before completing successfully, it is possible that JCLIN processing for the SYSMOD left structural information incomplete. Therefore, it would be dangerous to continue the current JCLIN processing.

System Action: JCLIN processing fails for the current JCLIN input stream.

Programmer Response: Finish the installation of SYSMOD *sysmod* and rerun the SMP/E command.

GIM41701E THE *statement* **JCLIN CONTROL STATEMENT IS NOT IN A PROPER PLACE IN THE JCLIN INPUT STREAM. JCLIN PROCESSING HAS FAILED.**

Explanation:

statement	**SMPE-ELSE** or **SMPE-END**

The indicated JCLIN control statement was not in the proper place in the JCLIN input stream. An **SMPE-ELSE** control statement must immediately follow the **SMPE-END** control statement that ends the **DO/END** group of the associated **SMPE-IF** control statement. An **SMPE-END** control statement must have an open **DO/END** group to close (the **DO/END** group was started on an **SMPE-IF** or **SMPE-ELSE** control statement).

System Action: JCLIN processing fails for the current JCLIN input stream.

Programmer Response: Check with the provider of the JCLIN input stream to determine why the JCLIN control statement is not in the proper place.

GIM41702E JCLIN PROCESSING FAILED BECAUSE SMP/E REACHED END-OF-FILE ON THE JCLIN INPUT STREAM WITHOUT FINDING AN SMPE-END CONTROL STATEMENT NEEDED TO CLOSE AN OPEN DO/END GROUP STARTED BY THE FOLLOWING JCLIN CONTROL STATEMENT - *statement*

Explanation:

statement	80-byte JCLIN control statement

SMP/E was looking for an **SMPE-END** control statement to close an open **DO/END** group in a JCLIN input stream. End-of-file for the JCLIN input stream was encountered before the **SMPE-END** control statement was found.

System Action: JCLIN processing fails for the current JCLIN input stream.

Programmer Response: Determine why the **SMPE-END** control statement was not found and correct the error. Then rerun the SMP/E command.

GIM41703I JCLIN PROCESSING IS SKIPPING JCLIN INPUT STREAM RECORDS FOUND AFTER THE FOLLOWING JCLIN CONTROL STATEMENT - *statement*

Explanation:

statement	80-byte JCLIN control statement

JCLIN processing has determined that it must skip JCLIN input stream records until the **DO/END** group

started by the indicated JCLIN control statement is closed by its corresponding **SMPE-END** control statement. While skipping records in the JCLIN input stream, JCLIN processing still syntax checks the new JCLIN control statements that may exist in the skipped section of the JCLIN input stream. Additionally, **DO/END** groups and **IF THEN/ELSE** clauses are matched up in the skipped section of the JCLIN input stream. If an error is detected when doing this syntax checking and matching, the appropriate error message is issued and JCLIN processing is terminated.

System Action: JCLIN processing continues looking for the needed **SMPE-END** control statement.

Programmer Response: None.

GIM41704I *number* **JCLIN INPUT STREAM RECORDS WERE SKIPPED BEFORE FINDING THE APPROPRIATE SMPE-END CONTROL STATEMENT.**

Explanation:

number a number

JCLIN processing was skipping JCLIN input stream records looking for an **SMPE-END** to close an open **DO/END** group. *number* records were skipped before the appropriate **SMPE-END** control statement was found.

System Action: JCLIN processing continues.

Programmer Response: None.

GIM41799E **A JCLIN CONTROL STATEMENT THAT IS SUPPORTED ONLY BY z/OS** *release* **SMP/E (OR HIGHER) HAS BEEN ENCOUNTERED IN THE JCLIN INPUT STREAM.**

Explanation:

release Minimum version and release (*vv.rr*) of z/OS SMP/E required to process this JCLIN input stream.

Note: SMP/E V3R1 is identical to z/OS V1R2 SMP/E.

During JCLIN processing, SMP/E found a JCLIN control statement (an SMPE-IF, SMPE-END, or SMPE-ELSE) that is defined for usage in the specified SMP/E level.

System Action: JCLIN processing is terminated for the current JCLIN input stream.

Programmer Response: Use the specified level of SMP/E to process the JCLIN input.

GIM41801I **INLINE JCLIN PROCESSING FAILED FOR SYSMOD** *sysmod***.**

Explanation:

sysmod SYSMOD ID

Processing of inline JCLIN failed for the indicated SYSMOD.

System Action: SYSMOD processing stops.

Programmer Response: Look at the previous messages from JCLIN processing to find the cause of the error. Then do the following:

1. Restore the SYSMOD.
2. Correct the inline JCLIN.
3. Receive and apply the SYSMOD again.

GIM41802I **INLINE JCLIN PROCESSING WAS SUCCESSFUL FOR SYSMOD** *sysmod***.**

Explanation:

sysmod SYSMOD ID

SMP/E successfully processed inline JCLIN for the indicated SYSMOD.

System Action: Processing continues.

Programmer Response: None.

GIM41901E **THE GLOBAL ZONE SYSMOD ENTRY FOR** *sysmod* **INDICATES THAT IT CONTAINS A ++***mcstype* **MCS. HOWEVER, THERE IS NO ++***mcstype* **MCS FOR** *sysmod* **IN THE SMPPTS DATA SET.**

Explanation:

sysmod SYSMOD ID
mcstype MCS types (++JCLIN, ++MOVE, ++DELETE, or ++RENAME)

SMP/E had set an indicator in the global zone SYSMOD entry to show that the indicated MCS was contained in the SYSMOD. However, SMP/E could not find the MCS in the SMPPTS MCS entry.

System Action: SYSMOD processing stops.

Programmer Response: Do the following:

1. Make sure the SMPPTS MCS entry was not changed after the SYSMOD was received.
2. Reject the SYSMOD and correct the error by either removing or adding the indicated MCS statement.
3. Receive the SYSMOD again so that the status indicator in the global zone SYSMOD entry correctly shows the contents of the SMPPTS MCS entry.

GIM42001W THE FOLLOWING CONDITIONS FOR SYSMOD *sysmod* WERE NOT SATISFIED, BUT WERE IGNORED BECAUSE THE BYPASS OPERAND WAS SPECIFIED. PROCESSING CONTINUES.

Explanation:

sysmod	SYSMOD ID

Processing would have failed for the indicated SYSMOD, but it did not fail because BYPASS was specified on the command being processed. However, certain requisites or holds were not satisfied for that SYSMOD. Message GIM359*xx* follows this message and names the requisites or holds that were not satisfied.

System Action: Processing continues.

Programmer Response: None.

GIM42003W SYSMOD *sysmod1* IN ZONE *zonename* HAS A CROSS-ZONE REQUISITE FOR SYSMOD *sysmod2*. THE CROSS-ZONE REQUISITE WILL NO LONGER BE SATISFIED WHEN SYSMOD *sysmod2* IS RESTORED.

Explanation:

sysmod1	SYSMOD ID
zonename	zone name that needs the cross-zone requisite
sysmod2	SYSMOD ID

Processing would have failed for SYSMOD *sysmod2*, but it did not fail because BYPASS(XZIFREQ) was specified on the RESTORE command. However, when SYSMOD *sysmod2* is restored, the cross-zone requisite stated by SYSMOD *sysmod1* in zone *zonename* will no longer be satisfied.

System Action: Processing continues.

Programmer Response: The SYSMOD in zone *zonename* with the cross-zone requisite for SYSMOD *sysmod2* should also be restored to ensure that the system will function properly.

GIM42100W AN ATTEMPT TO OBTAIN THE NAME OF THE PHYSICAL DATA SET CONTAINING THE DIRECTORY FOR *ddname* FAILED. THE RETURN CODE FROM THE *service* CALLABLE SERVICE WAS '*rc*'X AND THE REASON CODE WAS '*rsncode*'X.

Explanation:

ddname	ddname
service	callable service name
rc	return code from callable service in hexadecimal
rsncode	reason code from callable service in hexadecimal

SMP/E attempted to obtain the name of the physical data set containing the directory for the allocated file, but the callable service was unsuccessful.

System Action: Processing continues. However, the name of the physical data set is not shown in the file allocation report nor in the LCI records if LCI processing is in effect.

Programmer Response: None, but you may want to fix the error for future SMP/E jobs.

GIM42500W AN ATTEMPT TO OBTAIN THE NAME OF THE PHYSICAL DATA SET CONTAINING A SYMBOLIC LINK FOR *elmtype elmname* IN *library* FAILED. THE RETURN CODE FROM THE *service* CALLABLE SERVICE WAS '*rc*'X AND THE REASON CODE WAS '*rsncode*'X.

Explanation:

elmtype	element type
elmname	element name
library	ddname
service	callable service name
rc	return code from callable service in hexadecimal
rsncode	reason code from callable service in hexadecimal

SMP/E attempted to obtain the name of the physical data set containing a symbolic link for an element updated in library *library*, but the callable service was unsuccessful.

System Action: Processing continues. However, the name of the physical data set is not shown in the file allocation report nor in the LCI records if LCI processing is in effect.

Programmer Response: None, but you may want to fix the error for future SMP/E jobs.

GIM42201E THE UCLIN CHANGES WERE NOT MADE BECAUSE THE ++LMODIN INPUT CONTAINS MULTIPLE NAME STATEMENTS. ONLY ONE NAME STATEMENT IS ALLOWED.

Explanation: Only one NAME statement is allowed in the ++LMODIN input for a given UCL statement. That NAME statement must refer to the LMOD entry that contains the ++LMODIN input.

System Action: The requested UCLIN changes are not made.

Programmer Response: Do **one** of the following:

- Change the UCL statement so that the ++LMODIN input contains just one NAME statement for the associated LMOD entry.
- If you are changing more than one LMOD, split up the ++LMODIN input into separate UCLIN statements for the individual LMOD entries.

Then rerun the job.

GIM42202E THE UCLIN CHANGES WERE NOT MADE BECAUSE THE ++LMODIN INPUT IS INCORRECT. LINK–EDIT CONTROL STATEMENTS MAY NOT BE SELECTIVELY DELETED.

Explanation: You cannot selectively delete link-edit control statements from an LMOD entry. You can only delete them all. The following example shows the error:

```
SET BDY(ZOSZSET)  /* Set to zone processing */.
UCLIN             /* Start ULCIN            */.
DEL LMOD(LMODX)   /* Identify LMOD entry    */
++LMODIN          /* Delete link-edit       */
ENTRY START       /* control statements     */
ALIAS SECRET
++ENDLMODIN.
ENDUCL.           /* End ULCIN processing   */.
```

System Action: The requested UCLIN changes are not made.

Programmer Response: Remove the link-edit control statements from the UCLIN command. Then rerun the job. The following example shows the proper syntax for deleting the link-edit control statements from an LMOD entry:

```
SET BDY(ZOSZSET)  /* Set to zone processing */.
UCLIN             /* Start UCLIN            */.
DEL LMOD(LMODX)   /* Identify LMOD entry    */
++LMODIN          /* Delete link-edit       */
++ENDLMODIN       /* control statements     */.
ENDUCL.           /* End UCLIN processing   */.
```

Note: Using the REP command instead of the DEL command gives you the same result.

GIM42301S THE SPECIFIED DATE IS NOT IN THE EXPECTED RANGE.

Explanation: One of the following occurred:

- SMP/E did not recognize the date specified on the command being processed. It expected to see *mm dd yy* — the month (*mm*), followed by the day (*dd*), followed by the year (*yy*).
- The beginning date (*mm dd yy*) is later than the ending date (*mm dd yy*).

System Action: Command processing stops.

Programmer Response: Correct the date as follows:

- If SMP/E did not recognize the format of the date:
  - Specify *mm* as a number from 01 to 12.
  - Specify *dd* as a number from 01 to 31.
  - Specify *yy* as a number from 00 to 99.
- If the beginning date was later than the ending date, change the dates so that the ending date is later than the beginning date.

Then rerun the command.

GIM42401I THE FOLLOWING PARAMETERS WERE SPECIFIED ON THE EXEC STATEMENT FOR GIMSMP: *parameters*.

Explanation:

parameters parameters on the EXEC statement

This message lists the parameters that were specified on the EXEC statement for GIMSMP.

System Action: None.

Programmer Response: None.

GIM42601E SYSMOD *sysmod1* WAS NOT APPLIED BECAUSE IT DELETES SYSMOD *sysmod2*, WHICH WAS ALREADY APPLIED. *sysmod1* CAN ONLY BE APPLIED IF IT IS SPECIFIED ON THE SELECT OPERAND.

Explanation:

sysmod1 ID of the deleting SYSMOD
sysmod2 ID of the SYSMOD being deleted

SYSMOD *sysmod1* deletes another SYSMOD (SYSMOD *sysmod2*). Therefore, it will be processed only if it is specified on the SELECT operand.

System Action: SYSMOD processing stops. Message GIM37001S follows this message and shows that command processing also fails.

Programmer Response: Do **one** of the following:

- If you want to apply SYSMOD *sysmod1*, specify it on the SELECT operand.
- If you do not want to apply SYSMOD *sysmod1*, do **one** of the following:
  - Use the REJECT command to remove SYSMOD *sysmod1* from the global zone.
  - Specify SYSMOD *sysmod1* on the EXCLUDE operand to exclude it from processing.

Then rerun the APPLY command.

GIM42602E SYSMOD *sysmod1* WAS NOT ACCEPTED BECAUSE IT DELETES SYSMOD *sysmod2*, WHICH WAS ALREADY ACCEPTED. *sysmod1* CAN ONLY BE ACCEPTED IF IT IS SPECIFIED ON THE SELECT OPERAND.

Explanation:

sysmod1 ID of the deleting SYSMOD

sysmod2 ID of the SYSMOD being deleted

SYSMOD *sysmod1* deletes another SYSMOD (SYSMOD *sysmod2*). Therefore, it will be processed only if it is specified on the SELECT operand.

System Action: SYSMOD processing stops. Message GIM37001S follows this message and shows that command processing also fails.

Programmer Response: Do **one** of the following:
- If you want to accept SYSMOD *sysmod1*, specify it on the SELECT operand.
- If you do not want to accept SYSMOD *sysmod1*, do **one** of the following:
  - Use the REJECT command to remove SYSMOD *sysmod1* from the global zone.
  - Specify SYSMOD *sysmod1* on the EXCLUDE operand to exclude it from processing.

Then rerun the ACCEPT command.

GIM42701E SYSMOD *sysmod1* WAS NOT APPLIED BECAUSE IT IS MUTUALLY EXCLUSIVE (NPRE) WITH FUNCTION SYSMOD *sysmod2*, WHICH WAS ALREADY APPLIED.

Explanation:

sysmod1 ID of the SYSMOD being installed
sysmod2 ID of the SYSMOD that is already installed

Mutually exclusive SYSMODs cannot be installed on the same system. SYSMOD *sysmod1* was not applied because a mutually exclusive SYSMOD, SYSMOD *sysmod2*, had already been applied.

System Action: SYSMOD processing stops. Message GIM37001S follows this message and shows that command processing also fails.

Programmer Response: Do **one** of the following:
- If you do not want to process SYSMOD *sysmod1*, do **one** of the following:
  - Use the REJECT command to remove SYSMOD *sysmod1* from the global zone.
  - Specify SYSMOD *sysmod1* on the EXCLUDE operand to exclude it from processing.
- If you want to process SYSMOD *sysmod1*, use the RESTORE command to remove SYSMOD *sysmod2* from the target library and target zone.

Then rerun the APPLY command.

GIM42702E SYSMOD *sysmod1* WAS NOT ACCEPTED BECAUSE IT IS MUTUALLY EXCLUSIVE (NPRE) WITH FUNCTION SYSMOD *sysmod2*, WHICH WAS ALREADY ACCEPTED.

Explanation:

sysmod1 ID of the SYSMOD being installed
sysmod2 ID of the SYSMOD that is already installed

Mutually exclusive SYSMODs cannot be installed on the same system. SYSMOD *sysmod1* was not accepted because a mutually exclusive SYSMOD, SYSMOD *sysmod2*, had already been accepted.

System Action: SYSMOD processing stops. Message GIM37001S follows this message and shows that command processing also fails.

Programmer Response: Do **one** of the following:
- If you do not want to process SYSMOD *sysmod1*, do **one** of the following:
  - Use the REJECT command to remove SYSMOD *sysmod1* from the global zone.
  - Specify SYSMOD *sysmod1* on the EXCLUDE operand to exclude it from processing.
- If you want to process SYSMOD *sysmod1*, use UCLIN to delete SYSMOD *sysmod2*.

Then rerun the ACCEPT command.

GIM42800S THE *keyword* KEYWORD HAS BEEN SPECIFIED MORE THAN ONCE IN THE *dataset* DATA SET.

Explanation:

keyword duplicate keyword operand
dataset data set being parsed

The identified keyword operand was found more than once while SMP/E was parsing the identified data set.

System Action: Command processing stops.

Programmer Response: Correct the syntax by deleting the duplicate keyword, and rerun the command.

GIM42901E SYSMOD *sysmod1* WAS NOT RESTORED BECAUSE *elmtype elmname* IS IN BOTH SYSMOD *sysmod1* AND SYSMOD *sysmod2*, BUT *sysmod2* IS ACCEPTED IN ERROR.

Explanation:

sysmod1 SYSMOD ID
elmtype element type
elmname element name
sysmod2 SYSMOD ID

SYSMOD *sysmod1*, which is being restored, has an element in common with SYSMOD *sysmod2*. However, SMP/E cannot restore the element to the level of SYSMOD *sysmod2* because SYSMOD *sysmod2* was accepted in error.

For example, assume you have installed the following PTFs:

TARGET ZONE SYSMODs	DLIB ZONE SYSMODs
UZ00001 UZ00002	UZ00001 (in error)

Assume UZ00002 and UZ00001 have an element in common. If you tried to restore SYSMOD UZ00002, you would get this message because UZ00001, which supplied the previous version of the element, was not successfully accepted.

System Action: SYSMOD processing stops.

Programmer Response: Reaccept SYSMOD *sysmod2* so that it is accepted without errors. Then rerun the RESTORE command for SYSMOD *sysmod1*.

GIM42902E **SYSMOD** *sysmod1* **WAS NOT RESTORED BECAUSE** *elmtype elmname* **IS IN BOTH SYSMOD** *sysmod1* **AND SYSMOD** *sysmod2*, **BUT** *sysmod2* **HAS NOT BEEN ACCEPTED.**

Explanation:

sysmod1	SYSMOD ID
elmtype	element type
elmname	element name
sysmod2	SYSMOD ID

SYSMOD *sysmod1*, which is being restored, has an element in common with SYSMOD *sysmod2*. However, SMP/E cannot restore the element to the level of SYSMOD *sysmod2* because SYSMOD *sysmod2* has not been accepted.

For example, assume you have installed the following PTFs:

TARGET ZONE SYSMODs	DLIB ZONE SYSMODs
UZ00001 UZ00002 UZ00003	UZ00001

Assume UZ00003 and UZ00002 have an element in common. If you tried to restore SYSMOD UZ00003, you would get this message because UZ00002, which supplied the previous version of the element, has not been accepted and is not being restored.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Accept SYSMOD *sysmod2*, then rerun the RESTORE command for SYSMOD *sysmod1*.
- Restore both of the SYSMODs.

GIM42903E **SYSMOD** *sysmod1* **WAS NOT RESTORED BECAUSE** *elmtype elmname* **IS IN BOTH SYSMOD** *sysmod1* **AND SYSMOD** *sysmod2*, **BUT** *sysmod2* **HAS FAILED.**

Explanation:

sysmod1	SYSMOD ID
elmtype	element type
elmname	element type
sysmod2	SYSMOD ID

SYSMOD *sysmod1* and SYSMOD *sysmod2*, which are both being restored, have an element in common. Because RESTORE processing has failed for SYSMOD *sysmod2*, SYSMOD *sysmod1* cannot be restored.

System Action: SYSMOD processing stops.

Programmer Response: Determine and fix the cause of the RESTORE failure for SYSMOD *sysmod2*. Then rerun the RESTORE command for both SYSMOD *sysmod1* and SYSMOD *sysmod2*.

GIM43001I *zonename* **ZONE** *modid sysmod* **IS NOT APPLIED.**

Explanation:

zonename	target zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing. There is no SYSMOD entry for the indicated MODID, even though it appears in the target zone element entry that is being restored. RESTORE processing cannot continue without causing a potential mismatch between elements on the system.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:

- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43002I *zonename* **ZONE** *modid sysmod* **WAS NOT RESTORED BECAUSE RELATED SYSMODS FAILED.**

Explanation:

zonename	target zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

A target zone element contains multiple UMIDs because it was updated by several SYSMODs. Because SMP/E cannot restore the indicated UMID, it cannot restore any of the UMIDs for the element.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43003I *zonename* **ZONE** *modid sysmod* **IS APPLIED.**

Explanation:

zonename	target zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

The indicated SYSMOD has been applied, but it was not specified on the RESTORE command.

Assume UZ00003 and UZ00002 have an element in common. If you tried to restore SYSMOD UZ00003 you would get this message because UZ00002, which supplied the previous version of the element, is not being restored.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43004I *zonename* **ZONE** *modid sysmod* **IS SUPERSEDED.**

Explanation:

zonename	target zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

The indicated SYSMOD has been superseded. This can only happen in one of the following situations:
- A previous APPLY command failed while SMP/E was updating the target zone entries for the SYSMOD.
- UCLIN was used to change the element or SYSMOD entry.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43005I *zonename* **ZONE** *modid sysmod* **IS DELETED.**

Explanation:

zonename	zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

The indicated SYSMOD has been deleted. This can happen in one of the following situations:
- A previous APPLY command failed while SMP/E was updating the target zone entries for the SYSMOD.
- UCLIN was used to change the element or SYSMOD entry.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43006I *zonename* **ZONE** *modid sysmod* **IS ACCEPTED BUT NOT APPLIED.**

Explanation:

zonename	DLIB zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

The indicated SYSMOD is a MODID in a distribution zone element entry. It has been accepted but has not been applied. If you try to restore the element involved, SMP/E will effectively apply a SYSMOD that has not been applied to the target zone. (In this case, if the SYSMOD has been superseded, it will not be applied.)

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.

- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43007I *zonename* **ZONE** *modid sysmod* **IS ACCEPTED IN ERROR.**

Explanation:

zonename	DLIB zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

The indicated SYSMOD is a MODID in a distribution zone element entry, but an error occurred when it was accepted. Therefore, it cannot be used for RESTORE processing.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43008I *zonename* **ZONE** *modid sysmod* **IS ACCEPTED BUT SUPERSEDED ONLY.**

Explanation:

zonename	DLIB zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

The indicated SYSMOD is a FMID, RMID, or UMID in a distribution zone element entry, but it is superseded only in the target zone. For example, it may never have been applied, but a SYSMOD that superseded it was applied. Therefore, SMP/E cannot restore back to the level of this superseded SYSMOD.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43009I *sysmod* **IS THE FMID IN THE DLIB ZONE BUT NOT IN THE TARGET ZONE.**

Explanation:

sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

One of the following is true:
- The element entry does not exist in the target zone.
- The element entry exists but does not contain a valid FMID subentry.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43010I *zonename* **ZONE RMID** *sysmod* **DOES NOT SPECIFY ONE OF THE DLIB ZONE MODIDS AS A ++VER PRE OR ++VER SUP VALUE.**

Explanation:

zonename	target zone name
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

The indicated SYSMOD does not have a correct PRE or SUP relationship with any distribution zone RMID or UMID. To be correct, one of the following must be true:
- The indicated SYSMOD must supersede or be a prerequisite of a distribution zone RMID or UMID.
- The indicated SYSMOD must supersede or be a prerequisite of another SYSMOD that SMP/E is concurrently restoring and that has a correct PRE or SUP relationship with a distribution zone RMID or UMID.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43011I *zonename* **ZONE** *modid sysmod* **IS APPLIED BUT NOT ACCEPTED.**

Explanation:

zonename	target zone name
modid	FMID, RMID, or UMID
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing.

The indicated SYSMOD has been applied, but it has not been accepted and was not specified on the RESTORE command.

For example, assume you have installed the following PTFs:

TARGET ZONE SYSMODs	DLIB ZONE SYSMODs
UZ00001 UZ00002 UZ00003	UZ00001

Assume UZ00003 and UZ00002 have an element in common. If you tried to restore SYSMOD UZ00003, you would get this message because UZ00002, which supplied the previous version of the element, has not been accepted and is not being restored.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43012I *zonename* **ZONE UMID** *sysmod* **IS NOT SPECIFIED BY ONE OF THE TARGET ZONE MODIDS AS A ++VER PRE OR ++VER SUP VALUE.**

Explanation:

zonename	DLIB zone name
sysmod	SYSMOD ID

This message further explains the MODID error described by message GIM382*xx* for RESTORE processing. None of SYSMODs that are target zone MODIDs for the element being processed specify a relationship to the indicated distribution zone UMID value for that element. (For example, they do not supersede the UMID or specify it as a prerequisite.) As a result, SMP/E cannot determine whether it should restore the SYSMOD in message GIM382*xx*.

System Action: Processing of the SYSMOD specified in GIM382*xx* stops unless you specified **BYPASS(ID)**.

Programmer Response: Do **one** of the following:
- If the wrong set of SYSMODs was specified, use SELECT or GROUP to process the correct set.
- If the correct set of SYSMODs was specified, use BYPASS(ID) to restore the SYSMODs.

GIM43101E **THERE IS NO** *value* **VALUE FOR** *elmtype elmname* **IN SYSMOD** *sysmod***.**

Explanation:

value	DISTLIB or SYSLIB
elmtype	element type
elmname	element name
sysmod	SYSMOD ID

One of the following conditions has occurred:
- During ACCEPT processing, SMP/E found an element whose distribution library could not be determined from the distribution zone element entry.
- During APPLY processing, SMP/E found an element whose target library could not be determined from the target zone element entry.
- During APPLY processing, the DISTLIB for a module was missing from both the MOD entry and the ++MOD modification control statement.
- During APPLY processing, the SYSLIB for an hierarchical file system element was missing from the applicable element modification control statement.
- During RESTORE processing, either the distribution library was missing from the target zone element entry, or the target zone element entry itself was missing.

System Action: Command processing stops for the indicated SYSMOD.

Programmer Response: Use UCLIN or JCLIN to create or update the necessary target zone or distribution zone entries so that SMP/E can finish processing the element. Then rerun the command.

GIM43201T **PROGRAM** *progname* **FAILED WITH A** *abntype* **ABEND CODE OF** *abncode***.**

Explanation:

progname	program name
abntype	SYSTEM or USER
abncode	abend code

An abend occurred, and SMP/E invoked its recovery routine.

System Action: If the directories were in storage for write processing, SMP/E rewrote the directories of the affected SMP/E data sets to disk. SMP/E generates the SMP/E summary reports with the current element and SYSMOD status.

Programmer Response: Look at the following:

- The SMP/E summary reports to find the element and SYSMOD status
- The abend dump for problem determination.

GIM43301E SYSMOD *sysmod1* WAS NOT RESTORED BECAUSE *sysmod1* IS SUPERSEDED BY SYSMOD *sysmod2*, WHICH IS NOT BEING RESTORED.

Explanation:

sysmod1	ID of the superseded SYSMOD
sysmod2	ID of superseding SYSMOD

SMP/E could not restore SYSMOD *sysmod1* because it could not restore SYSMOD *sysmod2*. The SYSMOD entry for SYSMOD *sysmod1* indicated that SYSMOD *sysmod2* superseded SYSMOD *sysmod1*. (SYSMOD *sysmod2* was specified as a SUPBY value, which may have been set by UCLIN.) However, SMP/E could not restore SYSMOD *sysmod2* for one of these reasons:
- SYSMOD *sysmod2* has already been accepted.
- SYSMOD *sysmod2* has not been applied.

As a result, SMP/E also could not restore SYSMOD *sysmod1*.

System Action: Processing stops for SYSMOD *sysmod1*.

Programmer Response: Eliminate SYSMOD *sysmod1* from RESTORE processing.

GIM43401W *elmtype elmname* IN SYSMOD *sysmod* WAS NOT INSTALLED IN ANY TARGET LIBRARY.

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID

The indicated element was not applied to any target libraries. This is probably because no target library was specified for the element in either the JCLIN or on the SYSLIB operand of the element MCS.

Note: Even though the element was not installed in a target library, SMP/E updates the target zone element entry as follows:
- Creates an element entry (if one does not already exist) to indicate that it has processed the element.
- Updates the RMID value in the element entry.

System Action: SYSMOD processing continues.

Programmer Response: Do **one** of the following:
- If the element does not belong in a target library, do nothing.
- If the element belongs in a target library, check that the JCLIN or MCS defines the target library where the element should reside.

You can find out if the element belongs on a target library by doing one of the following:
- If the element is totally defined by an MCS (with no JCLIN) and the MCS does not specify a SYSLIB value, talk to the person who created the MCS (the product owner) to determine where the element should be installed.
  - If the element should be installed only in a distribution library, no further action is required.
  - If the element should be installed in a target library, the element was not packaged correctly. First, have the product owner add a SYSLIB value to the element MCS. Then reinstall the SYSMOD containing the element.
- If the element is defined by JCLIN:
  - Talk to the person who created the JCLIN.
  - Look at the JCLIN being processed to see if there is an INCLUDE statement for the element being applied. If there is no INCLUDE statement, the element does not belong in a target library.

    Note: If the element is part of a totally copied library, it will not show up in an INCLUDE statement.
  - Run the JCLIN and check the JCLIN Summary Report. This report will give you a complete list of all the INCLUDE statements that were processed. Check whether there is an INCLUDE statement for the module you were processing.

GIM43500S THE CALL TO THE *service* SERVICE FAILED. THE RETURN CODE WAS '*retcode*'X AND THE REASON CODE WAS '*rsncode*'X.

Explanation:

service	name of service that failed
retcode	return code from service in hexadecimal
rsncode	reason code from service in hexadecimal

The identified USS (UNIX System Service) service, ICSF (Integrated Cryptographic Service Facility) service, or other service returned to SMP/E with an error.

System Action: Command processing stops.

Programmer Response: Refer to the documentation for the identified service to determine the source of the error. For example, if the service name begins with BPX, look in USS documentation, or if the service name starts with CSN, look in ICSF documentation. Correct the error and rerun the command.

GIM43500W THE CALL TO THE *service* SERVICE FAILED. THE RETURN CODE WAS '*retcode*'X AND THE REASON CODE WAS '*rsncode*'X.

Explanation:

service name of service that failed
retcode return code from service in hexadecimal
rsncode reason code from service in hexadecimal

The identified USS (UNIX System Service) service, ICSF (Integrated Cryptographic Service Facility) service, or other service returned to SMP/E with an error.

System Action: Command processing stops.

Programmer Response: Refer to the documentation for the identified service to determine the source of the error. For example, if the service name begins with BPX, look in USS documentation, or if the service name starts with CSN, look in ICSF documentation. Correct the error and rerun the command.

GIM43601E THE VERNUM IN THE UCLIN CHANGES MUST MATCH THE EXISTING VERNUM VALUE IN THE SYSMOD ENTRY.

Explanation: The VERNUM operand was specified on a UCL statement, but the resulting update would have caused the SYSMOD to have subentries with different VERNUM values.

System Action: The requested UCLIN changes are not made.

Programmer Response: Do the following:
1. Check the existing subentries in the SYSMOD for their VERNUM values.
2. Do **one** of the following:
   - Specify the existing VERNUM values and rerun the UCLIN.
   - Replace all existing subentries that require the VERNUM value.

GIM43701T ESTAE ERROR RECOVERY PROCESSING FAILED WITH A RETURN CODE OF *rc*.

Explanation:

rc return code

An error occurred during ESTAE processing.

System Action: System processing stops.

Programmer Response: For more information about the ESTAE return code, see *z/OS MVS Programming: Assembler Services Reference ABE-HSP*.

GIM43801T SMP/E COULD NOT INITIALIZE ITS UTILITY INTERFACE. THE REASON CODE WAS *rc*.

Explanation:

rc two-digit reason code that shows why SMP/E could not initialize the utility interface

SMP/E could not initialize the SMP/E subtask that interfaces with utility programs. This is the reason code that was issued:

04 Subtask initialization failed because SMP/E was running on a non-MVS system.

System Action: SMP/E processing stops.

Programmer Response: Make sure that SMP/E is installed and running on an OS/390 or MVS system. Then rerun the job.

GIM43802T SMP/E COULD NOT INITIALIZE ITS UTILITY INTERFACE. THE REASON CODE WAS *rsncode*. THE RETURN CODE WAS *rtncode*.

Explanation:

rsncode
two-digit reason code that shows why SMP/E could not initialize the subtask
rc two-digit return code associated with the reason codes

SMP/E could not initialize the SMP/E subtask that interfaces with utility programs. These are the reason codes that may be issued:

01 IDENTIFY processing for the entry point of the subtask program issued a return code greater than 4. The return code given with this reason code is the return code from IDENTIFY. For an explanation of the IDENTIFY return codes, see *z/OS MVS Programming: Assembler Services Reference ABE-HSP*.

02 ATTACH processing for the subtask issued a nonzero return code. The return code given with this reason code is from ATTACH. For an explanation of the ATTACH return codes, see *z/OS MVS Programming: Assembler Services Reference ABE-HSP*.

03 Subtask initialization failed because the subtask could not establish its ESTAE coverage. The return code given with this reason code is from ESTAE. For an explanation of the ESTAE return codes, see *z/OS MVS Programming: Assembler Services Reference ABE-HSP*.

System Action: SMP/E processing stops.

Programmer Response: Fix the error and rerun the job.

GIM43901S RETRY PROCESSING FAILED FOR THE *library* LIBRARY.

Explanation:

library ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

SMP/E attempted retry processing after a B37–04, D37–04, or E37–04 abend for a library being processed by a utility. An error occurred, and the retry failed.

System Action: Command processing stops.

Programmer Response: Increase the size of the library being processed and rerun the job.

GIM43901T RETRY PROCESSING FAILED FOR THE *library* LIBRARY.

Explanation:

library ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

SMP/E attempted retry processing after a B37–04, D37–04, or E37–04 abend for a library being processed by a utility. An error occurred, and the retry failed.

System Action: SMP/E processing stops.

Programmer Response: Increase the size of the library being processed and rerun the job.

GIM43903I RETRY PROCESSING WILL BE ATTEMPTED FOR THE *library* LIBRARY.

Explanation:

library ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

Because a D37–04 or E37–04 abend occurred for a library being processed by a utility, SMP/E attempts retry processing.

System Action: SMP/E processing continues.

Programmer Response: None.

GIM43904I SMP/E WILL DEBATCH UPDATES AND PROCESS MEMBERS ONE AT A TIME FOR THE *library* LIBRARY.

Explanation:

library ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

A second *x*37 abend has occurred for the indicated library during retry processing. Because multiple members were being updated, SMP/E now attempts to update each member individually in an effort to get all updates into the space-constrained library.

System Action: SMP/E processing continues.

Programmer Response: None.

GIM43905I A POSSIBLE STORAGE SHORTAGE HAS OCCURRED FOR A LINK-EDIT SUBTASK. SMP/E WILL FREE AS MUCH STORAGE AS POSSIBLE AND RETRY THE FAILING SUBTASK FOR THE *library* LIBRARY.

Explanation:

library ddname of the library for APPLY or RESTORE processing

The link-edit utility has ended with a return code of 16 while processing against the indicated library. Because SMP/E is currently multitasking link-edit invocations, the return code 16 is likely caused by a storage shortage.

SMP/E frees up as much storage as possible by allowing all other link-edit subtasks to complete. Then SMP/E retries the batch of link-edits that failed with a return code 16.

System Action: SMP/E processing continues.

Programmer Response: None.

GIM44001I SYSTEM ABEND *abncode* OCCURRED WITH A REASON CODE OF *rsncode* AFTER SMP/E CALLED THE *utility* UTILITY. THE *library* LIBRARY RAN OUT OF SPACE.

Explanation:

abncode abend code
rsncode abend reason code
utility name of the load module for the utility
library ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The library is eligible for retry processing.

System Action: Retry processing is done.

Programmer Response: Your action depends on the success or failure of retry processing.

GIM44002S SYSTEM ABEND *abncode* OCCURRED WITH A REASON CODE OF *rsncode* AFTER SMP/E CALLED THE *utility* UTILITY TO PROCESS THE *library* LIBRARY.

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The system abend occurred when the indicated library was being processed by the utility.

System Action: Command processing stops.

Programmer Response: Your action depends on the system abend that occurred.

GIM44003S USER ABEND *abncode* OCCURRED WITH A REASON CODE OF *rsncode* AFTER SMP/E CALLED THE *utility* UTILITY TO PROCESS THE *library* LIBRARY.

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The user abend occurred when the indicated library was being processed by the utility.

System Action: Command processing stops.

Programmer Response: Your action depends on the user abend that occurred.

GIM44004S SYSTEM ABEND *abncode* OCCURRED WITH A REASON CODE OF *rsncode* AFTER SMP/E CALLED THE *utility* UTILITY. THE *library* LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT IS NOT THE LIBRARY BEING UPDATED BY SMP/E.

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library

The indicated library ran out of space during the utility invocation. The library is not eligible for retry processing, because it is not the library being updated by SMP/E.

System Action: Command processing stops.

Programmer Response: Increase the size of the library being processed, and rerun the job.

GIM44005S SYSTEM ABEND *abncode* OCCURRED AFTER SMP/E CALLED THE *utility* UTILITY TO PROCESS THE *library* LIBRARY. NO SYSTEM DIAGNOSTIC WORK AREA (SDWA) WAS PROVIDED TO THE ESTAE ROUTINE.

Explanation:

abncode	abend code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The system abend occurred when the indicated library was being processed by the utility.

System Action: Command processing stops.

Programmer Response: Your action depends on the system abend that occurred.

GIM44006S SYSTEM ABEND *abncode* OCCURRED WITH A REASON CODE OF *rsncode* AFTER SMP/E CALLED THE *utility* UTILITY TO PROCESS THE *library* LIBRARY. THE *library* LIBRARY RAN OUT OF SPACE. THE RETRY EXIT ROUTINE CANCELED RETRY PROCESSING.

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The retry exit routine canceled retry processing for the library.

System Action: Command processing stops.

Programmer Response: Determine why the exit routine ended the request. Increase the size of the library being processed, and rerun the job.

GIM44006T SYSTEM ABEND *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY TO PROCESS THE** *library* **LIBRARY. THE** *library* **LIBRARY RAN OUT OF SPACE. THE RETRY EXIT ROUTINE CANCELED RETRY PROCESSING.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The retry exit routine canceled retry processing for the library.

System Action: SMP/E processing stops.

Programmer Response: Determine why the exit routine ended the request. Increase the size of the library being processed, and rerun the job.

GIM44007E SYSTEM ABEND *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY. THE** *library* **LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT WAS NOT SPECIFIED IN THE RETRYDDN LIST.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The library is not eligible for retry processing because it was not in the RETRYDDN list explicitly or by default if ALL was specified.

System Action: The elements being processed are failed and the SYSMODs that supplied them are failed.

Programmer Response: Increase the size of the library being processed, and rerun the job.

GIM44007I SYSTEM ABEND *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY. THE** *library* **LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT WAS NOT SPECIFIED IN THE RETRYDDN LIST.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The library is not eligible for retry processing because it was not in the RETRYDDN list explicitly or by default if ALL was specified.

System Action: A subsequent message identifies the system action.

Programmer Response: Increase the size of the library being processed, and rerun the job if necessary.

GIM44008E SYSTEM ABEND *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY. THE** *library* **LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT WAS EXCLUDED VIA THE EXRTYDD LIST.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The library is not eligible for retry processing because it was explicit excluded by the EXRTYDD list in the OPTIONS entry that was in effect.

System Action: The elements being processed are failed and the SYSMODs that supplied them are failed.

Programmer Response: Increase the size of the library being processed and rerun the job.

GIM44008I **SYSTEM ABEND** *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY. THE** *library* **LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT WAS EXCLUDED VIA THE EXRTYDD LIST.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The library is not eligible for retry processing because it was explicitly excluded by the EXRTYDD list in the OPTIONS entry that was in effect.

System Action: A subsequent message identifies the system action

Programmer Response: Increase the size of the library being processed, and rerun the job if necessary.

GIM44009E **SYSTEM ABEND** *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY. THE** *library* **LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE RETRY(NO) WAS SPECIFIED ON THE** *command* **COMMAND.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library
command	SMP/E command

The indicated library ran out of space when it was being processed by the utility. The library is not eligible for retry processing because RETRY(NO) was specified on the current command.

System Action: The elements being processed are failed and the SYSMODs that supplied them are failed.

Programmer Response: Increase the size of the library being processed and rerun the job.

GIM44010S **SYSTEM ABEND** *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E INVOKED SHELL SCRIPT** *shellscr* **TO PROCESS** *elmtype elmname***.**

Explanation:

abncode	abend code
rsncode	abend reason code
shellscr	shell script name
elmtype	element type
elmname	element name

The system abend occurred when the indicated element was being processed by the shell script.

System Action: Command processing stops.

Programmer Response: Your action depends on the system abend that occurred.

GIM44012E **SYSTEM ABEND** *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY. THE SMPTLOAD LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT WAS ALLOCATED AS NEW AND THERE IS NO UNUSED SPACE TO BE REGAINED BY COMPRESSING THE LIBRARY.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility

The SMPTLOAD library ran out of space when it was being processed by the utility. The library is not eligible for retry processing because it was allocated as new and there is no unused space in the library that can be regained by compressing it.

System Action: Both the element being processed and the SYSMOD supplying the element are failed.

Programmer Response: Change the SMPTLOAD DDDEF so that more space is allocated and rerun the job.

GIM44013I **SYSTEM ABEND** *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY. THE** *library* **LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT IS A PDSE.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the

ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The library is not eligible for retry processing because it is a PDSE. No space can be reclaimed by compressing a library that is a PDSE.

System Action: A subsequent message identifies the system action

Programmer Response: Increase the size of the library being processed, and rerun the job if necessary.

GIM44014I SYSTEM ABEND *abncode* **OCCURRED WITH A REASON CODE OF** *rsncode* **AFTER SMP/E CALLED THE** *utility* **UTILITY. THE** *library* **LIBRARY RAN OUT OF SPACE. THE LIBRARY IS NOT ELIGIBLE FOR RETRY PROCESSING BECAUSE IT IS A DIRECTORY IN THE FILE SYSTEM.**

Explanation:

abncode	abend code
rsncode	abend reason code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The indicated library ran out of space when it was being processed by the utility. The library is not eligible for retry processing because it is a directory in the file system. Directories cannot be compressed.

System Action: A subsequent message identifies the system action

Programmer Response: Increase the size of the library being processed, and rerun the job if necessary.

GIM44015S SYSTEM ABEND *abncode* **OCCURRED AFTER SMP/E CALLED THE** *utility* **UTILITY TO PROCESS THE** *library* **LIBRARY.**

Explanation:

abncode	abend code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing

The system abend occurred when the indicated library was being processed by the utility.

System Action: Command processing stops.

Programmer Response: Your action depends on the system abend that occurred.

GIM44016S USER ABEND *abncode* **OCCURRED AFTER SMP/E CALLED THE** *utility* **UTILITY TO PROCESS THE** *library* **LIBRARY.**

Explanation:

abncode	abend code
utility	name of the load module for the utility
library	ddname of the library or the ddname/generated-ddname used during the cross-zone phase of APPLY or RESTORE processing.

The user abend occurred when the indicated library was being processed by the utility.

System Action: Command processing stops.

Programmer Response: Your action depends on the user abend that occurred.

GIM44017S SYSTEM ABEND *abncode* **OCCURRED AFTER SMP/E INVOKED SHELL SCRIPT** *shellscr* **TO PROCESS** *elmtype elmname***.**

Explanation:

abncode	abend code
shellscr	shell script name
elmtype	element type
elmname	element name

The system abend occurred when the indicated element was being processed by the shell script.

System Action: Command processing stops.

Programmer Response: Your action depends on the system abend that occurred.

GIM44101E SYSMOD *sysmod* **DOES NOT SUPERSEDE UMID** *umid* **FOR** *elmtype elmname***.**

Explanation:

sysmod	SYSMOD ID
umid	UMID specified on the element MCS
elmtype	element type
elmname	element name

The indicated SYSMOD specified a UMID on an element MCS but did not include that UMID on the ++VER SUP operand. The UMID operand on the element MCS indicates that the SYSMOD includes the changes made by the UMID SYSMOD. If that is the case, such as for a service-updated function, the UMID value should also be specified on the ++VER SUP operand.

System Action: SYSMOD processing stops.

Programmer Response: If the SYSMOD includes the update identified by the UMID on the element MCS, specify the UMID on the SYSMOD's ++VER SUP operand. Then rerun the job.

GIM44201I GIMMPVIA - AN ATTEMPT IS BEING MADE TO UPDATE A ZONE THAT IS OPEN FOR READ ONLY.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44202I GIMMPVIA - THE REQUIRED VPL CONNECTION FOR THIS FUNCTION IS NOT AVAILABLE.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44203I GIMMPVIA - A RECORD LENGTH OF ZERO OCCURRED WITH AN ADD OR UPDATE FUNCTION.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44204I GIMMPVIA - NO BUFFER IS AVAILABLE FOR A VPL FUNCTION THAT REQUIRES ONE.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44205I GIMMPVIA - NO KEYLENGTH IS SUPPLIED FOR A VPL FUNCTION THAT REQUIRES ONE.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44207I GIMMPVIA - AN UNRECOGNIZABLE VSAM I/O FUNCTION WAS ATTEMPTED.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44208I GIMMPVIA - A SHOWCB MACRO FAILED, EITHER FOR FEEDBACK CODE AFTER A FAILED I/O REQUEST, OR FOR LENGTH AFTER A SUCCESSFUL GET.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44209I GIMMPVIA - THE KEYLENGTH SUPPLIED IS GREATER THAN THE MAXIMUM ALLOWED.

System Action: See previous related messages.

Programmer Response: Make sure a valid KEYLENGTH value was used to define the CSI being accessed.

GIM44210I GIMMPVIA - A SEQUENCING ERROR OCCURRED UPDATING A VSAM RECORD.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44211I GIMMPVIA - A SEQUENCING ERROR OCCURRED DELETING A VSAM RECORD.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44212T GIMMPMRM - IOPTYPE *ioptype* IS NOT A MAP/REMAPPABLE ENTRY TYPE.

Explanation:

ioptype the TYPE field of the GIMMPIO parameter list

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44213T GIMMPMRM - THIS UNKNOWN IOPTYPE IS NOT MAP/REMAPPABLE.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44214I GIMMPMRM - THE IOPDSID IS UNKNOWN.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44215I GIMMPMRM - IOPFUNCT *iopfunct* **IS NOT SUPPORTED.**

Explanation:

iopfunct the FUNCTION field of the GIMMPIO parameter list

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44216I GIMMPMRM - THE IOPFUNCT IS UNKNOWN.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44217I GIMMPMRM - *modname* **+'** *offset* **' CALLED GIMMPIO FOR** *iopfunct* **FOR** *ioptype iopname***.**

Explanation:

modname name of the module that called GIMMPIO

offset the hexadecimal offset into that module

iopfunct IOPFUNCT value (such as BLDL, OPEN, READ)

ioptype IOPTYPE value (such as ASM, DLB, MOD)

iopname IOPNAME (element name or PTF number)

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44218T GIMMPMRM - NO VPLMSGP ERROR TEXT WAS RETURNED FROM GIMMPVIA.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44220T GIMMPMRM - THERE IS AN ERROR IN THE IOPTYPE VALUE PASSED TO GIMADTBL.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44221E GIMMPMRM - A DUPLICATE RECORD WAS DETECTED WHILE TRYING TO ADD TO THE *zonename* **ZONE.**

Explanation:

zonename zone name from the SET command

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44222I GIMMPVIA - THE FUNCTION REQUESTED (VPLFUNCT) IS NOT SUPPORTED.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44223I GIMMPVIA - THE VSAM ACCESS ID (VPLACID) IS INVALID.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44224I GIMMPVIA - THE CLUSTER IS NOT OPEN. ACCESS IS DENIED.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44225I GIMMPVIA - VSAM DATA SET *dataset* **IS OUT OF SPACE.**

Explanation:

dataset CSI data set name

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44227I GIMMPVIA - THE ZONE CANNOT BE OPENED. THE ZONE NAME (VPLZNAME) FOR THE SMPCSI MUST BE SMPCSI OR GLOBAL.

System Action: See previous related messages.

Programmer Response: Make sure the correct CSI data set was specified.

GIM44228I GIMMPVIA - THE *zonename* ZONE WAS ALREADY OPEN.

Explanation:
zonename name of zone to be opened

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44229I GIMMPVIA - ZONE *zonename* CANNOT BE OPENED FOR UPDATE BECAUSE THE PATH CONTAINING THE ZONE IS ALREADY OPENED WITH 'NOUPDATE'.

Explanation:
zonename name of zone to be opened

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44230I GIMMPVIA - THE PATH FOR THE ZONE CANNOT BE DETERMINED BECAUSE THE DATA SET NAME IS NOT AVAILABLE.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44231I GIMMPVIA - THE ACCESS ID IS INVALID FOR A ZONE OPEN FUNCTION.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44232I GIMMPVIA - DYNAMIC ALLOCATION FAILED FOR THE *zonename* ZONE, DATA SET *dataset*.

Explanation:
zonename zone name
dataset ddname of a data set required for processing in the indicated zone

System Action: See previous related messages.

Programmer Response: Add a DDDEF entry or DD statement for the indicated data set.

GIM44233I GIMMPVIA - A PREALLOCATED VSAM DATA SET WAS NOT FOUND FOR THE *zonename* ZONE.

Explanation:
zonename zone name

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44234I GIMMPVIA - SMPCSI DATA SET *dataset* IS BUSY.

Explanation:
dataset CSI data set name

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44236I GIMMPVIA - EITHER THE ACCESS ID (VPLACID) IS NOT ZERO, OR THE ZONE NAME (VPLZNAME) IS NOT BLANK. THIS COMBINATION IS INVALID FOR ZONE CLOSE.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44237I GIMMPVIA - A QUALIFIED ZONE MAY NOT HAVE A NAME OF SMPCSI OR GLOBAL. THIS IS A ZONE OPEN ERROR.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44238I GIMMPRPS - THE FUNCTION (RRLFUNCT) IS NOT RECOGNIZED.

System Action: See previous related messages.

Programmer Response: Report the problem to the IBM Support Center.

GIM44239I GIMMPRPS - THERE IS A LOGIC ERROR IN THE ASSERTION REQUEST. THIS ASSERTION IS UNSUPPORTED.

System Action: See previous related messages.

Programmer Response: Report the problem to the IBM Support Center.

GIM44240I GIMMPRPS - THE ZONE PATH ID IS INVALID.

System Action: See previous related messages.

Programmer Response: Report the problem to the IBM Support Center.

GIM44241I GIMMPRPS - THE CATEGORY IS INVALID.

System Action: See previous related messages.

Programmer Response: Report the problem to the IBM Support Center.

GIM44242I GIMMPRPS - A FREEPOOL ERROR OCCURRED. THE POOL IS MISSING.

System Action: See previous related messages.

Programmer Response: Report the problem to the IBM Support Center.

GIM44243I GIMMPRPS - THE RECORD TO BE DEASSIGNED IS MISSING.

System Action: See previous related messages.

Programmer Response: Report the problem to the IBM Support Center.

GIM44244W GIMVSMSG - NO REPRESENTATIONS MET INPUT REQUIREMENTS.

System Action: Processing continues.

Programmer Response: See previous related messages.

GIM44245E GIMVSMSG - THE MAIN RESOURCE IS NOT AVAILABLE.

Explanation: A required data set or zone is not available.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- If a data set was not available, make sure there are no missing DDDEF entries or DD statements, and that all data set names are specified correctly.
- If a zone was not available, find out why.

If necessary, rerun the job.

GIM44245S GIMVSMSG - THE MAIN RESOURCE IS NOT AVAILABLE.

Explanation: A required data set or zone is not available.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:

- If a data set was not available, make sure there are no missing DDDEF entries or DD statements, and that all data set names are specified correctly.
- If a zone was not available, find out why.

Then rerun the job.

GIM44246T GIMVSMSG - THERE IS A PROBABLE PHYSICAL I/O ERROR.

Explanation: A required device is not available.

System Action: SMP/E processing stops.

Programmer Response: Contact the IBM Support Center.

GIM44247T GIMVSMSG - THERE IS A PROBABLE LOGICAL I/O ERROR.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44248T GIMVSMSG - THE RETURN CODE FOR RRLFUNCT IS UNKNOWN.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44249S GIMVSMSG - AN UNKNOWN 'OUT OF SPACE' ERROR HAS OCCURRED.

Explanation: An error occurred for a data set required for processing. Module GIMMPVIA issued a return code of 12 but did not indicate that the CSI is busy or that dynamic allocation failed.

System Action: Command processing stops.

Programmer Response: Make sure there is a DDDEF entry or DD statement that defines the required data set.

GIM44250I GIMVSMSG - THE VSAM ERROR ANALYSIS OCCURRED FOR *modname* **+** *offset*, **RRLFUNCT='** *function* **', WITH RRLRETN='** *rtncode* **'.**

Explanation:

modname	module name
offset	offset into that module

function function value for RRLFUNCT
rtncode return code value for RRLRETN

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44250T GIMVSMSG - THE VSAM ERROR ANALYSIS OCCURRED FOR *modname* **+** *offset***, RRLFUNCT=‘** *function* **’, WITH RRLRETN=‘** *rtncode* **’.**

Explanation:
modname module name
offset offset into that module
function function value for RRLFUNCT
rtncode return code value for RRLRETN

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44251I GIMVSMSG - THERE IS NO AVAILABLE RRL MESSAGE TEXT.

Explanation: No information about the RRL error is available.

System Action: See previous related messages.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44252W GIMVSMSG - THE REPRESENTATION WAS NOT ADDED. A DUPLICATE EXISTS.

Explanation: SMP/E could not complete processing for the command being run.

System Action: See previous related messages.

Programmer Response: Make sure you specified the command correctly. If necessary, rerun the job.

GIM44253W GIMVSMSG - THE REPRESENTATION DID NOT MEET THE ASSERTION CRITERIA.

Explanation: The POSU (position for update) macro could not find a record.

System Action: Processing continues.

Programmer Response: Make sure the record to be processed actually exists. See previous related messages for more information.

GIM44253S GIMVSMSG - THE REPRESENTATION DID NOT MEET THE ASSERTION CRITERIA.

Explanation: The POSU (position for update) macro could not find a record.

System Action: Command processing stops.

Programmer Response: Make sure the record to be processed actually exists. See previous related messages for more information.

GIM44254W GIMVSMSG - NO REPRESENTATION WAS FOUND. THE END OF ASSERTION RANGE WAS SET USING POSITION.

Explanation: The EXTN (extract next) macro could not find a record.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44255W GIMVSMSG - THE REPRESENTATION WAS NOT DELETED. NONE EXISTED AT THE ASSERTED LEVEL.

Explanation: The BDEL (block delete) macro could not find anything to delete.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44256W GIMVSMSG - THE REPRESENTATION WAS NOT DELETED. NONE EXISTED IN THE ASSERTION GROUP.

Explanation: The EDEL (block delete with exclusions) macro could not find anything to delete.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44257S GIMVSMSG - A DUPLICATE STRING NAME WAS PREVIOUSLY ASSIGNED, OR THE CODE VALUES HAVE ALL BEEN ATTEMPTED.

Explanation: The ASGN (assign) macro failed for one of the indicated reasons.

System Action: Command processing stops.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44258E GIMVSMSG - A DISCONNECT FAILED. NO PRIOR CONNECTION WAS ESTABLISHED.

Explanation: A FREE (free) macro was attempted for a connection that was never made.

System Action: Command processing stops.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44259W GIMVSMSG - THE STRING IS UNKNOWN FOR THE ENCODE PROVIDED.

Explanation: A DECD (decode) macro was attempted and failed.

System Action: SYSMOD processing stops. See previous related messages for more information.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44260W GIMVSMSG - THE ENCODE IS UNKNOWN FOR THE STRING PROVIDED.

Explanation: An ENCD (encode) macro was attempted and failed.

System Action: SYSMOD processing stops. See previous related messages for more information.

Programmer Response: See previous related messages. If necessary, contact the IBM Support Center.

GIM44261W GIMVSMSG - ZONE *zonename* HAS NOT BEEN ASSIGNED.

Explanation:

zonename zone name

An attempt to open the zone failed.

System Action: See previous related messages.

Programmer Response: See previous related messages.

GIM44262E GIMVSMSG - NO ZONEINDEX EXISTS FOR THE ZONE.

System Action: Command processing stops.

Programmer Response: See previous related messages. If necessary, define a ZONEINDEX subentry for the zone.

GIM44263T GIMVSMSG - THERE WAS AN ERROR DURING AN OPEN OF THE ACB.

Explanation: One possible reason for the OPEN error is that the CSI data set may not be primed with a GIMZPOOL record.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44264I A VSAM ERROR OCCURRED IN MODULE *modname*. THE VSAM MACRO SHOWCB THAT REQUESTS THE *vsamcb* LENGTH FAILED.

Explanation:

modname an identifier for the module that issued this message

vsamcb name of a VSAM control block (ACB or RPL)

System Action: Processing continues.

Programmer Response: Report the error to the IBM Support Center.

GIM44264S A VSAM ERROR OCCURRED IN MODULE *modname*. THE VSAM MACRO SHOWCB THAT REQUESTS THE *vsamcb* LENGTH FAILED.

Explanation:

modname an identifier for the module that issued this message

vsamcb name of a VSAM control block (ACB or RPL)

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44265I A VSAM ERROR OCCURRED IN MODULE *modname*. THE VSAM MACRO GENCB THAT CREATES THE *vsamcb* FAILED.

Explanation:

modname an identifier for the module that issued this message

vsamcb name of a VSAM control block (ACB or RPL)

System Action: Processing continues.

Programmer Response: Report the error to the IBM Support Center.

GIM44265S A VSAM ERROR OCCURRED IN MODULE *modname*. THE VSAM MACRO GENCB THAT CREATES THE *vsamcb* FAILED.

Explanation:

modname an identifier for the module that issued this message

vsamcb name of a VSAM control block (ACB or RPL)

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44267S A VSAM ERROR OCCURRED IN MODULE *modname*. THE VSAM MACRO SHOWCB THAT REQUESTS THE KEY AND RECORD LENGTH OF A VSAM DATA SET FAILED.

Explanation:

modname an identifier for the module that issued this message

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44268S A VSAM ERROR OCCURRED IN MODULE *modname*. THE KEY LENGTH OF VSAM DATA SET *dataset* IS INVALID.

Explanation:

modname an identifier for the module that issued this message

dataset name of a CSI data set

System Action: Command processing stops.

Programmer Response: Specify a CSI data set that contains a valid key length. Then rerun the job.

GIM44269S A VSAM ERROR OCCURRED IN MODULE *modname*. THE RECORD LENGTH OF VSAM DATA SET *dataset* IS INVALID.

Explanation:

modname an identifier for the module that issued this message

dataset name of a CSI data set

System Action: Command processing stops.

Programmer Response: Specify a CSI data set that contains a valid record length. Then rerun the job.

GIM44270S A VSAM ERROR OCCURRED IN MODULE *modname*. *command* FAILED FOR VSAM DATA SET *dataset*.

Explanation:

modname an identifier for the module that issued this message

command ENDREC or VERIFY

dataset name of a CSI data set

System Action: Command processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44271S A VSAM ERROR OCCURRED IN MODULE *modname*. THE VSAM MACRO MODCB THAT MODIFIES THE *vsamcb* FAILED.

Explanation:

modname an identifier for the module that issued this message

vsamcb name of a VSAM control block (ACB or RPL)

System Action: Command processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44272S A VSAM ERROR OCCURRED IN MODULE *modname*. THE VSAM MACRO POINT FAILED TRYING TO POSITION TO THE BEGINNING OF VSAM DATA SET *dataset*.

Explanation:

modname an identifier for the module that issued this message

dataset name of a CSI data set

System Action: Command processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44273S A VSAM ERROR OCCURRED IN MODULE *modname*. A FAILURE OCCURRED TRYING TO GET THE NEXT RECORD FROM VSAM DATA SET *dataset*.

Explanation:

modname an identifier for the module that issued this message

dataset name of a CSI data set

System Action: Command processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44276I GIMMPVIA - A VSAM OPEN GETMAIN FAILED FOR AN ACB/RPL.

System Action: See previous related messages.

Programmer Response: Make sure the REGION size specified for SMP/E is large enough.

GIM44279I GIMMPVIA - VSAM OPEN FAILED. THE REASON CODE IS *rsncode*.

Explanation:

rsncode VSAM reason code for OPEN failure

System Action: See previous related messages.

Programmer Response: Make sure the REGION size specified for SMP/E is large enough. Your action depends on the reason code specified in the message. If necessary, contact the IBM Support Center.

GIM44280I GIMMPVIA - VSAM OPEN TERMINATED. A DATA SET IS EMPTY THAT REQUIRES AN INITIAL LOAD.

System Action: See previous related messages.

Programmer Response: Make sure you have primed the CSI data set with the GIMZPOOL record.

GIM44281I GIMMPVIA - A VSAM OPEN SHOWCB MACRO FAILED OBTAINING INITIAL STATISTICS. THE REASON CODE IS *rsncode*.

Explanation:

rsncode VSAM reason code for SHOWCB failure

System Action: See previous related messages.

Programmer Response: See previous related messages. Your action depends on the reason code specified in the message. If necessary, contact the IBM Support Center.

GIM44282I GIMMPVIA - A VSAM OPEN SHOWCB MACRO FAILED OBTAINING INDEX CI (CONTROL INTERVAL) SIZE. THE REASON CODE IS *rsncode*.

Explanation:

rsncode VSAM reason code for SHOWCB failure

System Action: See previous related messages.

Programmer Response: See previous related messages. Your action depends on the reason code specified in the message. If necessary, contact the IBM Support Center.

GIM44283I GIMMPVIA - A VSAM OPEN SHOWCB MACRO FAILED OBTAINING DATA CI (CONTROL INTERVAL) SIZE. THE REASON CODE IS *rsncode*.

Explanation:

rsncode VSAM reason code for SHOWCB failure

System Action: See previous related messages.

Programmer Response: See previous related messages. Your action depends on the reason code specified in the message. If necessary, contact the IBM Support Center.

GIM44284I GIMMPVIA - A VSAM OPEN FAILED WITH THE FOLLOWING INVALID: *keyword*

Explanation:

keyword KEYLENGTH, LRECL, or both

System Action: See previous related messages.

Programmer Response: Check the job used to allocate the CSI. Make sure the correct KEYLENGTH and LRECL values were used to define the CSI.

GIM44301E AN UNUSUAL CONDITION OCCURRED. *modname* - *elmtype elmname* TO BE RETRANSFORMED WAS NOT IN TRANSFORMED FORMAT.

Explanation:

modname an identifier for the module that issued this message
elmtype element type
elmname element name

SMP/E expects transformed elements to be in the format produced by service routine GIMDTS. However, the indicated element was not in the expected format.

System Action: SYSMOD processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44302E AN UNUSUAL CONDITION OCCURRED. *modname* - *elmtype elmname* TO BE RETRANSFORMED IS NOT ON THE SMPWRK6 DATA SET.

Explanation:

modname an identifier for the module that issued this message
elmtype element type
elmname element name

Normally, when SMP/E applies or accepts a transformed data element that was packaged inline, it copies the element to the SMPWRK6 data set before

processing it. However, when SMP/E was ready to process the indicated element, it was not on the SMPWRK6 data set.

System Action: SYSMOD processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44304S AN UNUSUAL CONDITION OCCURRED. *modname* - AN INVALID VALUE WAS SPECIFIED ON THE OPT PARAMETER PASSED TO MODULE GIMZNDEL.

Explanation:

modname an identifier for the module that issued this message

The OPT parameter specifies an incorrect value. This is probably due to an internal processing error that caused it to be overlaid.

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44305T AN UNUSUAL CONDITION OCCURRED. *modname* - THE RETURN CODE FOR A UCL DEL STATEMENT INDICATED THAT NOTHING WAS DELETED. THE STATEMENT WAS CHECKED IN *modname*.

Explanation:

modname an identifier for the module that issued this message

UCLIN processing tried to delete a subentry but could not find it. However, a previous check showed that the subentry existed.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44306E AN UNUSUAL CONDITION OCCURRED. *modname* - THE RETURN CODE IS NOT ZERO FOR THE VALUE BEING CHECKED IN THE DECODE TABLE. HOWEVER, THAT VALUE WAS PREVIOUSLY ACCEPTED AS VALID BY CHKRTN PROCESSING.

Explanation:

modname an identifier for the module that issued this message

SMP/E was checking a value against the decode table for data elements. Because the return code was not zero, it seems that the value was not found in the table. However, a previous check by CHKRTN processing showed that the value was in the table.

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44307E AN UNUSUAL CONDITION OCCURRED. *modname* - THE RETURN CODE IS NOT ZERO FOR THE VALUE BEING CHECKED IN THE DECODE TABLE. HOWEVER, THAT VALUE WAS PREVIOUSLY ACCEPTED AS VALID.

Explanation:

modname an identifier for the module that issued this message

SMP/E was checking a value against the decode table for data elements. Because the return code was not zero, it seems that the value was not found in the table. However, a previous check showed that the value was in the table.

System Action: SYSMOD processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44307T AN UNUSUAL CONDITION OCCURRED. *modname* - THE RETURN CODE IS NOT ZERO FOR THE VALUE BEING CHECKED IN THE DECODE TABLE. HOWEVER, THAT VALUE WAS PREVIOUSLY ACCEPTED AS VALID.

Explanation:

modname an identifier for the module that issued this message

SMP/E was checking a value against the decode table for data elements. Because the return code was not zero, it seems that the value was not found in the table. However, a previous check showed that the value was in the table.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44308T AN UNUSUAL CONDITION OCCURRED. *modname* - THE RETURN CODE FOR A UCL ADD STATEMENT INDICATED THAT NOTHING WAS ADDED. THE STATEMENT WAS CHECKED IN *modname*.

Explanation:

modname an identifier for the module that issued this message

UCLIN processing tried to add a subentry but a duplicate subentry exists. However, a previous check showed that the subentry did not exist.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44309E AN UNUSUAL CONDITION OCCURRED. *modname* - AN ERROR OCCURRED IN MODULE GIMMPREC. THE MCS ENTRY WAS NOT FOUND IN A CALL TO GIMADTBL.

Explanation:

modname	an identifier for the module that issued this message

An erroneous parameter was passed to GIMADTBL. This is probably due to an internal processing error that caused the parameter to be overlaid.

System Action: SYSMOD processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44310I AN UNUSUAL CONDITION OCCURRED. *modname1* - GIMDSMGR WAS CALLED FROM *modname2* + *offset*.

Explanation:

modname1	an identifier for the module that issued this message
modname2	an identifier for the module that called GIMDSMGR
offset	offset from the beginning of module *modname2*

This message accompanies other GIM443*xx* messages that were issued during processing by GIMDSMGR.

System Action: See the other GIM443*xx* messages issued with this message.

Programmer Response: See the other GIM443*xx* messages issued with this message.

GIM44311T AN UNUSUAL CONDITION OCCURRED. *modname* - THE FIRST PHASE PROCESSED IS NOT PHASE 1 FOR COMMAND *command*.

Explanation:

modname	an identifier for the module that issued this message
command	an SMP/E command

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310I is also issued to identify this module. Otherwise, message GIM44318I is issued along with this message.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44312T AN UNUSUAL CONDITION OCCURRED. *modname* - THE CURRENT PHASE SHOULD HAVE PRECEDED THE PREVIOUS PHASE FOR COMMAND *command*.

Explanation:

modname	an identifier for the module that issued this message
command	an SMP/E command

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310I is also issued to identify this module. Otherwise, message GIM44318I is issued along with this message.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44313T AN UNUSUAL CONDITION OCCURRED. *modname* - THE CLEANUP PHASE WAS NOT DONE FOR THE PREVIOUS COMMAND, *command1*. THE CURRENT COMMAND IS *command2*.

Explanation:

modname	an identifier for the module that issued this message
command1	an SMP/E command
command2	an SMP/E command

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310I is also issued to identify this module. Otherwise, message GIM44318I is issued along with this message.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44314T AN UNUSUAL CONDITION OCCURRED. *modname* - SMP/E COULD NOT GET EXTRA RESOURCES FOR COMMAND *command* BECAUSE THEY WERE HELD WITHOUT A REASON.

Explanation:

modname	an identifier for the module that issued this message
command	an SMP/E command

An abnormal condition occurred because of an internal processing error. If SMP/E can determine which module called GIMDSMGR, message GIM44310I is also issued to identify this module. Otherwise, message GIM44318I is issued along with this message.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44315T AN UNUSUAL CONDITION OCCURRED. *modname* - THE NAME OF THE DATA SET TO BE UPDATED WAS NOT PROVIDED.

Explanation:

modname	an identifier for the module that issued this message

An abnormal condition occurred because of an internal processing error. If SMP/E can determine which module called GIMDSMGR, message GIM44310I is also issued to identify this module. Otherwise, message GIM44318I is issued along with this message.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44316T AN UNUSUAL CONDITION OCCURRED. *modname* - THE CURRENT PHASE FOR COMMAND *command* IS UNKNOWN.

Explanation:

modname	an identifier for the module that issued this message
command	an SMP/E command

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310I is also issued to identify this module. Otherwise, message GIM44318I is issued along with this message.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44317T AN UNUSUAL CONDITION OCCURRED. *modname* - THE *phase1* PHASE WAS ATTEMPTED FOR COMMAND *command* BEFORE THE *phase2* PHASE WAS PROCESSED.

Explanation:

modname	an identifier for the module that issued this message
phase1	a phase of command processing
command	an SMP/E command
phase2	a phase of command processing

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310I is also issued to identify this module. Otherwise, message GIM44318I is issued along with this message.

System Action: SMP/E processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM44318I AN UNUSUAL CONDITION OCCURRED. *modname* - SMP/E COULD NOT DETERMINE WHICH MODULE CALLED GIMDSMGR.

Explanation:

modname	an identifier for the module that issued this message

This message accompanies other GIM443*xx* messages that were issued during processing by GIMDSMGR.

System Action: See the other GIM443*xx* messages issued with this message.

Programmer Response: See the other GIM443*xx* messages issued with this message.

GIM44319S AN UNUSUAL CONDITION OCCURRED. *modname* - AN ERROR HAS OCCURRED IN SUBROUTINE *subroutine*. AN UNKNOWN ENTRY OR SUBENTRY TYPE WAS FOUND IN A CALL TO MODULE GIMADTBL.

Explanation:

modname	an identifier for the module that issued this message.
subroutine	the subroutine name. If there is no subroutine, the module name is repeated.

This message may be issued if you are running the LIST command to obtain a list of entries for data elements or hierarchical file system elements, and if the level of SMP/E you are running is at a lower level than what was used to install the requested elements. Because the elements were added by a later release level of SMP/E, they are not recognized at a lower level.

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44319T AN UNUSUAL CONDITION OCCURRED. *modname* - AN ERROR HAS OCCURRED IN SUBROUTINE *subroutine*. AN UNKNOWN ENTRY OR SUBENTRY TYPE WAS FOUND IN A CALL TO MODULE GIMADTBL.

Explanation:

modname an identifier for the module that issued this message.

subroutine the subroutine name. If there is no subroutine, the module name is repeated.

This message may be issued if you are running the LIST command to obtain a list of entries for data elements or hierarchical file system elements, and if the level of SMP/E you are running is at a lower level than what was used to install the requested elements. Because the elements were added by a later release level of SMP/E, they are not recognized at a lower level.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44320E AN UNUSUAL CONDITION OCCURRED. *modname1* – AN INVALID PARAMETER WAS PASSED TO MODULE *modname2*.

Explanation:

modname1 an identifier for the module that called the module

modname2 an identifier for the module that issued this message

A parameter that was passed to the indicated module was either overlaid or invalid.

System Action: SYSMOD processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44320S AN UNUSUAL CONDITION OCCURRED. *modname* - AN INVALID PARAMETER WAS PASSED TO MODULE *modname*.

Explanation:

modname an identifier for the module that issued this message

A parameter that was passed to the indicated module was overlaid.

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44320T AN UNUSUAL CONDITION OCCURRED. *modname* - AN INVALID PARAMETER WAS PASSED TO MODULE *modname*.

Explanation:

modname an identifier for the module that issued this message

A parameter that was passed to the indicated module was overlaid.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44321T AN UNUSUAL CONDITION OCCURRED. *modname* - THE CALL FOR ANALYSIS SHOULD NOT HAVE BEEN MADE BECAUSE RRLRETN IS ZERO.

Explanation:

modname an identifier for the module that issued this message

A call to do error analysis was made. However, the previous return code was 0, which indicated that there was no error to process.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44322S AN UNUSUAL CONDITION OCCURRED. *modname* - AN ERROR HAS OCCURRED IN SUBROUTINE XRFPTF1. A RELATION ENTRY WAS NOT FOUND IN A CALL TO MODULE GIMADTBL.

Explanation:

modname an identifier for the module that issued this message

There is no relation entry for the specified element type.

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44323T AN UNUSUAL CONDITION OCCURRED. *modname* - GIMUTTBL IS TOO SMALL – LESS THAN 10 BYTES.

Explanation:

modname an identifier for the module that issued this message

There is a construction error in module GIMUTTBL.

System Action: SMP/E processing stops.

Programmer Response: Use macro GIMDFUT to replace GIMUTTBL. For more information, see the GIMUTTBL chapter in *SMP/E Reference*.

GIM44324T AN UNUSUAL CONDITION OCCURRED. *modname* - GIMUTTBL DOES NOT HAVE AN END-OF-LIST ENTRY.

Explanation:

modname	an identifier for the module that issued this message

There is a construction error in module GIMUTTBL.

System Action: SMP/E processing stops.

Programmer Response: Use macro GIMDFUT to replace GIMUTTBL. For more information, see the GIMUTTBL chapter in *SMP/E Reference*.

GIM44325I AN UNUSUAL CONDITION OCCURRED. *modname* - SMP/E COULD NOT ADD THE FMID OF SYSMOD *sysmod* TO THE GLOBAL ZONE, BECAUSE OF A VSAM ERROR.

Explanation:

modname	an identifier for the module that issued this message
sysmod	SYSMOD ID

When SMP/E receives a function SYSMOD, it adds the FMID of the function to the GLOBALZONE entry. However, SMP/E could not add the FMID of the indicated SYSMOD, either because the SMPCSI that contains the global zone is not available, or because a physical or logical I/O error occurred.

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44326I AN UNUSUAL CONDITION OCCURRED. *modname* - SMP/E COULD NOT FIND REGRESSED SYSMOD *sysmod* IN THE *zonetype* ZONE.

Explanation:

modname	an identifier for the module that issued this message
sysmod	SYSMOD ID of the SYSMOD being regressed
zonetype	TARGET or DLIB

The indicated SYSMOD was regressed by another SYSMOD that was installed. However, SMP/E could not find the regressed SYSMOD in the zone being processed. Use the SYSMOD Regression report to determine what elements are being regressed and which SYSMOD caused the regression.

System Action: Processing continues.

Programmer Response: The programmer response depends on why the message was issued.

- Except for the case described below, you should report the error to the IBM Support Center.
- If the message was issued during APPLY processing, a SYSMOD existing only in the distribution zone may have been regressed. For example, the RESTORE command may have introduced the distribution zone version of an element into the target libraries. Later, the restored SYSMOD may be reapplied with the BYPASS(ID) operand, regressing the SYSMOD that provided the distribution zone version of the affected element. If the regressed SYSMOD had never been applied to the target zone, SMP/E issues this message.

  In this case, you need to use the distribution zone entry for the regressed SYSMOD and the target zone entry for the SYSMOD causing the regression to determine whether any additional SYSMODs were regressed. Take the necessary actions to correct the regressed SYSMODs. For example, if a USERMOD is regressed, rework it to specify the appropriate relationship to the regressing SYSMOD (such as PRE), then reapply the USERMOD.

GIM44327T AN UNUSUAL CONDITION OCCURRED. *modname* - *action* WAS ATTEMPTED FOR A NONEXISTENT ENTRY - *entname*.

Explanation:

modname	an identifier for the module that issued this message
action	EXTEND, POSITION, or READ
entname	entry name

SMP/E tried to do the indicated type of processing but it could not find the entry to be processed. GIM44328 may also be issued to provide additional information.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44328I AN UNUSUAL CONDITION OCCURRED. *issuer* - GIMMPIO WAS CALLED FROM *modname* + *offset* WITH IOPFUNCT = *iopfunct*, IOPTYPE = *enttype* AND IOPNAME = *entname*.

Explanation:

issuer	an identifier for the module that issued this message
modname	name of the module that called GIMMPIO
offset	offset from the beginning of module *modname*
iopfunct	I/O function or UNKNOWN
enttype	entry type or UNKNOWN
entname	entry name

This message provides additional information about the error reported by message GIM44327.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44329E AN UNUSUAL CONDITION OCCURRED. *modname* - LMOD ENTRY *entname* HAS NO VALID SYSLIB VALUE.

Explanation:

modname	an identifier for the module that issued this message
entname	entry name

The indicated LMOD entry either contains no SYSLIB value or contains an incorrect SYSLIB value.

System Action: SYSMOD processing stops.

Programmer Response: Correct the SYSLIB value in the LMOD entry and rerun the command.

GIM44330T AN UNUSUAL CONDITION OCCURRED. *modname* - A RECORD IN THE *enttype* ENTRY CONTAINS AN UNKNOWN RELATIONSHIP. THE KEY OF THE RECORD IS *key*.

Explanation:

modname	an identifier for the module that issued this message
enttype	entry type
key	key of the record (in hexadecimal)

SMP/E was checking the records in the indicated entry and found a record that specified an unknown relationship.

System Action: SMP/E processing stops.

Programmer Response: Do **one** of the following:

- If you have run a higher level of SMP/E against the zone containing the indicated entry, it is likely that new information has been introduced that the level of SMP/E issuing the message does not recognize. Run the higher level of SMP/E and access the named entry. If no error occurs, you must now use the higher level of SMP/E to process the zone containing the new information. For more information on the compatibility of entries between SMP/E releases, see the Migration chapter in *SMP/E User's Guide*.
- Otherwise, report the situation to the IBM Support Center.

GIM44331W AN UNUSUAL CONDITION OCCURRED. *modname* - SYSMOD *sysmod* COULD NOT BE FOUND IN THE *zonetype* ZONE.

Explanation:

modname	an identifier for the module that issued this message
sysmod	SYSMOD ID
zonetype	type of zone (target or distribution)

SYSMOD *sysmod* has elements in common with a SYSMOD that failed (and was named in a GIM302*xx* message). However, SYSMOD *sysmod* could not be found in the indicated type of zone.

System Action: SYSMOD processing stops. Processing continues with the next SYSMOD.

Programmer Response: None.

GIM44332T AN UNUSUAL CONDITION OCCURRED. *modname1* - *modname2* WAS CALLED FROM *modname3* DURING *command* COMMAND PROCESSING.

Explanation:

modname1	an identifier for the module that issued this message
modname2	an identifier for the module that was called in error
modname3	an identifier for the module that called *modname2*
command	the command being processed at the time of the failure

An abnormal condition occurred because of an internal processing error. The identified module calling sequence should not have occurred during processing of the specified command.

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44333T AN UNUSUAL CONDITION OCCURRED. THE *function* FUNCTION CANNOT BE PROCESSED BECAUSE THE *ddname* DD SPECIFIES A PATH.

Explanation:

function	the requested I/O function
ddname	ddname specified by the I/O request

The specified I/O function cannot be processed, because the target DD of the operation is a path in a hierarchical file system.

System Action: SMP/E processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44334S AN UNUSUAL CONDITION OCCURRED. MODULE GIMZNDEL HAD A REQUEST TO DELETE THE GLOBAL ZONE FROM A COMMAND OTHER THAN ZONEIMPORT.

Explanation: Module ZIMZNDEL had a request to delete the global zone from a command other than ZONEIMPORT.

System Action: Command processing stops.

Programmer Response: Report the error to the IBM Support Center.

GIM44335E *operation* **FAILED FOR** *elmtype elmname* **IN SYSMOD** *sysmod*. **THE RETURN CODE FROM THE** *operation* **OPERATION WAS** *rtncode*.

Explanation:

operation	COMPACTION or EXPANSION
elmtype	element type
elmname	element name
sysmod	SYSMOD ID
rtncode	return code

SMP/E could not expand or compact the indicated element within SYSMOD *sysmod*. The *operation* operation failed.

System Action: SYSMOD processing stops.

Programmer Response: Report the error to your support group.

GIM44401W *asmname* **WAS NOT ASSEMBLED FOR** *enttype entname* **IN SYSMOD** *sysmod* **BECAUSE SMP/E COULD NOT FIND ASSEMBLER INPUT FOR** *asmname*.

Explanation:

asmname	name of an ASSEM entry or MOD entry
enttype	MAC or SRC
entname	name of the MAC or SRC entry used for the assembly
sysmod	ID of the SYSMOD that contains the MAC or SRC

There are several reasons this message is issued:

- Because of a macro update or replacement, SMP/E looked for the source that must be assembled to create the module. However, it could not find an ASSEM entry or SRC entry for the source. Therefore, it assumed that the module created from the assembly does not belong on your system.
- SMP/E found a SRC entry that did not contain an FMID and an RMID. As a result, SMP/E assumes that the source has not been installed.

System Action: SMP/E processing continues.

Programmer Response: Specify the assembler input by doing **one** of the following:

- If you are assembling a macro and the assembler input is in a library, an SRC entry, or an ASSEM entry, add the ASSEM and DISTSRC operands to the element MCS.
- Create a target or distribution zone ASSEM entry that creates the assembler input.
- Create a target or distribution zone SRC entry that creates the assembler input.

Then reinstall the SYSMOD.

GIM44402W *asmname* **WAS NOT ASSEMBLED FOR** *enttype entname* **IN SYSMOD** *sysmod* **BECAUSE THERE IS NO MOD ENTRY FOR** *asmname*. **SMP/E CANNOT DETERMINE THE TARGET LIBRARY FOR THE ASSEMBLER OUTPUT.**

Explanation:

asmname	name of an ASSEM entry or MOD entry
enttype	MAC or SRC
entname	name of the MAC or SRC entry used for the assembly
sysmod	ID of the SYSMOD that contains the MAC or SRC

The MAC or SRC entry being assembled does not indicate a target library. SMP/E assumed there was a MOD entry associated with the assembled output and that the MOD entry would indicate a target library. However, SMP/E could not find the indicated MOD entry. Therefore, it assumed that the module created from the assembly does not belong on your system.

System Action: SMP/E processing continues.

Programmer Response: Define a target or distribution zone MOD entry for the module. Then rerun the job.

GIM44500I VERIFICATION OF HASH VALUE OF FILE *filename* **FAILED. SMP/E WILL RETRY RETRIEVAL OF THE FILE.**

Explanation:

filename	name of file

After the identified file was transferred by SMP/E, the computed hash value for the file did not match the expected hash value. This could signal an error during transmission of the file.

System Action: SMP/E will retry the file transfer.

Programmer Response: None.

GIM44501W CHANGES FOR THE FOLLOWING USERMODS OR APAR FIXES WILL BE LOST BECAUSE SYSMOD *sysmod* IS BEING INSTALLED.

Explanation:

sysmod SYSMOD ID

When SMP/E was installing the indicated SYSMOD, SMP/E detected changes from USERMODs or APAR fixes that will be replaced by that SYSMOD. For example, you may get this message if a PTF (1) replaces an element that was last changed by a USERMOD applicable to another function and (2) specifies the VERSION operand to change the owner of the element.

System Action: SYSMOD processing continues. Message GIM44601I follows this message and lists the USERMODs and APAR fixes whose changes were replaced.

Programmer Response: None.

GIM44502W CHANGES FOR THE FOLLOWING USERMODS WILL BE LOST BECAUSE THE ASSOCIATED FUNCTION SYSMOD HAS BEEN DELETED.

Explanation:

When one function deletes another function, all the SYSMODs associated with the deleted function are also deleted. Although changes made by PTFs and APAR fixes may be incorporated into the deleting function, changes made by USERMODs are not and may be lost.

System Action: SYSMOD processing continues. Message GIM44601I follows this message and lists the USERMODs whose changes were replaced.

Programmer Response: None.

GIM44601I *elmtype sysmod* **IN** *elmtype elmname***.**

Explanation:

elmtype APAR or USERMOD
sysmod SYSMOD ID
elmtype element type
elmname element name

This message lists the USERMODs and APAR fixes whose changes have been replaced. The reason is given in message GIM44501W or GIM44502W.

System Action: None.

Programmer Response: None.

GIM44701I MODULE *modname* IN SYSMOD *sysmod* WAS NOT REASSEMBLED BECAUSE THE REUSE OPERAND WAS SPECIFIED. THE OBJECT MODULE SAVED IN SMPWRK3 WAS USED INSTEAD.

Explanation:

modname module name
sysmod SYSMOD ID

The indicated module was first assembled when SMP/E unsuccessfully tried to install SYSMOD *sysmod*. The SYSMOD is being reinstalled, and the REUSE operand was specified. As a result of the REUSE option, two things happen:

- The indicated module is not reassembled.
- The corresponding object module that was saved on SMPWRK3 is used instead.

System Action: SYSMOD processing continues.

Programmer Response: None.

GIM44801I *modname* **WAS ASSEMBLED BECAUSE THE ASSEMBLE SUBENTRY WAS SET IN THE MOD ENTRY.**

Explanation:

modname module name

The ASSEMBLE indicator is set in the MOD entry for the indicated module. As a result, whenever SMP/E processes a SYSMOD that affects this module, the module is reassembled. This assembly prevents regression of changes supplied by one of the following:

- A SYSMOD that provides macro changes for the module
- The SYSMODs that replace the module but do not have the macro changes.

The assemble indicator may have been set by SMP/E when processing an APAR or USERMOD containing a macro or when processing an assembly caused by a GENASM subentry on a macro.

System Action: Processing continues.

Programmer Response: Do **one** of the following:

- If the module no longer needs to be reassembled, use UCLIN to remove the ASSEMBLE indicator from the MOD entry.
- Otherwise, take no action.

GIM44802I *modname* **WAS ASSEMBLED BECAUSE ITS NAME MATCHES THE PREFIX VALUE ON A ++MAC OR ++MACUPD STATEMENT.**

Explanation:

modname module name

The PREFIX operand was specified on a ++MAC or ++MACUPD MCS. The PREFIX operand specifies the first characters of the names of modules that should be assembled when the macro change is installed. The indicated module was assembled because its name starts with the specified PREFIX value.

System Action: Processing continues.

Programmer Response: None.

GIM44803I *modname* **WAS ASSEMBLED BECAUSE IT WAS INCLUDED IN A NEW LOAD MODULE DURING LOAD MODULE BUILD PROCESSING.**

Explanation:

modname	module name

System Action: Processing continues.

Programmer Response: None.

GIM44901W **SMP/E ADDED THE** *subname* **SUBENTRY TO CREATE A VALID ENTRY.**

Explanation:

subname	name of the subentry that was added

After SMP/E processed the UCL statement, SMP/E checked data in the entry. If SMP/E had not added the subentry, the entry would have been invalid.

System Action: SMP/E adds the indicated subentry to the entry, and processing continues with the next command.

Programmer Response: Check whether the updated entry is correct. If necessary, correct the UCL statement and rerun it. These are some examples:

- If you are deleting the ACC subentry, you must also delete the ACCDATE subentry.
- SMP/E will not delete the RMID if an FMID is present in an element entry.

GIM45001I **AN I/O ERROR OCCURRED FOR** *enttype entname* **ON THE** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library

This message follows message GIM274*xx* when SMP/E can determine the entry that caused the I/O error.

System Action: SMP/E processing stops.

Programmer Response: Fix the I/O error and rerun the job. Here is one way to fix the error:

1. Use UCLIN to delete the entry that caused the I/O error.
2. Rebuild the entry with UCLIN.

Note: This may not always work because UCL processing may also get an I/O error when it tries to delete the entry.

GIM45101I **LOG RECORDING WAS SWITCHED TO** *dataset1* **BECAUSE** *dataset2* **IS FULL.**

Explanation:

dataset1	ddname of the LOG data set that is full
dataset2	ddname of the alternative LOG data set

Because the indicated LOG data set is full, SMP/E is writing LOG messages to an alternative data set.

System Action: SMP/E processing continues.

Programmer Response: Reallocate a larger SMPLOG or SMPLOGA data set for subsequent SMP/E runs.

GIM45200I *ftpcmd*

Explanation:

ftpcmd	FTP command sent to the server

This message echoes the commands sent by SMP/E to the FTP server.

System Action: Command processing continues.

Programmer Response: None.

GIM45201I *ftpreply*

Explanation:

ftpreply	reply from FTP server

This message echoes the reply received by SMP/E from the FTP server.

System Action: Command processing continues.

Programmer Response: None.

GIM45201S *ftpreply*

Explanation:

ftpreply	reply from FTP server

This message echoes the reply received by SMP/E from the FTP server. SMP/E has interpreted this reply as an error.

System Action: Command processing stops.

Programmer Response: Fix the problem identified in the reply and rerun the command.

GIM45301E DATA IS MISSING FROM THE ++HOLD FOR SYSMOD *sysmod*. *data* IS REQUIRED.

Explanation:

sysmod	SYSMOD ID
data	FMID, REASON, or TYPE

The ++HOLD MCS is missing one of the following:
- A hold type (ERROR, SYSTEM, or USER)
- The FMID operand
- The REASON operand

System Action: SYSMOD processing stops. Processing continues with the next SYSMOD.

Programmer Response: Provide the missing data and resubmit the job.

GIM45302E DATA IS MISSING FROM THE ++*mcstype* MCS FOR *mcsname*. *operand* IS REQUIRED.

Explanation:

mcstype	type of MCS (PRODUCT or FEATURE)
mcsname	name for the MCS (product-id and vrm, or feature name)
operand	missing MCS operand

The specified MCS is missing a required operand. For ++PRODUCT, either DESCRIPTION or SREL is missing. For ++FEATURE, either DESCRIPTION or PRODUCT is missing.

System Action: The indicated MCS is not received. Processing continues with the next MCS.

Programmer Response: Provide the missing operand and rerun the job.

GIM45401W *elmtype elmname* IN SYSMOD *sysmod* WAS NOT PROCESSED BECAUSE THE FMID ON THE ++VER STATEMENT DOES NOT MATCH THE FMID IN THE *enttype* ELEMENT ENTRY.

Explanation:

elmtype	APAR or USERMOD
elmname	element name
sysmod	SYSMOD ID
enttype	entry type

System Action: SYSMOD processing continues.

Programmer Response: Check that the FMID value specified in the SYSMOD being installed is the correct value for all the elements in the SYSMOD.

GIM45500S VERIFICATION OF HASH VALUE OF FILE *filename* FAILED. SMP/E WILL NOT RETRY FILE RETRIEVAL.

Explanation:

filename	name of file

After the identified file was transferred by SMP/E, the computed hash value for the file did not match the expected hash value. This could signal an error during transmission of the file. SMP/E makes no more attempts at transferring the file.

System Action: Command processing stops.

Programmer Response: Contact the software provider to verify the integrity of identified file on the provider's server.

GIM45601W THE SMPTLIB DDDEF VOLUME SUBENTRY *subname1* WAS NOT UPDATED BECAUSE VOLUME SUBENTRY *subname2* ALREADY EXISTS IN THE VOLUME LIST.

Explanation:

subname1	an existing volume subentry value
subname2	the new volume subentry value

A ZONEEDIT CHANGE statement tried to change a volume subentry in the SMPTLIB entry to an already existing volume subentry.

System Action: SYSMOD processing continues.

Programmer Response: Do **one** of the following:
- If you want to use the ZONEEDIT command to change the indicated volume, specify a new value that does not currently exist in the SMPTLIB entry. Then rerun the command.
- If you want to delete the indicated volume, use the UCLIN command.
- If you do not want to change the volume, take no action.

GIM45700S SMP/E COULD NOT RESOLVE SERVER HOST NAME *hostname* TO AN IP ADDRESS.

Explanation:

hostname	Host name specified in SERVER data set

An error occurred when SMP/E attempted to resolve the specified host name to an IP address. Either the TCP/IP GETHOSTBYNAME service could not be loaded or it could not resolve the host name to an IP address.

System Action: RECEIVE FROMNETWORK processing terminates.

Programmer Response: Specify the host in the

SERVER data set in dotted decimal format and rerun the RECEIVE.

GIM45800S PACKAGE *package* WAS NOT FOUND IN THE SMPNTS.

Explanation:
package package id

SMP/E could not locate the request package in the SMPNTS directory.

System Action: Command processing stops.

Programmer Response: Check that you have specified the correct package id.

GIM45900S PACKAGE *package* DOES NOT CONTAIN *datatype* DATA.

Explanation:
package package id
datatype SMPPTFIN or SMPHOLD

The options you have specified on the RECEIVE command require SMPPTFIN data, SMPHOLD data, or both. The identified package does not contain that data.

System Action: Command processing stops.

Programmer Response: Check that you have specified the correct package id and or the correct RECEIVE command operands.

GIM46000S AN ERROR OCCURRED WHILE SMP/E WAS EXPLODING MEMBER *member* IN THE ARCHIVE FILE *filename* FROM PACKAGE *package*.

Explanation:
member member in the archive file
filename archive file name
package package id

SMP/E encountered an error while trying to expand the identified member from the identified archive file in the identified package.

System Action: Command processing stops.

Programmer Response: Check the SYSPRINT data set to determine what error was encountered while expanding the member.

GIM46100I PACKAGE *package* HAS BEEN DELETED FROM THE SMPNTS.

Explanation:
package package id

SMP/E deleted the identified package.

System Action: Command processing continues.

Programmer Response: None.

GIM46200I RECEIVE PROCESSING HAS FAILED BECAUSE THE FTP SERVER DETECTED AN ERROR.

Explanation: SMP/E received a reply from the FTP server during a RECEIVE FROMNETWORK operation that SMP/E has interpreted as an error.

System Action: RECEIVE command processing stops.

Programmer Response: Refer to the preceding GIM45201S message to determine the error.

GIM46700I *progname* PACKAGE CONTROL STATEMENT PROCESSING FAILED.

Explanation:
progname program name (GIMZIP or GIMUNZIP)

An error occurred during processing of the Package Control Statements for the GIMZIP or GIMUNZIP routine.

System Action: A prior message identifies the system action. GIMZIP or GIMUNZIP processing stops.

Programmer Response: Fix the error and retry the operation.

GIM46701I PROCESSING HAS FAILED FOR THE *dataset* DATA SET ALLOCATED TO THE *ddname* DDNAME.

Explanation:
dataset data set being parsed
ddname ddname

An error occurred during processing of the SERVER or CLIENT data set. Previous messages describe the specific error.

The SERVER data set contains information about a TCP/IP connected host running an FTP server. The CLIENT data set contains information about the TCP/IP Client environment of the local machine.

System Action: Processing stops.

Programmer Response: Correct the error and rerun the job.

GIM46702I PROCESSING HAS FAILED FOR THE FILE ATTRIBUTE FILE IN ARCHIVE *filename*.

Explanation:
filename name of archive file

An error occurred during processing of the file attribute file. Previous messages describe the specific error.

The file attribute file contains the file control statements that describe the attributes of a file and information about how the file was created. The file attribute file is named 'GIMFAF.XML' and is one of the component files in the indicated archive.

System Action: Processing stops.

Programmer Response: Correct the error and rerun the job.

GIM46703I PROCESSING HAS FAILED FOR PACKAGE ATTRIBUTE FILE *filename*.

Explanation:

filename	name of the package attribute file

An error occurred during processing of the identified package attribute file. Previous messages describe the specific error.

The package attribute file contains the package attribute statements that describe the contents of the package and information about how the package was created.

System Action: Processing stops.

Programmer Response: Correct the error and rerun the job.

GIM46800I DATA SET *dataset* WAS EXTRACTED FROM ARCHIVE *archive*.

Explanation:

dataset	data set name of the extracted file
archive	pathname of the archive

GIMUNZIP successfully processed the named archive. The named archive was expanded in the hierarchical file system. The data set name indicated in the message was successfully extracted from the named archive. GIMUNZIP processing continues.

Programmer Response: None.

GIM47000T *progname* PROCESSING IS TERMINATED. THE *dataset* DATA SET CANNOT BE OPENED.

Explanation:

progname	program name (GIMZIP or GIMUNZIP)
dataset	ddname associated with the data set name

The indicated data set is required for the named program. However, the data set could not be opened.

System Action: Processing is terminated.

Programmer Response: Do one of the following:

- If the JCL was missing a DD statement for the data set, add one.
- If the data set name was incorrect, specify the correct name.

Then rerun the job.

GIM47100S *progname* PROCESSING FAILED BECAUSE IT WAS UNABLE TO OBTAIN THE REQUIRED FILE ATTRIBUTES FOR DATA SET *dataset*.

Explanation:

progname	program name (GIMZIP or GIMUNZIP)
dataset	data set name

Processing failed because a required file attribute for a data set could not be obtained.

System Action: Processing stops.

Programmer Response: Ensure that the data sets that you are processing have the required file attributes specified. Correct any errors and rerun the job.

GIM47400S THE *ddname* DD STATEMENT MUST BE ALLOCATED TO A PATH.

Explanation:

ddname	DD or DDDEF name.

The identified DD must be allocated to a path in the file system.

System Action: Processing stops.

Programmer Response: Allocate the identified DD or DDDEF to a path and rerun the command or job.

GIM47500I DATA SET *infile* WAS ARCHIVED INTO *outfile*.

Explanation:

infile	data set name of input file
outfile	pathname of output file

GIMZIP successfully processed the named input file. The named input file was archived into the hierarchical file system into the pathname of the output file indicated in the message.

System Action: GIMZIP processing continues.

Programmer Response: None.

GIM47501I DATA SET *infile* WAS COPIED INTO *outfile*.

Explanation:

infile	data set name of input file
outfile	name of output file

GIMZIP successfully copied the named data set (as a text file) into the hierarchical file system as the name of the output file indicated in the message.

System Action: GIMZIP processing continues.

Programmer Response: None.

GIM47600I PACKAGE *package* WAS SUCCESSFULLY STAGED TO THE SMPNTS.

Explanation:
package package id.

The identified package has been staged successfully into the SMPNTS.

System Action: Processing continues.

Programmer Response: None.

GIM47601I PACKAGE *package* WAS PARTIALLY STAGED TO THE SMPNTS.

Explanation:
package package id.

Part of the identified package has been staged into the SMPNTS. An error prevented the complete package from being staged.

System Action: Processing stops because of the previously identified error.

Programmer Response: Refer to the previously identified error. Fix the error and rerun the RECEIVE.

GIM47700S PACKAGE *package* ALREADY EXISTS IN THE SMPNTS.

Explanation:
package package id.

A package with the specified id already exists in the SMPNTS directory. SMP/E will not overlay this existing package.

System Action: Processing stops.

Programmer Response: If you want to overlay the existing package, delete the files within the package id subdirectory and rerun the RECEIVE. If you want to keep the existing package, specify a unique package id and rerun the RECEIVE.

GIM47800S AN ERROR OCCURRED WHILE *progname* WAS PROCESSING THE ARCHIVE NAMED *filename*.

Explanation:
progname SMP/E routine that failed (GIMUNZIP or GIMZIP)
filename name of archive file

The indicated SMP/E routine failed because it could not successfully create or retrieve an archive file. If the routine that failed was GIMZIP, the archive was being created. If the routine that failed was GIMUNZIP, the archive was being retrieved.

System Action: Processing stops.

Programmer Response: If a previous message in SMPOUT does not identify the problem that must be resolved, review the associated SYSPRINT data set to identify the error. In either case, correct the error and rerun the job. If the error persists, call your IBM support representative.

GIM47801S AN ERROR OCCURRED WHILE COPYING *infile* INTO *outfile*.

Explanation:
infile data set name of input file
outfile name of output file

An error occurred while trying to copy the named data set into the named file in the hierarchical file system.

System Action: Processing stops.

Programmer Response: Review other messages in the SMPOUT data set to identify the error. Correct the error and rerun the job. If the error persists, call your IBM support representative.

GIM47900S REQUIRED COMPONENT FILE *filename1* WAS NOT FOUND IN ARCHIVE FILE *filename2*.

Explanation:
filename1 name of component file
filename2 name of archive file

The GIMUNZIP routine failed because the required component file was not found in the HFS after the archive was expanded.

System Action: Processing stops.

Programmer Response: Review the associated SYSPRINT data set to identify the error. If the name of the missing component file was 'GIMFAF.XML', GIMUNZIP cannot process the archive.

GIM48000S THE *keyword* KEYWORD HAS BEEN SPECIFIED MORE THAN ONCE IN A FILE OR ARCHIVE DEFINITION GROUP IN THE *dataset* DATA SET.

Explanation:
keyword duplicate keyword operand
dataset ddname of the data set

The identified keyword operand was found more than once while GIMZIP was processing a file definition

group or while GIMUNZIP was processing an archive definition group in the indicated data set.

System Action: Command processing stops.

Programmer Response: Correct the syntax by deleting the duplicate keyword, and rerun the job.

GIM48100S **GIMZIP PROCESSING FAILED. THE** *dataset* **DATA SET IS NOT VALID FOR THE SPECIFIED FILE TYPE OF** '*filetype*'**.**

Explanation:

dataset	data set name
filetype	GIMZIP file type specified

GIMZIP processing failed because the input data set attributes were invalid for the file type specified. The input data set attributes were not valid based on the following requirements.

- If the file type specified was 'README', the data set must be sequential and it must not have a record format of VS (variable spanned).
- If the file type specified was 'SMPPTFIN' or 'SMPHOLD', the data set must be sequential and it must have a record format of FB (fixed block) and a logical record length of eighty (80) bytes.
- If the file type specified was 'SMPRELF', the data set must be partitioned (PO), or the data set must be an IEBCOPY unloaded format partitioned data set.

System Action: Command processing stops.

Programmer Response: Change the appropriate data set attributes and/or file types based on the valid GIMZIP combinations and rerun the job.

GIM48200W *progname* **PROCESSING WAS NOT ABLE TO REMOVE THE** *directory* **DIRECTORY.**

Explanation:

progname	SMP/E routine that failed (RECEIVE, GIMUNZIP, or GIMZIP)
directory	name of directory

The indicated SMP/E routine was unable to remove the specified directory.

System Action: Processing continues.

Programmer Response: If this message is preceded by message GIM43500S, use the information in that message to identify the error. Otherwise, review the associated SYSPRINT data set to identify the error. Delete the directory before rerunning the job.

GIM48200S *progname* **PROCESSING WAS NOT ABLE TO REMOVE THE** *directory* **DIRECTORY.**

Explanation:

progname	SMP/E routine that failed (RECEIVE, GIMUNZIP, or GIMZIP)
directory	name of directory

The indicated SMP/E routine was unable to remove the specified directory.

System Action: Processing stops.

Programmer Response: If this message is preceded by message GIM43500S, use the information in that message to identify the error. Otherwise, review the associated SYSPRINT data set to identify the error. Delete the directory before rerunning the job.

GIM48300S *enttype entname* **COULD NOT BE SAVED IN THE** *library* **LIBRARY BECAUSE IT RAN OUT OF SPACE.**

Explanation:

enttype	entry type
entname	entry name
library	ddname of the library

The indicated library ran out of space.

System Action: Command processing stops.

Programmer Response: Do any of the following:

- Increase the size of the library being processed.
- Define spill data sets for the SMPPTS. A spill data set can be defined using a JCL DD statement or a DDDEF entry. See *SMP/E Reference* for information on defining SMPPTS spill data sets.
- Reduce the number of SYSMODs being saved in the SMPPTS by doing a REJECT or by accepting SYSMODs that have already been applied.

Then rerun the job.

GIM48400S *enttype entname* **COULD NOT BE SAVED BECAUSE THE SMPPTS AND ALL OF ITS SPILL DATA SETS RAN OUT OF SPACE.**

Explanation:

enttype	entry type
entname	entry name

The specified member could not be save because the SMPPTS and all of its spill data sets are out of space or directory blocks.

System Action: Command processing stops.

Programmer Response: Do any of the following:

- Increase the size of the SMPPTS or one or more of the spill data sets.

- Define additional SMPPTS spill data sets. See *SMP/E Reference* for information on defining SMPPTS spill data sets.
- Reduce the number of SYSMODs being saved in the SMPPTS and its spill data sets by doing a REJECT or by accepting SYSMODs that have already been applied.

Then rerun the job.

GIM48500I THE *oldlib* LIBRARY RAN OUT OF SPACE. SMP/E WILL INSTEAD ATTEMPT TO STORE *enttype entname* IN THE *newlib* LIBRARY.

Explanation:

oldlib	ddname of library that ran out of space
enttype	entry type
entname	entry name
newlib	ddname of new library

oldlib ran out of space. An attempt is made to save the member in library *newlib* instead.

System Action: Processing continues.

Programmer Response: See subsequent messages to determine what action, if any, should be taken.

GIM48600S THE DIRECTORY ASSOCIATED WITH THE SMPDIR DD STATEMENT (*directory*) MUST BE EMPTY FOR GIMZIP PROCESSING TO CONTINUE.

Explanation:

directory	name of directory

The indicated directory will be used by GIMZIP to generate a network transportable software package. It must be empty at the start of GIMZIP processing.

System Action: Processing stops.

Programmer Response: Delete the contents of this directory and rerun the job.

GIM48800I DIRECTORY SPACE FOR THE *oldlib* LIBRARY WAS EXCEEDED DURING AN ATTEMPT TO STORE *enttype* ENTRY *entname*. SMP/E WILL INSTEAD ATTEMPT TO STORE *enttype entname* IN THE *newlib* LIBRARY.

Explanation:

oldlib	ddname of library being processed
enttype	entry type
entname	entry name
newlib	ddname of new library

When SMP/E tried to store the element in the indicated library, it exceeded the number of directory blocks allocated to the library. An attempt is made to save the member in library *newlib* instead.

System Action: SMP/E does not store the member. Messages that follow this message show the actions taken by SMP/E.

Programmer Response: Increase the directory block allocation for the indicated library if possible. See subsequent messages to determine what other actions, if any, should be taken.

GIM48900S THE PACKAGE ATTRIBUTE FILE DOES NOT CONTAIN THE REQUIRED ARCHIVE NAME OR HASH VALUE FOR THE ARCHIVE FILE *filename*.

Explanation:

filename	archive file name

GIMUNZIP could not locate an entry for the indicated archive name or its hash value in the package attribute file.

System Action: Processing stops.

Programmer Response: Ensure that the archive names listed in the GIMUNZIP SYSIN input stream have a matching entry in the package attribute file within the directory pointed to by the SMPDIR DD statement. Also, ensure that the directory pointed to by the SMPDIR DD statement contains the appropriate package attribute file for the archives listed in the GIMUNZIP SYSIN input stream.

GIM49000S *progname* COULD NOT LOCATE THE REQUIRED PACKAGE ATTRIBUTE FILE '*filename*' WITHIN THE DIRECTORY NAMED ON THE *ddname* DD STATEMENT.

Explanation:

progname	program name
filename	file name
ddname	ddname

The indicated program could not locate the package attribute file, which is required for processing.

System Action: Processing stops.

Programmer Response: Ensure that the required package attribute file exists within the parent HFS directory pointed to by the SMPDIR DD statement.

GIM49100S VERIFICATION OF HASH VALUE FAILED FOR ARCHIVE *filename*.

Explanation:

filename	name of archive file

The computed hash value of the indicated archive did not match its hash value listed in the package attribute file. The archive's data integrity may have been compromised.

System Action: Processing stops.

Programmer Response: Contact the provider of the indicated archive.

GIM49200S DATA SET *dataset* IS EMPTY. AN EMPTY DATA SET CANNOT BE PART OF A PACKAGE.

Explanation:

dataset	data set name

The indicated data set was part of a package defined for GIMZIP processing. GIMZIP found that the data set was empty and this is not allowed. Note that empty means that the data set had no records if it is sequential or that it had no members if it is a PDS or PDSE.

System Action: GIMZIP processing stops.

Programmer Response: It is likely that the wrong data set was specified as part of the package for GIMZIP processing. If this is the case, correct the name of the data set and rerun the job. If the right data set name was specified, determine why it is empty and correct the problem.

GIM50001E THERE IS AN ERROR IN *systype* SYSMOD *sysmod*.

Explanation:

systype	SYSMOD type
sysmod	SYSMOD ID or blank, if unknown

Note: This message immediately follows message GIM203*xx*, which is immediately preceded by the line that has the syntax error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Look for the specified SYSMOD. Fix the syntax error and rerun the job.
- If you cannot find the SYSMOD in error, use the RECEIVE LIST option to rerun your SMPPTFIN stream. SMP/E will print out the entire SMPPTFIN stream. Fix the syntax error and rerun the job.

GIM50002E THERE IS AN ERROR IN AN UNKNOWN TYPE OF SYSMOD *sysmod*.

Explanation:

sysmod	SYSMOD ID or blank, if unknown

Note: This message immediately follows message GIM203*xx*, which is immediately preceded by the line that has the syntax error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Look for the specified SYSMOD ID. Fix the syntax error and rerun the job.
- If you cannot find the SYSMOD in error, use the RECEIVE LIST option to rerun your SMPPTFIN stream. SMP/E will print out the entire SMPPTFIN stream. Fix the syntax error and rerun the job.

GIM50003E THERE IS AN ERROR IN A ++*mcstype* MCS FOR ELEMENT *elmname* IN *systype* SYSMOD *sysmod*.

Explanation:

mcstype	MCS type or blank, if unknown
elmname	element name or blank, if unknown
systype	SYSMOD type
sysmod	SYSMOD ID or blank, if unknown

Note: This message immediately follows message GIM203*xx*, which is immediately preceded by the line that has the syntax error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Look for the specified SYSMOD and check the format of the operand on the specified MCS. Fix the syntax error and rerun the job.
- If you cannot find the SYSMOD in error, use the RECEIVE LIST option to rerun your SMPPTFIN stream. SMP/E will print out the entire SMPPTFIN stream. Fix the syntax error and rerun the job.

GIM50004E THERE IS AN ERROR IN AN UNKNOWN MCS FOR *systype* SYSMOD *sysmod*.

Explanation:

systype	SYSMOD type
sysmod	SYSMOD ID or blank, if unknown

Note: This message immediately follows message GIM203*xx*, which is immediately preceded by the line that has the syntax error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Look for the specified SYSMOD and check the format of the MCS-type operand on the MCS statement. Fix the syntax error and rerun the job.
- If you cannot find the SYSMOD in error, use the RECEIVE LIST option to rerun your SMPPTFIN stream. SMP/E will print out the entire SMPPTFIN stream. Fix the syntax error and rerun the job.

GIM50005E THERE IS AN ERROR IN A ++*mcstype* MCS FOR AN UNKNOWN ELEMENT IN *systype* SYSMOD *sysmod*.

Explanation:

mcstype	MCS type
systype	SYSMOD type
sysmod	SYSMOD ID or blank, if unknown

Note: This message immediately follows message GIM203*xx*, which is immediately preceded by the line that has the syntax error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Look for the specified SYSMOD and check the format of the element name on the specified MCS. Fix the syntax error and rerun the job.
- If you cannot find the SYSMOD in error, use the RECEIVE LIST option to rerun your SMPPTFIN stream. SMP/E will print out the entire SMPPTFIN stream. Fix the syntax error and rerun the job.

GIM50006E THERE IS AN ERROR IN A ++IF MCS ASSOCIATED WITH ++VER MCS *number* IN *systype* SYSMOD *sysmod*.

Explanation:

number	number of the ++VER MCS
systype	SYSMOD type
sysmod	SYSMOD ID or blank, if unknown

Note: This message immediately follows message GIM203*xx*, which is immediately preceded by the line that has the syntax error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Look for the specified SYSMOD and check the specified MCS. Fix the syntax error and rerun the job.
- If you cannot find the SYSMOD in error, use the RECEIVE LIST option to rerun your SMPPTFIN stream. SMP/E will print out the entire SMPPTFIN stream. Fix the syntax error and rerun the job.

GIM50007E THERE IS AN ERROR IN ++VER MCS *number* IN *systype* SYSMOD *sysmod*.

Explanation:

number	number of the ++VER MCS
systype	SYSMOD type
sysmod	SYSMOD ID or blank, if unknown

Note: This message immediately follows message GIM203*xx*, which is immediately preceded by the line that has the syntax error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Look for the specified SYSMOD and check the specified MCS. Fix the syntax error and rerun the job.
- If you cannot find the SYSMOD in error, use the RECEIVE LIST option to rerun your SMPPTFIN stream. SMP/E will print out the entire SMPPTFIN stream. Fix the syntax error and rerun the job.

GIM50008E THERE IS AN ERROR IN A ++*mcstype* MCS FOR *systype* SYSMOD *sysmod*.

Explanation:

mcstype	MCS type
systype	SYSMOD type
sysmod	SYSMOD ID or blank, if unknown

Note: This message immediately follows message GIM203*xx*, which is immediately preceded by the line that has the syntax error.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- Look for the specified SYSMOD and check the specified MCS. Fix the syntax error and rerun the job.
- If you cannot find the SYSMOD in error, use the RECEIVE LIST option to rerun your SMPPTFIN stream. SMP/E will print out the entire SMPPTFIN stream. Fix the syntax error and rerun the job.

GIM50050I *command* PROCESSING FOR SYSMOD *sysmod* ENCOUNTERED THE DESCRIPTION OPERAND ON THE HEADER MCS. THIS OPERAND IS IGNORED SINCE IT IS SUPPORTED ONLY BY OS/390 R7 SMP/E OR HIGHER.

Explanation:

command	SMP/E command
sysmod	SYSMOD name

A SYSMOD MCS contains a description operand, but you are running a level of SMP/E that does not support the description operand.

System Action: Command processing continues.

Programmer Response: If you wish to make use of the description operand, use OS/390 SMP/E Release 7 or higher. Otherwise, ignore this message.

GIM50051W *command* PROCESSING ENCOUNTERED A ++*mcstype* MCS. THIS MCS IS IGNORED SINCE IT IS SUPPORTED ONLY BY OS/390 RELEASE 7 SMP/E, OR HIGHER.

Explanation:

command	SMP/E command
mcstype	MCS type

The MCS type being processed is either a ++PRODUCT or ++FEATURE, but you are running a level of SMP/E that does not support that MCS type.

System Action: The ++PRODUCT or ++FEATURE MCS is ignored. Processing continues with the next MCS statement.

Programmer Response: If you wish to process a ++FEATURE or ++PRODUCT, use OS/390 SMP/E Release 7 or higher. Otherwise, ignore this message.

GIM50052E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE IT CONTAINS THE ++***mcstype* **MCS. THIS MCS IS ONLY SUPPORTED BY OS/390 RELEASE 7 SMP/E, OR HIGHER.**

Explanation:

command	SMP/E command
sysmod	SYSMOD name
mcstype	MCS type

The SMP/E command being processed has encountered a SYSMOD that contains an MCS type which is only supported by SMP/E OS/390 Release 7 or higher. You are running a level of SMP/E below OS/390 Release 7.

System Action: SYSMOD processing stops.

Programmer Response: If you wish to process this SYSMOD, use OS/390 SMP/E Release 7 or higher. Otherwise, ignore this message.

GIM50053E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE THE ++***mcstype* **FOR ELEMENT** *elmname* **CONTAINS THE** *operand* **OPERAND. THIS OPERAND IS SUPPORTED ONLY BY OS/390 RELEASE 7 SMP/E, OR HIGHER.**

Explanation:

command	SMP/E command
sysmod	SYSMOD name
mcstype	MCS type
elmname	element name
operand	operand

SMP/E processing encountered an operand on an MCS statement which is only supported by SMP/E OS/390 Release 7 or higher.

System Action: SYSMOD processing stops.

Programmer Response: If you wish to process this SYSMOD, use OS/390 SMP/E Release 7 or higher. Otherwise, ignore this message.

GIM50101S *operand1* **IS NOT ALLOWED BECAUSE** *operand2* **WAS NOT SPECIFIED.**

Explanation:

operand1	the first operand
operand2	the second operand

The command being processed specified the first operand without the second one. This is not allowed.

System Action: Command processing fails. SMP/E continues to check the syntax in SMPCNTL.

Programmer Response: For more information about which operands must be specified together, see *SMP/E Commands*.

Remove or add the desired operands and rerun the job.

GIM50201E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE** *operand* **WAS NOT SPECIFIED.**

Explanation:

command	an SMP/E command
sysmod	SYSMOD ID
operand	operand

The indicated SYSMOD was not processed because an operand that would have allowed it to be processed was not specified.

System Action: SYSMOD processing stops. Processing continues with the next SYSMOD.

Programmer Response: If you want SMP/E to process the indicated SYSMOD, add the operand stated in the message and rerun the job.

GIM50301E *value* **WAS NOT PROCESSED BECAUSE IT IS NOT AN FMIDSET DEFINED IN THE GLOBAL ZONE.**

Explanation:

value	value

SMP/E tried to determine whether the indicated value is an FMID or an FMIDSET name. The value is not 7 characters long, so it is not an FMID. Therefore, SMP/E checked whether there was an FMIDSET entry defined in the global zone with the same name as this value. However, it did not find any.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- If you meant to specify an FMIDSET name, either correct the value you specified or create the desired FMIDSET entry. Then rerun the job.
- Otherwise, take no action.

GIM50302E *value* **WAS NOT PROCESSED BECAUSE IT IS NOT AN FMIDSET DEFINED IN THE GLOBALZONE CONTAINED IN THE CSI SPECIFIED ON THE FROMCSI OPERAND.**

Explanation:
value value from FORFMID operand

SMP/E tried to determine whether the indicated value is an FMID or an FMIDSET name. The value is not 7 characters long, so it is not an FMID. Therefore, SMP/E checked whether there was an FMIDSET entry defined in the originating GLOBAL zone with the same name as this value. However, it did not find any.

System Action: GZONEMERGE processing continues with the next FORFMID value.

Programmer Response: Do **one** of the following:

- If you meant to specify an FMIDSET name, either correct or remove the value you specified or create the desired FMIDSET entry. Then rerun the job.
- Otherwise, take no action.

GIM50401T THE CSI PARAMETER IS NOT ALLOWED ON THE GIMSMP EXEC STATEMENT BECAUSE THE SMPCSI DD STATEMENT IS SPECIFIED.

Explanation: There are two ways to have SMP/E allocate the SMPCSI data set that contains the global zone (the master SMPCSI):

- Specify the CSI parameter on the GIMSMP EXEC statement.
- Specify an SMPCSI DD statement in the job.

You can do one or the other, but not both.

System Action: SMP/E processing stops.

Programmer Response: SMP/E cannot determine which SMPCSI it should use. Do **one** of the following:

- If SMP/E should use the SMPCSI DD statement, rerun the job without the CSI parameter on the EXEC statement.
- If SMP/E should use the value on the CSI parameter, rerun the job without the SMPCSI DD statement.

GIM50501E CONNECT PROCESSING FAILED FOR THE *zonename* **ZONE.**

Explanation:
zonename zone name

System Action: SYSMOD processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50501S CONNECT PROCESSING FAILED FOR THE *zonename* **ZONE.**

Explanation:
zonename zone name

System Action: Command processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50501T CONNECT PROCESSING FAILED FOR THE *zonename* **ZONE.**

Explanation:
zonename zone name

System Action: SMP/E processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50502E FREE PROCESSING FAILED FOR THE *zonename* **ZONE.**

Explanation:
zonename zone name

A physical I/O error caused FREE processing to fail for the indicated zone.

System Action: SYSMOD processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50502T FREE PROCESSING FAILED FOR THE *zonename* **ZONE.**

Explanation:
zonename zone name

A physical I/O error caused FREE processing to fail for the indicated zone.

System Action: SMP/E processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50504W OPEN PROCESSING FAILED FOR THE *zonename* **ZONE.**

Explanation:
zonename zone name

During the cross-zone update phase of the APPLY or RESTORE command, SMP/E could not open the indicated zone.

System Action: The cross-zone updates are not made for the zone named in the message. Processing continues with the next cross-zone.

Programmer Response: Make sure you specified the

correct SMPCSI data set name for the indicated target or DLIB zone. This name may be specified either on a DD statement or in a ZONEINDEX subentry in the GLOBALZONE entry.

Fix the error. The Cross-Zone Summary report and the MOVE/RENAME/DELETE report identify the cross-zone updates that need to be done for the zone named in this message. Use a combination of the following to complete the unfinished cross-zone updates identified:
- The LINK command
- UCLIN updates
- The linkage editor (outside of SMP/E)

GIM50504E OPEN PROCESSING FAILED FOR THE *zonename* ZONE.

Explanation:
zonename zone name

SMP/E could not open the indicated zone.

System Action: SYSMOD processing stops.

Programmer Response: Make sure you specified the correct SMPCSI data set name for the indicated target or DLIB zone. This name may be specified either on a DD statement or in a ZONEINDEX subentry in the GLOBALZONE entry.

Fix the error and rerun the job.

GIM50504S OPEN PROCESSING FAILED FOR THE *zonename* ZONE.

Explanation:
zonename zone name

SMP/E could not open the indicated zone.

System Action: Command processing stops.

Programmer Response: Make sure you specified the correct SMPCSI data set name for the indicated target or DLIB zone. This name may be specified either on a DD statement or in a ZONEINDEX subentry in the GLOBALZONE entry.

Fix the error and rerun the job.

GIM50504T OPEN PROCESSING FAILED FOR THE *zonename* ZONE.

Explanation:
zonename zone name

SMP/E could not open the indicated zone (the global zone).

System Action: SMP/E processing stops.

Programmer Response: Make sure you specified the correct SMPCSI data set name for the indicated zone. This name may be specified either on the SMPCSI DD statement or on the CSI parameter of the EXEC statement for GIMSMP.

Fix the error and rerun the job.

GIM50505S CLOSE PROCESSING FAILED FOR THE *zonename* ZONE.

Explanation:
zonename zone name

A physical I/O error caused CLOSE processing to fail for the indicated zone.

System Action: Command processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50505T CLOSE PROCESSING FAILED FOR THE *zonename* ZONE.

Explanation:
zonename zone name

A physical I/O error caused CLOSE processing to fail for the indicated zone.

System Action: SMP/E processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50506E LMOD BUILD PROCESSING WAS UNSUCCESSFUL BECAUSE OPEN PROCESSING FAILED FOR DISTRIBUTION ZONE *zonename*.

Explanation:
zonename zone name

System Action: Command processing stops.

Programmer Response: Make sure you specified the correct SMPCSI data set name for the distribution zone. This name may be specified either on a DD statement or in a ZONEINDEX subentry in the GLOBALZONE entry. Fix the error and rerun the job.

GIM50507T SMP/E CANNOT PROCESS ZONE *zonename* BECAUSE THE SMPCSI DATA SET CONTAINS TOO MANY ZONES.

Explanation:
zonename zone name

System Action: SMP/E processing stops.

Programmer Response: If you are trying to define a new zone, define it in a different SMPCSI data set. Remember to update the associated zone index in the global zone.

If you are trying to update an existing zone, move it to a

different SMPCSI data set. These are some SMP/E commands you can use to do this:

- To add an existing zone to another SMPCSI:
  ZONECOPY
  ZONEEXPORT, UCLIN for the zone index, and
  ZONEIMPORT
- To delete an existing zone from an SMPCSI:
  ZONEDELETE
  ZONEEXPORT

For more information about these commands and when to use them, see *SMP/E Commands*.

GIM50508E *command* **PROCESSING FAILED.**

Explanation:
command CLOSE, CONNECT, or FREE

System Action: SYSMOD processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50508T *command* **PROCESSING FAILED.**

Explanation:
command CLOSE, CONNECT, or FREE

System Action: SMP/E processing stops.

Programmer Response: Fix the error and rerun the job.

GIM50509T AN I/O ERROR OCCURRED WHEN SMP/E WAS TRYING TO LOCATE A RECORD FOR ZONE *zonename***.**

Explanation:
zonename zone name

A physical or logical I/O error occurred when SMP/E was trying to find a record for the indicated zone.

System Action: SMP/E processing stops.

Programmer Response: Subsequent messages explain SMP/E's actions. Find the cause of the error from these messages. Then fix the error and rerun the job.

GIM50601E LINK-EDIT PROCESSING FAILED FOR LMOD *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rtncode***. DELETION WAS ATTEMPTED FOR THE FOLLOWING CSECTS:**

Explanation:
loadmod load module name
library ddname of the library
rtncode return code

Link-edit processing to remove a module from the indicated load module failed. Message GIM50701 follows this message and identifies which modules and CSECTs were being deleted.

System Action: Processing stops for the SYSMOD that caused the module to be deleted.

Programmer Response: Do the following:
1. Check the linkage editor output in SYSPRINT to find the cause of error.

   Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:
   - The PRINT subentry specifies a DDDEF of DUMMY.
   - The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.
2. Reapply the failing SYSMOD.

GIM50602I LINK-EDIT PROCESSING WAS SUCCESSFUL FOR LMOD *loadmod* **IN THE** *library* **LIBRARY. THE RETURN CODE WAS** *rtncode***. THE FOLLOWING CSECTS HAVE BEEN DELETED:**

Explanation:
loadmod load module name
library ddname of the library
rtncode return code

Link-edit processing to remove a module from the indicated load module was successful. Message GIM50701 follows this message and identifies the modules and CSECTs that were deleted.

System Action: Processing continues.

Programmer Response: None.

Note: A return code of 12 is normal when the linkage editor replaces a dummy CSECT.

GIM50603E LINK-EDIT PROCESSING FAILED FOR LMOD *loadmod* **IN THE** *library* **LIBRARY. ABEND** *abncode* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE. DELETION WAS ATTEMPTED FOR THE FOLLOWING CSECTS:**

Explanation:
loadmod load module name
library ddname of the library
abncode abend code

Link-edit processing to remove a module from the indicated LMOD failed. Message GIM50701 follows this message and identifies which modules and CSECTs were being deleted.

System Action: Processing stops for the SYSMOD that caused the module to be deleted.

Programmer Response: Do the following:
1. Increase the size of the indicated library.
2. Reapply the failing SYSMOD.

GIM50701I CSECT *csect* **IN DLIB MODULE** *modname* **IN SYSMOD** *sysmod***.**

Explanation:

csect	CSECT name
modname	module name
sysmod	SYSMOD ID

This message lists the module and CSECTs within that module that were deleted from the load module identified in message GIM50601E or GIM50602I. SYSMOD *sysmod* is the SYSMOD that caused the module to be deleted from the load module.

System Action: None.

Programmer Response: None.

GIM50801W ZONE *zonename* **WAS NOT USED BECAUSE IT IS NOT DEFINED BY A ZONEINDEX SUBENTRY IN THE GLOBAL ZONE.**

Explanation:

zonename	zone name

During the cross-zone update phase of the APPLY or RESTORE command, SMP/E could not find a ZONEINDEX subentry for the zone specified on an SMP/E command—for example:
- The BOUNDARY operand on the SET command
- The RELATED subentry (in a TARGETZONE or DLIBZONE entry) used during APPLY command processing
- The XZLMOD operand of a MOD entry

System Action: Cross-zone updates are not made for the zone named in the message. Processing continues with the next cross-zone.

Programmer Response: The Cross-Zone Summary report and the MOVE/RENAME/DELETE report identify what cross-zone updates need to be done for the zone named in this message. Use a combination of the following to complete the unfinished cross-zone updates identified:
- The LINK command
- UCLIN updates
- The linkage editor (outside of SMP/E)

GIM50801E ZONE *zonename* **WAS NOT USED BECAUSE IT IS NOT DEFINED BY A ZONEINDEX SUBENTRY IN THE GLOBAL ZONE.**

Explanation:

zonename	zone name

The zone listed was specified on the API QUERY zone parameter. A ZONEINDEX subentry could not be found for this zone. Therefore, the query could not be processed.

System Action: Command processing stops after syntax checking is complete.

Programmer Response: Fix the zone parameter or use UCLIN to define a ZONEINDEX subentry for the zone. Then rerun the calling program.

GIM50801S ZONE *zonename* **WAS NOT USED BECAUSE IT IS NOT DEFINED BY A ZONEINDEX SUBENTRY IN THE GLOBAL ZONE.**

Explanation:

zonename	zone name

SMP/E could not find a ZONEINDEX subentry for a zone needed for an SMP/E command—for example:
- The BOUNDARY operand of the SET command
- The FORZONE, ZONES, or ZONESET operand of the REPORT command
- Within a ZONESET used with the ACCEPT, APPLY, REJECT, or RESTORE command
- The RELATED subentry (in a TARGETZONE or DLIBZONE entry) used during APPLY, ACCEPT, or RESTORE command processing
- The input zone on the ZONECOPY command
- The originating zone on the ZONEMERGE command
- The XZGROUP operand on an APPLY, ACCEPT, or RESTORE command
- The XZLMOD subentry of a MOD entry
- The ZONEGROUP operand of the RECEIVE command
- The RECZGRP or RECEXZGRP subentries of the OPTIONS entry.

System Action: Command processing stops. Processing continues with the next valid SET or RESETRC command.

Programmer Response: Do one of the following:
- If you want SMP/E to use the indicated zone, use UCLIN to define a ZONEINDEX subentry for the zone. It may also be necessary to set up the zone itself. Once the preparation work is complete, rerun the job.
- If you don't want SMP/E to use the indicated zone, remove the zone from the area causing the command termination and rerun the job.

GIM50802E LOAD MODULE BUILD PROCESSING STOPPED BECAUSE DISTRIBUTION ZONE *zonename* IS NOT DEFINED BY A ZONEINDEX SUBENTRY IN THE GLOBAL ZONE.

Explanation:
zonename zone name

During APPLY processing, SMP/E needed to check the related distribution zone for a module that is part of a load module being built. However, the distribution zone defined as the related zone is not defined by a ZONEINDEX subentry in the global zone.

System Action: SMP/E also issues the following messages:
- GIM67401E or GIM67402E, which lists the load modules that failed
- GIM67501I, which lists the modules required to complete the load modules
- GIM22601I or GIM22601E, which lists the SYSMODs that failed because the load module failed

Programmer Response: Do **one** of the following:
- If you want SMP/E to process the indicated zone, use UCLIN to define a ZONEINDEX subentry for the zone. Then rerun the job.
- Otherwise, take no action.

GIM50803I A NULL LIST WAS SPECIFIED FOR THE XZGROUP OPERAND. NO CROSS-ZONE REQUISITE CHECKING WILL BE DONE.

Explanation: Because the user has specifically requested that no zone group is to be established, cross-zone requisite checking is not being done.

System Action: Command processing continues.

Programmer Response: None.

GIM50806W NO ZONES WERE APPLICABLE FOR CROSS-ZONE REQUISITE CHECKING FOR THE *command* COMMAND.

Explanation:
command ACCEPT, APPLY, or RESTORE

After checking all zones and ZONESETs for the command, SMP/E could not find any applicable zones for the current command. SMP/E uses only one type of zone for each command:
- APPLY uses only TARGET zones
- ACCEPT uses only DLIB zones
- RESTORE uses only TARGET zones

System Action: Command processing continues.

Programmer Response: Ensure that the ZONESETs and zones specified or used implicitly have the correct zones in them.

GIM50808I *zonetype* ZONE *zonename* WILL BE USED FOR CROSS-ZONE REQUISITE CHECKING.

Explanation:
zonetype TARGET or DLIB
zonename zone name

The indicated zone will be used for cross-zone requisite checking.

System Action: Command processing continues.

Programmer Response: None.

GIM50809S XZGROUP MAY NOT BE SPECIFIED AS A NULL LIST IF THE XZREQ OPERAND IS ALSO SPECIFIED.

Explanation: The XZGROUP operand was specified as a null list (that is, **XZGROUP()**) and the XZREQ operand was also specified. A zone group must be established to make XZREQ effective.

System Action: Command processing terminates.

Programmer Response: Recode the command so that the conflict does not exist and resubmit it.

GIM50810W THE XZREQ OPERAND WAS SPECIFIED ON THE *command* COMMAND BUT SINCE NO ZONES WERE APPLICABLE FOR CROSS-ZONE REQUISITE CHECKING, THE XZREQ OPERAND WILL BE IGNORED.

Explanation:
command ACCEPT or APPLY

After checking all zones and ZONESETs for the command, SMP/E could not find any applicable zones for the current command. Because processing for the XZREQ operand requires a non-empty zone group, XZREQ processing is ignored for the current command.

System Action: Command processing continues.

Programmer Response: Ensure that the ZONESETs and zones specified or used implicitly have the correct zones in them.

GIM50901S A SET COMMAND IS REQUIRED BEFORE THE *command* COMMAND.

Explanation:
command an SMP/E command

There was no SET command before the indicated command. Therefore, SMP/E does not know which zone to process.

System Action: Command processing fails. SMP/E checks the syntax of the subsequent commands.

Processing continues with the next SET command.

Programmer Response: Put a SET command in SMPCNTL before the command that requires it. This will point SMP/E to the zone that it is going to process. Then rerun the job.

GIM50902S SET COMMAND PROCESSING FAILED. DATA IN SMPCNTL UP TO THE NEXT SET COMMAND WILL ONLY BE SYNTAX-CHECKED.

Explanation: An error occurred when SMP/E tried to process a SET command in the SMPCNTL data set. Other messages describe the specific error.

System Action: Command processing fails. SMP/E checks the syntax of the subsequent commands. Processing continues with the next SET command.

Programmer Response: Fix the error for the SET command and rerun the job.

GIM50903I PROCESSING FOR THE *member* MEMBER IN THE SMPPARM DATA SET HAS FAILED.

Explanation:

member	member name, either GIMDDALC or GIMEXITS

An error occurred when SMP/E tried to process the specified member in the SMPPARM data set. Previous messages describe the specific error.

The GIMDDALC member is used to define allocations for certain types of data sets to be used by SMP/E. The GIMEXITS member is used to identify the exit routine programs to be given control at SMP/E exit points. See the section on GIMEXITS in *SMP/E Reference* for details.

System Action: Command processing fails.

Programmer Response: Correct the error and rerun the job.

GIM50904S ALLOCATION FAILED FOR DATA SET *dataset* SPECIFIED ON A GIMEXITS CONTROL STATEMENT.

Explanation:

dataset	data set name

The indicated data set could not be allocated. The data set name was specified on a GIMEXITS control statement in the GIMEXITS member of the SMPPARM data set. Previous messages describe the specific allocation error.

The GIMEXITS member is used to identify the exit routine programs to be given control at SMP/E exit points. See the section on GIMEXITS in *SMP/E Reference* for details.

System Action: Command processing fails.

Programmer Response: Correct the error and rerun the job.

GIM50905S EXIT ROUTINE *exitname* SPECIFIED ON A GIMEXITS CONTROL STATEMENT WAS NOT FOUND.

Explanation:

exitname	exit routine name

The indicated exit routine could not be found. The exit routine was specified on a GIMEXITS control statement in the GIMEXITS member of the SMPPARM data set. If a data set name was also specified on the control statement, then the exit routine could not be found in the specified data set. If no data set name was specified, the exit routine could not be found using the normal program search (STEPLIB, JOBLIB, link list datasets).

The GIMEXITS member is used to identify the exit routine programs to be given control at SMP/E exit points. See the section on GIMEXITS in *SMP/E Reference* for details.

System Action: Command processing fails.

Programmer Response: Do one of the following:

- If you specified an incorrect exit routine name, or an incorrect data set name, correct one or both of them and rerun the job.
- If the exit routine is not available, either specify the data set in which the exit routine resides on the GIMEXITS control statement, or add the exit routine to the JOBLIB, STEPLIB, or a link list dataset. Then rerun the job.

GIM51001E THE *subtype* SUBENTRY IS NOT IN *enttype* ENTRY *entname*.

Explanation:

subtype	subentry type
enttype	entry type
entname	entry name

The indicated subentry was specified on a UCL statement but does not currently exist in the entry.

System Action: Processing for this UCL statement stops. UCLIN processing continues with the next statement.

Programmer Response: Fix the error and rerun the statement.

GIM51101S SMP/E PROCESSING STOPPED BECAUSE IT COULD NOT DETERMINE WHICH ZONE TO UPDATE FOR UCLIN PROCESSING.

Explanation: Module GIMMPUCD could not determine the zone to update for UCLIN processing.

System Action: SMP/E processing stops.

Programmer Response: Report the error to your support group.

GIM51201E THE SPECIFIED ENTRY TYPE IS ONLY ALLOWED FOR A *zonetype* ZONE.

Explanation:
zonetype zone type

One of the entry types specified is not allowed for the zone being processed. It is only allowed for the zone type indicated in the message.

System Action: Processing for the UCL statement stops.

Programmer Response: Check *SMP/E Commands* to determine which entry types can be used for which zones.

Do **one** of the following:
- Change the zone specified on the SET command.
- Change the operand on the command being processed.

Then rerun the job.

GIM51202S THE UTILITY OPERAND IS ONLY ALLOWED FOR THE GLOBAL ZONE.

Explanation: The UTILITY operand was specified on the ZONEEDIT command, but the global zone was not specified on the SET command. Because UTILITY entries only exist in the global zone, the global zone must be specified.

System Action: Command processing stops.

Programmer Response: Change the zone specified on the SET command. Then rerun the job.

GIM51301E THE *operand* OPERAND EXCEEDS THE 26-CHARACTER MAXIMUM.

Explanation:
operand DSPREFIX or TLIBPREFIX

This message is issued during UCLIN processing when a UCLIN statement specified a DSPREFIX or a TLIBPREFIX that has more than 26 characters.

System Action: Processing stops for the statement in error. Processing continues with the next UCLIN statement.

Programmer Response: Fix the UCLIN statement and rerun the job.

GIM51301S THE *operand* OPERAND EXCEEDS THE 26-CHARACTER MAXIMUM.

Explanation:
operand RFPREFIX

This message is issued during RECEIVE processing when the RFPREFIX operand has more than 26 characters.

System Action: Command processing stops.

Programmer Response: Fix the RFPREFIX and rerun the job.

GIM51401W THERE IS NO *enttype* ENTRY FOR *entname*.

Explanation:
enttype entry type (OPTIONS)
entname entry name

SMP/E did not find the indicated OPTIONS entry in the zone definition entry. (An OPTIONS entry is needed for certain commands to set up page length and PEMAX values.) This error occurred either during processing for the LIST, UNLOAD, or UCLIN command, or one of the zone commands, or during the cross-zone phase of APPLY or RESTORE processing.

System Action: **One** of the following occurs:
- If the error did not occur during cross-zone processing, command processing continues using the default values.
- If the error occurred during cross-zone processing, any updates to the cross-zone will not be done. Processing continues with the next zone.

Programmer Response: Do **one** of the following:
- If the error did not occur during cross-zone processing, no action is required. However, you may want to either define the missing entry in the global zone or change the zone definition entry to specify an existing entry.
- If the error occurred during cross-zone processing, see the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to determine which cross-zone updates were not done. Use the information from those reports to create an SMP/E LINK command (or, for ++RENAME, a UCLIN command) to update the indicated load modules.

GIM51401S THERE IS NO *enttype* ENTRY FOR *entname*.

Explanation:
enttype entry type
entname entry name

SMP/E did not find the indicated entry. (For example, an OPTIONS entry is needed for certain commands to set up page length and PEMAX values.) Here are some typical errors:

- An OPTIONS entry specified on the SET command could not be found.
- The OPTIONS entry needed for JCLIN processing could not be found.
- An entry other than an OPTIONS entry could not be found.

System Action: Command processing stops. SMP/E continues to check the SMPCNTL output for syntax.

Programmer Response: Use UCLIN to create the missing entry. Then rerun the job.

GIM51501I *enttype entname* **WAS MOVED TO A NEW DISTLIB AND DELETED FROM DISTLIB** *distlib***.**

Explanation:

enttype	entry type
entname	entry name
distlib	name of the old distribution library

The indicated element was moved to a new distribution library when a function SYSMOD was installed.

System Action: The element is deleted from the indicated data set.

Programmer Response: None.

GIM51601S **THE** *command* **COMMAND WAS NOT PROCESSED BECAUSE THE SET COMMAND SPECIFIED A** *zonetype* **ZONE.**

Explanation:

command	an SMP/E command
zonetype	zone type

The indicated command cannot be processed for the type of zone that was specified on the SET command.

System Action: Command processing stops. SMP/E continues to check SMPCNTL input for syntax.

Programmer Response: Do the following:
1. Determine which types of zones may be processed by the subsequent commands.
2. Make any necessary changes to the SET command or subsequent commands.
3. Rerun the job.

GIM51602S **THE** *command* **COMMAND DID NOT PROCESS THE GLOBAL ZONE BECAUSE THE SET COMMAND SPECIFIED A** *zonetype* **ZONE.**

Explanation:

command	an SMP/E command
zonetype	zone type

The SET command did not specify the global zone, but the indicated command specified the global zone as the input zone. This is not allowed.

System Action: Command processing stops. SMP/E continues to check SMPCNTL input for syntax.

Programmer Response: Do the following:
1. Determine which types of zones may be processed by the subsequent commands.
2. Make any necessary changes to the SET command or subsequent commands.
3. Rerun the job.

GIM51603S **THE** *command* **COMMAND WAS NOT PROCESSED BECAUSE OF A CONFLICT IN ZONE TYPES. THE SET COMMAND SPECIFIED A** *zonetype1* **ZONE, BUT THE INPUT ZONE IS A** *zonetype2* **ZONE.**

Explanation:

command	an SMP/E command
zonetype1	zone type
zonetype2	zone type

The input zone and the SET BOUNDARY zone are incompatible for the specified command. The following chart shows the combinations that are allowed:

SET BOUNDARY Zone Types	**Input Zone Types**	
	Target	**DLIB**
Global		
Target	X	
DLIB	X	X

System Action: Command processing stops. SMP/E continues to check SMPCNTL input for syntax.

Programmer Response: Specify a valid combination of zones and rerun the job.

GIM51605E *value* **WAS NOT PROCESSED BECAUSE** *value* **IS NOT DEFINED AS AN FMID SUBENTRY IN THE GLOBALZONE ENTRY IN THE GLOBAL ZONE CONTAINED IN THE CSI SPECIFIED ON THE FROMCSI OPERAND.**

Explanation:

value	value from FORFMID operand

A value specified on the FORFMID operand of the GZONEMERGE command was 7 characters long and was not an FMIDSET value. Therefore, SMP/E interprets the value as a function SYSMOD name. However, when SMP/E tried to validate this value by

checking to see if it existed in the FMID subentry list of the GLOBALZONE entry of the originating CSI, a matching FMID subentry was not found.

Note: The originating CSI is the CSI specified on the FROMCSI operand.

System Action: GZONEMERGE processing continues with the next FORFMID value.

Programmer Response: Check the input specified on the FORFMID operand of the GZONEMERGE command and correct the indicated value(s).

Rerun the GZONEMERGE command with the corrected FORFMID value(s).

GIM51606S THE DATA SET NAME SPECIFIED ON THE FROMCSI OPERAND CANNOT BE THE SAME AS THE DATA SET CONTAINING THE DESTINATION GLOBAL ZONE.

Explanation: The CSI data set name specified on the FROMCSI operand of the GZONEMERGE command was the same as the CSI data set name that contains the current GLOBAL zone specified on the SET command.

System Action: Command processing stops.

Programmer Response: Fix the GZONEMERGE command so that the originating GLOBAL zone data set name specified on the FROMCSI operand points to a different GLOBAL zone CSI data set name than the set-to GLOBAL zone CSI data set name. Then rerun the job.

GIM51607S THE FROMCSI OPERAND IS REQUIRED ON THE GZONEMERGE COMMAND.

Explanation: The FROMCSI operand is missing from the GZONEMERGE command.

System Action: Command processing stops.

Programmer Response: Add the required FROMCSI operand to the GZONEMERGE command and return the job.

GIM51608S THE ORIGINATING SMPPTS DATA SET CANNOT BE ALLOCATED FOR GZONEMERGE PROCESSING. THERE IS NO DDDEF ENTRY FOR SMPPTS DEFINED IN THE GLOBAL ZONE IN CSI *fromcsi_value***.**

Explanation:

fromcsi_value value CSI data set name containing originating GLOBAL zone

The GZONEMERGE command requires a DDDEF for the SMPPTS data set in the originating GLOBAL zone when either the CONTENT or FORFMID operands are specified, or when all entries are being merged.

System Action: Command processing stops.

Programmer Response: Add the required FROMCSI operand to the GZONEMERGE command and return the job.

GIM51609S THE ORIGINATING SMPPTS DATA SET AND ANY ORIGINATING SMPPTS SPILL DATA SETS MUST BE DIFFERENT FROM THE DESTINATION SMPPTS DATA SET AND ANY DESTINATION SMPPTS SPILL DATA SETS.

Explanation:

The originating SMPPTS data set or one of the originating SMPPTS spill data sets matches the destination SMPPTS data set or one of the destination SMPPTS spill data sets.

System Action: Command processing stops.

Programmer Response: Change the SMPPTS data set names so that they are different. Then rerun the job.

GIM51610S ALLOCATION FAILED FOR THE ORIGINATING *dddef* **DATA SET. THE DATA SET DEFINED USING DDDEF** *dddef* **IN THE ORIGINATING GLOBAL ZONE IN CSI** *csiname* **COULD NOT BE ALLOCATED.**

Explanation:

dddef DDDEF name

csiname Name of CSI data set that contains the originating GLOBAL zone

An error occurred while allocating the originating SMPPTS data set or an originating SMPPTS spill data set. Previous messages contain specific information about the problem.

System Action: Command processing stops.

Programmer Response: Fix the error and rerun the job.

GIM51611S ALLOCATION FAILED FOR THE DATA SET SPECIFIED ON THE FROMCSI OPERAND *operand*

Explanation:

operand Name of CSI data set that contains the originating GLOBAL zone.

An allocation error occurred for the data set specified on the FROMCSI operand of the GZONEMERGE command. Previous messages contain specific information about the problem.

System Action: Command processing stops.

Programmer Response: Fix the error and rerun the job.

GIM51701S THE *command* COMMAND WAS NOT PROCESSED BECAUSE NO SREL WAS SPECIFIED IN THE ZONE DEFINITION ENTRY.

Explanation:

command	SMP/E command

SMP/E could not process the indicated command because no SREL is defined in the zone definition entry.

System Action: Command processing fails. SMP/E continues to check SMPCNTL input for syntax.

Programmer Response: Use UCLIN to define the proper SREL values in the zone definition entry. Then rerun the job.

GIM51702E THE *command* COMMAND WAS NOT PROCESSED BECAUSE NO RELATED ZONE WAS SPECIFIED IN THE ZONE DEFINITION ENTRY.

Explanation:

command	SMP/E command

SMP/E could not process the indicated command because no related zone is defined in the zone definition entry.

System Action: Processing continues. SMP/E continues to check SMPCNTL input for syntax.

Programmer Response: Use UCLIN to define the proper RELATED value in the zone definition entry. Then rerun the job.

GIM51702S THE *command* COMMAND WAS NOT PROCESSED BECAUSE NO RELATED ZONE WAS SPECIFIED IN THE ZONE DEFINITION ENTRY.

Explanation:

command	SMP/E command

SMP/E could not process the indicated command because no related zone is defined in the zone definition entry.

System Action: Command processing stops. SMP/E continues to check SMPCNTL input for syntax.

Programmer Response: Use UCLIN to define the proper RELATED value in the zone definition entry. Then rerun the job.

GIM51703E THE *command* COMMAND WAS NOT PROCESSED BECAUSE NO *zonetype* ZONE WAS SPECIFIED ON THE *operand* OPERAND.

Explanation:

command	SMP/E command
zonetype	zone type
operand	operand

The command could not be processed because no zone of the required type was specified on the indicated operand. For example, this message is issued for the LIST command when the NOACCEPT operand does not specify a distribution zone and the SET command does not specify a target zone.

System Action: Command processing fails.

Programmer Response: Specify the required type of zone on the indicated operand or on the SET command, as appropriate. Then rerun the job.

GIM51801W ZONE *zonename* IS DEFINED AS A *zonetype1* ZONE IN THE ZONEINDEX SUBENTRY. THIS CONFLICTS WITH THE ZONE TYPE OF *zonetype2*.

Explanation:

zonename	zone name
zonetype1	zone type defined in the ZONEINDEX subentry
zonetype2	zone type

During the cross-zone update phase of the APPLY or RESTORE command, SMP/E determined that the zone type specified in the ZONEINDEX subentry conflicts with the zone type indicated in SMP/E's internal records.

System Action: Cross-zone updates are not made for the zone named in the message. Processing continues with the next cross-zone.

Programmer Response: Determine which of the zone types is correct. Then, if necessary, do **one** of the following:

- If the zone type specified in the ZONEINDEX subentry is incorrect, use UCLIN to fix the ZONEINDEX subentry. Then rerun the job.
- If the zone type in SMP/E's internal records is incorrect, use ZONEDELETE to delete the zone, followed by UCLIN to recreate the zone with the correct zone type. Then rerun the job.
- The Cross-Zone Summary report and the MOVE/RENAME/DELETE report identifies which cross-zone updates need to be done for the zone named in this message. Use a combination of the following to complete the unfinished cross-zone updates identified:
  - The LINK command
  - UCLIN updated

– The linkage editor (outside of SMP/E)

GIM51801S ZONE *zonename* IS DEFINED AS A *zonetype1* ZONE IN THE ZONEINDEX SUBENTRY. THIS CONFLICTS WITH THE ZONE TYPE OF *zonetype2*.

Explanation:

zonename	zone name
zonetype1	zone type defined in the ZONEINDEX subentry
zonetype2	zone type

The zone type specified in the ZONEINDEX subentry conflicts with the zone type indicated in SMP/E's internal records.

System Action: Command processing stops.

Programmer Response: Determine which of the zone types is correct. Then do **one** of the following:

- If the zone type specified in the ZONEINDEX subentry is incorrect, use UCLIN to fix the ZONEINDEX subentry. Then rerun the job.
- If the zone type in SMP/E's internal records is incorrect, use ZONEDELETE to delete the zone, followed by UCLIN to recreate the zone with the correct zone type. Then rerun the job.
- Rerun the job.

GIM51901E A ++*mcstype* MCS IS NOT ALLOWED WITHIN A SYSMOD.

Explanation:

mcstype	MCS type (++RELEASE or ++NULL)

During RECEIVE processing, SMP/E found the indicated MCS within a SYSMOD. This MCS is only allowed in SMPHOLD.

System Action: SMP/E does not receive the SYSMOD.

Programmer Response: Remove the indicated MCS from the SYSMOD and, if necessary, add it to SMPHOLD. Then run RECEIVE again.

GIM52001E SYSMOD *sysmod1* CONTAINS A ++HOLD MCS WHICH SPECIFIES SYSMOD *sysmod2*. SYSMOD *sysmod2* IS NOT SUPERSEDED BY *sysmod1*.

Explanation:

sysmod1	SYSMOD ID
sysmod2	SYSMOD ID

During RECEIVE processing, SMP/E found a ++HOLD MCS within a SYSMOD. However, the SYSMOD specified on that MCS was not the same as the SYSMOD that SMP/E was receiving.

You can use a ++HOLD within a SYSMOD only to hold the containing SYSMOD. The SYSMOD specified on the ++HOLD must be the same as the SYSMOD specified on the header MCS or it must be superseded on all ++VER MCS.

System Action: SMP/E does not receive the SYSMOD.

Programmer Response: If you mean to hold the containing SYSMOD, change the SYSMOD specified on the ++HOLD MCS so that it is either the same as the SYSMOD specified on the header MCS of the containing SYSMOD or is the same as a SYSMOD superseded by the containing SYSMOD (if there are multiple ++VER MCS in the SYSMOD, then each must supersede the SYSMOD).

If you don't want to hold the containing SYSMOD, remove the ++HOLD MCS.

Once you have corrected the error, rerun the RECEIVE job.

GIM52101E THE ++*mcstype* MCS SPECIFIES MORE THAN ONE HOLD CATEGORY. THIS IS NOT ALLOWED.

Explanation:

mcstype	MCS type (++HOLD or ++RELEASE)

The indicated MCS specified one of the following:

- More than one HOLD category (ERROR, SYSTEM, or USER)
- The same HOLD category more than once

System Action:

- If SMP/E was processing a ++HOLD within a SYSMOD, SYSMOD processing stops.
- If SMP/E was processing the MCS from SMPHOLD, MCS processing stops.

Programmer Response: Fix the indicated MCS and rerun the command.

GIM52301E THE *command* COMMAND SPECIFIES OPERANDS THAT ARE NOT ALLOWED FOR THE ZONE SPECIFIED ON THE SET COMMAND.

Explanation:

command	LIST or UNLOAD

The indicated command specifies entries that cannot exist in the zone being processed. Therefore, the operands for these entries are not allowed.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Do **one** of the following:

- To process the entries that were specified, specify the correct zone on the SET command.

- To process the zone specified on the SET command, change the entries that are specified on the indicated command.

The following list shows the entries that **cannot** be specified for the indicated zone types. Make sure **not** to specify the operands with the zone types under which they are listed. For more information about which operands are allowed for which types of zones, see *SMP/E Commands*.

- **Operands Not Allowed for Global Zone:**
  - ASSEM
  - BACKUP
  - BYP,BYPASS
  - Data element entries
  - DEL,DELETE
  - DLIB
  - DZONE,DLIBZONE
  - LMOD
  - MAC
  - MOD
  - NOSUP
  - RES,RESTORE
  - SRC
  - SUP
  - TZONE,TARGETZONE
- **Operands Not Allowed for Target Zone:**
  - DZONE,DLIBZONE
  - FEATURE
  - FMSET,FMIDSET
  - GZONE,GLOBALZONE
  - OPTIONS
  - PRODUCT
  - UTILITY
  - ZONESET
- **Operands Not Allowed for Distribution Zone:**
  - FEATURE
  - FMSET,FMIDSET
  - GZONE,GLOBALZONE
  - OPTIONS
  - PRODUCT
  - RES,RESTORE
  - TZONE,TARGETZONE
  - UTILITY
  - ZONESET

GIM52401I UCLIN PROCESSING IS COMPLETE FOR *zonetype* ZONE *zonename*.

Explanation:

zonetype	zone type
zonename	zone name

UCLIN finished processing for the indicated zone.

System Action: Processing continues.

Programmer Response: None.

GIM52402I UCLIN PROCESSING IS COMPLETE FOR THE GLOBAL ZONE.

Explanation: UCLIN finished processing for the global zone.

System Action: Processing continues.

Programmer Response: None.

GIM52501W A PERIOD IS MISSING AFTER THE LAST *command* COMMAND. ONE IS ASSUMED.

Explanation:

command	ENDUCL or ENDZONEEDIT

The indicate command was followed by an end-of-file. The period after the command is missing.

System Action: All prior UCLIN or ZONEEDIT processing was completed. A message follows to show whether command processing was successful. Processing continues.

Programmer Response: Do **one** of the following:

- If a subsequent message indicates that an error occurred, fix the error and rerun the job.
- Otherwise, take no action.

GIM52601E THE UCL STATEMENT AND THE SET COMMAND REFER TO DIFFERENT ZONE NAMES. THE ZONE NAMES MUST BE THE SAME.

Explanation: The UCL statement points to a zone name that is not the same as the zone name specified on the SET command.

System Action: SMP/E processing for this statement stops. Processing continues with the next UCL statement.

Programmer Response: Do **one** of the following:

- Change the zone specified UCL statement and rerun the job.
- Change the zone specified on the SET BOUNDARY command and rerun the job.

GIM52602E THE UCL STATEMENT AND THE SET COMMAND REFER TO DIFFERENT ZONE TYPES. THE ZONE TYPES MUST BE THE SAME.

Explanation: The UCL statement points to a zone type that is not the same as the zone type specified on the SET command.

System Action: SMP/E processing for this statement stops. Processing continues with the next UCL statement.

Programmer Response: Do **one** of the following:

- Change the zone specified UCL statement and rerun the job.
- Change the zone specified on the SET BOUNDARY command and rerun the job.

GIM52701E THE DSNAME EXCEEDS THE 44-CHARACTER MAXIMUM.

Explanation: An operand on a UCL statement specifies a data set name that has more than 44 characters.

System Action: Processing stops for the UCL statement in error. Processing continues with the next UCL statement.

Programmer Response: Fix the operand, then rerun the job.

GIM52701S THE DSNAME EXCEEDS THE 44-CHARACTER MAXIMUM.

Explanation: An operand on a command specifies a data set name that has more than 44 characters.

System Action: Processing stops for the command in error. Processing continues with the next command.

Programmer Response: Fix the operand, then rerun the job.

GIM52702I THE CSI PARAMETER ON THE GIMSMP EXEC STATEMENT SPECIFIES A DATA SET NAME THAT EXCEEDS THE 44-CHARACTER MAXIMUM.

Explanation: The CSI parameter specifies a data set name that has more than 44 characters.

System Action: SMP/E processing stops.

Programmer Response: Correct the CSI parameter on the EXEC statement, then rerun the job.

GIM52703S PATHNAME *pathname* TO BE DYNAMICALLY ALLOCATED TO DDNAME *ddname* EXCEEDS THE 255-CHARACTER MAXIMUM.

Explanation:

pathname	pathname that is to be dynamically allocated to a DD statement
ddname	ddname to which the path was to be allocated

The indicated pathname needs to be allocated to a ddname for SMP/E processing, but exceeds the 255-character limit allowed by dynamic allocation. This error may occur when:

- GIMUNZIP needs to allocate the GIMPAF.XML file or a GIMFAF.XML file and the pathname specified with the SMPDIR DD statement is too long.
- RECEIVE processing needs to allocate a GIMPAF.XML file or an SMPPTFIN or SMPHOLD file within a network package and the combination of the pathname specified with the SMPNTS DD statement plus the user assigned package id subdirectory name is too long.

System Action: The component of SMP/E issuing this message ends.

Programmer Response:

- For GIMUNZIP processing, shorten the pathname specified by the SMPDIR DD statement. This can be done by:
  - copying or moving the network package from its current location in the hierarchical file system to a new location where the length of the pathname needed to access the network package is shorter.
  - creating a shorter symbolic link for the pathname used on the SMPDIR DD and then substituting the symbolic link for the original pathname value.

  Implement the solution best for your system and rerun the GIMUNZIP job.
- For RECEIVE processing, shorten the combination of the pathname specified by the SMPNTS DD statement plus the value of the package id subdirectory. This can be done by:
  - specifying a different pathname for the SMPNTS DD statement, a shorter package id value, or both, for RECEIVE processing.
  - creating a shorter symbolic link for the place-name used on the SMPNTS DD statement and possibly also specifying a shorter package id value for RECEIVE processing.

  Implement the solution best for your system and rerun the RECEIVE job.

Note: As a rule of thumb, if the pathname specified by the SMPDIR or SMPNTS DD statement is no more than 100 characters long, then there should be no problem dynamically allocating files within the specified path of the hierarchical file system for GIMUNZIP or RECEIVE processing.

GIM52801E THE LOW-LEVEL QUALIFIER OF THE SMPCSI DATA SET NAME MUST BE CSI.

Explanation: The lowest-level qualifier for an SMPCSI data set name must be CSI. The SMPCSI data set name specified on a UCL statement did not meet this requirement.

System Action: Processing stops for the UCLIN statement in error. Processing continues with the next UCL statement.

Programmer Response: Fix the UCL statement. Then rerun the job.

GIM52801S THE LOW-LEVEL QUALIFIER OF THE SMPCSI DATA SET NAME MUST BE CSI.

Explanation: The lowest-level qualifier for an SMPCSI data set name must be CSI. The SMPCSI data set name specified on a command did not meet this requirement.

System Action: Processing stops for the command in error. Processing continues with the next command.

Programmer Response: Fix the command. Then rerun the job.

GIM52801T THE LOW-LEVEL QUALIFIER OF THE SMPCSI DATA SET NAME MUST BE CSI.

Explanation: The lowest-level qualifier for an SMPCSI data set name must be CSI. The SMPCSI data set name specified on the EXEC statement for SMP/E did not meet this requirement.

System Action: SMP/E processing stops.

Programmer Response: Fix the EXEC parameter. Then rerun the job.

GIM52901E THE UCL DEL STATEMENT CANNOT DELETE ZONES. USE THE ZONEDELETE COMMAND INSTEAD.

Explanation: The UCL DEL statement seemed to indicate that an entire zone should be deleted. The UCL DEL statement can delete entries within zones, and subentries within entries. However, it cannot delete an entire zone.

System Action: Processing stops for the statement in error. Processing continues with the next UCL statement.

Programmer Response: Do **one** of the following:
- If you meant to delete a target or distribution zone, use the ZONEDELETE command instead.
- If you meant to delete an entry or subentry, fix the error and rerun this statement.

GIM53901E THE LIST NOACCEPT COMMAND MUST SPECIFY A DLIB ZONE. THE LIST NOAPPLY COMMAND MUST SPECIFY A TARGET ZONE.

Explanation: A zone was specified on the indicated LIST command, but it was not the correct type.

System Action: Command processing stops.

Programmer Response: Specify the correct zone type or operand and rerun the command.

GIM54001E AN ERROR REASON ID IS NOT ALLOWED ON A ++HOLD MCS WITHIN A SYSMOD.

Explanation: When SMP/E was receiving a SYSMOD, it encountered a ++HOLD MCS that specified an ERROR reason ID. This is not allowed. The only type of reason ID allowed within a SYSMOD is a SYSTEM hold.

System Action: SMP/E did not receive the SYSMOD.

Programmer Response: Move the ++HOLD MCS to SMPHOLD and rerun the RECEIVE command.

GIM54002E A USER REASON ID IS NOT ALLOWED ON A ++HOLD MCS WITHIN A SYSMOD.

Explanation: When SMP/E was receiving a SYSMOD, it encountered a ++HOLD MCS that specified a USER reason ID. This is not allowed. The only type of reason ID allowed within a SYSMOD is a SYSTEM hold.

System Action: SMP/E did not receive the SYSMOD.

Programmer Response: Move the ++HOLD MCS to SMPHOLD and rerun the RECEIVE command.

GIM54101I DYNAMIC ALLOCATION FOR *library* **WAS SUCCESSFUL –** *parameters*

Explanation:

library	ddname of the library
parameters	allocation parameters

SMP/E successfully allocated the indicated data set.

System Action: Processing continues.

Programmer Response: None.

GIM54201I DYNAMIC ALLOCATION FOR *library1* **WAS SUCCESSFUL. THE FOLLOWING LIBRARIES WERE CONCATENATED:** *ddname,...,ddname***.**

Explanation:

library1	ddname of the library that was allocated
ddname,...,ddname	ddnames of the libraries that were concatenated

The indicated libraries were successfully concatenated during dynamic allocation of the specified library. If there are more than 13 libraries concatenated, this message shows the first 13 libraries, followed by **(PART 1)**. Additional messages list the remaining libraries, followed by **(PART 2)**, **(PART 3)**, and so on, until all the libraries have been listed.

System Action: Processing continues.

Programmer Response: None.

GIM54301I DYNAMIC DEALLOCATION WAS SUCCESSFUL FOR *library*.

Explanation:
library ddname of the library

SMP/E successfully freed the indicated library.

System Action: Processing continues.

Programmer Response: None.

GIM54401W DYNAMIC DEALLOCATION FAILED FOR *dataset*.

Explanation:
dataset one of the following:
- If the SMPTLIB data set is being deallocated, this is the data set name.
- If another data set is being deallocated, this is the ddname of the data set.

SMP/E attempted to free a data set but the return code from dynamic allocation indicated that the data set was not freed.

System Action: Processing continues.

Programmer Response: None.

GIM54501I ALLOCATION FAILED FOR *library* BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN THE GLOBAL ZONE.

Explanation:
library ddname of the library

If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary in the future, do **one** of the following, then rerun the job:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

GIM54501W ALLOCATION FAILED FOR *library* BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN THE GLOBAL ZONE.

Explanation:
library ddname of the library

If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary, do **one** of the following, then rerun the job:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

GIM54501E ALLOCATION FAILED FOR *library* BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN THE GLOBAL ZONE.

Explanation:
library ddname of the library

If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM54501S ALLOCATION FAILED FOR *library* BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN THE GLOBAL ZONE.

Explanation:
library ddname of the library

If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM54501T **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN THE GLOBAL ZONE.**

Explanation:

library ddname of the library

If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM54502I **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library ddname of the library

If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.

zonetype TARGET or DLIB

zonename zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary in the future, do **one** of the following, then rerun the job:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

GIM54502W **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library ddname of the library

If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.

zonetype TARGET or DLIB

zonename zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary, do **one** of the following, then rerun the job:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

GIM54502E **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library ddname of the library

If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.

zonetype TARGET or DLIB

zonename zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM54502S **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	ddname of the library If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM54502T **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	ddname of the library If this is in the form SMP#####, the library is part of a concatenation. Message GIM54601 shows the ddname of the library for which the concatenation was done.
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM54601I **DYNAMIC ALLOCATION FAILED FOR** *library1* **BECAUSE** *library2* **IS PART OF THE** *library1* **CONCATENATION BUT WAS NOT ALLOCATED.**

Explanation:

library1	ddname of the library being allocated
library2	ddname of a library that was part of the concatenation

The indicated library could not be allocated because allocation failed for one of the libraries that is concatenated to create the indicated library. Additional messages, such as GIM545*xx*, GIM54701, or GIM57601, indicate why the concatenated library failed.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary in the future, fix the error for the *library2* library and rerun the job.

GIM54601W **DYNAMIC ALLOCATION FAILED FOR** *library1* **BECAUSE** *library2* **IS PART OF THE** *library1* **CONCATENATION BUT WAS NOT ALLOCATED.**

Explanation:

library1	ddname of the library being allocated
library2	ddname of a library that was part of the concatenation

The indicated library could not be allocated because allocation failed for one of the libraries that is concatenated to create the indicated library. Additional messages, such as GIM545*xx*, GIM54701, or GIM57601, indicate why the concatenated library failed.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary, fix the error for the *library2* library and rerun the job.

GIM54601E DYNAMIC ALLOCATION FAILED FOR *library1* BECAUSE *library2* IS PART OF THE *library1* CONCATENATION BUT WAS NOT ALLOCATED.

Explanation:

library1	ddname of the library being allocated
library2	ddname of a library that was part of the concatenation

The indicated library could not be allocated because allocation failed for one of the libraries that is concatenated to create the indicated library. Additional messages, such as GIM545*xx*, GIM54701, or GIM57601, indicate why the concatenated library failed.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error for the *library2* library and rerun the job.

GIM54601S DYNAMIC ALLOCATION FAILED FOR *library1* BECAUSE *library2* IS PART OF THE *library1* CONCATENATION BUT WAS NOT ALLOCATED.

Explanation:

library1	ddname of the library being allocated
library2	ddname of a library that was part of the concatenation

The indicated library could not be allocated because allocation failed for one of the libraries that is concatenated to create the indicated library. Additional messages, such as GIM545*xx*, GIM54701, or GIM57601, indicate why the concatenated library failed.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error for the *library2* library and rerun the job.

GIM54601T DYNAMIC ALLOCATION FAILED FOR *library1* BECAUSE *library2* IS PART OF THE *library1* CONCATENATION BUT WAS NOT ALLOCATED.

Explanation:

library1	ddname of the library being allocated
library2	ddname of a library that was part of the concatenation

The indicated library could not be allocated because allocation failed for one of the libraries that is concatenated to create the indicated library. Additional messages, such as GIM545*xx*, GIM54701, or GIM57601, indicate why the concatenated library failed.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error for the *library2* library and rerun the job.

GIM54701I *svc99fct* FAILED FOR *library* – *errmsg*.

Explanation:

svc99fct	SVC99 function that failed
library	ddname of the library
errmsg	the text of the error message from the message processing facility of SVC 99 that shows the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary in the future, fix the error and rerun the job.

Note: Multiple SYSPRINT data sets may be allocated for an APPLY or RESTORE command that has cross-zone work to be done. If allocation fails for SYSPRINT, you need to use the File Allocation report to determine if the SYSPRINT allocation was for the set-to zone or whether it was for a cross-zone. Fix the SYSPRINT allocation that was in error.

GIM54701W *svc99fct* FAILED FOR *library* – *errmsg*.

Explanation:

svc99fct	SVC99 function that failed
library	ddname of the library
errmsg	the text of the error message from the message processing facility of SVC 99 that shows the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary, fix the error and rerun the job.

Note: Multiple SYSPRINT data sets may be allocated for an APPLY or RESTORE command that has cross-zone work to be done. If allocation fails for SYSPRINT, you need to use the File Allocation

report to determine if the SYSPRINT allocation was for the set-to zone or whether it was for a cross-zone. Fix the SYSPRINT allocation that was in error.

GIM54701E *svc99fct* **FAILED FOR** *library* **–** *errmsg***.**

Explanation:

svc99fct	SVC99 function that failed
library	ddname of the library
errmsg	the text of the error message from the message processing facility of SVC 99 that shows the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error and rerun the job.

Note: Multiple SYSPRINT data sets may be allocated for an APPLY or RESTORE command that has cross-zone work to be done. If allocation fails for SYSPRINT, you need to use the File Allocation report to determine if the SYSPRINT allocation was for the set-to zone or whether it was for a cross-zone. Fix the SYSPRINT allocation that was in error.

GIM54701S *svc99fct* **FAILED FOR** *library* **–** *errmsg***.**

Explanation:

svc99fct	SVC99 function that failed
library	ddname of the library
errmsg	the text of the error message from the message processing facility of SVC 99 that shows the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error and rerun the job.

Note: Multiple SYSPRINT data sets may be allocated for an APPLY or RESTORE command that has cross-zone work to be done. If allocation fails for SYSPRINT, you need to use the File Allocation report to determine if the SYSPRINT allocation was for the set-to zone or whether it was for a cross-zone. Fix the SYSPRINT allocation that was in error.

GIM54701T *svc99fct* **FAILED FOR** *library* **–** *errmsg***.**

Explanation:

svc99fct	SVC99 function that failed
library	ddname of the library
errmsg	the text of the error message from the message processing facility of SVC 99 that shows the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error and rerun the job.

Note: Multiple SYSPRINT data sets may be allocated for an APPLY or RESTORE command that has cross-zone work to be done. If allocation fails for SYSPRINT, you need to use the File Allocation report to determine if the SYSPRINT allocation was for the set-to zone or whether it was for a cross-zone. Fix the SYSPRINT allocation that was in error.

GIM54801I **ALLOCATION WAS NOT ATTEMPTED FOR** *library* **BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.**

Explanation:

library	ddname of the library

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary in the future, do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used for SMP/E.

Then issue another SET command and rerun the failing command.

GIM54801W ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation:
library ddname of the library

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: If necessary, do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used for SMP/E.

Then issue another SET command and rerun the failing command.

GIM54801E ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation:
library ddname of the library

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used for SMP/E.

Then issue another SET command and rerun the failing command.

GIM54801S ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation:
library ddname of the library

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used for SMP/E.

Then issue another SET command and rerun the failing command.

GIM54801T ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation:
library ddname of the library

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the specified zone.

- Include the DD statement in the JCL used for SMP/E.

Then issue another SET command and rerun the failing command.

GIM54901I *library* **WAS ALLOCATED BY THE USER AS SYSIN/SYSOUT.**

Explanation:
library ddname of the library

The indicated data set was allocated outside of SMP/E, instead of dynamically by SMP/E.
- For background jobs, SMP/E found a DD statement in the JCL used to run SMP/E.
- For foreground processing, SMP/E found a file that was already allocated.

System Action: None.

Programmer Response: None. SMP/E writes this message to the SMPLOG data set for record-keeping purposes only. The message lets you use SMPLOG to verify that SMP/E directed updates to the correct data sets and volumes at any point in time.

GIM54902I *library* **WAS ALLOCATED BY THE USER AS A DUMMY DATA SET.**

Explanation:
library ddname of the library

The indicated data set was allocated outside of SMP/E, instead of dynamically by SMP/E.
- For background jobs, SMP/E found a DD statement in the JCL used to run SMP/E.
- For foreground processing, SMP/E found a file that was already allocated.

System Action: Processing continues.

Programmer Response: No action is required.

SMP/E writes this message to the SMPLOG data set. The message enables you to use SMPLOG to verify that SMP/E directed updates to the correct data sets and volumes at any point in time.

GIM54902S *library* **WAS ALLOCATED BY THE USER AS A DUMMY DATA SET.**

Explanation:
library ddname of the library

An enqueue was attempted on a dummy data set. The indicated data set was allocated outside of SMP/E, instead of dynamically by SMP/E.
- For background jobs, SMP/E found a DD statement in the JCL used to run SMP/E.
- For foreground processing, SMP/E found a file that was already allocated.

System Action: Command processing stops.

Programmer Response: Specify a valid data set name for the data set and rerun the job.

SMP/E writes this message to the SMPLOG data set. The message enables you to use SMPLOG to verify that SMP/E directed updates to the correct data sets and volumes at any point in time. (This message is also written to SMPOUT.)

GIM54903I *library* **WAS ALLOCATED BY THE USER –** *parameters*

Explanation:
library ddname of the library
parameters allocation parameters

The indicated data set was allocated outside of SMP/E, instead of dynamically by SMP/E.
- For background jobs, SMP/E found a DD statement in the JCL used to run SMP/E.
- For foreground processing, SMP/E found a file that was already allocated.

System Action: None.

Programmer Response: None. SMP/E writes this message to the SMPLOG data set for record-keeping purposes only. The message enables you to use SMPLOG to verify that SMP/E directed updates to the correct data sets and volumes at any point in time.

GIM55101I **ZONE** *zonename* **WAS SUCCESSFULLY DELETED.**

Explanation:
zonename zone name

SMP/E deleted from the SMPCSI data set all the entries that were in the indicated zone.

System Action: Processing continues.

Programmer Response: None.

GIM55201I **THE ZONEINDEX SUBENTRY FOR** *zonename* **WAS DELETED.**

Explanation:
zonename zone name

SMP/E deleted the indicated ZONEINDEX subentry.

System Action: Processing continues.

Programmer Response: None.

GIM55202W **THE ZONEINDEX SUBENTRY FOR** *zonename* **WAS NOT DELETED. SMP/E COULD NOT OBTAIN ACCESS TO THE GLOBAL ZONE.**

Explanation:
zonename zone name

SMP/E was not able to delete the indicated ZONEINDEX subentry.

System Action: Processing continues.

Programmer Response: Use UCLIN to delete the ZONEINDEX subentry.

GIM55301W SMP/E COULD NOT ALLOCATE THE SMPPARM DATA SET TO CHECK FOR OPCODE VALUES. SMP/E USED THE DEFAULT SET OF OPCODE DEFINITIONS.

Explanation: During JCLIN processing, SMP/E could not allocate the SMPPARM data set, which contains members that list valid assembler OPCODEs. SMP/E uses this list of OPCODEs when it is scanning the assembler steps to distinguish valid OPCODEs from macro invocations.

Because SMP/E could not get the OPCODE lists from SMPPARM, SMP/E continues processing, using SMP/E's default set of OPCODE definitions. .

System Action: Processing continues.

Programmer Response: Do **one** of the following:

- Rerun the job with an SMPPARM DD statement.
- If message GIM55503I appears, determine whether the default method caused any errors. If so, use UCLIN to delete the incorrect entry.

  The following is an example of how to use UCLIN to delete a MAC entry:

  ```
  UCLIN.
  DEL MAC(macname)
      ...
      GENASM(genasm).
  ENDUCL.
  ```

  where:
  - *macname* is the name of the macro from GIM55503I.
  - *asment* is the name of the ASSEM entry from GIM55503I.

GIM55401W MEMBER *member* IS NOT IN THE SMPPARM DATA SET. SMP/E USED THE DEFAULT SET OF OPCODE DEFINITIONS.

Explanation:
member name of an OPCODE member

During JCLIN processing of an assembler step, SMP/E needed to determine whether a character string was a macro or an OPCODE. The OPCODE member was not found. Therefore, SMP/E used the default set of OPCODE definitions.

System Action: Processing continues.

Programmer Response: Do **one** of the following:

- Check the member name, correct it if necessary, and rerun the job.
- If the required member is missing, code one, and rerun the job.
- Check the SMPPARM DD statement to ensure that it identifies the proper data set name. If not, correct it and rerun the job.
- If message GIM55503I appears, determine whether the default method caused any errors. If so, use UCLIN to delete the incorrect entry.

  The following is an example of how to use UCLIN to delete a MAC entry:

  ```
  UCLIN.
  DEL MAC(macname)
      ...
      GENASM(genasm).
  ENDUCL.
  ```

  where:
  - *macname* is the name of the macro from GIM55503I.
  - *genasm* is the name of the ASSEM entry from GIM55503I.

GIM55402W MEMBER *member* IS NOT IN THE SMPPARM DATA SET. SMP/E USED ONLY THE OPCODES IN THE GIMOPCDE MEMBER.

Explanation:
member name of an OPCODE member

During JCLIN processing of an assembler step, SMP/E needed to determine whether a character string was a macro or an OPCODE. In this case, the user specified to look in member *macname* in SMPPARM. However, SMP/E could not find member *macname*, but it did find GIMOPCDE. As a result, SMP/E used only the OPCODEs in the GIMOPCDE member, not the member *macname* specified on the JCLIN command or the ++JCLIN MCS.

System Action: Processing continues.

Programmer Response: Do **one** of the following:

- Check the member name, correct it if necessary, and rerun the job.
- Add the member to the SMPPARM data set and rerun the job.
- Check the SMPPARM DD statement to ensure that it identifies the proper data set name. If not, correct it and rerun the job.
- If message GIM55503I appears, determine whether the default method caused any errors. If so, use UCLIN to delete the incorrect entry.

  The following is an example of how to use UCLIN to delete a MAC entry:

```
UCLIN.
DEL MAC(macname)
    ...
    GENASM(genasm).
ENDUCL.
```

where:
- *macname* is the name of the macro from GIM55503I.
- *genasm* is the name of the ASSEM entry from GIM55503I.

GIM55503I *charstr* **FOUND IN ASSEMBLY** *asmname* **INPUT IS CONSIDERED A MACRO BECAUSE IT IS NOT A DEFINED OPCODE.**

Explanation:

charstr	character string
asmname	name of the assembly

An assembler instruction could not be found either in the list of definitions provided by a user-supplied OPCODE member or in the SMP/E default set of OPCODE definitions.

System Action: Processing continues.

Programmer Response: Do **one** of the following:
- Fix the error shown in GIM55301W, GIM55401W, or GIM55402W.
- Determine whether the default method caused any errors. If so, use UCLIN to delete the incorrect entry.

  The following is an example of how to use UCLIN to delete a MAC entry:

```
UCLIN.
DEL MAC(macname)
    ...
    GENASM(genasm).
ENDUCL.
```

  where:
  - *macname* is the name of the macro.
  - *genasm* is the name of the ASSEM entry.

GIM55601I *dataset* **WAS NOT DEFINED. PROCESSING CONTINUES USING SYSPRINT INSTEAD.**

Explanation:

dataset	ddname of the missing library or the ddname/zone where the allocation was used

During the invocation of a utility program, SMP/E used the PRINT value in the associated UTILITY entry to determine what SYSPRINT data set to use. However, there was no DD statement or DDDEF entry to allocate the indicated data set. Instead of stopping utility processing, SMP/E used the SYSPRINT data set for the output.

This is normal processing because the PRINT subentry is optional in the UTILITY entry.

System Action: Processing continues.

Programmer Response: You do not need to do anything for the current SMP/E run. However, you can fix the problem for subsequent processing by doing one of the following for the missing data set:
- Add a DDDEF entry to the applicable zones
- Add a DD statement to the JCL used to run SMP/E

Note: If this message was issued during the cross-zone processing of APPLY or RESTORE, the only way to fix the problem for future SMP/E runs is to add a DDDEF entry. Adding a DD statement will **not** resolve the problem.

GIM55801S **THE** *command* **COMMAND SPECIFIED THE ZONE NAME** *zonename***. THIS DOES NOT MATCH THE ZONE SPECIFIED ON THE PREVIOUS SET COMMAND.**

Explanation:

command	SMP/E command
zonename	zone name

System Action: Command processing stops.

Programmer Response: Check which zone name is correct. Fix the error and rerun the job.

GIM55802S **THE** *command* **COMMAND SPECIFIED A** *zonetype* **ZONE. THIS CONFLICTS WITH THE ZONE TYPE OF THE ZONE SPECIFIED ON THE PREVIOUS SET COMMAND.**

Explanation:

command	SMP/E command
zonetype	DLIB or TARGET

The zone type specified on the command being processed conflicts with the zone type specified on the previous SET command. This message may be issued when an operand conflicts with the previous SET command — for example, when the ZONEDELETE command specifies the TARGETZONE operand and the previous SET command specifies a DLIB zone.

System Action: Command processing stops.

Programmer Response: Check which zone type is correct. Fix the error and rerun the job.

GIM55901E **THE CONCAT OPERAND IS MUTUALLY EXCLUSIVE WITH ALL OTHER DDDEF ENTRY OPERANDS.**

Explanation: A UCL statement for a DDDEF entry specified the CONCAT operand along with other

operands. This is not allowed.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Decide whether you want to define a real library or a concatenation. Specify the appropriate operands and rerun the job.

GIM56001E SMP/E DOES NOT ALLOW MORE THAN 123 DDDEFS ON THE CONCAT OPERAND.

Explanation: More than 123 names were specified on the CONCAT operand. This is more than SMP/E allows to be defined in a DDDEF.

Note: DFSMS/MVS sets its own limits on how many data sets can be concatenated. The actual limit depends on the kind of data sets to be concatenated (partitioned data sets or PDSEs) and the total number of extents for partitioned data sets. The DFSMS/MVS limit will very likely be less than 123 data sets. Therefore, although specifying 123 or fewer names on the CONCAT operand will prevent SMP/E from issuing message GIM56001E, other error messages (such as GIM54701E) may be issued later if the DFSMS/MVS limit is exceeded.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Fix the statement to reduce the number of DDDEF names specified in the CONCAT operand to the **lesser** of 123 or the limit set by DFSMS/MVS. (See *z/OS DFSMS: Using Data Sets*, SC26-7410, to determine the maximum number of data sets you can concatenate.) Then rerun the job.

GIM56101S A ZONE SPECIFIED ON THE *operand* OPERAND MUST BE A *zonetype1* ZONE. *zonename* IS A *zonetype2* ZONE.

Explanation:

operand	PURGE or TARGETZONE
zonetype1	DLIB or TARGET
zonename	zone name
zonetype2	TARGET or DLIB

The indicated zone is not the correct type for the operand on which it was specified.

System Action: Command processing stops.

Programmer Response: Either:
- Remove the zone from the indicated operand, or
- Replace it with a zone of the correct type for that operand, or
- Replace the indicated operand with an operand compatible with the specified zone.

Then rerun the job.

GIM56301S RELATED ZONE *zonename* MUST BE A *zonetype* ZONE.

Explanation:

zonename	zone name
zonetype	zone type

System Action: Command processing stops.

Programmer Response: Check that the related zone for the zone being processed is the correct type:
- If a DLIB zone is being processed, the related zone must be a target zone.
- If a target zone is being processed, the related zone must be a DLIB zone.

Specify the correct type of zone as the related zone. Then rerun the job.

GIM56401S *name* HAS ALREADY BEEN SPECIFIED ON EITHER THE SELECT OR EXCLUDE OPERAND.

Explanation:

name	SYSMOD ID or FMIDSET

The indicated SYSMOD or FMIDSET name was repeated in one of the following ways:

1. It was specified more than once on the SELECT operand,
2. It was specified more than once on the EXCLUDE operand, or
3. It was specified on both the SELECT and EXCLUDE operands.

None of these are allowed.

System Action: Command processing stops.

Programmer Response: Specify the SYSMOD or FMIDSET name only once. Then rerun the job.

GIM56402E SYSMOD *sysmod* WAS SPECIFIED MORE THAN ONCE ON A ++ASSIGN MCS. THIS IS NOT ALLOWED.

Explanation:

sysmod	SYSMOD ID

System Action: Processing stops for the ++ASSIGN statement. Processing continues with the next MCS statement.

Programmer Response: Use UCLIN to add the source ID to the entry for the affected SYSMOD(s).

GIM56403S SYSMOD ID *sysmod* HAS ALREADY BEEN SPECIFIED FOR THE *reason* REASON ID ON THE HOLDSYSTEM BYPASS OPTION.

Explanation:

sysmod	SYSMOD ID

reason reason ID

The indicated SYSMOD ID was specified more than once for the indicated reason ID on the HOLDSYSTEM BYPASS option. If the reason ID is specified more than once for the HOLDSYSTEM BYPASS option, be sure that the SYSMOD ID appears in only one of the SYSMOD ID lists associated with each reason ID specification.

System Action: Command processing stops.

Programmer Response: Remove the incorrect value. Then rerun the job.

GIM56404S *value* **HAS ALREADY BEEN SPECIFIED ON THE** *option* **BYPASS OPTION.**

Explanation:
value reason ID, SYSMOD ID or class
option BYPASS option

The indicated value was specified more than once on the indicated BYPASS option.

Notes:

1. For the HOLDSYSTEM BYPASS option, the reason ID may be specified multiple times only if each occurrence identifies a list of exception SYSMOD IDs. A reason ID may not appear both by itself and with a list of SYSMOD IDs. Additionally, a SYSMOD ID may not appear more than once for any given reason ID.
2. For the XZIFREQ BYPASS option, the SYSMOD id may be specified multiple times only if each occurrence identifies an associated zone name. A SYSMOD ID may not appear both by itself and as part of a SYSMOD/zone pair. Additionally, each SYSMOD/zone pair must be unique.

System Action: Command processing stops.

Programmer Response: Remove the incorrect value. Then rerun the job.

GIM56405S **ZONE** *zonename* **HAS ALREADY BEEN SPECIFIED FOR THE** *sysmod* **SYSMOD ID ON THE XZIFREQ BYPASS OPTION.**

Explanation:
zonename zone name
sysmod SYSMOD ID

The indicated zone name was specified more than once for the indicated SYSMOD ID on the XZIFREQ BYPASS option.

System Action: Command processing stops.

Programmer Response: Remove the incorrect value. Then rerun the job.

GIM56501W **THE** *subtype* **SUBENTRY WAS ADDED INSTEAD OF REPLACED BECAUSE IT DID NOT EXIST.**

Explanation:
subtype subentry type

SMP/E could not find the subentry specified on the UCLIN REP command. It, therefore, assumes you wanted to add the subentry.

System Action: SMP/E adds the subentry.

Programmer Response: None.

GIM566I **JOB** *jobname* **HAS BEEN WAITING** *mm* **MINUTES FOR PACKAGE ID RESOURCE** *package***.**

Explanation:
jobname job name
mm time the task has been waiting (a multiple of 30 minutes)
package package id value

The PROCESS=WAIT parameter was specified on the EXEC statement for SMP/E, which indicates that SMP/E should wait for a required resource if it is not immediately available. The job being processed needs the indicated package id resource, which is currently being used by another iteration of SMP/E. The job will continue to wait until the resource is available or until you cancel the job.

System Action: None.

Programmer Response: None, unless you decide to cancel the job.

GIM56601E **ENTRY** *entname* **WAS NOT ADDED BECAUSE IT ALREADY EXISTS.**

Explanation:
entname entry name

The indicated entry already exists, so SMP/E cannot add it.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Specify the correct entry name and rerun the job.

GIM56701E **THERE IS AN ERROR IN A ++***mcstype* **MCS FOR SYSMOD** *sysmod* **IN SMPHOLD.**

Explanation:
mcstype MCS type (++HOLD or ++RELEASE)
sysmod SYSMOD ID

Message GIM203*xx* immediately precedes this message and shows that the line immediately before that message has a syntax error.

System Action: Processing stops for the MCS that is in error.

Programmer Response: Check the operands on the indicated MCS. Fix the error and rerun the job.

GIM56702E THERE IS AN ERROR IN A ++*mcstype* MCS IN SMPHOLD. THE SYSMOD IS UNKNOWN.

Explanation:

mcstype MCS type (++HOLD or ++RELEASE)

Message GIM203*xx* immediately precedes this message and shows that the line immediately before that message has a syntax error.

System Action: Processing stops for the MCS that is in error.

Programmer Response: Check the operands on the indicated MCS. Fix the error and rerun the job.

GIM56703E THERE IS AN ERROR IN A ++NULL MCS IN SMPHOLD.

Explanation: Message GIM203*xx* immediately precedes this message and shows that the line immediately before that message has a syntax error.

System Action: Processing stops for the MCS that is in error.

Programmer Response: Check the operands on the indicated MCS. Fix the error and rerun the job.

GIM56704E THERE IS AN ERROR IN AN UNKNOWN MCS IN SMPHOLD.

Explanation: Message GIM203*xx* immediately precedes this message and shows that the line immediately before that message has a syntax error.

System Action: Processing stops for the MCS that is in error.

Programmer Response: Check the operands on the indicated MCS. Fix the error and rerun the job.

GIM56801E A SUBENTRY VALUE WAS SPECIFIED MORE THAN ONCE FOR THE *subtype* SUBENTRY. THIS IS NOT ALLOWED.

Explanation:

subtype subentry type

The same value was specified more than once on the indicated subentry.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Remove the duplicate value and rerun the statement.

GIM56901S A UCL STATEMENT FOR THE ZONEINDEX SUBENTRY CONTAINS AN ERROR. THE REMAINING UCL STATEMENTS UP TO THE ENDUCL COMMAND WERE NOT PROCESSED.

Explanation: SMP/E found an error while processing a ZONEINDEX change. Because it considers this a critical error that can cause unpredictable results, it did not process the subsequent UCL statements.

System Action: All UCLIN processing stops. However, SMP/E checks the syntax of the subsequent UCL statements.

Programmer Response: Fix the statement in error and rerun all the statements that were not processed.

GIM57001E THE PARM SUBENTRY EXCEEDS THE 100-CHARACTER MAXIMUM.

Explanation: More than 100 characters were specified on the PARM subentry. This is not allowed.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Fix the statement in error and rerun the job.

GIM57001T THE PARM SUBENTRY EXCEEDS THE 100-CHARACTER MAXIMUM.

Explanation: More than 100 characters were specified on the PARM subentry. This is not allowed.

System Action: SMP/E processing stops.

Programmer Response: Fix the statement in error and rerun the job.

GIM57101E CONCAT IS NOT ALLOWED WHEN OTHER SUBENTRIES ALREADY EXIST IN THE DDDEF ENTRY.

Explanation: The CONCAT operand was specified on a UCL statement to update an existing DDDEF entry that contains other subentries. This is not allowed.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Fix the statement in error and rerun the job.

GIM57201E OTHER SUBENTRIES ARE NOT ALLOWED WHEN CONCAT ALREADY EXISTS IN THE DDDEF ENTRY.

Explanation: Other operands were specified on a UCL statement to update an existing DDDEF entry that

contains the CONCAT subentry. This is not allowed.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Fix the statement in error and rerun the job.

GIM57301E THE UCL STATEMENT FOR THE *enttype* ENTRY MUST CONTAIN AT LEAST ONE SUBENTRY OPERAND.

Explanation:
enttype entry type

The indicated statement does not contain enough information for UCLIN processing.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Add the desired operands to the statement and rerun the job.

GIM57302S THE GIMDDALC CONTROL STATEMENT FOR THE *dataset* DATA SET MUST CONTAIN AT LEAST ONE OPERAND.

Explanation:
dataset ddname

A GIMDDALC control statement does not contain enough information for SMP/E to use the control statement. This message is immediately preceded by the line with the error.

Note: SMP/E checks only columns 1 through 72. If any data is specified beyond column 72, SMP/E ignores it and indicates an error in the statement.

System Action: Command processing stops.

Programmer Response: Correct the control statement and rerun the job.

GIM57303S THE GIMDDALC CONTROL STATEMENT FOR THE *dataset* DATA SET HAS ALREADY BEEN FOUND. ONLY ONE GIMDDALC CONTROL STATEMENT IS ALLOWED FOR EACH DATA SET.

Explanation:
dataset ddname

A GIMDDALC control statement for the indicated data set was already found in the GIMDDALC member. Only one control statement may be specified for each data set.

System Action: Command processing stops.

Programmer Response: Remove the duplicate control statement and rerun the job.

GIM57304S THE GIMEXITS CONTROL STATEMENT FOR THE *exitpnt* EXIT POINT HAS ALREADY BEEN FOUND. ONLY ONE GIMEXITS CONTROL STATEMENT IS ALLOWED FOR EACH EXIT POINT.

Explanation:
exitpnt exit point identifier

A GIMEXITS control statement for the indicated exit point was already found in the GIMEXITS member. Only one control statement may be specified for each exit point.

System Action: Command processing stops.

Programmer Response: Remove the duplicate control statement and rerun the job.

GIM57501E THE BACKUP OPERAND IS NOT ALLOWED WITH OTHER LIST OPERANDS.

Explanation: If BACKUP is specified on the LIST command, no other operands are allowed.

System Action: Command processing stops.

Programmer Response: Fix the error and rerun the job. For more information about the syntax of SMP/E commands, see *SMP/E Commands*.

GIM57601I DYNAMIC ALLOCATION FAILED FOR *library1* BECAUSE IT CONCATENATES THE *library2* DDDEF ENTRY, WHICH CONCATENATES OTHER DATA SETS. THIS IS NOT ALLOWED.

Explanation:
library1 ddname of the library being allocated
library2 ddname of the library being concatenated

The DDDEF entry for the library being allocated concatenated another library, which also concatenated libraries. This is not allowed. You cannot nest concatenated libraries.

System Action: Processing continues.

Programmer Response: If necessary in the future, fix the DDDEF entry for the library being allocated so that it does not include other concatenation DDDEF entries. Then rerun the job.

GIM57601W DYNAMIC ALLOCATION FAILED FOR *library1* BECAUSE IT CONCATENATES THE *library2* DDDEF ENTRY, WHICH CONCATENATES OTHER DATA SETS. THIS IS NOT ALLOWED.

Explanation:

library1	ddname of the library being allocated
library2	ddname of the library being concatenated

The DDDEF entry for the library being allocated concatenated another library, which also concatenated libraries. This is not allowed. You cannot nest concatenated libraries.

System Action: Processing continues.

Programmer Response: If necessary, fix the DDDEF entry for the library being allocated so that it does not include other concatenation DDDEF entries. Then rerun the job.

GIM57601E DYNAMIC ALLOCATION FAILED FOR *library1* BECAUSE IT CONCATENATES THE *library2* DDDEF ENTRY, WHICH CONCATENATES OTHER DATA SETS. THIS IS NOT ALLOWED.

Explanation:

library1	ddname of the library being allocated
library2	ddname of the library being concatenated

The DDDEF entry for the library being allocated concatenated another library, which also concatenated libraries. This is not allowed. You cannot nest concatenated libraries.

System Action: SYSMOD processing stops.

Programmer Response: Fix the DDDEF entry for the library being allocated so that it does not include other concatenation DDDEF entries. Then rerun the job.

GIM57601S DYNAMIC ALLOCATION FAILED FOR *library1* BECAUSE IT CONCATENATES THE *library2* DDDEF ENTRY, WHICH CONCATENATES OTHER DATA SETS. THIS IS NOT ALLOWED.

Explanation:

library1	ddname of the library being allocated
library2	ddname of the library being concatenated

The DDDEF entry for the library being allocated concatenated another library, which also concatenated libraries. This is not allowed. You cannot nest concatenated libraries.

System Action: Command processing stops.

Programmer Response: Fix the DDDEF entry for the library being allocated so that it does not include other concatenation DDDEF entries. Then rerun the job.

GIM57601T DYNAMIC ALLOCATION FAILED FOR *library1* BECAUSE IT CONCATENATES THE *library2* DDDEF ENTRY, WHICH CONCATENATES OTHER DATA SETS. THIS IS NOT ALLOWED.

Explanation:

library1	ddname of the library being allocated
library2	ddname of the library being concatenated

The DDDEF entry for the library being allocated concatenated another library, which also concatenated libraries. This is not allowed. You cannot nest concatenated libraries.

System Action: SMP/E processing stops.

Programmer Response: Fix the DDDEF entry for the library being allocated so that it does not include other concatenation DDDEF entries. Then rerun the job.

GIM57701T AN I/O ERROR OCCURRED DURING *command* PROCESSING FOR THE *dataset* DATA SET.

Explanation:

command	type of processing being done (READ, WRITE, STOW, BLDL, OPEN, CLOSE)
dataset	ddname of the data set.

SMP/E was processing a non-VSAM data set (a sequential data set or the directory of a partitioned data set) when it encountered an I/O error. This error may have been indicated by one of the following:

- The return code from a STOW or BLDL operation
- The SYNAD exit for READ and WRITE operations

System Action: SMP/E's action depends on the following:

- The data set where SMP/E detected the I/O error
- The processing that SMP/E had already done

Subsequent messages show SMP/E's action. The action may range from stopping SYSMOD processing to stopping SMP/E processing.

Programmer Response: Find the cause of the error from the message. Check for correct DCB information on the input DD statement (especially for a non-labeled tape). Fix the error and rerun the SMP/E job.

GIM57702T AN I/O ERROR OCCURRED DURING *command* PROCESSING FOR *memtype* MEMBER *memname* IN THE *dataset* DATA SET.

Explanation:

command	type of processing being done (READ, WRITE, STOW, BLDL, OPEN, CLOSE)
memtype	member type
memname	member name
dataset	ddname of the data set

SMP/E was processing a non-VSAM data set (a member of a partitioned data set) when it encountered an I/O error. This error may have been indicated by one of the following:
- The return code from a STOW or BLDL operation
- The SYNAD exit for READ and WRITE operations

Note that if the member name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: SMP/E's action depends on the following:
- The data set where SMP/E detected the I/O error
- The processing that SMP/E had already done

Subsequent messages show SMP/E's action. The action may range from stopping SYSMOD processing to stopping SMP/E processing.

Programmer Response: Find the cause of the error from the message. Fix the error and rerun the SMP/E job.

GIM57703T AN I/O ERROR OCCURRED DURING *command* PROCESSING FOR *memtype* MEMBER *memname* IN THE *dataset* DATA SET. APPROXIMATE TTR=*ttr*.

Explanation:

command	type of processing being done (READ, WRITE, STOW, BLDL, OPEN, CLOSE)
memtype	member type
memname	member name
dataset	ddname of the data set
ttr	TTR where the error was detected: • For READ operations, the TTR is from the first record for the indicated member. • For WRITE operations, the TTR is from the DCB where SMP/E did the last write.

SMP/E was processing a non-VSAM data set (a member of a partitioned data set) when it encountered an I/O error. This error may have been indicated by one of the following:
- The return code from a STOW or BLDL operation
- The SYNAD exit for READ and WRITE operations

System Action: SMP/E's action depends on the following:
- The data set where SMP/E detected the I/O error
- The processing that SMP/E had already done

Subsequent messages show SMP/E's action. The action may range from continuing processing to stopping SMP/E processing.

Programmer Response: Find the cause of the error from the message. Fix the error and rerun the SMP/E job.

GIM57801E THE *operand* OPERAND IS NOT ALLOWED ON THE *command* COMMAND.

Explanation:

operand	operand
command	SMP/E command

An operand that is not allowed was specified on the indicated command.

System Action: Command processing stops.

Programmer Response: Remove the operand from the command and rerun the job.

GIM57802I THE *operand* OPERAND IS NO LONGER ALLOWED FOR THE *command* COMMAND.

Explanation:

operand	operand
command	SMP/E command

The operand was supported in past releases of SMP/E, but is no longer used.

System Action: Processing continues.

Programmer Response: To prevent this message, remove the operand from the command.

GIM57802E THE *operand* OPERAND IS NO LONGER ALLOWED FOR THE *command* COMMAND.

Explanation:

operand	operand
command	SMP/E command

The operand was supported in past releases of SMP/E, but is no longer used.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Remove the operand, and rerun the command.

GIM57901S SMPCSI DATA SET *dataset* IS OUT OF SPACE.

Explanation:
dataset data set name

This message is issued in either of the following cases:
- The indicated SMPCSI data set is full.
- There is not enough room on the DASD where the indicated SMPCSI data set resides.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:
- If the indicated SMPCSI data set is full, use IDCAMS (the AMS utility) to obtain additional space. You have several options:
  - **REPRO.** Allocate a new, larger CSI and use the AMS REPRO command to copy the old data set that is out of space into the new one.
  - **ADDVOLUMES.** Use the AMS ALTER ADDVOLUMES command to add another volume to the CSI.
  - **EXPORT and IMPORT.** Use the AMS IMPORT and EXPORT commands to move the old CSI to a new, larger CSI.
    1. Export the SMPCSI data set to a sequential data set.
    2. Delete the original SMPCSI data set.
    3. Define a new SMPCSI data set with the same name and more space.
    4. Import the exported data set into the new data set.
- If there is not enough room on the DASD, delete and compress data sets on that DASD as appropriate to free up space.

Then rerun the job.

Note: If the job fails again after you have taken steps to get more space, the reason could be that SMP/E updated or created CSI entries before processing stopped for the original abend. If this was the case:
1. Check which element entries (including LMOD entries) were affected by the processing. The Element Summary report from your first attempt to run the job will help you with this step.
2. Use UCLIN to undo changes to updated entries and to delete any entries that were created.
3. Then rerun the job.

GIM58101I *progname* IS NOT ALLOWED FOR THE *utiltype* UTILITY BECAUSE IT IS NOT DEFINED IN GIMUTTBL.

Explanation:
progname program name
utiltype utility type

System Action: Processing continues.

Programmer Response: No action is required.

GIM58101W *progname* IS NOT ALLOWED FOR THE *utiltype* UTILITY BECAUSE IT IS NOT DEFINED IN GIMUTTBL.

Explanation:
progname program name
utiltype utility type

System Action: Cross-zone updates are not made for the zone named in the message. Processing continues with the next cross-zone.

Programmer Response: If you want to define the program for the indicated utility, add it to GIMUTTBL. For more information, see the GIMUTTBL chapter in *SMP/E Reference*. Then use the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to determine which cross-zone updates were not done. Create an SMP/E LINK command (or, for ++RENAME, a UCLIN command) to update the indicated load modules.

GIM58101S *progname* IS NOT ALLOWED FOR THE *utiltype* UTILITY BECAUSE IT IS NOT DEFINED IN GIMUTTBL.

Explanation:
progname program name
utiltype utility type

System Action: Command processing stops.

Programmer Response: If you want to define the program for the indicated utility, add it to GIMUTTBL. For more information, see the GIMUTTBL chapter in *SMP/E Reference*. Then rerun the job.

GIM58201S ZONE *zonename* WAS NOT RENAMED BECAUSE THERE IS NO ZONEINDEX SUBENTRY FOR IT.

Explanation:
zonename zone name

System Action: ZONERENAME processing stops.

Programmer Response: Do **one** of the following:
- Make sure the zone is specified correctly on the command.
- Make sure there is a ZONEINDEX subentry for the zone in the GLOBALZONE entry.

Then rerun the command.

GIM58301S *zonename1* **WAS NOT RENAMED BECAUSE THERE IS ALREADY A ZONEINDEX SUBENTRY FOR THE NEW NAME** *zonename2***.**

Explanation:

zonename1	current zone name
zonename2	new zone name

SMP/E tried to rename the indicated zone but could not. The new zone name is already defined by a ZONEINDEX subentry.

System Action: ZONERENAME processing stops.

Programmer Response: Do **one** of the following:
- Make sure the zone is specified correctly on the command.
- Check whether the indicated ZONEINDEX subentry defines an existing zone.
  - If so, choose a different new name for the zone being renamed.
  - If not, use UCLIN to delete the ZONEINDEX subentry for the new zone name.

Then rerun the job.

GIM58401S *zonename1* **WAS NOT RENAMED BECAUSE THE NEW ZONE NAME** *zonename2* **ALREADY EXISTS IN SMPCSI DATA SET** *dataset***.**

Explanation:

zonename1	name of the zone being renamed
zonename2	new zone name
dataset	SMPCSI data set name associated with the ddname specified on the NEWDATASET operand

SMP/E tried to rename the zone that was copied into the indicated data set. However, there are already records for a zone with the new name in that data set.

System Action: ZONERENAME processing stops.

Programmer Response: To get rid of the unwanted records, do the following:
1. Use UCLIN to add a ZONEINDEX subentry for the new zone name and the indicated data set.
2. Make sure that you do not need the records in the indicated data set.
3. Use ZONEDELETE to delete the new zone name from the data set.
4. Rerun the ZONERENAME command.

GIM58501I **ZONE** *zonename1* **CANNOT BE RENAMED TO** *zonename2***.**

Explanation:

zonename1	current zone name
zonename2	new zone name

The indicated zone could not be renamed. Previous messages show why not.

System Action: ZONERENAME processing stops.

Programmer Response: Correct the errors shown by previous error messages and rerun the ZONERENAME command.

GIM58601I **ZONE** *zonename1* **WAS RENAMED TO** *zonename2***.**

Explanation:

zonename1	old zone name
zonename2	new zone name

The indicated zone was successfully renamed.

System Action: Processing continues.

Programmer Response: None.

GIM58701I **ZONE** *zonename1* **WAS RENAMED TO** *zonename2* **IN SMPCSI DATA SET** *dataset***.**

Explanation:

zonename1	old zone name
zonename2	new zone name
dataset	SMPCSI data set name associated with the ddname specified on the NEWDATASET operand

The indicated zone was successfully renamed in the data set specified on the NEWDATASET operand.

System Action: Processing continues.

Programmer Response: None.

GIM58801S **THERE IS A ZONEINDEX SUBENTRY FOR** *zonename* **BUT THE ZONE DOES NOT EXIST IN THE SMPCSI DATA SET.**

Explanation:

zonename	zone name

SMP/E found a ZONEINDEX subentry for the indicated zone. However, there are no entries for the zone (no zone definition entry and no other types of entries) in the SMPCSI data set that SMP/E checked.
- If NEWDATASET was specified, SMP/E checked the specified SMPCSI data set.
- Otherwise, SMP/E checked the SMPCSI data set specified in the ZONEINDEX subentry.

System Action: ZONERENAME processing stops.

Programmer Response: Make sure to specify the correct zone to be renamed. Use the LIST command to check the following:
- If you specify the NEWDATASET operand, make sure the zone exists in the SMPCSI data set specified by NEWDATASET.

- If you do not specify NEWDATASET, make sure the zone exists in the SMPCSI data set specified by the ZONEINDEX subentry in the global zone.

Then rerun the job.

GIM59001S ZONE *zonename* WAS NOT RENAMED BECAUSE THE NEW ZONE NAME AND THE OLD ZONE NAME ARE THE SAME.

Explanation:
zonename zone name

System Action: ZONERENAME processing stops.

Programmer Response: Specify a different value for the new name of the zone and rerun the job.

GIM59201S THE GLOBAL ZONE CANNOT BE RENAMED.

Explanation: The ZONERENAME command tried to change the name of the global zone. This is not allowed. The name of the global zone must be *GLOBAL*.

System Action: ZONERENAME processing stops.

Programmer Response: None. You cannot rename the global zone.

GIM59202S "GLOBAL" IS NOT ALLOWED AS THE NEW NAME FOR A ZONE.

Explanation: The ZONERENAME command tried to change the name of a zone to *GLOBAL*. This is not allowed. Only the global zone may have that name.

System Action: ZONERENAME processing stops.

Programmer Response: Change the value specified as the new name for the zone and rerun the job.

GIM59203E "GLOBAL" IS NOT ALLOWED AS A ZONESET NAME.

Explanation: *GLOBAL* was specified as a ZONESET name on the command being processed. This is not allowed.

System Action: Processing of the UCL statement stops. Processing continues with the next UCL statement.

Programmer Response: Specify a valid ZONESET name and rerun the job.

GIM59301S THE TOTYPE(TARGET) OPERAND WAS NOT PROCESSED BECAUSE *zonename* IS ALREADY DEFINED AS A TARGET ZONE IN THE ZONEINDEX SUBENTRY.

Explanation:
zonename zone name

The TOTYPE(TARGET) operand was specified on the ZONERENAME command to change the zone being renamed from a distribution zone to a target zone. However, the ZONEINDEX subentry for that zone indicates that it is already a target zone.

System Action: ZONERENAME processing stops.

Programmer Response: Do **one** of the following:

- If the zone you want to rename is a distribution zone, change the ZONEINDEX subentry to show that the zone is a distribution zone.
- If the zone you want to rename is a target zone, remove the TOTYPE(TARGET) operand from the ZONERENAME command.

Then rerun the job.

GIM59302S THE TOTYPE(TARGET) OPERAND WAS NOT PROCESSED BECAUSE *zonename* IS ALREADY DEFINED BY A TARGET ZONE ENTRY.

Explanation:
zonename zone name

The TOTYPE(TARGET) operand was specified on the ZONERENAME command to change the zone being renamed from a distribution zone to a target zone. However, there is a TARGETZONE entry for that zone, which indicates that it is already a target zone.

System Action: ZONERENAME processing stops.

Programmer Response: Do **one** of the following:

- If the zone you want to rename is a distribution zone, delete the TARGETZONE entry and create a DLIBZONE entry to show that the zone is a distribution zone.
- If the zone you want to rename is a target zone, remove the TOTYPE(TARGET) operand from the ZONERENAME command.

Then rerun the job.

GIM59401S THE NEWDATASET OPERAND MUST NOT SPECIFY THE DATA SET THAT CURRENTLY CONTAINS THE ZONE TO BE RENAMED.

Explanation: The NEWDATASET operand indicates that a zone was copied into a new data set, and that the copy is to be renamed. However, the value that was specified on the NEWDATASET operand is the same as the data set that currently contains the original zone. This is not allowed.

System Action: ZONERENAME processing stops.

Programmer Response: Do **one** of the following:

- If you want to rename the original zone in its current data set, delete the NEWDATASET operand from the ZONERENAME command and use the SAMEDATASET operand instead.
- If you want to rename the copy of the zone in the new data set, make sure the new data set name is different from the name of the data set that contains the original zone.

Then rerun the job.

GIM59501I OPTIONS SUBENTRY *subentval* WAS ADDED.

Explanation:
subentval subentry value

When SMP/E renamed the zone being processed, it added the indicated OPTIONS subentry to the zone definition entry.

System Action: None.

Programmer Response: None.

GIM59502I THE OPTIONS SUBENTRY WAS CHANGED TO *subentval*.

Explanation:
subentval new subentry value

When SMP/E renamed the zone being processed, it also changed the OPTIONS subentry in the zone definition entry.

System Action: None.

Programmer Response: None.

GIM59503I RELATED SUBENTRY *subentval* WAS ADDED.

Explanation:
subentval subentry value

When SMP/E renamed the zone being processed, it added the indicated RELATED subentry to the zone definition entry.

System Action: None.

Programmer Response: None.

GIM59504I THE RELATED SUBENTRY WAS CHANGED TO *subentval*.

Explanation:
subentval new subentry value

When SMP/E renamed the zone being processed, it also changed the RELATED subentry in the zone definition entry.

System Action: None.

Programmer Response: None.

GIM59505I A ZONE DEFINITION ENTRY WAS ADDED FOR *zonename*.

Explanation:
zonename zone name

When SMP/E processed the indicated zone, it found that the zone was not empty but did not contain a zone definition entry. Because the OPTIONS or RELATED operand was specified on the ZONERENAME command, SMP/E added a zone definition entry when it renamed the zone.

System Action: None.

Programmer Response: None.

GIM59601I ENQ WAS INITIATED FOR EXCLUSIVE USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of SMPCSI or SMPPTS data set or a package id value
command SMP/E command

GIMSMP issues this message to SMPLOG. The message indicates that the named SMP/E command is about to try to get the named resource, but that there is a possibility that another SMP/E command may already have the resource.

System Action: Processing continues.

Programmer Response: None.

GIM59602I ENQ WAS SUCCESSFUL FOR EXCLUSIVE USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of SMPCSI or SMPPTS data set or a package id value
command SMP/E command

GIMSMP issues this message to SMPLOG. The message indicates that the named SMP/E command has acquired the named resource.

System Action: Processing continues.

Programmer Response: None.

GIM59603I ENQ WAS SUCCESSFUL FOR SHARED USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of the SMPCSI data set or SMPPTS data set

command SMP/E command

SMP/E writes this message to the SMPLOG data set each time it enqueues an SMPCSI data set.

System Action: Processing continues.

Programmer Response: None.

GIM59604S ENQ FAILED FOR EXCLUSIVE USE OF *resource* FOR *command* PROCESSING.

Explanation:
resource resource name (a package id value)
command SMP/E command

The indicated SMP/E command needed the named resource in order to perform the transfer for a network package. Since another iteration of SMP/E was using the resource, the current command cannot use it and must therefore stop its processing.

System Action: Processing stops.

Programmer Response: If the job was not submitted by accident, rerun it when the resource is not being used by another iteration of SMP/E.

GIM59604W ENQ FAILED FOR EXCLUSIVE USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of the SMPCSI data set or SMPPTS data set
command SMP/E command

SMP/E writes this message to the SMPLOG data set each time it enqueues an SMPCSI data set.

System Action: The current phase of the command being processed stops. Command processing may stop.

Programmer Response: Check that the data set is not being used by another user.

GIM59605S ENQ FAILED FOR SHARED USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of the CSI data set or the SMPPTS data set
command SMP/E or GIMAPI command

SMP/E writes this message to the SMPLOG for batch command processing or places it in a message buffer for GIMAPI command processing if another user had exclusive use of the data set.

System Action: The current phase of the command being processed stops. Command processing may stop.

Programmer Response:

Check that the data set is not being used by another user.

GIM59605W ENQ FAILED FOR SHARED USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of the SMPCSI data set or SMPPTS data set
command SMP/E command

SMP/E writes this message to the SMPLOG data set if another user had exclusive use of the data set.

System Action: The current phase of the command being processed stops. Command processing may stop.

Programmer Response: Check that the data set is not being used by another user.

GIM59606I DEQ WAS SUCCESSFUL FOR EXCLUSIVE USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of SMPCSI or SMPPTS data set or a package id value
command SMP/E command

GIMSMP issues this message to SMPLOG. The message indicates that the named SMP/E command has freed a previously gotten resource.

System Action: Processing continues.

Programmer Response: None.

GIM59607I DEQ WAS SUCCESSFUL FOR SHARED USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of the SMPCSI data set or SMPPTS data set
command SMP/E command

SMP/E writes this message to the SMPLOG data set each time it dequeues an SMPCSI data set.

System Action: Processing continues.

Programmer Response: None.

GIM59608I ENQ WAS INITIATED FOR SHARED USE OF *dataset* FOR *command* PROCESSING.

Explanation:
dataset name of the SMPCSI data set or SMPPTS data set
command SMP/E command

SMP/E writes this message to the SMPLOG data set each time it enqueues an SMPCSI data set.

System Action: Processing continues.

Programmer Response: None.

GIM59701I THE GLOBAL ZONE WAS UPDATED WITH CHANGES FROM THE *zonename* ZONE.

Explanation:
zonename zone name

If this message was issued for APPLY, ACCEPT, or RESTORE processing, SMP/E updated the global zone with changes for this command.

If this message was issued for SET processing, then during previous processing for the indicated zone, SMP/E was not able to get access to the global zone in order to make the required updates. As a result, these pending changes had been saved in the zone. The zone was processed again by the SET command, and this time SMP/E was able to get access to the global zone. These changes have now been added to the global zone.

System Action: None.

Programmer Response: None.

GIM59801W THE GLOBAL ZONE WAS NOT UPDATED WITH CHANGES FROM THE *zonename* ZONE.

Explanation:
zonename zone name

At the end of processing for the indicated zone, SMP/E was not able to get access to the global zone in order to make the required updates. As a result, these pending changes have been saved in the zone. The next time the zone is processed, SMP/E will try to add the changes to the global zone.

System Action: None.

Programmer Response: None is required. However, if you do not want to wait to synchronize the indicated zone with the global zone, do the following now:
1. Specify the indicated zone on the SET command.
2. Run a command (such as LIST) to process that zone.

GIM59801E THE GLOBAL ZONE WAS NOT UPDATED WITH CHANGES FROM THE *zonename* ZONE.

Explanation:
zonename zone name

At the end of processing for the indicated zone, SMP/E was not able to get access to the global zone in order to make the required updates. As a result, these pending changes have been saved in the zone. The next time the zone is processed, SMP/E will try to add the changes to the global zone.

System Action: None.

Programmer Response: None is required. However, if you do not want to wait to synchronize the indicated zone with the global zone, do the following now:
1. Specify the indicated zone on the SET command.
2. Run a command (such as LIST) to process that zone.

GIM59801S THE GLOBAL ZONE WAS NOT UPDATED WITH CHANGES FROM THE *zonename* ZONE.

Explanation:
zonename zone name

At the end of processing for the indicated zone, SMP/E was not able to get access to the global zone in order to make the required updates. As a result, these pending changes have been saved in the zone. The next time the zone is processed, SMP/E will try to add the changes to the global zone.

System Action: None.

Programmer Response: None is required. However, if you do not want to wait to synchronize the indicated zone with the global zone, do the following now:
1. Specify the indicated zone on the SET command.
2. Run a command (such as LIST) to process that zone.

GIM59801T THE GLOBAL ZONE WAS NOT UPDATED WITH CHANGES FROM THE *zonename* ZONE.

Explanation:
zonename zone name

At the end of processing for the indicated zone, SMP/E was not able to get access to the global zone in order to make the required updates. As a result, these pending changes have been saved in the zone. The next time the zone is processed, SMP/E will try to add the changes to the global zone.

System Action: None.

Programmer Response: None is required. However, if you do not want to wait to synchronize the indicated zone with the global zone, do the following now:
1. Specify the indicated zone on the SET command.
2. Run a command (such as LIST) to process that zone.

GIM599I **JOB** *jobname* **HAS BEEN WAITING** *mm* **MINUTES FOR DATA SET** *dataset***.**

Explanation:

jobname	job name
mm	time the task has been waiting (a multiple of 30 minutes)
dataset	name of the SMPCSI data set or the SMPPTS data set

The PROCESS=WAIT parameter was specified on the EXEC statement for SMP/E, which indicates that SMP/E should wait for a required SMPCSI or SMPPTS data set if it is not available.

The job being processed needs the indicated data set, which is currently being used. The job will continue to wait until all the required data sets are available or until you cancel the job.

System Action: None.

Programmer Response: None, unless you decide to cancel the job.

GIM60001S *command* **PROCESSING FAILED BECAUSE AN SMPCSI OR SMPPTS DATA SET WAS NOT AVAILABLE.**

Explanation:

command	SMP/E command

The PROCESS=END parameter was specified on the EXEC statement for SMP/E, which indicates that SMP/E should wait 10 minutes for a required data set and fail if the data set is not available after that time is up. One of the following happened:

- The job being processed needs one or more SMPCSI data sets. However, one or more of these data sets has been in use for the past 10 minutes.
- The job being processed needs the SMPPTS data set or an SMPPTS spill data set. However, the data set has been in use for the past 10 minutes.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:

- Wait until the required data set is available.
- Change **PROCESS=END** to **PROCESS=WAIT.**
- Remove the **PROCESS=END** parameter and allow the default of **PROCESS=WAIT** to take effect.

Then rerun the job.

GIM60101S **PURGE IS NOT ALLOWED ON A ZONEEXPORT COMMAND FOR THE GLOBAL ZONE.**

Explanation: PURGE was specified on a ZONEEXPORT command for the global zone. The PURGE operand indicates that when the specified zone is exported, SMP/E should delete that zone from its SMPCSI data set. However, PURGE is not allowed for the global zone.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:

- If you want to export the global zone, remove the PURGE operand from the command and rerun the job.
- If you want to delete the global zone, you must use Access Method Services to delete the SMPCSI data set that contains it.

GIM60201I **ZONE** *zonename* **WAS SUCCESSFULLY WRITTEN TO OUTFILE** *dataset***.**

Explanation:

zonename	zone name
dataset	ddname of the output data set

SMP/E successfully copied the indicated zone to the OUTFILE data set.

You should be aware that after this message is issued, an error may occur when SMP/E is deleting the exported zone. However, this does not affect the data that was copied to the OUTFILE data set. This information can still be processing by the ZONEIMPORT command.

System Action: None.

Programmer Response: None.

GIM60301S **THE ZONE TYPES ARE INCOMPATIBLE. THE RECEIVING ZONE** *zonename1* **IS A** *zonetype1* **ZONE AND THE INPUT ZONE** *zonename2* **IS A** *zonetype2* **ZONE.**

Explanation:

zonename1	first zone name
zonetype1	first zone type
zonename2	second zone name
zonetype2	second zone type

The indicated zone types are not a valid combination. The following combinations are allowed for input zones and receiving zones:

Input Zone	Receiving Zone
Distribution zone	Distribution zone
Distribution zone	Target zone
Target zone	Target zone

Notes:

1. You cannot copy or import a target zone into a distribution zone.
2. A global zone cannot be an input or receiving zone.

System Action: Command processing stops. No data is moved.

Programmer Response: Specify a valid combination of zones and rerun the job.

GIM60401S THE SET BOUNDARY ZONE *zonename1* AND THE INTO ZONE *zonename2* HAVE DIFFERENT NAMES.

Explanation:
zonename1 first zone name
zonename2 second zone name

The INTO zone for the command being processed is supposed to be the same as the zone specified on the SET command. However, the names are different.

System Action: Command processing stops.

Programmer Response: Specify the same zone on the INTO operand and the SET command. Then rerun the job.

GIM60402S THE SET BOUNDARY ZONE *zonename1* AND THE INPUT ZONE *zonename2* HAVE DIFFERENT NAMES.

Explanation:
zonename1 first zone name
zonename2 second zone name

The input zone for the command being processed is supposed to be the same as the zone specified on the SET command. However, the names are different.

System Action: Command processing stops.

Programmer Response: Specify the same zone on the SET command and the command being processed. Then rerun the job.

GIM60501S THE INTO ZONE *zonename* ALREADY EXISTS IN SMPCSI DATA SET *dataset*.

Explanation:
zonename zone name
dataset data set name

When SMP/E is importing a zone, the receiving zone (INTO zone) cannot already exist in the SMPCSI data set being processed.

System Action: Command processing stops. SMP/E does not move any data.

Programmer Response: Do **one** of the following:
- Specify a different receiving zone on the command.
- Delete the indicated receiving zone.

Then rerun the command.

GIM60601S THE GLOBAL ZONE WAS NOT IMPORTED INTO *dataset* BECAUSE THAT DATA SET ALREADY CONTAINS A ZONE.

Explanation:
dataset data set name

When the global zone is being imported, no other zones may be defined in the receiving SMPCSI data set. However, the indicated SMPCSI data set already contains a zone.

System Action: Command processing stops. SMP/E does not move any data.

Programmer Response: Do **one** of the following:
- Use a different receiving SMPCSI.
- Delete the existing zones.

Then rerun the command.

GIM60701S ZONEIMPORT PROCESSING FOR ZONE *zonename* FROM INFILE *dataset* IS INCOMPLETE.

Explanation:
zonename zone name
dataset ddname of the INFILE data set

An end-of-file on the INFILE data set occurred before SMP/E read the special terminating data record. This could be due to an I/O error or because ZONEEXPORT stopped before it was complete.

System Action: Command processing stops. SMP/E deleted all data that it imported.

Programmer Response: Do **one** of the following:
- If a previous error shows that an I/O error on the INFILE data set caused the problem, use a different input device to rerun the job.
- If SMP/E produced an I/O error, check the results of the ZONEEXPORT processing that created the INFILE data set. If the zone was not successfully exported, rerun the ZONEEXPORT command to create a new INFILE data set.
- If you cannot determine the problem, export the original zone to create a new INFILE data set.

Then rerun the ZONEIMPORT command.

GIM60801I THE ZONE TYPE FOR *zonename* WAS CHANGED FROM DLIB TO TARGET.

Explanation:
zonename zone name

The zone being processed was changed from a DLIB zone to a target zone.

System Action: None.

Programmer Response: None.

GIM60801W THE ZONE TYPE OF *zonename* WAS CHANGED FROM DLIB TO TARGET.

Explanation:
zonename zone name

The zone being processed was changed from a DLIB zone to a target zone.

System Action: None.

Programmer Response: None.

GIM60901W THERE IS NO ZONE DEFINITION ENTRY FOR ZONE *zonename*.

Explanation:
zonename zone name

During the cross-zone update phase of the APPLY or RESTORE command, SMP/E could not find a zone definition entry for the zone being processed.

System Action: Cross-zone updates are not made for the zone named in the message. Processing continues with the next cross-zone.

Programmer Response: Do the following:

1. Make sure the indicated zone is the one you want to process.
2. If necessary, create a zone definition entry for the zone.
3. The Cross-Zone Summary report and the MOVE/RENAME/DELETE report will identify what cross-zone updates need to be done for the zone named in this message. Use a combination of the following to complete the unfinished cross-zone updates identified:
   - The LINK command
   - UCLIN updates
   - The linkage editor (outside of SMP/E)

GIM60901S THERE IS NO ZONE DEFINITION ENTRY FOR ZONE *zonename*.

Explanation:
zonename zone name

SMP/E could not find a zone definition entry for the zone being processed.

System Action: Command processing stops.

Programmer Response: Do the following:

1. Make sure the indicated zone is the one you want to process.
2. If necessary, create a zone definition entry for the zone.
3. Rerun the command.

GIM61001S *enttype* IS NOT ALLOWED ON THE ZONEEDIT COMMAND. THE ENTRY TYPE MUST BE DDDEF, UTILITY, OR XZENTRIES.

Explanation:
enttype entry type

The only entry types that may be specified on a ZONEEDIT command are DDDEF, UTILITY, and XZENTRIES. None of these was specified on the command being processed.

System Action: Command processing stops.

Programmer Response: Do **one** of the following:

- To change DDDEF, UTILITY, or XZENTRIES information, reenter the ZONEEDIT command with the correct entry type.
- To change any other type of entry, use the UCLIN command.

GIM61101S THE *operand* OPERAND IS NOT ALLOWED WHEN ZONEEDIT *enttype* IS REQUESTED.

Explanation:
operand operand name
enttype entry type

The command being processed specified an operand that is not valid for the entry type specified on the ZONEEDIT command.

System Action: Command processing stops. No changes were made.

Programmer Response: Specify an operand that is allowed for the entry type chosen on the ZONEEDIT command and rerun the job. For more information on the operands that may be specified for a particular entry, see *SMP/E Commands*.

GIM61201E *elmname* WAS NOT MOVED BECAUSE SYSMOD *sysmod* CONTAINS MORE THAN ONE ++MOVE MCS TO MOVE *elmname* INTO THE SAME LIBRARY *library*. THIS IS NOT ALLOWED.

Explanation:
elmname element name
sysmod SYSMOD ID
library ddname of the library

The same element cannot be moved from two different libraries into the same library.

System Action: SYSMOD processing stops.

Programmer Response:

- If there are two copies of the element, keep the ++MOVE MCS for one of them. Then delete the other by doing the following:

- For an LMOD, use a ++DELETE MCS.
- For other elements, specify the DELETE operand on the ++MOD, ++MAC, or ++SRC MCS.

- If there is only one copy of the element, remove the extra ++MOVE MCS.

Then rerun the job.

GIM61301S THE NAME OF THE ZONE ON THE ZONEIMPORT COMMAND (*zonename1*) IS DIFFERENT FROM THE NAME OF THE ZONE IN THE INFILE DATA SET (*zonename2*). THE NAMES MUST BE THE SAME.

Explanation:

zonename1	first zone name
zonename2	second zone name

SMP/E cannot import the indicated zone because there are two different zone names. This may happen if:

- The wrong zone was specified on the ZONEIMPORT command
- The wrong ddname was specified on the INFILE operand

System Action: ZONEIMPORT processing stops.

Programmer Response: Do **one** of the following:

- Specify the correct zone name on the ZONEIMPORT command.
- Specify the ddname of the INFILE data set that contains the correct zone.

Then rerun the job.

GIM61401S THE INFILE DATA SET IS NOT IN THE FORMAT EXPECTED FOR THE ZONEIMPORT COMMAND.

Explanation: SMP/E cannot process the INFILE data set because it is not in the expected format. For example, it may not have been created by the ZONEEXPORT command, or an error may have occurred while it was being created by the ZONEEXPORT command.

System Action: ZONEIMPORT processing stops.

Programmer Response: Do **one** of the following:

- If the INFILE operand does not specify the intended data set, specify the correct data set and rerun the job.
- If the INFILE operand does specify the intended data set, use ZONEEXPORT to recreate the data set. Then rerun the job using the new output from ZONEEXPORT.

GIM61501W THE NAME OF THE IMPORTED ZONE WAS CHANGED FROM *zonename1* TO *zonename2*.

Explanation:

zonename1	old zone name
zonename2	new zone name

The INTO operand on the ZONEIMPORT command specifies a different zone name from the zone in the INFILE data set. Therefore, when SMP/E imported the zone, it changed the zone name to the value specified on the INTO operand.

System Action: None.

Programmer Response: None.

GIM61601S *command* PROCESSING FAILED BECAUSE THE INPUT ZONE AND RECEIVING ZONE HAVE THE SAME NAME (*zonename*).

Explanation:

command	SMP/E command
zonename	zone name

For the specified command, the input zone and receiving (INTO) zone must have different names. However, the same name was specified for both the input zone and the receiving zone.

System Action: Command processing stops.

Programmer Response: Specify a different zone name on the INTO operand and rerun the command.

GIM61701E *elmtype elmname* WAS NOT MOVED BY SYSMOD *sysmod* BECAUSE IT DOES NOT EXIST IN THE SPECIFIED *library* LIBRARY.

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID
library	ddname of the library

There was an entry for the indicated element in the target or distribution zone, but the element does not exist in the specified library. It may have been moved, renamed, or deleted outside of SMP/E, or its entry may have been created using UCLIN or JCLIN. Note that if an element name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: SYSMOD processing stops.

Programmer Response: First determine why the element does not exist in the library.

- If it was moved, deleted, or renamed outside of SMP/E, reconstruct it in the specified library.

- If the entry was created using UCLIN or JCLIN, decide whether to create the member in the library or to delete the entry from the zone.

Then rerun the job.

GIM61702W *enttype entname* **WAS NOT RENAMED BY SYSMOD** *sysmod* **BECAUSE** *entname* **DOES NOT EXIST IN THE SPECIFIED** *library* **LIBRARY.**

Explanation:

enttype	entry type
entname	entry name
sysmod	SYSMOD ID
library	ddname of the library

There was an entry for the indicated member in the target zone, but the member does not exist in the specified library. It may have been moved, renamed, or deleted outside of SMP/E, or its entry may have been created using UCLIN or JCLIN.

If the message identifies an LMOD and indicates SMPLTS as the library, the LMOD entry has CALLLIBS subentries, and the LMOD should exist in the SMPLTS data set. It may have been moved, renamed, or deleted from the SMPLTS library outside of SMP/E, or its CALLLIBS subentries may have been created using UCLIN or JCLIN.

If the message identifies a side deck, the LMOD entry for the same-named load module has a SIDE DECK LIBRARY subentry, and the side deck should exist in the specified library. It may have been moved, renamed, or deleted from the library outside of SMP/E, or its side deck library subentry may have been created using UCLIN or JCLIN.

System Action: SYSMOD processing continues, but SMP/E cannot rename the identified member.

Programmer Response:

- If an LMOD does not exist in the SMPLTS library, determine if the CALLLIBS subentries in the LMOD entry are valid.
  - If they **are not** valid, use UCLIN to delete them.
  - If they **are** valid, SMP/E attempts to reconstruct the SMPLTS version of the LMOD the next time the LMOD is serviced.
- If a definition side deck member does not exist in the specified library, determine if the side deck library subentry in the LMOD entry is valid.
  - If it **is not** valid, use UCLIN to delete it.
  - If it **is** valid, the binder will reconstruct the definition side deck the next time the LMOD is link-edited by SMP/E.

GIM61702E **LMOD** *elmname* **WAS NOT RENAMED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **DOES NOT EXIST IN THE SPECIFIED** *library* **LIBRARY.**

Explanation:

elmname	element name
sysmod	SYSMOD ID
library	ddname of the library

There was an entry for the indicated LMOD in the target zone, but the LMOD does not exist in the specified library. It may have been moved, renamed, or deleted outside of SMP/E, or its entry may have been created using UCLIN or JCLIN.

System Action: SYSMOD processing stops.

Programmer Response: Determine why the LMOD is missing from its target library:

- If it was moved, deleted, or renamed outside of SMP/E, reconstruct it in the specified library.
- If the entry was created using UCLIN or JCLIN, decide whether to create the member in the library or to delete the entry from the zone.

Then rerun the job.

GIM61703W *enttype entname* **WAS NOT DELETED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **DOES NOT EXIST IN THE SPECIFIED** *library* **LIBRARY.**

Explanation:

enttype	ALIAS or LMOD
entname	entry name
sysmod	SYSMOD ID
library	ddname of the library

There was an entry for the indicated alias or LMOD in the target or distribution zone, but the alias or LMOD does not exist in the specified library. It may have been moved, renamed, or deleted outside of SMP/E, or its entry may have been created using UCLIN or JCLIN.

System Action: SYSMOD processing stops.

Programmer Response: First determine why the element does not exist in the library.

- If it was moved, deleted, or renamed outside of SMP/E, reconstruct it in the specified library.
- If the entry was created using UCLIN or JCLIN, decide whether to create the member in the library or to delete the entry from the zone.

Then rerun the job.

GIM61704E *elmname* **WAS NOT MOVED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **DOES NOT EXIST IN THE SPECIFIED** *library* **LIBRARY.**

Explanation:

elmname	element name
sysmod	SYSMOD ID
library	ddname of the library

The indicated element does not exist in the specified library. (Because of an internal SMP/E error, SMP/E could not determine the element type.) It may have been moved, renamed, or deleted outside of SMP/E, or its entry may have been created using UCLIN or JCLIN.

System Action: SYSMOD processing stops.

Programmer Response: First determine why the element does not exist in the library.

- If it was moved, deleted, or renamed outside of SMP/E, reconstruct it in the specified library.
- If the entry was created using UCLIN or JCLIN, decide whether to create the member in the library or to delete the entry from the zone.

Then rerun the job.

GIM61705E **ALIAS** *alias* **WAS NOT DELETED BY SYSMOD** *sysmod* **BECAUSE THE ALIAS IS GREATER THAN EIGHT CHARACTERS IN LENGTH AND RESIDES IN LIBRARY** *library***, WHICH IS A PDSE.**

Explanation:

alias	alias name
sysmod	SYSMOD that contains the ++DELETE MCS
library	ddname of the library

++DELETE processing failed for the indicated alias. Note that if the alias name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: SYSMOD processing stops for the indicated SYSMOD.

Programmer Response: A ++DELETE MCS cannot be used to delete an alias that is greater than eight characters in length from a PDSE. In order to delete the alias, the SYSMOD must be built in the following manner:

- Supply JCLIN that defines the load module whose alias needs to be deleted. The JCLIN must not specify an ALIAS control statement for the alias to be deleted.
- Supply a ++MOD for a module that is included in the load module.

This SYSMOD causes SMP/E to link-edit the load module, which results in the long name alias being deleted, because it was not specified on an ALIAS control statement.

GIM61706E **LMOD** *loadmod* **WAS NOT MOVED FROM** *library1* **TO** *library2* **FOR SYSMOD** *sysmod* **BECAUSE ITS LINK-EDIT CONTROL STATEMENTS INDICATE THAT IT HAS ASSOCIATED SYMBOLIC LINKS.**

Explanation:

loadmod	load module name
library1	from library name
library2	to library name
sysmod	SYSMOD ID

The indicated load module resides in the hierarchical file system and cannot be moved because it has symbolic links. The symbolic links may become invalid if the load module is moved.

System Action: SYSMOD processing stops.

Programmer Response: If you are trying to install an IBM-supplied SYSMOD, call IBM for guidance in resolving the problem.

If you are installing a ++USERMOD, do one of the following:

- Use ++DELETE to entirely delete the load module and then redefine it with JCLIN and rebuild it by supplying at least one of its modules (more may be needed if SMP/E cannot find all the needed modules).
- Move the load module outside of SMP/E processing and properly resolve its symbolic links for its new location. Use UCLIN to correct the LMOD entry in the CSI so that its SYSLIB and symbolic link information is correct.

GIM61707E **LMOD** *loadmod* **WAS NOT RENAMED FOR SYSMOD** *sysmod* **BECAUSE ITS LINK-EDIT CONTROL STATEMENTS INDICATE THAT IT HAS ASSOCIATED SYMBOLIC LINKS.**

Explanation:

loadmod	load module name
sysmod	SYSMOD ID

The indicated load module resides in the hierarchical file system and cannot be renamed because it has symbolic links. The symbolic links may become invalid if the load module is renamed.

System Action: SYSMOD processing stops.

Programmer Response: If you are trying to install an IBM supplied SYSMOD, call IBM for guidance in resolving the problem.

If you are installing a ++USERMOD, do one of the following:

- Use ++DELETE to entirely delete the load module and then redefine it with its new name with updated JCLIN and rebuild it by supplying at least one of its modules (more may be needed if SMP/E cannot find all the needed modules).
- Rename the load module outside of SMP/E processing and properly resolve its symbolic links for its new name. Use UCLIN to delete the old LMOD entry and to create a new LMOD entry that has the correct name and symbolic link information.

GIM61801E THE ++MOVE MCS FOR *member* **IN SYSMOD** *sysmod* **SPECIFIES THE SAME LIBRARY (***library***) FOR BOTH** *libtypes***. THIS IS NOT ALLOWED.**

Explanation:

member	member name
sysmod	SYSMOD ID
library	ddname of the library
libtypes	″DISTLIB AND TODISTLIB″ or ″SYSLIB AND TOSYSLIB″

SMP/E cannot move a member from one library to that same library.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- If the member is supposed to be moved, specify different ddnames for the current and new libraries.
- If the member is not supposed to be moved, delete the ++MOVE MCS.

Then rerun the job.

GIM61901I *elmtype elmname* **WAS NOT MOVED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID
zonename	zone name

SMP/E did not find an entry for the element in the indicated zone, so no processing was done.

System Action: Processing continues.

Programmer Response: No action is needed.

GIM61901W *elmtype elmname* **WAS NOT MOVED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID
zonename	zone name

SMP/E did not find an entry for the element in the indicated zone, so no processing was done.

System Action: Processing continues.

Programmer Response: No action is needed.

GIM61902W *elmtype elmname* **WAS NOT RENAMED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID
zonename	zone name

SMP/E did not find an entry for the element in the indicated zone, so no processing was done.

System Action: The system action depends on the processing being done:

- If you are restoring an entry for a renamed LMOD, processing stops.
- Otherwise, processing continues.

Programmer Response: If processing stops, add the missing entry and rerun the job. Otherwise, no action is needed.

GIM61903W *elmtype elmname* **WAS NOT DELETED BY SYSMOD** *sysmod* **BECAUSE IT IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID
zonename	zone name

SMP/E did not find an entry for the element in the indicated zone, so there was nothing to delete. Note that if the element name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: Processing continues.

Programmer Response: None.

GIM61904I *elmname* **WAS NOT MOVED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmname	element name
sysmod	SYSMOD ID
zonename	zone name

SMP/E did not find an entry for the element in the indicated zone, so no processing was done. (Because of an internal SMP/E error, SMP/E could not determine the element type.)

System Action: Processing continues.

Programmer Response: No action is needed.

GIM61904W *elmname* **WAS NOT MOVED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmname	element name
sysmod	SYSMOD ID
zonename	zone name

SMP/E did not find an entry for the element in the indicated zone, so no processing was done. (Because of an internal SMP/E error, SMP/E could not determine the element type.)

System Action: Processing continues.

Programmer Response: If you want to move the missing entry, add it to the zone and rerun the job. Otherwise, no action is needed.

GIM61904E *elmname* **WAS NOT MOVED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmname	element name
sysmod	SYSMOD ID
zonename	zone name

During ACCEPT processing for an element with DISTLIB changes, SMP/E did not find an entry for the element in the indicated zone, so no processing was done. (Because of an internal SMP/E error, SMP/E could not determine the element type.)

System Action: SYSMOD processing stops.

Programmer Response: Add the missing entry and rerun the job.

GIM61905W *elmname* **WAS NOT RENAMED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmname	element name
sysmod	SYSMOD ID
zonename	zone name

SMP/E did not find an entry for the element in the indicated zone, so no processing was done. (Because of an internal SMP/E error, SMP/E could not determine the element type.)

System Action: The system action depends on the processing being done:

- If you are restoring an entry for a renamed LMOD, processing stops.
- Otherwise, processing continues.

Programmer Response: If processing stops, add the missing entry and rerun the job. Otherwise, no action is needed.

GIM61906W *elmname* **WAS NOT DELETED BY SYSMOD** *sysmod* **BECAUSE** *elmname* **IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmname	element name
sysmod	SYSMOD ID
zonename	zone name

SMP/E did not find an entry for the element in the indicated zone, so there was nothing to delete. (Because of an internal SMP/E error, SMP/E could not determine the element type.)

System Action: Processing continues.

Programmer Response: None.

GIM62001I *elmtype elmname* **WAS MOVED BY SYSMOD** *sysmod* **FROM** *librtype library1* **TO** *library2***.**

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID
librtype	DISTLIB or SYSLIB
library1	ddname of the old library
library2	ddname of the new library

The element was successfully moved from the old library to the new one. Note that if the element name is greater than 300 characters, only the first 300 characters are placed in the message.

System Action: None.

Programmer Response: None.

GIM62002I *elmname* **WAS MOVED BY SYSMOD** *sysmod* **FROM** *librtype library1* **TO** *library2***.**

Explanation:

elmname	element name
sysmod	SYSMOD ID
librtype	DISTLIB or SYSLIB
library1	ddname of the old library
library2	ddname of the new library

The element was successfully moved from the old library to the new one. (Because of an internal SMP/E error, SMP/E could not determine the element type.)

System Action: None.

Programmer Response: None.

GIM62101E THE *operand1* OPERAND IS REQUIRED BECAUSE THE *operand2* OPERAND WAS SPECIFIED ON THE ++MOVE MCS IN SYSMOD *sysmod*.

Explanation:

operand1	missing operand
operand2	specified operand
sysmod	SYSMOD ID

The indicated operand, *operand1*, is required because the second operand, *operand2*, was specified. However, *operand1* was missing from the ++MOVE MCS.

System Action: SYSMOD processing stops.

Programmer Response: Add the missing operand to the ++MOVE MCS, then rerun the job.

GIM62102E THE *operand1* OPERAND IS NOT ALLOWED BECAUSE THE *operand2* OPERAND WAS SPECIFIED ON THE ++MOVE MCS IN SYSMOD *sysmod*.

Explanation:

operand1	first operand
operand2	second operand
sysmod	SYSMOD ID

The indicated operands are mutually exclusive. However, they were specified on the same ++MOVE MCS.

System Action: SYSMOD processing stops.

Programmer Response: Specify the correct operands on the ++MOVE MCS, then rerun the job.

GIM62201E THE ++*mcstype* MCS IS OUT OF SEQUENCE.

Explanation:

mcstype	MCS type.

The indicated MCS is out of order in the SMPPTFIN data set.

System Action: SYSMOD processing stops.

Programmer Response: See *SMP/E Reference* or *MVS Packaging Rules* to determine the correct order for the MCS statements. Make any necessary changes, then rerun the job.

GIM62401E A MEMBER TYPE OPERAND IS REQUIRED ON THE ++*mcstype* MCS FOR *elmname* IN SYSMOD *sysmod*.

Explanation:

mcstype	MCS type
elmname	element name
sysmod	SYSMOD ID

The indicated MCS did not specify the type of member to be moved.

System Action: SYSMOD processing stops. The SYSMOD is not received.

Programmer Response: Add the required operand and rerun the job.

GIM62402E A LIBRARY TYPE OPERAND IS REQUIRED ON THE ++*mcstype* MCS FOR *elmname* IN SYSMOD *sysmod*.

Explanation:

mcstype	MCS type
elmname	element name
sysmod	SYSMOD ID

The indicated MCS did not specify the libraries to be processed.

System Action: SYSMOD processing stops. The SYSMOD is not received.

Programmer Response: Add the required operand and rerun the job.

GIM62403E A *operand* OPERAND IS REQUIRED ON THE ++*mcstype* MCS FOR *elmname* IN SYSMOD *sysmod*.

Explanation:

operand	operand
mcstype	MCS type
elmname	element name
sysmod	SYSMOD ID

The indicated MCS statement did not specify a required value:

- A ++DELETE MCS must specify the library from which the load module should be deleted.
- A ++RENAME MCS must specify the new name of the load module.

System Action: SYSMOD processing stops. The SYSMOD is not received.

Programmer Response: Add the required operand and rerun the job.

GIM62404E THE *operand1* OPERAND IS REQUIRED BECAUSE THE *operand2* OPERAND WAS SPECIFIED ON THE ++*mcstype* MCS IN SYSMOD *sysmod*.

Explanation:

operand1	required operand
operand2	specified operand
mcstype	MCS type
sysmod	SYSMOD ID

The indicated MCS statement specified the RFDSNPFX operand. This operand is valid only for SYSMODs packaged in RELFILE format. However, the MCS did not specify the FILES operand, which is how you indicate to SMP/E that a SYSMOD is packaged in RELFILE format.

System Action: SYSMOD processing stops. The SYSMOD is not received.

Programmer Response: Add the required operand and rerun the job.

GIM62405E THE *operand1* OPERAND IS REQUIRED BECAUSE THE *operand2* OPERAND WAS SPECIFIED ON THE ++*mcstype* MCS FOR *elmname* IN SYSMOD *sysmod*.

Explanation:

operand1	required operand
operand2	specified operand
mcstype	MCS type
elmname	element name
sysmod	SYSMOD ID

The indicated MCS for the indicated element specified the *operand2* operand. When the *operand2* operand is specified on this MCS, the *operand1* operand must also be specified. For instance, if the SYMLINK operand is specified on any hierarchical file system element MCS, then the SYMPATH operand must also be specified (the reverse of this is also true).

System Action: SYSMOD processing stops.

Programmer Response: Add the required operand and rerun the job.

GIM62501T *value* IS NOT A DEFINED DUMP POINT.

Explanation:

value	dump ID or VPLFUNCT value

The dump ID or VPLFUNCT value specified on the DEBUG command is not a defined dump point.

System Action: Command processing stops.

Programmer Response: Specify a valid dump ID or VPLFUNCT value on the DEBUG command and rerun the job.

GIM62502T MESSAGE ID *msgid* IS NOT AN SMP/E MESSAGE.

Explanation:

msgid	message ID

The message ID specified on the DEBUG command is not an SMP/E message. For example, it did not start with **GIM**.

System Action: Command processing stops.

Programmer Response: Specify a valid message ID on the DEBUG command and rerun the job.

GIM62601E CURRENT DATA SET NAME *dataset1* WAS NOT CHANGED BECAUSE THE NEW NAME *dataset2* EXCEEDS THE 44-CHARACTER MAXIMUM.

Explanation:

dataset1	current data set name
dataset2	new data set name

A data set name can have up to 44 characters, including periods and the user ID, if it is appended to the beginning. The new name would have been too long, so the data set name was not changed.

System Action: Command processing stops.

Programmer Response: Specify a new data set name that is less than or equal to 44 characters and rerun the job.

GIM62602S THE DATA SET NAME DERIVED BY REPLACING THE HIGH-LEVEL QUALIFIER OF FILE *filename* WITH THE PREFIX VALUE OF *prefix* CREATES A NAME WHICH EXCEEDS THE 44-CHARACTER MAXIMUM.

Explanation:

filename	file name value found in file attribute file of archived file
prefix	prefix value specified in an archive definition group

A data set name can have up to 44 characters, including periods. The data set name derived by replacing the high-level qualifier of the file name value with the prefix value given causes the data set name to be too long.

System Action: GIMUNZIP processing stops.

Programmer Response: Use a shorter prefix value (or no prefix value at all) to extract the archived file.

GIM62701W THE ++*mcstype* MCS FOR *enttype* ENTRY *entname* IN SYSMOD *sysmod* WAS NOT PROCESSED BECAUSE THE *operand* VALUE DOES NOT MATCH THE EXISTING *operand* VALUE.

Explanation:

mcstype	MCS type
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
operand	operand

SMP/E detected a value specified on the indicated MCS that did not match the value currently in the indicated

entry. Note that if the entry name is greater than 300 characters, then only the first 300 characters are placed in the message.

System Action: SYSMOD processing continues. However, when processing a ++DELETE MCS, the load module will not be deleted from any of its system libraries.

Programmer Response: No action is required.

GIM62701E THE ++*mcstype* MCS FOR *enttype* ENTRY *entname* IN SYSMOD *sysmod* WAS NOT PROCESSED BECAUSE THE *operand* VALUE DOES NOT MATCH THE EXISTING *operand* VALUE.

Explanation:

mcstype	MCS type
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
operand	operand

SMP/E detected a value specified on the indicated MCS that did not match the value currently in the indicated entry.

System Action: SYSMOD processing stops.

Programmer Response: Specify the correct value on the MCS and rerun the job.

GIM62702W THE ++*mcstype* MCS FOR ENTRY *entname* IN SYSMOD *sysmod* WAS NOT PROCESSED BECAUSE THE *operand* VALUE DOES NOT MATCH THE EXISTING *operand* VALUE.

Explanation:

mcstype	MCS type
entname	entry name
sysmod	SYSMOD ID
operand	operand

SMP/E detected a value specified on the indicated MCS that did not match the value currently in the indicated entry. (Because of an internal SMP/E error, SMP/E could not determine the entry type.)

System Action: SYSMOD processing continues. However, when a ++DELETE MCS is being processed, the load module will not be deleted from any of its system libraries.

Programmer Response: No action is required.

GIM62702E THE ++*mcstype* MCS FOR ENTRY *entname* IN SYSMOD *sysmod* WAS NOT PROCESSED BECAUSE THE *operand* VALUE DOES NOT MATCH THE EXISTING *operand* VALUE.

Explanation:

mcstype	MCS type
entname	entry name
sysmod	SYSMOD ID
operand	operand

SMP/E detected a value specified on the indicated MCS that did not match the value currently in the indicated entry. (Because of an internal SMP/E error, SMP/E could not determine the entry type.)

System Action: SYSMOD processing stops.

Programmer Response: Specify the correct value on the MCS and rerun the job.

GIM62801I SMP/E WILL NOT USE STORAGE ABOVE 16MB BECAUSE SMP/E WAS LINK-EDITED WITH A LINKAGE EDITOR THAT DOES NOT RECOGNIZE AMODE.

Explanation: SMP/E will not use storage above the 16-megabyte virtual line because the SMP/E load module (GIMSMP) was not link-edited using a linkage editor that supports the AMODE parameter.

System Action: SMP/E continues processing, using storage below 16 megabytes.

Programmer Response: To allow SMP/E to use storage above the 16 megabyte line, the SMP/E load module (GIMSMP) must be link-edited using a linkage editor that supports the AMODE parameter, such as an IBM DFP linkage editor.

GIM62901I LMOD *loadmod1* WAS RENAMED TO *loadmod2* IN THE *library* LIBRARY BY SYSMOD *sysmod*.

Explanation:

loadmod1	old LMOD name
loadmod2	new LMOD name
library	ddname of the library
sysmod	SYSMOD ID

The LMOD was successfully renamed.

System Action: None.

Programmer Response: None.

GIM63201I *enttype entname* **WAS DELETED FROM THE** *library* **LIBRARY BY SYSMOD** *sysmod***.**

Explanation:

enttype	ALIAS or LMOD
entname	entry name
library	ddname of the library
sysmod	SYSMOD ID

The alias or LMOD was successfully deleted. Note that if the alias name is greater than 300 characters in length, only the first 300 characters are placed in the message.

System Action: None.

Programmer Response: None.

GIM63301E SYSMOD *sysmod* **WAS NOT RESTORED BECAUSE IT CONTAINS A ++DELETE MCS.**

Explanation:

sysmod	SYSMOD ID

When the indicated SYSMOD was installed, SMP/E deleted one or more load modules. Because load modules cannot be rebuilt, they cannot be restored.

System Action: RESTORE processing stops.

Programmer Response: None.

GIM63401S ZONESET *zoneset* **IS NOT DEFINED IN THE GLOBAL ZONE.**

Explanation:

zoneset	ZONESET name

The indicated ZONESET is not defined in the global zone.

System Action: Command processing stops.

Programmer Response: Make sure the ZONESET name you specified is correct. If the ZONESET name is incorrect, enter a different ZONESET name or fix the spelling. If the ZONESET name is correct, use the UCLIN command or the Administration Dialogs to define the ZONESET entry in the global zone.

GIM63501S THE ZONE WAS NOT COPIED BECAUSE THE INPUT SMPCSI AND THE RECEIVING SMPCSI ARE THE SAME.

Explanation: The zone was not copied because the input and receiving zones must be in different SMPCSI data sets.

System Action: ZONECOPY processing stops.

Programmer Response: Do **one** of the following:

- Define the receiving zone in another SMPCSI, then rerun the ZONECOPY command.
- Use ZONEMERGE or ZONEEXPORT and ZONEIMPORT to copy the zone in the same SMPCSI.

GIM63901S THE *operand* **OPERAND IS REQUIRED ON THE** *command* **COMMAND.**

Explanation:

operand	operand
command	SMP/E command

The indicated command could not be processed because a required operand was missing.

System Action: Command processing stops.

Programmer Response: Add the missing operand and rerun the job.

GIM64001E JOBCARD MEMBER *memname* **IS NOT IN THE** *library* **LIBRARY. A JCL COMMENT WILL BE GENERATED EACH TIME A JOB CARD IS REQUIRED.**

Explanation:

memname	member name
library	ddname of the library

The JOBCARD operand was specified on the GENERATE command to indicate the library that contains the job card to be used for the GENERATE output. SMP/E checked the library for the specified member (or a member called JOBCARD, if no member name was specified), but no member was found. As a result, a comment will be inserted in the GENERATE output each time a job card is required.

System Action: GENERATE processing continues.

Programmer Response: Do **one** of the following:

- Add a job card member to the indicated data set and rerun the GENERATE command.
- Edit the GENERATE output to find the JCL comments inserted by SMP/E. (The format of the JCL comment is described under the GENERATE command in *SMP/E Commands*.) Replace the JCL comment with a valid job card.

GIM64101E NO ASSEMBLER STEP WAS GENERATED FOR MOD *modname* **BECAUSE THERE IS NO ASSOCIATED ASSEM OR SRC ENTRY.**

Explanation:

modname	module name

SMP/E found a MOD entry with a DISTLIB of SYSPUNCH, showing that the module is not in any distribution library. Instead, the module is in source

format, which SMP/E must assemble each time it processes the module. SMP/E normally gets the source text from one of the following:

- A target zone ASSEM entry with the same name as the MOD entry
- A target zone SRC entry with the same name as the MOD entry

For the indicated module, SMP/E could not find an ASSEM or SRC entry. Therefore, SMP/E could not generate the assembler step for the module.

System Action: GENERATE processing continues.

Programmer Response: Do **one** of the following:

- If the MOD entry is no longer required, delete it.
- If the MOD entry is still required, run the appropriate JCLIN to add an ASSEM entry or SRC entry to the target zone. Then rerun GENERATE.

GIM64201E MOD *modname* **WAS NOT INCLUDED IN THE LINK-EDIT STEP FOR LMOD** *loadmod* **BECAUSE** *modname* **CANNOT BE ASSEMBLED.**

Explanation:

modname	module name
loadmod	load module name

SMP/E could not assemble the indicated module, as shown in message GIM64101E. As a result, there is no object deck to link-edit into the indicated load module.

System Action: SMP/E will generate a link-edit step for the load module without the module.

Programmer Response: See message GIM64101E to find out why the module was not assembled and do **one** of the following:

- Correct the problem and rerun the GENERATE command.
- Add an assembly step to the GENERATE output and link-edit the INCLUDE card in the load module link step.

GIM64301W A JOB STEP WAS GENERATED TO COPY *elmtype elmname* **FROM** *distlib* **TO** *targlib***.** *elmtype* **ENTRY** *elmname* **HAS NO FMID AND THEREFORE MIGHT NOT HAVE BEEN INSTALLED. AN ERROR WILL OCCUR IF** *elmname* **IS NOT IN** *distlib***.**

Explanation:

elmtype	element type
elmname	element name
distlib	ddname of the distribution library
targlib	ddname of the target library

SMP/E assumes that every element in the target zone was part of a function that SMP/E installed in the target zone. Each element entry identifies the function (FMID) for which it was installed.

In this case, the indicated element entry does not have an FMID subentry. Therefore, SMP/E is not sure whether the element was ever really installed.

System Action: SMP/E generates a copy step for the indicated element. However, you may get an error when you run the GENERATE output if the element is not in the distribution library.

Programmer Response: Do **one** of the following:

- If the element is still required, add the appropriate FMID to the element entry.
- If the element is no longer required, delete the element entry.

GIM64401W A JOB STEP WAS GENERATED TO COPY MOD *elmname* **TO LMOD** *loadmod* **IN LIBRARY** *library***. MOD ENTRY** *elmname* **HAS NO FMID AND THEREFORE MIGHT NOT HAVE BEEN INSTALLED. AN ERROR WILL OCCUR IF MOD** *elmname* **IS NOT IN THE DISTRIBUTION LIBRARY.**

Explanation:

elmname	element name
loadmod	load module name
library	ddname of the library where the load module resides

SMP/E assumes that every element in the target zone was part of a function that SMP/E installed in the target zone. Each element entry identifies the function (FMID) for which it was installed.

In this case, the indicated element entry does not have an FMID subentry. Therefore, SMP/E is not sure whether the element was ever really installed.

System Action: SMP/E generates a copy step for the indicated element. However, you may get an error when you run the GENERATE output if the element is not in the distribution library.

Programmer Response: Do **one** of the following:

- If the function that owns the module is installed in the target zone being processed, add the appropriate FMID to the element entry. Enter **LIST MOD XREF** to find out which SYSMODs affect the module. Then enter **LIST SYSMOD** for these SYSMODs to find out which FMID they apply to. This is the FMID you should add to the element entry.
- If the function that owns the module is installed in a different target zone, do nothing. This situation exists when one function includes a module from another function. During JCLIN processing, SMP/E creates a

MOD entry for the module. However, because SMP/E never installs the module in this zone, SMP/E never adds the FMID subentry.

- If the element is no longer required, delete the element entry.

GIM64402W A JOB STEP WAS GENERATED TO LINK-EDIT MOD *elmname* TO LMOD *loadmod* IN LIBRARY *library*. MOD ENTRY *elmname* HAS NO FMID AND THEREFORE MIGHT NOT HAVE BEEN INSTALLED. AN ERROR WILL OCCUR IF MOD *elmname* IS NOT IN THE DISTRIBUTION LIBRARY.

Explanation:

elmname	element name
loadmod	load module name
library	ddname of the library where the load module resides

SMP/E assumes that every element in the target zone was part of a function that SMP/E installed in the target zone. Each element entry identifies the function (FMID) for which it was installed.

In this case, the indicated element entry does not have an FMID subentry. Therefore, SMP/E is not sure whether the element was ever really installed.

System Action: SMP/E generates a link-edit step to include the element in the specified load module. However, you may get an error when you run the GENERATE output if the element is not in the distribution library.

Programmer Response: Do **one** of the following:

- If the function that owns the module is installed in the target zone being processed, add the appropriate FMID to the element entry. Enter **LIST MOD XREF** to find out which SYSMODs affect the module. Then enter **LIST SYSMOD** for these SYSMODs to find out which FMID they apply to. This is the FMID you should add to the element entry.
- If the function that owns the module is installed in a different target zone, do nothing. This situation exists when one function includes a module from another function. During JCLIN processing, SMP/E creates a MOD entry for the module. However, because SMP/E never installs the module in this zone, SMP/E never adds the FMID subentry.
- If the element is no longer required, delete the element entry.

GIM64501W NO LINK-EDIT STEPS WERE GENERATED FOR LMOD *loadmod* BECAUSE NO MOD ENTRIES SPECIFY *loadmod* AS AN LMOD SUBENTRY.

Explanation:

loadmod	load module name

An LMOD entry is in the target zone, but none of the MOD entries show that the module was part of that load module.

System Action: Because SMP/E does not know what modules make up the load module, SMP/E cannot schedule a link-edit. GENERATE processing continues.

Programmer Response: Do **one** of the following:

- Add LMOD subentries to the appropriate MOD entries and rerun the GENERATE command.
- If SMP/E does not need the load module anymore, delete it.

GIM64601E A COMMENT WAS GENERATED FOR THE *jclname* JCL STATEMENT IN STEP *stepname* OF JOB *jobname* BECAUSE THERE IS NO DDDEF ENTRY FOR *dddef*.

Explanation:

jclname	name of the JCL statement
stepname	step name
jobname	job name
dddef	DDDEF name

SMP/E could not get the information to generate the specified JCL statement.

System Action: SMP/E generated a JCL comment to replace the JCL statement it needed. The format of the JCL comment is described in the chapter on the GENERATE command in *SMP/E Commands*. GENERATE processing continues, but the output job will not run unless you fix the error.

Programmer Response: Do **one** of the following:

- Add the required DDDEF entry and rerun the GENERATE command.
- Edit the GENERATE output to add the required JCL statement in place of the comments.

GIM64602E A COMMENT WAS GENERATED FOR THE *jclname* JCL STATEMENT IN STEP *stepname* OF JOB *jobname* BECAUSE THE DDDEF ENTRY FOR *stepname* DOES NOT CONTAIN A PATH SUBENTRY.

Explanation:

jclname	name of the JCL statement
stepname	step name
jobname	job name

The HFSINST job copies hierarchical file system elements to hierarchical file system target libraries. Therefore, the DDDEF entry defining the target library must contain the PATH subentry that indicates a hierarchical file system library.

System Action: SMP/E generated a JCL comment to replace the JCL statement it needed. The format of the

JCL comment is described in the chapter on the GENERATE command in *SMP/E Commands*. GENERATE processing continues, but the output job will not run unless you fix the error.

Programmer Response: Do **one** of the following:

- Add the required PATH subentry to the *stepname* DDDEF entry, then rerun the GENERATE command.
- Edit the GENERATE output to add the required PATH operand in place of the comments.

GIM64603E NO SELECT STATEMENT WAS GENERATED FOR *elmtype elmname* **IN STEP** *stepname* **OF JOB** *jobname***.** *enttype* **ENTRY** *entname* **IS NEEDED TO PROCESS** *elmtype elmname***, BUT IS NOT IN THE** *zonename* **ZONE.**

Explanation:

elmtype	element type
elmname	element name
stepname	step name
jobname	job name
enttype	entry type (DDDEF or SHELLSCR)
entname	entry name
zonename	zone name

The indicated element specifies a shell script to be invoked to complete the installation of the element. Either no entry exists in the zone for the specified shell script element, or the DDDEF entry for the library that contains the shell script is not in the zone. Both entries are needed to process the element.

System Action: SMP/E does not generate a GIMIAP SELECT statement for the indicated element.

Programmer Response: Add the required entry and rerun the GENERATE command.

GIM64604E NO SELECT STATEMENT WAS GENERATED FOR *elmtype elmname* **IN STEP** *stepname* **OF JOB** *jobname***. DDDEF ENTRY** *entname* **IS NEEDED TO PROCESS** *elmtype elmname***, BUT DDDEF ENTRY** *entname* **DOES NOT CONTAIN A PATH SUBENTRY.**

Explanation:

elmtype	element type
elmname	element name
stepname	step name
jobname	job name
entname	DDDEF entry name

The indicated element specifies a shell script to be invoked to complete the installation of the element, but the shell script does not reside in the hierarchical file system. The DDDEF entry for the library that contains the shell script must specify a path in the hierarchical file system.

System Action: SMP/E does not generate a GIMIAP SELECT statement for the indicated element.

Programmer Response: Ensure that the indicated DDDEF entry specifies a path in the hierarchical file system. Rerun the GENERATE command.

GIM64605E NO SELECT STATEMENT WAS GENERATED FOR *elmtype elmname* **IN STEP** *stepname* **OF JOB** *jobname* **BECAUSE THE LENGTH OF THE EXECUTION PARAMETER STRING (I.E., THE EPARM VALUE) EXCEEDS THE MAXIMUM LENGTH OF 65,535 BYTES.**

Explanation:

elmtype	element type
elmname	element name
stepname	step name
jobname	job name

SMP/E attempted to process the indicated element, but when SMP/E built the execution parameter string for invoking the hierarchical file system copy utility, the 65,535 byte limit for the EPARM string was exceeded.

System Action: SMP/E does not generate a GIMIAP SELECT statement for the indicated element.

Programmer Response: Contact the IBM Support Center.

GIM64801E NO JOB STEP WAS GENERATED FOR *enttype entname* **BECAUSE** *entname* **HAS NO DISTLIB SUBENTRY.**

Explanation:

enttype	entry type
entname	entry name

SMP/E assumes that all elements exist in a distribution library. However, SMP/E found an entry that did not have a DISTLIB subentry. Because the output of the GENERATE command indicates how to create elements in the target libraries by using elements in the distribution libraries, SMP/E could not process the indicated element.

System Action: SMP/E did not generate a job step for the indicated element.

Programmer Response: Use UCLIN to add the DISTLIB subentry to the element entry, then rerun the GENERATE command.

GIM64901E NO JOBS WERE GENERATED BECAUSE NO ELEMENTS WERE SELECTED FOR THE FORFMID OPERAND.

Explanation: The FORFMID operand was specified to limit which elements are included in the jobs produced by the GENERATE command. However, SMP/E found

no elements owned by the indicated FMIDs. As a result, it did not generate any jobs.

System Action: Processing continues.

Programmer Response: Specify a valid FMID or delete the FORFMID operand, then rerun the GENERATE command.

GIM64902E NO JOBS WERE GENERATED BECAUSE NO ELEMENTS IN THE TARGET ZONE WERE SELECTED FOR INSTALLATION.

Explanation: As part of GENERATE processing, SMP/E checked the target zone for elements to be reinstalled. However, either none of the elements in the target zone were eligible, or the target zone contained no elements at all. As a result, SMP/E did not generate any jobs.

System Action: Processing continues.

Programmer Response: Do **one** of the following:

- If the SET command does not specify the correct target zone, specify the correct zone and rerun the GENERATE command.
- If the SET command does specify the correct target zone, use JCLIN to add the required entries to the zone. The best way to process the JCLIN is to build a SYSMOD that contains a ++JCLIN MCS, and either include the JCLIN inline after the MCS, or put it in a data set pointed to by the TXLIB or RELFILE operand on the MCS. That way, if you have a problem with those JCLIN changes, you can remove them by restoring the SYSMOD in which they were packaged.

GIM65001I STEP *stepname* IN JOB *jobname* WAS NOT PROCESSED BECAUSE *progname* WAS NOT SPECIFIED AS A UTILITY ON THE JCLIN COMMAND.

Explanation:

stepname	step name
jobname	job name
progname	program or procedure name

SMP/E encountered the indicated program or procedure name in the JCLIN being processed, but it did not recognize the name as a valid utility.

System Action: This step is not processed. Processing continues with the next step.

Programmer Response:

- Make sure the name is spelled correctly.
- Check whether it is necessary for SMP/E to recognize the name. During JCLIN processing, SMP/E needs to process only information used to create or update element entries. Utilities for other types of statements do not need to be recognized. If the program listed in the message is one of the following types of utilities, you must define it to SMP/E:
  - Assembler
  - Linkage editor
  - Copy
  - Update

  For more information about defining a utility to SMP/E, see the descriptions of UTILITY entries and OPTIONS entries in *SMP/E Reference*.

  If the program does not need to be recognized by SMP/E at this point, no action is necessary.
- If SMP/E should process this step and update the target zone with information from the utility statements, specify the program on the appropriate JCLIN operand, and rerun the job.

GIM65001W STEP *stepname* IN JOB *jobname* WAS NOT PROCESSED BECAUSE *progname* WAS NOT SPECIFIED AS A UTILITY ON THE JCLIN COMMAND.

Explanation:

stepname	step name
jobname	job name
progname	program or procedure name

SMP/E encountered the indicated program or procedure name in the JCLIN being processed, but it did not recognize the name as a valid utility.

System Action: This step is not processed. Processing continues with the next step.

Programmer Response:

- Make sure the name is spelled correctly.
- Check whether it is necessary for SMP/E to recognize the name. During JCLIN processing, SMP/E needs to process only information used to create or update element entries. Utilities for other types of statements do not need to be recognized. If the program listed in the message is one of the following types of utilities, you must define it to SMP/E:
  - Assembler
  - Linkage editor
  - Copy
  - Update

  For more information about defining a utility to SMP/E, see the descriptions of UTILITY entries and OPTIONS entries in *SMP/E Reference*.

  If the program does not need to be recognized by SMP/E at this point, no action is necessary.
- If SMP/E should process this step and update the target zone with information from the utility

statements, specify the program on the appropriate JCLIN operand, and rerun the job.

GIM65002I STEP *stepname* **IN JOB** *jobname* **WAS NOT PROCESSED BECAUSE** *progname* **WAS NOT SPECIFIED AS A UTILITY ON THE ++JCLIN MCS IN SYSMOD** *sysmod***.**

Explanation:

stepname	step name
jobname	job name
progname	program or procedure name
sysmod	SYSMOD ID

SMP/E encountered the indicated program or procedure name in the JCLIN being processed, but it did not recognize the name as a valid utility.

System Action: This step is not processed. Processing continues with the next step.

Programmer Response:

- Make sure the name is spelled correctly.
- Check whether it is necessary for SMP/E to recognize the name. During JCLIN processing, SMP/E needs to process only information used to create or update element entries. Utilities for other types of statements do not need to be recognized. If the program listed in the message is one of the following types of utilities, you must define it to SMP/E:
  - Assembler
  - Linkage editor
  - Copy
  - Update

  For more information about defining a utility to SMP/E, see the descriptions of UTILITY entries and OPTIONS entries in *SMP/E Reference*.

  If the program does not need to be recognized by SMP/E at this point, no action is necessary.
- If SMP/E should process this step and update the target zone with information from the utility statements, specify the program on the appropriate JCLIN operand, and rerun the job.

GIM65002W STEP *stepname* **IN JOB** *jobname* **WAS NOT PROCESSED BECAUSE** *progname* **WAS NOT SPECIFIED AS A UTILITY ON THE ++JCLIN MCS IN SYSMOD** *sysmod***.**

Explanation:

stepname	step name
jobname	job name
progname	program or procedure name
sysmod	SYSMOD ID

SMP/E encountered the indicated program or procedure name in the JCLIN being processed, but it did not recognize the name as a valid utility.

System Action: This step is not processed. Processing continues with the next step.

Programmer Response:

- Make sure the name is spelled correctly.
- Check whether it is necessary for SMP/E to recognize the name. During JCLIN processing, SMP/E needs to process only information used to create or update element entries. Utilities for other types of statements do not need to be recognized. If the program listed in the message is one of the following types of utilities, you must define it to SMP/E:
  - Assembler
  - Linkage editor
  - Copy
  - Update

  For more information about defining a utility to SMP/E, see the descriptions of UTILITY entries and OPTIONS entries in *SMP/E Reference*.

  If the program does not need to be recognized by SMP/E at this point, no action is necessary.
- If SMP/E should process this step and update the target zone with information from the utility statements, specify the program on the appropriate JCLIN operand, and rerun the job.

GIM65101S ZONE *zonename* **IS AN EMPTY ZONE.**

Explanation:

zonename	zone name

The command being processed specifies a zone that has no data.

System Action: Command processing stops.

Programmer Response: Check the zone name and do **one** of the following:

- If the zone name is correct, build the zone and rerun the command.
- If the zone name is incorrect, change the zone name and rerun the command.

GIM65201S CSI *dataset* **DOES NOT CONTAIN A GIMZPOOL RECORD.**

Explanation:

dataset	data set name

The indicated data set cannot be processed because it was not initialized with the GIMZPOOL record.

System Action: Command processing stops.

Programmer Response: Check the data set name and do **one** of the following:

- If you specified the correct name, do the following:
  1. Initialize the data set with a GIMZPOOL record.
  2. Create the zones to be processed.

3. Rerun the job.

- If you specified the incorrect name, change the data set name and rerun the job.

GIM65301S THE *operand* OPERAND WAS SPECIFIED MORE THAN ONCE ON A ZONEEDIT CHANGE STATEMENT. THIS IS NOT ALLOWED.

Explanation:
operand operand name

The same operand was specified more than once on a single ZONEEDIT CHANGE statement. This is not allowed.

System Action: Command processing stops. No changes were made.

Programmer Response: Fix the CHANGE statement to specify the value only once. If there is more than one change for the same subentry, code a CHANGE statement for each change. Then rerun the job.

GIM65401W THE ZONE DESCRIPTION EXCEEDS THE 500-BYTE MAXIMUM. ONLY THE FIRST 500 BYTES WERE SAVED.

Explanation: SMP/E only allows 500 bytes of data in a zone description. This includes blanks, as well as shift-in and shift-out characters for double-byte character set input (such as kanji characters). The zone description being processed was too long. These are some possible causes:

- More than 500 bytes of data were entered (including blanks between words and shift-in and shift-out characters).
- SMP/E added shift-in and shift-out characters to double-byte character set input because the input spanned multiple records. This caused the zone description to go over 500 bytes.

System Action: SMP/E saves the first 500 bytes of the zone description, and processing continues.

Programmer Response: To check what was saved as the zone description, run the LIST command for the zone definition entry that contains the zone description. If the zone description that was saved is acceptable, no action is required. Otherwise, enter **UCLIN REP** to replace the saved description with a new, shorter one.

GIM65402I THE size ATTRIBUTE FOR FILE *filename* WILL NOT BE GENERATED IN THE PACKAGE ATTRIBUTE FILE SINCE THE VALUE IS GREATER THAN 99,999,999,999.

Explanation:
filename name of file

GIMZIP is omitting the size attribute for the indicated file in the package attribute file, because the number of bytes in the file is greater than the maximum value for a size attribute (which is 99,999,999,999).

System Action: Processing continues.

Programmer Response: None.

GIM65403I THE size ATTRIBUTE FOR THE PACKAGE WILL NOT BE GENERATED IN THE PACKAGE ATTRIBUTE FILE SINCE THE VALUE IS GREATER THAN 99,999,999,999.

Explanation: GIMZIP is omitting the size attribute for the package in the package attribute file, because the number of bytes in the package is greater than the maximum value for a size attribute (which is 99,999,999,999).

System Action: Processing continues.

Programmer Response: None.

GIM65404I THE files ATTRIBUTE FOR THE PACKAGE WILL NOT BE GENERATED IN THE PACKAGE ATTRIBUTE FILE SINCE THE VALUE IS GREATER THAN 9,999.

Explanation: GIMZIP is omitting the files attribute for the package in the package attribute file, because the number of files in the package is greater than the maximum value for a files attribute (which is 9,999).

System Action: Processing continues.

Programmer Response: None.

GIM65405I THE originalsize ATTRIBUTE FOR FILE *filename* WILL NOT BE GENERATED IN THE PACKAGE ATTRIBUTE FILE SINCE THE VALUE IS GREATER THAN 99,999,999,999.

Explanation:
filename name of file

The original file *filename* that was placed in the archive file is allocated to more than 99,999,999,999 bytes. Because the maximum value for an *originalsize* attribute is 99,999,999,999, GIMZIP has omitted the *originalsize* attribute for file *filename* in the package attribute file.

System Action: Processing continues.

Programmer Response: None.

GIM65406I THE originalsize ATTRIBUTE FOR THE PACKAGE WILL NOT BE GENERATED IN THE PACKAGE ATTRIBUTE FILE SINCE THE VALUE IS GREATER THAN 99,999,999,999.

Explanation: The sum of the values for *originalsize* attributes on ARCHDEF tags in the package exceeds 99,999,999,999. Because the maximum value for an *originalsize* attribute is 99,999,999,999, GIMZIP has omitted the *originalsize* attribute from the package attribute file.

System Action: Processing continues.

Programmer Response: None.

GIM65501S ZONE *zonename* IN THE FORZONE OPERAND IS NOT DEFINED IN THE *zoneset* ZONESET.

Explanation:

zonename	zone name
zoneset	entry name

The indicated zone was specified in the FORZONE operand of the REPORT command but is not defined in the specified ZONESET. All zones in the FORZONE operand must be defined in the ZONESET being used.

System Action: REPORT processing stops.

Programmer Response: Do **one** of the following:
- Remove the zone name from the FORZONE operand.
- Add the zone name to the ZONESET using UCLIN.

Then rerun the REPORT command.

GIM65601S SMP/E ASSUMED THAT ALL THE ZONES IN THE ZONESET WERE *zonetype1* ZONES. *zonename* IS A *zonetype2* ZONE.

Explanation:

zonetype1	DLIB or TARGET
zonename	zone name
zonetype2	TARGET or DLIB

When no zone type operand is specified on the REPORT command, all the zones in the ZONESET being used must be the same type. However, the indicated zone is not the same type as other zones in the ZONESET.

System Action: REPORT processing stops.

Programmer Response: Do **one** of the following:
- Specify the TARGETZONE or DLIBZONE operand on the REPORT command to indicate which type of zones in the ZONESET should be processed.
- Redefine the ZONESET so that all the zones in it are the same type.

Then rerun the REPORT command.

GIM65701E LOAD MODULE BUILD PROCESSING STOPPED BECAUSE DISTRIBUTION ZONE *zonename* WAS NOT AVAILABLE.

Explanation:

zonename	zone name

SMP/E is trying to build a load module in a target library using modules in the related distribution library. However, the distribution zone for that library is not available because another user already has access to the SMPCSI data set that contains it.

System Action: SMP/E stops building the load modules for which it needs the distribution zone. It also issues the following messages:
- GIM674*xx*, which lists the load modules that failed
- GIM67501I, which lists the modules SMP/E requires to complete the load modules
- GIM22601I or GIM22601E, which lists the SYSMODs that failed because load modules failed

Programmer Response: When the other user is done with the data set that contains the required distribution zone, rerun the job.

GIM65801E MODULE *modname* IS NEEDED TO BUILD THE FOLLOWING LOAD MODULES FOR SYSMOD *sysmod* BUT IS NOT FOUND. THE MODULE IS NOT IN DISTRIBUTION ZONE *zonename*.

Explanation:

modname	module name
sysmod	SYSMOD ID
zonename	zone name

SMP/E required the specified module to build load modules, but SMP/E could not find the module in the distribution zone.

SMP/E also did not search in the SMPPTS data set for the SYSMOD that last replaced the module in the target system because the module was either previously assembled, or was updated since its last replacement (contains a UMID). If the SYSMOD that last replaced the module could be used and was found in the SMPPTS, the copy of the module within the SYSMOD would be included during link edit operations.

Message GIM65905 follows this message and identifies the load module that could not be built.

System Action: SYSMOD processing stops.

Programmer Response: If the SYSMOD containing a copy of the indicated module has not yet been accepted into the specified distribution zone, accept it so SMP/E can find the module in the distribution zone, then rerun the job.

GIM65802E **MODULE** *modname* **IS NEEDED TO BUILD THE FOLLOWING LOAD MODULES FOR SYSMOD** *sysmod1* **BUT IS NOT FOUND. THE MODULE IS NOT IN DISTRIBUTION ZONE** *zonename***, AND SYSMOD** *sysmod2* **WHICH LAST REPLACED THE MODULE IS NOT IN THE SMPPTS.**

Explanation:

modname	module name
sysmod1	SYSMOD ID
zonename	zone name
sysmod2	module's target zone RMID

SMP/E required the specified module to build load modules, but SMP/E could not find the module in the distribution zone.

SMP/E also could not find in the SMPPTS data set SYSMOD *sysmod2*, which last replaced the module in the target system. This SYSMOD is the module's target zone RMID. If the SYSMOD was found in the SMPPTS, the copy of the module within the SYSMOD would be included during link edit operations.

Message GIM65905 follows this message and identifies the load modules that could not be built.

System Action: SYSMOD processing stops.

Programmer Response: Do one of the following, then rerun the job.

1. If the SYSMOD containing a copy of the indicated module has not yet been accepted into the specified distribution zone, then accept it so SMP/E can find the module in the distribution zone.
2. RECEIVE into the global zone and SMPPTS data set SYSMOD *sysmod2*, which last replaced the module in the target system. SMP/E will then find and use the module in the SMPPTS data set.

GIM65901E **MODULE** *modname* **IS NEEDED TO BUILD THE FOLLOWING LOAD MODULES FOR SYSMOD** *sysmod* **BUT CAN NOT BE USED. ITS DISTRIBUTION ZONE** *umidtype dzumid* **IS DIFFERENT FROM THE TARGET ZONE** *umidtype tzumid***.**

Explanation:

modname	module name
sysmod	SYSMOD ID
umidtype	FMID, RMID, or UMID
dzumid	distribution zone FMID or RMID. This field is blank if *umidtype* is UMID.
tzumid	target zone FMID or RMID. This field is blank if *umidtype* is UMID.

SMP/E required the specified module to build load modules. It found the module in the distribution zone, but the module was at the wrong service level. A module may have different service levels in the target zone and the distribution zone if any service is applied to the target zone and not accepted in the distribution zone.

SMP/E also did not search in the SMPPTS data set for the SYSMOD that last replaced the module in the target system because the module was either previously assembled, or was updated since its last replacement (contains a UMID). If the SYSMOD that last replaced the module could be used and was found in the SMPPTS, the copy of the module within the SYSMOD would have been included during link edit operations.

Message GIM65905 follows this message and identifies the load modules that could not be built.

System Action: SYSMOD processing stops.

Programmer Response: List the module in the target and distribution zones and compare the FMID, RMID, and UMIDs. (If the message indicated an FMID or RMID mismatch, the UMIDs may also mismatch.) Then you may be able to use the ACCEPT or RESTORE commands to synchronize the target zone and distribution zone either restoring or accepting the module's target zone FMID, RMID, or UMID (whichever was indicated as causing the mismatch). This will allow SMP/E to find a usable copy of the module in the distribution zone when you rerun the job.

GIM65902E **MODULE** *modname* **IS NEEDED TO BUILD THE FOLLOWING LOAD MODULES FOR SYSMOD** *sysmod1* **BUT CAN NOT BE USED. ITS DISTRIBUTION ZONE** *umidtype dzumid* **IS DIFFERENT FROM THE TARGET ZONE** *umidtype tzumid***, AND SYSMOD** *sysmod2* **WHICH LAST REPLACED THE MODULE IS NOT IN THE SMPPTS.**

Explanation:

modname	module name
sysmod1	SYSMOD ID
umidtype	FMID, RMID, or UMID
dzumid	distribution zone FMID or RMID. This field is blank if *umidtype* is UMID.
tzumid	target zone FMID or RMID. This field is blank if *umidtype* is UMID.
sysmod2	module's target zone RMID

SMP/E required the specified module to build load modules. It found the module in the distribution zone, but the module was at the wrong service level. A module may have different service levels in the target zone and the distribution zone if any service is applied to the target zone and not accepted in the distribution zone.

SMP/E also could not find in the SMPPTS data set SYSMOD *sysmod2* that last replaced the module in the

target system. This SYSMOD is the module's target zone RMID. If the SYSMOD was found in the SMPPTS, the copy of the module within the SYSMOD would be included during link edit operations.

Message GIM65905 follows this message and identifies the load modules that could not be built.

System Action: SYSMOD processing stops.

Programmer Response: Do one of the following and rerun the job:

1. RECEIVE into the Global zone and SMPPTS data set SYSMOD *sysmod2* that last replaced the module in the target system. SMP/E will then find and use the module in the SMPPTS data set when you rerun the job.
2. List the module in the target and distribution zones and compare the FMID, RMID, and UMIDs. (If the message indicated an FMID or RMID mismatch, the UMIDs may also mismatch.) You may then be able to use the ACCEPT or RESTORE commands to synchronize the target zone and distribution zone either restoring or accepting the module's target zone FMID, RMID, or UMID (whichever was indicated as causing the mismatch). This will allow SMP/E to find a usable copy of the module in the distribution zone when you rerun the job.

GIM65903I **MODULE** *modname* **IS NEEDED TO BUILD THE FOLLOWING LOAD MODULES FOR SYSMOD** *sysmod* **BUT THE SMPTLIB DATA SET COULD NOT BE ALLOCATED.**

Explanation:

modname	module name
sysmod	SYSMOD ID

SMP/E required the specified module to build load modules, but the library that contains the module could not be allocated.

SMP/E attempted to use the copy of the module within the SYSMOD that last replaced the module in the target system. This SYSMOD is the module's target zone RMID. The SYSMOD exists in the SMPPTS data set and the module was packaged on a RELFILE data set, therefore SMP/E attempted to allocate the corresponding SMPTLIB data set. If allocated, the module would be included from the SMPTLIB data set during link edit operations. However, the data set could not be allocated.

Message GIM65905 follows this message and identifies the load modules that could not be built.

System Action: SYSMOD processing stops.

Programmer Response: Correct the allocation error described in previous messages and rerun the job.

GIM65904I *membtype membname* **IN THE** *library* **LIBRARY IS NEEDED TO BUILD THE FOLLOWING LOAD MODULES FOR SYSMOD** *sysmod* **BUT** *library* **COULD NOT BE ALLOCATED.**

Explanation:

membtype	member type
membname	member name
library	library ddname
sysmod	SYSMOD ID

SMP/E required the specified member to build load modules, but the library that contains the member could not be allocated. This member could be any one of the following:

Source element
In this case, SMP/E attempted to assemble the source element to obtain a copy of a module to build the indicated load module. However, the library where the source element resides could not be allocated.

Module element
In this case, either:
- SMP/E attempted to use the copy of the module within the module's distribution library to build the indicated load module. However, the distribution library could not be allocated.
- SMP/E attempted to use the copy of the module within the SYSMOD that last replaced the module in the target system. This SYSMOD is the module's target zone RMID. The SYSMOD exists in the SMPPTS data set and the module was packaged in either an LKLIB or TXLIB data set, therefore SMP/E attempted to allocate the data set. If allocated, the module would be included from the data set during link edit operations. However, the data set could not be allocated.

Load module
In this case, load module *membname* contains only one module. This module is needed to build other load modules, therefore SMP/E attempted to use this module. However, the library that contains load module *membname* could not be allocated.

GIM65905I **– LOAD MODULE** *loadmod***.**

Explanation:

loadmod	load module name

This message identifies a load module that could not be completely built because of a condition described by the preceding messages.

System Action: None.

Programmer Response: None.

GIM66001I THE JCLIN CROSS-REFERENCE REPORT WAS FORMATTED IN ONE COLUMN BECAUSE SMP/E COULD NOT OBTAIN ENOUGH STORAGE FOR TWO COLUMNS.

Explanation: SMP/E does not have enough storage to format the JCLIN Cross-Reference report in two columns. Single-column format will be used.

System Action: None.

Programmer Response: None.

GIM66101I THE JCLIN CROSS-REFERENCE REPORT WAS FORMATTED IN ONE COLUMN BECAUSE SMP/E COULD NOT OBTAIN ENOUGH STORAGE FOR TWO COLUMNS.

Explanation: SMP/E does not have enough storage to format the JCLIN Cross-Reference report in two columns. Single column format will be used.

System Action: None.

Programmer Response: None.

GIM66201S THERE ARE NO *enttype* ENTRIES IN THE *zonename* ZONE.

Explanation:

enttype	DDDEF or UTILITY
zonename	zone name

The DDDEF or UTILITY entry to be changed does not exist in the indicated zone.

System Action: Command processing stops.

Programmer Response:

1. Check whether the specified zone is the one you want to change.
2. If the zone is correct, check to see if the job or command to define the entries was successful. If not, then either rerun that job or command, or use UCLIN to create the entries.

GIM66301E THE ACCJCLIN SUBENTRY IS ONLY ALLOWED FOR A DLIBZONE ENTRY.

Explanation: ACCJCLIN was specified on a UCL statement to control whether inline JCLIN is saved at ACCEPT time. However, the UCL statement did not specify a DLIBZONE entry.

System Action: SMP/E does not change the ACCJCLIN value. Processing continues with the next command.

Programmer Response: Specify the appropriate DLIBZONE entry on the UCL statement and run the UCLIN again.

GIM66501E *value* WAS SPECIFIED MORE THAN ONCE ON THE *operand* OPERAND OF THE ++*mcstype* MCS FOR *elmname* IN SYSMOD *sysmod*. THIS IS NOT ALLOWED.

Explanation:

value	operand value
operand	operand type
mcstype	MCS type
elmname	element name
sysmod	SYSMOD ID

The indicated MCS specifies the same operand value more than once.

System Action: The SYSMOD is not received.

Programmer Response: Remove the duplicate value and rerun the job.

GIM66601E *operand* VALUE *value* IS THE SAME AS THE ELEMENT NAME ON THE ++*mcstype* MCS FOR *elmname* IN SYSMOD *sysmod*. THESE VALUES MUST BE DIFFERENT.

Explanation:

operand	operand type
value	operand value
mcstype	MCS type
elmname	element name
sysmod	SYSMOD ID

The indicated operand value must be different from the element name.

System Action: The SYSMOD is not received.

Programmer Response: Change the operand value or the element name specified on the indicated MCS, as appropriate, and rerun the job.

GIM66701T DUMPON IS REQUIRED WITH THE SNAP OPERAND.

Explanation: To get a SNAP dump, you must enter DUMPON, a dump ID, and SNAP on the DEBUG command. However, DUMPON was missing from the command being processed.

System Action: DEBUG processing stops.

Programmer Response: Add the DUMPON operand, then rerun the job.

GIM66801E *enttype1* **ENTRY** *entname* **WAS NOT DELETED FROM THE** *dataset* **DATA SET BECAUSE THERE IS NO CORRESPONDING** *enttype2* **ENTRY FOR** *entname* **IN THE** *zonename* **TARGET ZONE.**

Explanation:

enttype1	MTSMAC or STSSRC
entname	entry name
dataset	SMPMTS or SMPSTS
enttype2	MAC or SRC
zonename	zone name

During CLEANUP processing, SMP/E could not delete the specified entry, because no corresponding entry existed in the target zone.

System Action: CLEANUP processing continues.

Programmer Response: None.

GIM66802E **BACKUP ENTRIES FOR** *sysmod* **WERE NOT DELETED FROM THE SMPSCDS DATA SET BECAUSE THERE IS NO CORRESPONDING SYSMOD ENTRY FOR** *sysmod* **IN THE** *zonename* **TARGET ZONE.**

Explanation:

sysmod	SYSMOD ID
zonename	zone name

During CLEANUP processing, SMP/E could not delete the specified entry, because no corresponding entry existed in the target zone.

System Action: CLEANUP processing continues.

Programmer Response: None.

GIM66901E **THE** *operand* **OPERAND EXCEEDS THE** *number***-VALUE MAXIMUM ON THE** *mcstype* **MCS FOR** *elmname* **IN SYSMOD** *sysmod***.**

Explanation:

operand	FMID or SYSLIB
number	10 (for FMID) or 2 (for SYSLIB)
mcstype	++MOVE (for FMID) or ++DELETE (for SYSLIB)
elmname	element name
sysmod	SYSMOD ID

The number of values entered on the specified MCS statement exceeds the maximum allowed.

System Action: SYSMOD processing stops.

Programmer Response: Fix the MCS statement so that it specifies an allowable number of values for the indicated operand. Then rerun the job.

GIM67001E **THE SAME VALUE (***zonename***) WAS SPECIFIED FOR THE RELATED ZONE AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.**

Explanation:

zonename	zone name

On the UCLIN command used to define a TARGETZONE or DLIBZONE entry, the RELATED zone name was the same as the zone name specified on the SET command.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Correct the values so they are not the same. Then rerun the command.

GIM67002S **THE SAME VALUE (***value***) WAS SPECIFIED FOR THE OUTFILE DDNAME AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.**

Explanation:

value	specified value

On the ZONEEXPORT command, the OUTFILE ddname was the same as the zone name specified on the SET command.

System Action: Command processing stops.

Programmer Response: Correct the values so they are not the same. Then rerun the command.

GIM67003S **THE SAME VALUE (***value***) WAS SPECIFIED FOR THE INFILE DDNAME AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.**

Explanation:

value	specified value

On the ZONEIMPORT command, the INFILE ddname was the same as the zone name specified on the SET command.

System Action: Command processing stops.

Programmer Response: Correct the values so they are not the same. Then rerun the command.

GIM67101I THE REMAINING CHANGE STATEMENTS UP TO THE ENDZONEEDIT COMMAND WILL NOT BE PROCESSED.

Explanation: A severe error occurred during ZONEEDIT processing (the return code was 12 or higher).

System Action: SMP/E skips to the ENDZONEEDIT command without processing any more ZONEEDIT CHANGE statements.

Programmer Response: See "ZONEEDIT Return Codes" on page 284 for a description of ZONEEDIT return codes. Fix the indicated problem and rerun the job.

GIM67201E LOAD MODULE BUILD PROCESSING STOPPED BECAUSE NO RELATED DISTRIBUTION ZONE IS DEFINED FOR TARGET ZONE *zonename*.

Explanation:

zonename	zone name

While building a load module in a target library, SMP/E needed to check the related distribution zone for a required module. However, no related distribution zone is defined for the target zone.

System Action: SMP/E stops building load modules that need the missing distribution zone. It also issues the following messages:
- GIM674*xx*, which lists the load modules that failed
- GIM67501I, which lists the modules SMP/E requires to complete the load modules
- GIM22601I or GIM22601E, which lists the SYSMODs that failed because load modules failed

Programmer Response: Specify the related distribution zone in the TARGETZONE entry and rerun the job.

GIM67301W LOAD MODULE *loadmod* IN SYSLIB *syslib* DOES NOT INCLUDE MODULE *modname* BECAUSE *modname* HAS NOT BEEN INSTALLED.

Explanation:

loadmod	load module name
syslib	SYSLIB value
modname	module name

A module that SMP/E needs to build the indicated load module has not been installed. As a result, the load module is link-edited without the missing module.

System Action: Processing continues.

Programmer Response: Link-edit the module into the load module outside SMP/E.

GIM67302W LOAD MODULE *loadmod* IN SYSLIBS *syslib1* AND *syslib2* DOES NOT INCLUDE MODULE *modname* BECAUSE *modname* HAS NOT BEEN INSTALLED.

Explanation:

loadmod	load module name
syslib1	SYSLIB value
syslib2	SYSLIB value
modname	module name

A module that SMP/E needs to build the indicated load module has not been installed. As a result, the load module is link-edited without the missing module.

System Action: Processing continues.

Programmer Response: Link-edit the module into the load module outside SMP/E.

GIM67401E LOAD MODULE *loadmod* WAS NOT BUILT IN SYSLIB *syslib*.

Explanation:

loadmod	load module name
syslib	SYSLIB value

SMP/E stopped building the indicated load module because of a previous error. Check for messages (such as GIM67201E) that explain this error.

System Action: Load module processing ends. SMP/E issues the following related messages:
- GIM67501I, which lists the modules SMP/E requires to complete the load module
- GIM22601I or GIM22601E, which lists the SYSMODs that failed because load modules failed

Programmer Response: Correct the error indicated in the other messages and rerun the job.

GIM67402E LOAD MODULE *loadmod* WAS NOT BUILT IN SYSLIBS *syslib1* AND *syslib2*.

Explanation:

loadmod	load module name
syslib1	SYSLIB value
syslib2	SYSLIB value

SMP/E stopped building the indicated load module because of a previous error. Check for messages (such as GIM67201E) that explain this error.

System Action: Load module processing ends. SMP/E issues the following related messages:
- GIM67501I, which lists the modules SMP/E requires to complete the load module
- GIM22601I or GIM22601E, which lists the SYSMODs that failed because load modules failed

Programmer Response: Correct the error indicated in the other messages and rerun the job.

GIM67501I MODULE *modname* IS REQUIRED FOR LOAD MODULE *loadmod*.

Explanation:

modname	module name
loadmod	load module name

Because of a previous error, SMP/E cannot get the modules it needs to build the indicated load module. This message lists these modules. A previous message (such as GIM65701E or GIM67201E) tells what error is preventing SMP/E from getting the modules.

System Action: Processing for the indicated load module stops. Processing continues for other load modules and SYSMODs.

Programmer Response: Correct the error indicated in the other messages and rerun the job.

GIM67601I SMPTLIB *dataset* WAS CATALOGED ON VOLUME *volser*.

Explanation:

dataset	data set name
volser	volume serial number

During RECEIVE processing, SMP/E cataloged an existing SMPTLIB data set on the indicated volume.

System Action: Processing continues.

Programmer Response: None.

GIM67602I SMPTLIB *dataset* WAS DELETED FROM VOLUME *volser*.

Explanation:

dataset	data set name
volser	volume serial number

During command processing, SMP/E deleted an existing SMPTLIB from the indicated volume.

System Action: Processing continues.

Programmer Response: None.

GIM67701I LOG RECORDING WAS SWITCHED TO SMPOUT BECAUSE SMPLOGA DOES NOT HAVE THE SAME BLKSIZE VALUE AS SMPLOG.

Explanation: Because the SMPLOG data set is full, SMP/E tried to record log data in SMPLOGA. However, SMPLOGA does not have the same block size as SMPLOG. SMP/E is therefore using SMPOUT.

System Action: Processing continues.

Programmer Response: Make sure the SMPLOG and SMPLOGA data sets have the same block size.

GIM67801T DUMPOFF IS NOT ALLOWED FOR *value* BECAUSE *value* IS ALREADY OFF.

Explanation:

value	dump ID or VPLFUNCT value

The DEBUG command specified DUMPOFF for a dump ID or VPLFUNCT value that was already turned off.

System Action: DEBUG processing stops.

Programmer Response: Rerun the job without turning off the specified dump points.

GIM67802T DUMPOFF IS NOT ALLOWED FOR MSGIDS BECAUSE DUMPMSG IS ALREADY OFF.

Explanation: The DEBUG command specified DUMPOFF for all message IDs, but DUMPMSG was already turned off.

System Action: DEBUG processing stops.

Programmer Response: Rerun the job without turning off the specified dump points.

GIM67901W MOD ENTRY *entname* CONTAINS THE SAME VALUE (*subval*) FOR THE TALIAS AND LMOD SUBENTRIES. THE TALIAS SUBENTRY WAS IGNORED AND SHOULD BE DELETED.

Explanation:

entname	entry name
subval	subentry value

The indicated MOD entry specified the same value for the TALIAS and LMOD subentries. This is not allowed.

System Action: GENERATE ignores the TALIAS subentry and builds a step to copy the module to the indicated LMOD.

Programmer Response: Use the UCLIN DEL statement to delete the TALIAS subentry from the MOD entry.

GIM68001I THE DUMP ID FOR *msgid* IS *dumpid*.

Explanation:

msgid	SMP/E message number
dumpid	dump ID

This is an informational message issued for the DEBUG DUMPMSG command. SMP/E will associate the indicated dump ID with the dump that was requested for the SMP/E message.

System Action: Processing continues.

Programmer Response: None.

GIM69000E THE DATA WAS NOT TRANSFORMED BECAUSE THE OPEN MACRO FAILED FOR THE *dataset* DATA SET.

Explanation:
dataset ddname of the data set that could not be opened

The indicated data set is required for GIMDTS. However, the specified data set name is incorrect.

System Action: GIMDTS processing stops.

Programmer Response: Do **one** of the following:
- If the JCL was missing a DD statement for the data set, add one.
- If the data set name was incorrect, specify the correct name.

Then rerun the job.

GIM69001E THE DATA WAS NOT TRANSFORMED BECAUSE THE *dataset* DATA SET IS MISSING.

Explanation:
dataset ddname of the missing data set

The indicated data set is required for GIMDTS. However, the JCL contained no DD statement for that data set.

System Action: GIMDTS processing stops.

Programmer Response: Add a DD statement for the data set and rerun the job.

GIM69002E THE DATA WAS NOT TRANSFORMED BECAUSE THE ATTRIBUTES OF THE *dataset* DATA SET ARE NOT SUPPORTED.

Explanation:
dataset ddname of the data set

GIMDTS does not transform data if the SYSUT1 or SYSUT2 data set does not have the correct attributes.

The SYSUT1 data set must have the following attributes:
- It must be a sequential data set or a member of a PDS.
- The record format (RECFM) must be F, FA, FM, FB, FBA, FBM, V, VA, VM, VB, VBA, or VBM.
- The input must not contain spanned records.

The SYSUT2 data set must have the following attributes:
- It must be a sequential data set or a member of a PDS.
- It must be on DASD.
- The record format (RECFM) must be F or FB.
- The logical record length (LRECL) must be 80.

System Action: GIMDTS processing stops.

Programmer Response: Redefine the data set with the correct attributes and rerun the job.

GIM69003E THE DATA WAS NOT TRANSFORMED BECAUSE THE GETMAIN MACRO FAILED FOR THE WORK AREAS.

Explanation: GIMDTS was not able to obtain the storage required to process the input records.

System Action: GIMDTS processing stops.

Programmer Response: Increase the value specified for the REGION parameter on the JOB or EXEC statement. Then rerun the job.

GIM69004T THE DATA WAS NOT TRANSFORMED BECAUSE OF AN I/O ERROR. SYNAD INFORMATION FOLLOWS – *jobname,stepname,unit address,device type,ddname,operation attempted,error description,actual track address and block number,access method*

Explanation: An I/O error occurred when GIMDTS tried to get records from SYSUT1 or write records to SYSUT2.

System Action: GIMDTS processing stops.

Programmer Response: Use the error description and other information supplied in the message to find the cause of the error. Fix the error and rerun the job.

GIM69005W THE SYSUT2 DATA SET WAS NOT BLOCKED OR HAD A BLOCKSIZE THAT WAS NOT A MULTIPLE OF 80. IT HAS BEEN BLOCKED WITH A BLOCKSIZE OF 3200.

Explanation: GIMDTS requires a SYSUT2 data set that has fixed-block records and a BLKSIZE value that is a multiple of 80. If the record format of the SYSUT2 data set is fixed but not blocked (RECFM=F), GIMDTS changes the RECFM value to FB and the BLKSIZE value to 3200. If the BLKSIZE value of the SYSUT2 data set is not a multiple of 80, GIMDTS changes the BLKSIZE value to 3200.

System Action: GIMDTS processing continues.

Programmer Response: None.

GIM69006I *sysut1* WAS TRANSFORMED INTO *sysut2*.

Explanation:
sysut1 data set name of the SYSUT1 data set

sysut2 data set name of the SYSUT2 data set

GIMDTS successfully processed the records from the SYSUT1 data set. It either transformed the input from the SYSUT1 data set into the SYSUT2 data set, or copied the input from SYSUT1 into SYSUT2, if the records did not need to be transformed.

System Action: GIMDTS processing continues.

Programmer Response: None.

GIM69007E THE DATA WAS NOT TRANSFORMED BECAUSE THE SYSUT1 DATA SET WAS EMPTY.

Explanation: The data set indicated by the SYSUT1 DD statement did not contain any records for GIMDTS to process.

System Action: GIMDTS processing stops.

Programmer Response: Make sure the SYSUT1 DD statement specifies the data set that contains the input for GIMDTS. Then rerun the job.

GIM69008I GIMDTS PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE IS *rtncode*.

Explanation:

rtncode
the highest return code from GIMDTS processing. GIMDTS may issue the following return codes:

- **00** The input data was processed successfully.
- **04** The SYSUT2 data set was reblocked.
- **08** The input data was not processed successfully.
- **16** An I/O error occurred.

System Action: If the return code was 00 or 04, GIMDTS processing continued normally. If the return code was 08 or 16, GIMDTS processing stopped.

Programmer Response: See the other messages that were issued by GIMDTS to determine what action to take.

GIM69009E *elmtype elmname* COULD NOT BE RETRANSFORMED BECAUSE THE *attribute* OF *value1* FOR LIBRARY *library* CONFLICTS WITH THE ORIGINAL *attribute* OF *value2* FOR THE DATA.

Explanation:

elmtype element type
elmname element name
attribute attribute type (RECFM or LRECL)
value1 attribute value for the library
library library ddname
value2 attribute value for the transformed element

When SMP/E was installing the indicated element, it compared the attributes of the library where the element should be installed and the original attributes of element, which are indicated in the records included with the transformed element. Because these attributes do not match, SMP/E did not retransform the element back to its original format and did not install the element in the indicated library.

System Action: SYSMOD processing stops.

Programmer Response: Check the program directory for the product that supplies the element to find out the correct RECFM and LRECL values required for the library where the element should be installed. Specify these values for the library and rerun the job.

GIM69010E AN ERROR OCCURRED IN RETRANSFORMATION OF *elmtype elmname* ON THE *library* LIBRARY. DATA HAS BEEN LOST.

Explanation:

elmtype element type
elmname element name
library data set name of the library where the element should be installed

When SMP/E was installing the indicated element, it compared the actual amount of transformed data and the amount indicated in the records included with the transformed element. Because these numbers do not match, SMP/E did not retransform the element back to its original format and did not install the element in the indicated library.

System Action: SYSMOD processing stops.

Programmer Response: This error usually indicates the transformed data has been truncated. Try obtaining a new copy of the SYSMOD that contains transformed data and rerunning the job. If the error still occurs, then report the error to your support group.

GIM69011I RETRANSFORMATION OF *elmtype elmname* ON LIBRARY *library* WAS UNSUCCESSFUL FOR SYSMOD *sysmod*.

Explanation:

elmtype element type
elmname element name
library data set name of the library where the element should be installed
sysmod SYSMOD ID that supplied the element

The indicated element had been transformed by GIMDTS and was packaged inline. When SMP/E tried

to install the element, it did not retransform the element back to its original format. A previous message gives the reason for this failure.

System Action: SYSMOD processing stops.

Programmer Response: See the description of the related previous message for the action you should take.

GIM69012I RETRANSFORMATION OF *elmtype elmname* ON LIBRARY *library* WAS SUCCESSFUL.

Explanation:

elmtype	element type
elmname	element name
library	data set name of the library where the element was installed

The indicated element had been transformed by GIMDTS and was packaged inline. SMP/E retransformed the element back to its original format and installed it in the indicated library.

System Action: Processing continues.

Programmer Response: None.

GIM69013S THE ZONES TO BE COMPARED HAVE NO MATCHING SREL.

Explanation: The zone definitions specified on the REPORT SYSMODS command did not contain any matching SREL values. As a result, none of the SYSMODs in the input zone are applicable to the comparison zone.

System Action: Command processing stops.

Programmer Response: Check whether the correct zones were specified on the command being processed.

- If the zones were specified correctly, no report can be produced. No further action is required.
- If the zones were not specified correctly, specify the correct zone names and rerun the command.

GIM69014S THE *zonename* ZONE WAS SPECIFIED ON THE *operand* OPERAND BUT IT DOES NOT EXIST.

Explanation:

zonename	zone name
operand	COMPAREDTO or INZONE

The specified zone does not exist for one of the following reasons:

- There is no ZONEINDEX subentry for the zone in the global zone.
- The zone is not defined in the data set indicated by its ZONEINDEX subentry.

System Action: Command processing stops.

Programmer Response: Check whether the correct zone was specified on the indicated operand.

- If the zone was specified correctly, then perhaps the wrong global zone was being used. If so, specify the correct CSI data set for the global zone on the EXEC statement for SMP/E or on the SMPCSI DD statement and rerun the command.
- If the zone was not specified correctly, specify the correct zone name and rerun the command.

GIM69015E THE GLOBAL ZONE IS NOT ALLOWED AS THE VALUE OF THE *operand* OPERAND.

Explanation:

operand	RECZGRP or RECEXZGRP

GLOBAL is not allowed as the zone name specified on the indicated operand. You must specify the name of the target or distribution zone.

System Action: Processing stops for the UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM69015S THE GLOBAL ZONE IS NOT ALLOWED AS THE VALUE OF THE *operand* OPERAND.

Explanation:

operand	COMPAREDTO, INZONE, or XZGROUP

GLOBAL is not allowed as a value for the indicated operand. For the INZONE and COMPAREDTO operands, you must specify the name of a target or distribution zone. For the XZGROUP operand, you must specify the name of a target zone, a distribution zone, or a ZONESET entry.

System Action: Command processing stops.

Programmer Response: Specify a valid value for the indicated operand and rerun the command.

GIM69016S THE INZONE AND COMPAREDTO OPERANDS SPECIFIED THE SAME ZONE NAME.

Explanation: The same zone name was specified on both the INZONE and COMPAREDTO operands. The REPORT SYSMODS command does not compare a zone to itself.

System Action: Command processing stops.

Programmer Response: Specify different zone names on the INZONE and COMPAREDTO operands. (Make sure these are names of target or distribution zones.) Then rerun the command.

GIM69017S THE *zonename* ZONE WAS SPECIFIED ON THE *operand* OPERAND BUT IT CONTAINED NO SYSMOD ENTRIES.

Explanation:

zonename	zone name
operand	COMPAREDTO or INZONE

The indicated zone contains no SYSMOD entries. The REPORT SYSMODS command only compares zones that contain at least one SYSMOD entry.

System Action: Command processing stops.

Programmer Response: Check whether the correct zone was specified on the indicated operand.

- If the zone was specified correctly, no report can be produced. No further action is required.
- If the zone was not specified correctly, specify the correct zone name and rerun the command.

GIM69031E SMP/E COULD NOT ADD SYSLIB *syslib* TO *enttype entname* BECAUSE THE MAXIMUM NUMBER OF SYSLIB SUBENTRIES WOULD HAVE BEEN EXCEEDED.

Explanation:

syslib	SYSLIB name
enttype	entry type (DLIB, LMOD, or element)
entname	entry name

SMP/E tried to add a new SYSLIB subentry to the indicated entry. However, by adding this new SYSLIB subentry, the maximum number of SYSLIB subentries would have been exceeded.

- DLIB entries may contain one or two SYSLIB subentries.
- LMOD entries may contain one or two SYSLIB subentries.
- Element entries may contain only one SYSLIB subentry.

As a result, SMP/E did not add the new SYSLIB subentry to the entry.

System Action: SMP/E ignores the duplicate UCL statement, and processing continues with the next statement.

Programmer Response: If you want to replace an existing SYSLIB subentry with the new SYSLIB value, use the UCL REP statement instead of the UCL ADD statement.

- If the entry contains only one SYSLIB subentry, replace the existing value with the new value. For example:

```
UCLIN.
REP MAC(MAC01)
    SYSLIB(new-value)
ENDUCL.
```

- If the entry contains two SYSLIB subentries, replace the existing list with the new list. For example:

```
UCLIN.
REP LMOD(MOD01)
    SYSLIB(old-value,new-value)
ENDUCL.
```

GIM69032S SOURCEID *sourceid* HAS ALREADY BEEN SPECIFIED ON EITHER THE SOURCEID OR EXSRCID OPERAND.

Explanation:

sourceid	SOURCEID value

The indicated SOURCEID value was repeated in one of these ways:

- It was specified more than once on the SOURCEID operand.
- It was specified more than once on the EXSRCID operand.
- It was specified on both the SOURCEID and EXSRCID operands.

None of these are allowed.

System Action: Command processing stops.

Programmer Response: Specify the SOURCEID value only once. Then rerun the job.

GIM69033E *enttype entname* WAS NOT RENAMED BY SYSMOD *sysmod* BECAUSE *entname* ALREADY EXISTS IN THE SPECIFIED *library* LIBRARY.

Explanation:

enttype	entry type
entname	entry name
sysmod	SYSMOD ID
library	ddname of the library

There was no entry for the indicated member in the target or distribution zone, but the member does exist in the specified library. It may have been added, moved, or renamed outside of SMP/E, or its entry may have been deleted by UCLIN.

System Action: SYSMOD processing stops.

Programmer Response: First determine why the member is in the library.

- If it was added, moved, or renamed outside of SMP/E, reconstruct the entry in the appropriate zones.
- If it was added by SMP/E using a different element type, decide whether to move the member to a new library, or delete it from the library.
- If the entry was deleted by UCLIN, decide whether to reconstruct the entry or delete the member from the library.

Then rerun the job.

GIM69034S ZONE *zonename1* WAS NOT RENAMED BECAUSE THE NEW ZONE NAME (*zonename2*) IS THE SAME AS THE NAME FOR A TIEDTO ZONE.

Explanation:

zonename1	name of zone to be renamed
zonename2	new zone name

The new zone name is the same as the name for a TIEDTO zone. This is not allowed.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Choose a different new name for the zone to be renamed. Then rerun the command.

GIM69035E THE SAME VALUE (*zonename*) WAS SPECIFIED FOR THE TIEDTO ZONE AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.

Explanation:

zonename	zone name

On the UCLIN command used to define a TARGETZONE entry, the TIEDTO zone name was the same as the zone name specified on the SET command.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Correct values so they are not the same.

GIM69036E THE *subtype1* SUBENTRY FOR *enttype entname* WILL NOT BE DELETED BECAUSE *entname* CONTAINS *subtype2* SUBENTRIES.

Explanation:

subtype1	XZLMODP or XZMODP
enttype	MOD or LMOD
entname	MOD name or LMOD name
subtype2	XZLMOD or XZMOD

The XZLMODP and XZMODP subentry indicators tell SMP/E that a module is included in a LMOD in a different zone or that an LMOD contains modules from another zone. The deletion of the XZLMODP or the XZMODP subentries is not allowed if SMP/E determines that (a) the module is included in a LMOD in a different zone, or (b) the LMOD contains modules from a different zone, because this deletion causes the entry to be incorrect.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Rework the command so that the XZMODP or XZLMODP subentry is not deleted. Then rerun the command, if necessary.

GIM69037E THE *subtype1* SUBENTRY FOR *enttype entname* WILL NOT BE ADDED BECAUSE AFTER PROCESSING IS COMPLETE NO *subtype2* SUBENTRY WILL EXIST FOR *entname*.

Explanation:

subtype1	XZLMODP or XZMODP
enttype	MOD or LMOD
entname	MOD name or LMOD name
subtype2	XZLMOD or XZMOD

The XZLMODP and XZMODP subentry indicators tell SMP/E that a module is included in a LMOD in a different zone or that an LMOD contains modules from another zone. The addition of the XZLMODP or the XZMODP subentries is not allowed if SMP/E determines that (a) the module is not included in a LMOD in a different zone, or (b) the LMOD does not contain modules from a different zone, because this addition causes the entry to be incorrect.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Rework the command so that the XZMODP or XZLMODP subentry is not added. Then rerun the command, if necessary.

GIM69038E THE SAME VALUE (*zonename*) WAS SPECIFIED FOR THE ZONE CONTAINING THE CROSS-ZONE *entname* AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.

Explanation:

zonename	zone name
entname	LMOD or MOD

On the UCLIN command used to update the XZLMOD subentry of the MOD entry or the XZMOD subentry of the LMOD entry, the zone containing the cross-zone MOD or LMOD is the same as the zone specified on the previous SET command.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Correct the zone values so they are not the same. Then rerun the command.

GIM69039E UCLIN PROCESSING FAILED BECAUSE EXISTING XZMOD SUBENTRIES INDICATE THAT MODULE *modname* EXISTS IN ZONE *zonename1*, BUT THE CURRENT COMMAND INDICATES THAT ZONE *zonename2* SHOULD CONTAIN MODULE *modname*.

Explanation:

modname	MOD name
zonename1	zone name from XZMOD subentries
zonename2	zone name from command

This error occurs when an XZMOD record already exists in the zone for the LMOD. This indicates that the module resides in a different zone from the one specified on the UCLIN statement.

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Correct values so that only one zone is specified for the module. Then rerun the command.

GIM69040E THE *operand* OPERAND IS ONLY ALLOWED FOR A *zonetype* ZONE.

Explanation:

operand	operand
zonetype	zone type

An operand specified on the command being processed is not valid for the type of zone specified on the previous SET command.

System Action: SYSMOD processing stops.

Programmer Response: Either change the operand, or change the zone specified on the previous SET command. Then rerun the job.

GIM69040S THE *operand* OPERAND IS ONLY ALLOWED FOR A *zonetype* ZONE.

Explanation:

operand	operand
zonetype	zone type

An operand specified on the command being processed is not valid for the type of zone specified on the previous SET command.

System Action: Command processing stops.

Programmer Response: Either change the operand, or change the zone specified on the previous SET command. Then rerun the job.

GIM69041S CONDITIONAL CHANGES ARE NOT ALLOWED WHEN ZONEEDIT *enttype* IS SPECIFIED.

Explanation:

enttype	entry type

Conditional ZONEEDIT CHANGE statements are not allowed for the specified entry.

System Action: Command processing stops.

Programmer Response: Fix the CHANGE statement so that it is no longer conditional. Then rerun the job.

GIM69042S AN ASTERISK ("*") IS NOT ALLOWED FOR THE FROM VALUE OF A CHANGE STATEMENT FOR ZONEVALUE.

Explanation: Specifying an asterisk for the *from-value* causes SMP/E to change the existing value to the new value. However, when the CHANGE statement is for ZONEVALUE, it is possible to have more that one existing zone name. As a result, if an asterisk were specified for the *from-value*, SMP/E would not be able to determine which of the existing values to update.

System Action: Command processing stops.

Programmer Response: Fix the CHANGE statement so that an asterisk is no longer specified for the *from-value*. Then rerun the job.

GIM69043S THE SAME VALUE (*zonename*) WAS SPECIFIED FOR THE CROSS-ZONE NAME AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.

Explanation:

zonename	zone name

On the ZONEEDIT command for ZONEVALUE, the *to-value* is the same as the zone name specified on the SET command. This is not allowed.

System Action: Command processing stops. Processing continues with the next command.

Programmer Response: Correct the values so they are not the same. Then rerun the command.

GIM69044W A TIEDTO SUBENTRY WAS FOUND FOR ZONE *zonename*.

Explanation:

zonename	zone name

This message serves as a reminder that corrective action may be needed to keep the zone identified in the message and the current zone connected properly.

System Action: Processing continues.

Programmer Response: To keep the zones properly connected, you can use the ZONEEDIT command or UCLIN to update the cross-zone information in either the zone identified by the message, the current (set-to) zone, or both zones. For details on ZONEEDIT and UCLIN, see *SMP/E Commands*.

GIM69045W *enttype1 entname1* **IN ZONE** *zonename1* **CONTAINS AN** *subtype* **SUBENTRY INDICATING A CROSS-ZONE RELATIONSHIP WITH** *enttype2 entname2* **IN ZONE** *zonename2***.**

Explanation:

enttype1	MOD or LMOD
entname1	MOD name or LMOD name
zonename1	zone name
subtype	XZLMOD or XZMOD
enttype2	LMOD or MOD
entname2	LMOD name or MOD name
zonename2	cross-zone name

This message serves as a reminder that corrective action may be needed to keep the cross-zone identified in the message and the current zone connected properly.

System Action: Processing continues.

Programmer Response: To keep the zones properly connected, you can use the ZONEEDIT command or UCLIN to update the cross-zone information in either the zone identified by the message, the current (set-to) zone, or both zones. For details on ZONEEDIT and UCLIN, see *SMP/E Commands*.

GIM69046W **LMOD** *loadmod* **CONTAINS MOD** *modname* **FROM ZONE** *zonename***. MOD** *modname* **WILL NOT BE INCLUDED IN THE LINK-EDIT STEP FOR LMOD** *loadmod***.**

Explanation:

loadmod	name of the load module containing cross-zone information
modname	name of the module that is controlled by another zone
zonename	name of the zone that controls the module

During GENERATE processing, a load module was encountered that contains a module from another zone. GENERATE will not include the cross-zone module in the load module.

System Action: Processing continues.

Programmer Response: If you want the cross-zone module to be part of the load module, use the LINK command to link-edit the cross-zone module into the load module created by running the generated JCL.

GIM69047E *enttype1 entname1* **CONTAINS AN** *subtype* **SUBENTRY INDICATING A CROSS-ZONE RELATIONSHIP WITH** *enttype2 entname2* **IN ZONE** *zonename***. THIS SUBENTRY WAS NOT ADDED TO THE INTO ZONE BECAUSE THE CROSS-ZONE NAME MATCHES THE INTO ZONE NAME (***zonename***).**

Explanation:

enttype1	MOD or LMOD
entname1	MOD name or LMOD name
subtype	XZLMOD or XZMOD
enttype2	LMOD or MOD
entname2	LMOD name or MOD name
zonename	cross-zone name (same as INTO zone name)

This message indicates that if the subentry had been added, the zone specified on the INTO operand would have been the same as one of its cross-zones. This is not allowed.

System Action: Command processing continues. The subentry is not added to the INTO zone.

Programmer Response: Do the following:
1. Use the ZONEDELETE command to delete the INTO zone.
2. List all the remaining ZONEINDEX subentries (to avoid duplicates).
3. Define a new ZONEINDEX subentry for the INTO zone (with a new and unique zone name).

GIM69048S *enttype* **ENTRY** *entname* **IS NEEDED FOR LINK PROCESSING BUT IS NOT IN ZONE** *zonename***.**

Explanation:

enttype	entry type (MOD or LMOD)
entname	MOD name or LMOD name
zonename	cross-zone name, set-to zone name, or name of the DLIB zone related to the cross-zone

Entry *entname* is needed to process the LINK command, but it either is not in the indicated zone, or it exists only as an entry containing just XZMOD or XZLMOD subentries.

System Action: LINK processing stops.

Programmer Response: Make sure the correct entry name is specified on the LINK command.

GIM69049S *enttype entname* **IS NEEDED FOR LINK PROCESSING BUT THE ENTRY IN** *zonetype* **ZONE** *zonename* **DOES NOT CONTAIN BOTH AN FMID AND AN RMID.**

Explanation:

enttype	entry type (MOD)

entname entry name
zonetype zone type
zonename zone name

Entry *entname* exists in the indicated zone, but it does not contain both an FMID value and an RMID value. Therefore, it is not really installed in the zone and cannot be processed by the LINK command.

System Action: LINK processing stops.

Programmer Response: Install *entname* and rerun the LINK command.

GIM69050S LMOD *loadmod* FOR MOD *modname* IN ZONE *zonename* IS NEEDED FOR LINK PROCESSING BUT IS NOT INSTALLED IN A SYSTEM LIBRARY.

Explanation:
loadmod LMOD name
modname MOD name
zonename cross-zone name from FROMZONE operand

SMP/E determined that the indicated LMOD contained a usable copy of module *modname* in the indicated cross-zone. However, the load module does not exist in any of the SYSLIBs defined in the LMOD entry.

System Action: LINK processing stops.

Programmer Response: Make sure that LMOD entry *loadmod* contains the correct SYSLIB subentries.

GIM69051S LMOD *loadmod* IS NEEDED FOR LINK PROCESSING BUT IS MARKED AS BEING COPIED.

Explanation:
loadmod LMOD name

LMOD *loadmod* is a single-module load module. The LINK command does not add another module to a copied module. Copied load modules do not have any linkage editor control statements that allow for the proper management of a multiple-module load module.

System Action: LINK processing stops.

Programmer Response: Check that the correct LMOD name has been specified on the LINK command:

- If the LMOD name was specified correctly, it cannot be link-edited.
- If the LMOD name was not specified correctly, specify the correct LMOD name and rerun the command.

GIM69052S SMP/E WILL NOT LINK MOD *modname* FROM ZONE *zonename1* INTO LMOD *loadmod* BECAUSE THIS MOD WAS PREVIOUSLY INCLUDED FROM ZONE *zonename2*.

Explanation:
modname module name
zonename1 FROMZONE value specified on LINK command
loadmod load module name
zonename2 current cross-zone value for module

You cannot link MOD *modname* into LMOD *loadmod* because another MOD by the same name (but from a different zone) was previously linked into that LMOD.

System Action: LINK processing stops.

Programmer Response: Check that the correct MOD name and FROMZONE value are specified on the LINK command.

GIM69053I LINK PROCESSING WAS SUCCESSFUL FOR LMOD *loadmod* IN SYSLIB *syslib*. THE RETURN CODE WAS *rtncode*.

Explanation:
loadmod LMOD name
syslib SYSLIB value
rtncode return code

LINK processing completed successfully with the indicated return code.

Note: Date, time, and sequence number information is **not** generated for this utility completion message, because the volume of utility output for the LINK command is expected to be small.

System Action: Processing continues.

Programmer Response: No action is required.

GIM69053W LINK PROCESSING WAS SUCCESSFUL FOR LMOD *loadmod* IN SYSLIB *syslib*. THE RETURN CODE WAS *rtncode*.

Explanation:
loadmod LMOD name
syslib SYSLIB value
rtncode return code

LINK processing completed successfully with the indicated return code.

Note: Date, time, and sequence number information is **not** generated for this utility completion message, because the volume of utility output for the LINK command is expected to be small.

System Action: Processing continues.

Programmer Response: No action is required. However, because the return code indicates that an error may have occurred, you should check the utility output to verify that the results are acceptable.

GIM69054E LINK PROCESSING FAILED FOR LMOD *loadmod* **IN SYSLIB** *syslib***. THE RETURN CODE WAS** *rtncode***.**

Explanation:

loadmod	LMOD name
syslib	SYSLIB value
rtncode	return code

LINK processing failed with the indicated return code.

Note: Date, time, and sequence number information is **not** generated for this utility completion message, because the volume of utility output for the LINK command is expected to be small.

System Action: The return code determines the system action.

Programmer Response: See the utility output for the cause of the error. Fix the error and rerun the LINK command for this load module.

Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:

- The PRINT subentry specifies a DDDEF of DUMMY.
- The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

GIM69055W MOD *modname* **IN ZONE** *zonename1* **WILL NOT BE LINKED INTO LMOD** *lmodname* **IN ZONE** *zonename2* **BECAUSE CROSS-ZONE UPDATES ARE BEING DEFERRED.**

Explanation:

modname	MOD name
zonename1	zone specified on current SET command
lmodname	LMOD name
zonename2	cross-zone name for load module

The indicated module has been changed by SMP/E and is contained in a cross-zone load module. However, the cross-zone load module was not updated with the new version of this module because the TARGETZONE entry for the load module zone indicates that cross-zone updates should be deferred.

System Action: Command processing continues.

Programmer Response: Use the LINK command to update the cross-zone load module.

GIM69056W CROSS-ZONE PROCESSING WILL NOT BE DONE FOR LMOD *loadmod* **IN ZONE** *zonename* **BECAUSE LMOD** *loadmod* **DOES NOT EXIST IN ZONE** *zonename***.**

Explanation:

loadmod	LMOD name
zonename	cross-zone name for load module

At least one module in the set-to zone refers to the indicated LMOD and was updated by the current APPLY or RESTORE command. SMP/E attempted to update the cross-zone LMOD but could not find it in the indicated cross-zone.

System Action: Command processing continues.

Programmer Response: Use the Cross-Zone Summary report to determine what cross-zone work was not done and the action you may need to take to complete the cross-zone work.

GIM69057W LMOD *loadmod* **IN ZONE** *zonename1* **DOES NOT REFER TO MOD** *modname* **IN ZONE** *zonename2***. CROSS-ZONE UPDATE PROCESSING TO LMOD** *loadmod* **FOR MOD** *modname* **WILL NOT BE ATTEMPTED.**

Explanation:

loadmod	LMOD name
zonename1	cross-zone name for load module
modname	MOD name
zonename2	zone specified on the current SET command

The indicated module has been changed by SMP/E and is contained in a cross-zone load module. However, the cross-zone load module entry does not indicate that it contains the module. The cross-zone load module was not updated with the new version of the module.

System Action: Command processing continues.

Programmer Response: If the cross-zone load module should contain the module, use the LINK command to update the LMOD with the indicated MOD.

GIM69058W CROSS-ZONE UPDATE PROCESSING WILL NOT BE ATTEMPTED FOR LMOD *loadmod* **IN ZONE** *zonename* **BECAUSE SYSLIB** *library* **COULD NOT BE ALLOCATED.**

Explanation:

loadmod	LMOD name
zonename	cross-zone name
library	library for load module

Cross-zone updates were not made to the indicated LMOD because SMP/E could not allocate a library for the LMOD.

System Action: Command processing continues.

Programmer Response: Fix the allocation problem. Use the Cross-Zone Summary report to determine what cross-zone work was not done for the LMOD and the action you may need to take to complete the cross-zone work.

GIM69059W **MOD** *modname* **IN ZONE** *zonename1* **WILL NOT BE LINKED INTO LMOD** *lmodname* **IN ZONE** *zonename2* **BECAUSE SMP/E COULD NOT FIND A USABLE COPY OF THE MODULE CONTAINING THE ZAP.**

Explanation:

modname	MOD name
zonename1	zone specified on the current SET command
lmodname	LMOD name
zonename2	cross-zone name for load module

The indicated module has been changed by SMP/E and is contained in a cross-zone load module. However, SMP/E could not find a usable copy of the module containing the ZAP to include in the cross-zone load module. As a result, the load module was not updated to include the ZAP.

System Action: Command processing continues.

Programmer Response: If the cross-zone load module should contain the updated module, use the LINK command to update the cross-zone LMOD with an updated copy of the indicated module.

Note: One way to obtain a copy of the updated module from the distribution library is to accept the SYSMOD that contains the module with the ZAP.

GIM69060W **MOD** *modname* **IN ZONE** *zonename1* **WILL BE LINKED INTO LMOD** *lmodname* **IN ZONE** *zonename2* **BUT UPDATES TO THIS MOD BY SYSMOD** *sysmod* **WILL NOT BE INCLUDED.**

Explanation:

modname	MOD name
zonename1	zone specified on the current SET command
lmodname	LMOD name
zonename2	cross-zone name for load module
sysmod	ID of SYSMOD that supplied the ZAP

The indicated module was both updated and replaced by SMP/E and is contained in a cross-zone load module. However, the module does not exist on the target library as a single-CSECT load module. As a result, the copy of the module that SMP/E used to update the cross-zone load module does not contain the updates made to the module by the indicated SYSMOD.

System Action: Command processing continues.

Programmer Response: If the cross-zone load module should contain the updated module, use the LINK command to update the cross-zone LMOD with an updated copy of the indicated module.

Note: One way to obtain a copy of the updated module from the distribution library is to accept the SYSMOD that contained the module with the ZAP.

GIM69061W **LINK-EDIT PROCESSING FAILED FOR MOD** *modname* **IN LMOD** *loadmod* **IN ZONE** *zonename* **IN SYSLIB** *syslib***. THE RETURN CODE WAS** *rtncode***.**

Explanation:

modname	MOD name
loadmod	LMOD name
zonename	cross-zone name for load module
syslib	SYSLIB ddname
rtncode	return code

An error occurred during link-edit processing for the indicated module and load module.

Note: Date, time, and sequence number information is **not** generated for this utility completion message, because the volume of utility output for the cross-zone processing of the APPLY and RESTORE commands is expected to be small.

System Action: Command processing continues.

Programmer Response: See the utility output for the cause of the error. Fix the error and use the LINK command to update the LMOD with the indicated MOD.

Note: If you did not get any utility output, check the value specified for the PRINT subentry in the UTILITY entry. This ddname is used for the SYSPRINT data set. Here are some typical reasons for not getting any utility output:

- The PRINT subentry specifies a DDDEF of DUMMY.
- The PRINT subentry specifies a data set that is sent to a SYSOUT class that suppresses output.

GIM69062I **LINK-EDIT PROCESSING WAS SUCCESSFUL FOR MOD** *modname* **IN LMOD** *loadmod* **IN ZONE** *zonename* **IN SYSLIB** *syslib***. THE RETURN CODE WAS** *rtncode***.**

Explanation:

modname MOD name
loadmod LMOD name
zonename cross-zone name for load module
syslib SYSLIB ddname
rtncode return code

Link-edit processing completed successfully for the indicated module and load module.

Note: Date, time, and sequence number information is **not** generated for this utility completion message, because the volume of utility output for the cross-zone processing of the APPLY and RESTORE commands is expected to be small.

System Action: Command processing continues.

Programmer Response: No action is required.

GIM69063W MOD *modname* **WILL NOT BE DELETED FROM LMOD** *loadmod* **IN ZONE** *zonename* **BECAUSE CROSS-ZONE UPDATES ARE BEING DEFERRED.**

Explanation:
modname MOD name
loadmod LMOD name
zonename cross-zone name for load module

SMP/E processing has deleted the indicated module, which is contained in a cross-zone load module. However, because the indicated target zone specifies that cross-zone updates should be deferred, the module was not deleted from the cross-zone load module.

System Action: Command processing continues.

Programmer Response: If desired, delete the module from the cross-zone load module outside of SMP/E.

GIM69064W MODULE *modname* **WILL BE LINK-EDITED BUT ITS SERVICE LEVEL OR FMID IN THE DISTRIBUTION ZONE IS DIFFERENT FROM THAT IN THE TARGET ZONE.**

Explanation:
modname module name

The LINK command required the specified module to build a load module. It found the module in the distribution zone, but the module was either at a different service level or was supplied by a different function SYSMOD. In spite of this discrepancy, the LINK command included the module.

System Action: The load module was link-edited with the module on the distribution zone. The load module may be regressed or have an incorrect module in it.

Programmer Response: If the load module was regressed or is incorrect because of the inclusion of the module from the distribution zone, link-edit the correct version of the module into the LMOD outside SMP/E.

GIM69065W PROGRAM *progname* **WAS REQUIRED FOR SMP/E** *command* **CROSS-ZONE PROCESSING BUT WAS NOT AVAILABLE.**

Explanation:
progname name of utility program
command command name

To do cross-zone processing, SMP/E checks the OPTIONS entry for the cross-zone to determine the names of the UTILITY entries to be used for cross-zone processing. SMP/E determined that the indicated program is required for cross-zone processing, but is currently not available.

System Action: Cross-zone processing for the current zone fails. Cross-zone processing continues with the next zone.

Programmer Response: Use the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to determine which cross-zone updates were not done. With the information in those reports, create an SMP/E LINK command (or, for ++RENAME changes, a UCLIN command) to update the indicated load modules.

GIM69066I LMOD *loadmod* **FOR MOD** *modname* **IS NOT IN ZONE** *zonename***.**

Explanation:
loadmod LMOD name
modname MOD name
zonename cross-zone name

Entry *loadmod* is needed for LINK processing but does not exist in the indicated zone.

System Action: Command processing continues.

Programmer Response: Check that the MOD entry specifies the correct LMOD.

- If the MOD is contained in the LMOD, add the LMOD entry to the cross-zone.
- If the MOD is not in the LMOD, delete the LMOD subentry from the MOD entry in the cross-zone.

GIM69067S AN ERROR OCCURRED WHEN SMP/E TRIED TO UPDATE MOD ENTRY *modname* **IN ZONE** *zonename* **TO SHOW THAT LMOD** *loadmod1* **WAS RENAMED TO** *loadmod2***.**

Explanation:
modname module name
zonename cross-zone
loadmod1 old LMOD name
loadmod2 new LMOD name

SMP/E ++RENAME processing has renamed the indicated load module that contains a cross-zone module. However, SMP/E could not update the cross-zone module entry to rename the cross-zone load module subentry.

System Action: Cross-zone processing fails.

Programmer Response: Update the cross-zone module entry using UCLIN.

GIM69068W **MOD** *modname* **IN ZONE** *zonename* **WAS NOT UPDATED TO SHOW THAT LMOD** *loadmod1* **WAS RENAMED TO** *loadmod2* **BECAUSE CROSS-ZONE UPDATES ARE BEING DEFERRED.**

Explanation:

modname	module name
zonename	cross-zone name for module
loadmod1	old load module name
loadmod2	new load module name

SMP/E ++RENAME processing has renamed the indicated load module, which contains a cross-zone module. However, SMP/E did not update the cross-zone MOD entry to rename the cross-zone load module subentry because the TARGETZONE entry specifies that cross-zone updates should be deferred.

System Action: Command processing continues.

Programmer Response: Update the cross-zone MOD entry using UCLIN.

GIM69069W **CROSS-ZONE PROCESSING WILL NOT BE ATTEMPTED FOR LMOD** *loadmod* **IN ZONE** *zonename* **BECAUSE THE CALLLIBS FOR LMOD** *loadmod* **COULD NOT BE ALLOCATED**

Explanation:

loadmod	load module name
zonename	cross-zone name

Cross-zone updates were not attempted for the indicated load module because SMP/E could not allocate the load module's CALLLIBS.

System Action: Processing continues.

Programmer Response: Correct the allocation error identified in earlier messages. Use the Cross-Zone Summary report to determine what cross-zone processing was not performed for the load module, and the action you may need to take to complete the cross-zone work.

GIM69070S *operand value* **HAS ALREADY BEEN SPECIFIED.**

Explanation:

operand	operand
value	operand value

The indicated operand value was previously specified on the command.

System Action: Command processing stops.

Programmer Response: Remove the duplicate specification and rerun the job.

GIM69071S **THE SAME VALUE (***zonename***) WAS SPECIFIED FOR THE FROMZONE OPERAND AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.**

Explanation:

zonename	FROMZONE value

The FROMZONE value specified on the LINK command cannot be the same as the zone specified on the current SET command.

System Action: LINK processing stops.

Programmer Response: Correct the syntax of the LINK command and rerun the job.

GIM69072S **THE** *command* **COMMAND WAS NOT PROCESSED BECAUSE THE FROMZONE IS NOT A TARGET ZONE.**

Explanation:

command	LINK

The FROMZONE value specified on the LINK command must be for a target zone.

System Action: LINK processing stops.

Programmer Response: Correct the syntax of the LINK command and rerun the job.

GIM69073I **MOD** *modname* **IN ZONE** *zonename* **WAS SUCCESSFULLY UPDATED TO SHOW THAT LMOD** *loadmod1* **WAS RENAMED TO** *loadmod2***.**

Explanation:

modname	module name
zonename	cross-zone name
loadmod1	old load module name
loadmod2	new load module name

SMP/E ++RENAME processing renamed the indicated load module, which contains a cross-zone module. SMP/E updated the cross-zone MOD entry to rename the XZLMOD subentry.

System Action: Command processing continues.

Programmer Response: No action is required.

GIM69074S *enttype entname* **IS NEEDED FOR LINK PROCESSING BUT IS NOT INSTALLED IN THE** *library* **LIBRARY.**

Explanation:

enttype	entry type (MOD)
entname	module name
library	ddname of the distribution library

SMP/E was unable to find a copy of *entname* to use for LINK processing.

System Action: LINK processing stops.

Programmer Response: Install *entname* in the indicated library and rerun the LINK command.

GIM69075W **LMOD** *loadmod* **IN ZONE** *zonename* **DOES NOT EXIST IN SYSLIB** *syslib***. CROSS-ZONE UPDATE PROCESSING WILL NOT BE ATTEMPTED FOR LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

loadmod	load module name
zonename	cross-zone name
syslib	ddname of SYSLIB data set

SMP/E could not update the cross-zone load module because it does not exist in the indicated SYSLIB data set.

System Action: Command processing continues.

Programmer Response: Determine why the LMOD does not exist in the SYSLIB and fix the problem.

Use the Cross-Zone Summary report to determine the cross-zone work that was not done for the LMOD and what action you may have to take to complete the cross-zone work.

GIM69076E **UCLIN PROCESSING FAILED BECAUSE THE XZMOD OPERAND SPECIFIED THE SAME MODULE (***modname***) FOR MORE THAN ONE ZONE.**

Explanation:

modname	module name

Two or more zones were specified to contain the same module. Here is an example of a UCLIN command that can create this error:

```
UCLIN.
ADD LMOD(lmod1) XZMOD((mod1,zone1), (mod1,zone2)).
ENDUCL.
```

System Action: Processing stops for the statement in error. Processing continues with the next statement.

Programmer Response: Correct the UCL statement so only one zone is specified for the module. Then rerun the command.

GIM69077S **AN I/O ERROR OCCURRED WHEN SMP/E WAS TRYING TO LOCATE** *enttype entname* **IN CROSS-ZONE** *zonename***.**

Explanation:

enttype	entry type (LMOD)
entname	load module name
zonename	cross-zone name

SMP/E was unable to complete cross-zone update processing for the indicated load module because of an I/O error.

System Action: Command processing fails.

Programmer Response: Fix the I/O error.

GIM69078W **LMOD** *loadmod* **CONTAINS A MODULE CONTROLLED BY ANOTHER ZONE AND MODULES THAT WERE NOT SELECTED TO BE INSTALLED. THE CURRENT LOAD MODULE WILL BE INCLUDED IN THE LINK-EDIT STEP FOR THE NEW LOAD MODULE.**

Explanation:

loadmod	name of the load module containing cross-zone information and nonselected modules

During GENERATE processing, a load module was encountered that contains a module from another zone. This was indicated by message GIM69046W. Additionally, not all the modules in that load module were selected to be installed. Because there are nonselected modules, the link-edit step generated for this load module will have a control card to include the current load module. The inclusion of the current load module may make the cross-zone module part of the new load module.

System Action: Processing continues.

Programmer Response: If you want the cross-zone module to be part of the load module, there are two possible responses:

- If the level of the cross-zone module in the current load module is correct, no action is needed.
- If the level of the cross-zone module is not correct, use the LINK command to link-edit the cross-zone module into the load module created by running the generated JCL.

GIM69079S *command* **PROCESSING STOPPED BECAUSE NO RELATED DISTRIBUTION ZONE IS DEFINED FOR TARGET ZONE** *zonename***.**

Explanation:

command	command name
zonename	zone name

SMP/E could not process the indicated command because no related distribution zone was defined in the TARGETZONE entry.

System Action: Command processing stops.

Programmer Response: Determine the name of the related distribution zone and use UCLIN or the SMP/E Administration dialogs to add the correct RELATED value to the TARGETZONE entry. Then rerun the job.

GIM69080S *command* **PROCESSING STOPPED BECAUSE DISTRIBUTION ZONE** *zonename* **WAS NOT AVAILABLE.**

Explanation:

command	command name
zonename	zone name

SMP/E could not process the indicated command because another user already has access to the CSI data set containing the required distribution zone.

System Action: Command processing stops.

Programmer Response: When the other user is done with the CSI data set containing the required distribution zone, rerun the job.

GIM69081S SMP/E WILL NOT LINK MOD *modname* **FROM ZONE** *zonename* **INTO LMOD** *loadmod* **BECAUSE THIS LMOD ALREADY CONTAINS A MOD WITH THE SAME NAME.**

Explanation:

modname	module name
zonename	cross-zone name
loadmod	load module name

You cannot link the indicated module into the load module because a module with the same name is already included in this LMOD.

System Action: LINK processing stops.

Programmer Response: Check that the correct MOD name was specified on the LINK command.

GIM69082W LMOD *loadmod* **IS NEEDED FOR LINK PROCESSING BUT IS NOT INSTALLED IN SYSLIB** *syslib***.**

Explanation:

loadmod	load module name
syslib	SYSLIB ddname

The LINK command will not link-edit any cross-zone modules into the indicated load module because the load module was not found in the indicated SYSLIB.

System Action: Command processing continues.

Programmer Response: Check that the LMOD entry specifies the correct SYSLIB.

GIM69083S NO LINKS WERE PROCESSED FOR THIS LINK COMMAND.

Explanation: The link-edit utility was not invoked to do any processing for this LINK command because of circumstances identified in previous messages.

System Action: LINK processing stops.

Programmer Response: Refer to previous messages.

GIM69084E THE *subtype* **SUBENTRY FOR** *enttype entname* **FROM ZONE** *zonename* **WAS NOT ADDED BECAUSE IT ALREADY EXISTS.**

Explanation:

subtype	XZMOD or XZLMOD
enttype	MOD or LMOD
entname	name of cross-zone module or load module
zonename	cross-zone name

A UCL statement tried to add the indicated subentry to an entry, but the subentry was already there. Either the entry or subentry is incorrect.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM69085E THE *subtype* **SUBENTRY FOR** *enttype entname* **FROM ZONE** *zonename* **WAS NOT DELETED BECAUSE IT DOES NOT EXIST.**

Explanation:

subtype	XZMOD or XZLMOD
enttype	MOD or LMOD
entname	name of cross-zone module or load module
zonename	cross-zone name

A UCL statement tried to delete the indicated subentry from an entry, but the subentry was not there.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Rerun the UCL statement without the specified subentry.

GIM69086I **MOD** *modname* **FROM SYSMOD** *sysmod* **WILL BE DELETED FROM LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
sysmod	ID of SYSMOD containing the module
loadmod	load module name
zonename	cross-zone name

The SYSMOD being processed deletes the indicated module, which contains XZLMOD subentries. As a result, SMP/E has selected the indicated module as a candidate for cross-zone updates during APPLY processing. If processing is successful, the module is deleted from the indicated load module. (During CHECK processing, this message indicates the cross-zone update processing that would occur.)

System Action: Command processing continues.

Programmer Response: Use the Element Summary report, the Cross-Zone Summary report, and the messages issued during the cross-zone phase of APPLY processing to determine if the cross-zone work identified by this message was done. If the cross-zone work was not done, you may have to complete the cross-zone work using a combination of the LINK command, the UCLIN command, or the linkage editor outside of SMP/E.

If the cross-zone work was not done but the Element Summary report indicates that MOD *modname* from SYSMOD *sysmod* was **not** successfully deleted, no intervention is required to complete the cross-zone work. The cross-zone work is done automatically once the module has been successfully deleted by a subsequent APPLY job.

If the cross-zone work was not done and the Element Summary report indicates that MOD *modname* from SYSMOD *sysmod* was successfully deleted, you may have to complete the cross-zone work identified by this message. Use messages issued during the cross-zone phase of the APPLY command to determine what action should be taken.

GIM69087I **MOD** *modname* **FROM SYSMOD** *sysmod* **WILL BE REPLACED IN LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
sysmod	ID of SYSMOD containing the module
loadmod	load module name
zonename	cross-zone name

The SYSMOD being processed replaces the indicated module, which contains XZLMOD subentries. As a result, SMP/E has selected the indicated module as a candidate for cross-zone updates during APPLY processing. If processing is successful, the module is replaced in the indicated load module. (During CHECK processing, this message indicates the cross-zone update processing that would occur.)

System Action: Command processing continues.

Programmer Response: Use the Element Summary report, the Cross-Zone Summary report, and the messages issued during the cross-zone phase of APPLY processing to determine if the cross-zone work identified by this message was done. If the cross-zone work was not done, you may have to complete the cross-zone work using a combination of the LINK command, the UCLIN command, or the linkage editor outside of SMP/E.

If the cross-zone work was not done but the Element Summary report indicates that MOD *modname* from SYSMOD *sysmod* was **not** successfully applied, no intervention is required to complete the cross-zone work. The cross-zone work is done automatically once the module has been successfully applied.

If the cross-zone work was not done and the Element Summary report indicates that MOD *modname* from SYSMOD *sysmod* was successfully applied, you may have to complete the cross-zone work identified by this message. Use messages issued during the cross-zone phase of the APPLY command to determine what action should be taken.

GIM69088I **ZAP FOR MOD** *modname* **FROM SYSMOD** *sysmod* **WILL BE INCLUDED IN LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
sysmod	ID of SYSMOD containing the module
loadmod	load module name
zonename	cross-zone name

The SYSMOD being processed updates the indicated module, which contains XZLMOD subentries. As a result, SMP/E has selected the indicated module as a candidate for cross-zone updates during APPLY or RESTORE processing. If processing is successful, the module is updated in the indicated load module. (During CHECK processing, this message indicates the cross-zone update processing that would occur.)

System Action: Command processing continues.

Programmer Response: Use the Element Summary report, the Cross-Zone Summary report, and the messages issued during the cross-zone phase of APPLY or RESTORE processing to determine if the cross-zone work identified by this message was done. If

the cross-zone work was not done, you may have to complete the cross-zone work using a combination of the LINK command, the UCLIN command, or the linkage editor outside of SMP/E.

If the cross-zone work was not done but the Element Summary report indicates that the ZAP for MOD *modname* from SYSMOD *sysmod* was **not** successfully applied or restored, no intervention is required to complete the cross-zone work. The cross-zone work is done automatically once the ZAP for the module has been successfully applied or restored.

If the cross-zone work was not done and the Element Summary report indicates that the ZAP for MOD *modname* from SYSMOD *sysmod* was successfully applied or restored, you may have to complete the cross-zone work identified by this message. Use messages issued during the cross-zone phase of command processing to determine what action should be taken.

GIM69089I MOD *modname* **FROM SYSMOD** *sysmod* **WILL NOT BE DELETED FROM LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
sysmod	ID of SYSMOD containing the module
loadmod	load module name
zonename	cross-zone name

The SYSMOD being processed deletes the indicated module, which contains XZLMOD subentries. However, because the module is no longer a candidate for APPLY processing, it is no longer a candidate for cross-zone update processing. As a result, it is not deleted from the indicated load module.

System Action: Command processing continues.

Programmer Response: No action is required. See the APPLY output for more information about the module status.

GIM69090I MOD *modname* **FROM SYSMOD** *sysmod* **WILL NOT BE REPLACED IN LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
sysmod	ID of SYSMOD containing the module
loadmod	load module name
zonename	cross-zone name

The SYSMOD being processed replaces the indicated module, which contains XZLMOD subentries. However, because the module is no longer a candidate for APPLY processing, it is no longer a candidate for cross-zone update processing. As a result, it is not replaced in the indicated load module.

System Action: Command processing continues.

Programmer Response: No action is required. See the APPLY output for more information about the module status.

GIM69091I ZAP FOR MOD *modname* **FROM SYSMOD** *sysmod* **WILL NOT BE INCLUDED IN LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
sysmod	ID of SYSMOD containing the module
loadmod	load module name
zonename	cross-zone name

The SYSMOD being processed updates the indicated module, which contains XZLMOD subentries. However, the module is no longer a candidate for cross-zone update processing because it is no longer a candidate for APPLY or RESTORE processing. As a result, it is not updated in the indicated load module.

System Action: Command processing continues.

Programmer Response: No action is required. See the APPLY or RESTORE output for more information about the module status.

GIM69092I CROSS-ZONE PROCESSING WILL NOT BE DONE BECAUSE THERE ARE NO CROSS-ZONE UPDATES REQUIRED.

Explanation: APPLY or RESTORE processing found no modules that needed to be deleted, updated, or replaced in load modules residing in zones other than the current set-to zone.

System Action: Command processing continues.

Programmer Response: No action is required.

GIM69093W MOD *modname* **IN ZONE** *zonename* **WILL NO LONGER BE INCLUDED IN LMOD** *loadmod***. A LINK COMMAND IS NEEDED TO INCLUDE MOD** *modname* **IN LMOD** *loadmod***.**

Explanation:

modname	module name
zonename	cross-zone name
loadmod	load module name

The indicated load module is being created by JCLIN. This load module previously existed and contained a module from a different zone. JCLIN processing does not include the cross-zone module when rebuilding the module.

System Action: Command processing continues.

Programmer Response: If it is necessary to include the indicated cross-zone module in this load module,

create and run an SMP/E LINK command to reestablish the linkage.

GIM69094I **MOD** *modname* **IN ZONE** *zonename* **WILL BE UPDATED TO SHOW THAT LMOD** *loadmod1* **WAS RENAMED TO** *loadmod2***.**

Explanation:

modname	module name
zonename	cross-zone name
loadmod1	old load module name
loadmod2	new load module name

The indicated load module has been renamed in the set-to zone. However, this load module contains cross-zone modules. The cross-zone subentries in the MOD entries in the cross-zone will be updated to show the new load module name.

System Action: Command processing continues.

Programmer Response: No action is required if APPLY or RESTORE processing is successful.

If an error occurs (an SMP/E failure or a system failure) and SMP/E does not automatically make this change, use the information in this message to update the XZLMOD information

---

GIM69095W *enttype1 entname1* **IN ZONE** *zonename1* **CONTAINS AN** *subtype* **SUBENTRY INDICATING A CROSS-ZONE RELATIONSHIP WITH** *enttype2 entname2* **IN ZONE** *zonename2***. THIS SUBENTRY WAS NOT MERGED INTO ZONE** *zonename3* **BECAUSE** *enttype1 entname1* **ONLY EXISTS AS A STUB ENTRY IN ZONE** *zonename1***.**

Explanation:

enttype1	MOD or LMOD
entname1	MOD name or LMOD name
zonename1	FROM zone name
subtype	XZLMOD or XZMOD
enttype2	LMOD or MOD
entname2	LMOD name or MOD name
zonename2	cross-zone name
zonename3	INTO zone name

During ZONEMERGE processing, SMP/E found a stub entry in the FROM zone.

- If a MOD entry is being processed, this stub indicates that module *entname1* used to be part of the load module named in the message. (SMP/E assumes that the module no longer exists because no base entry was found for the module in the FROM zone.)
- If an LMOD entry is being processed, this stub indicates that LMOD *entname1* used to contain the module named in the message. (SMP/E assumes that the load module no longer exists because no base entry was found for the LMOD in the FROM zone.)

The XZMOD or XZLMOD subentry from the stub entry is not merged into the INTO zone because it is not part of an existing base LMOD or MOD entry.

System Action: Processing continues.

Programmer Response: No action is required.

GIM69096W *enttype1 entname1* **IN ZONE** *zonename1* **CONTAINS AN** *subtype* **SUBENTRY INDICATING A CROSS-ZONE RELATIONSHIP WITH** *enttype2 entname2* **IN ZONE** *zonename2***. THIS SUBENTRY WAS MERGED INTO ZONE** *zonename3***.**

Explanation:

enttype1	MOD or LMOD
entname1	MOD name or LMOD name
zonename1	FROM zone name
subtype	XZLMOD or XZMOD
enttype2	LMOD or MOD
entname2	LMOD name or MOD name
zonename2	cross-zone name
zonename3	INTO zone name

This message serves as a reminder that corrective action may be needed to keep the cross-zone identified in the message and the INTO zone connected properly.

System Action: Processing continues.

Programmer Response: To keep the zones properly connected, you may need to update cross-zone information in either or both of the following:

- The INTO zone
- The cross-zone identified by the message

GIM69097E *enttype1 entname1* **IN ZONE** *zonename1* **CONTAINS AN** *subtype* **SUBENTRY INDICATING A CROSS-ZONE RELATIONSHIP WITH** *enttype2 entname2* **IN ZONE** *zonename2***. THIS SUBENTRY WAS NOT MERGED INTO ZONE** *zonename3* **BECAUSE THE CROSS-ZONE NAME MATCHES THE INTO ZONE NAME.**

Explanation:

enttype1	MOD or LMOD
entname1	MOD name or LMOD name
zonename1	FROM zone name
subtype	XZLMOD or XZMOD
enttype2	LMOD or MOD
entname2	LMOD name or MOD name
zonename2	cross-zone name
zonename3	INTO zone name

During ZONEMERGE processing, SMP/E found the indicated cross-zone subentry in the FROM zone. The cross-zone specified by this subentry to contain the LMOD or the module is the same as the INTO zone. This is not allowed; therefore, the cross-zone subentry was not merged into the INTO zone.

System Action: Processing continues. SMP/E attempts to merge the rest of the data in the zone.

Programmer Response: The wrong zone was specified for either the INTO or the FROM zone. Correct either or both of these values and rerun the job.

Note: You may have to replace the INTO zone with a backup copy from tape, because it is likely that the INTO zone has been updated by the ZONEMERGE command.

GIM69098E THE TARGET ZONE CONTAINS A TIEDTO SUBENTRY FOR ZONE *zonename***. THIS SUBENTRY WAS NOT ADDED TO THE INTO ZONE BECAUSE THE CROSS-ZONE NAME MATCHES THE INTO ZONE NAME (**_zonename_**).**

Explanation:

zonename	cross-zone name and INTO zone name

This message indicates that if the subentry had been added, the zone specified on the INTO operand would have been the same as one of its TIEDTO zones. This is not allowed.

System Action: Command processing continues. The subentry is not added to the INTO zone.

Programmer Response: Do the following:
1. Use the ZONEDELETE command to delete the INTO zone.
2. List all the remaining ZONEINDEX subentries (to avoid duplicates).
3. Define a new ZONEINDEX subentry for the INTO zone and use a new, unique zone name.

GIM69099S *enttype1 entname1* **CONTAINS AN** *subtype* **SUBENTRY INDICATING A CROSS-ZONE RELATIONSHIP WITH** *enttype2 entname2* **IN ZONE** *zonename***. ZONERENAME PROCESSING FAILED BECAUSE THE CROSS-ZONE NAME MATCHES THE NEW ZONE NAME (**_zonename_**).**

Explanation:

enttype1	MOD or LMOD
entname1	MOD name or LMOD name
subtype	XZLMOD or XZMOD
enttype2	LMOD or MOD
entname2	LMOD name or MOD name
zonename	cross-zone name and new zone name

This message indicates that if ZONERENAME processing had been done, the new zone name would have been the same as one of its cross-zones. This is not allowed.

System Action: Command processing fails. Processing continues with the next command.

Programmer Response: Do the following:
1. List all the remaining ZONEINDEX subentries (to avoid duplicates).
2. Choose a new, unique zone name for the new name of the zone.

GIM69100S THE ZONE VALUE IN AN *subtype* **SUBENTRY FOR** *enttype1 entname1* **WAS NOT CHANGED TO** *zonename***.** *enttype1 entname1* **ALREADY CONTAINS AN** *subtype* **SUBENTRY FOR** *enttype2 entname2* **IN ZONE** *zonename***.**

Explanation:

subtype	XZLMOD or XZMOD
enttype1	MOD or LMOD
entname1	MOD name or LMOD name
zonename	new ZONEVALUE name
enttype2	MOD or LMOD
entname2	MOD name or LMOD name

The ZONEEDIT command was issued to change the zone name associated with cross-zone subentries in the current target zone. However, the MOD or LMOD entry already contains both of the following:
- A cross-zone subentry with the old ZONEVALUE specified on the ZONEEDIT CHANGE statement
- A cross-zone subentry with the new ZONEVALUE specified on the ZONEEDIT CHANGE statement

Changing the subentry with the old ZONEVALUE creates a duplicate cross-zone subentry.

System Action: Command processing stops. The ZONEEDIT Summary report is issued to show which entries and subentries were updated.

Programmer Response: Specify that the correct ZONEVALUE was specified.
- If the correct ZONEVALUE was specified, use the SMP/E administration dialogs or the UCLIN command to correct any erroneous cross-zone information.
- If an incorrect ZONEVALUE was specified, use the SMP/E administration dialogs or the UCLIN command to undo changes in all entries that were updated because of the current ZONEEDIT CHANGE statement. (Use the ZONEEDIT Summary report to determine which entries were changed.)

  Then rerun the ZONEEDIT command and specify the correct ZONEVALUE on the ZONEEDIT CHANGE statement.

GIM69101S THE ZONE VALUE IN A TIEDTO SUBENTRY WAS NOT CHANGED TO *zonename1*. ZONE *zonename2* ALREADY CONTAINS A TIEDTO SUBENTRY FOR ZONE *zonename1*.

Explanation:

zonename1	new ZONEVALUE name
zonename2	name of current target zone

The ZONEEDIT command was issued to change the zone name associated with cross-zone subentries in the current target zone. However, the current target zone already contains both of the following:

- A TIEDTO subentry for the old ZONEVALUE specified on the ZONEEDIT CHANGE statement
- A TIEDTO subentry for the new ZONEVALUE specified on the ZONEEDIT CHANGE statement

Changing the subentry with the old ZONEVALUE would have created a duplicate TIEDTO subentry.

System Action: Command processing stops. The ZONEEDIT Summary report is issued to show which entries were updated.

Programmer Response: Specify that the correct ZONEVALUE was specified.

- If the correct ZONEVALUE was specified, use the SMP/E administration dialogs or the UCLIN command to correct any erroneous cross-zone information.
- If an incorrect ZONEVALUE was specified, use the SMP/E administration dialogs or the UCLIN command to undo changes in all entries that were updated because of the current ZONEEDIT CHANGE statement. (Use the ZONEEDIT Summary report to determine what entries were changed.)

Then rerun the ZONEEDIT command and specify the correct ZONEVALUE on the ZONEEDIT CHANGE statement.

GIM69102I ALLOCATION FAILED FOR *library* BECAUSE THERE IS NO DDDEF ENTRY FOR *ddname* IN *zonetype* ZONE *zonename*.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: If you want to avoid this problem in the future, add the required DDDEF to the zone named in the message before running this job. (See subsequent messages for details.)

Note: Adding a JCL DD statement for the specified library would **not** satisfy the allocation request. A DDDEF entry **must** be used.

GIM69102W ALLOCATION FAILED FOR *library* BECAUSE THERE IS NO DDDEF ENTRY FOR *ddname* IN *zonetype* ZONE *zonename*.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: If you need to correct this problem, add the required DDDEF to the zone named in the message. Then rerun the job if necessary. (See subsequent messages for details.)

Note: Adding a JCL DD statement for the specified library would **not** satisfy the allocation request. A DDDEF entry **must** be used.

GIM69102E ALLOCATION FAILED FOR *library* BECAUSE THERE IS NO DDDEF ENTRY FOR *ddname* IN *zonetype* ZONE *zonename*.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Add the required DDDEF to the zone named in the message. Then rerun the job if necessary. (See subsequent messages for details.)

Note: Adding a JCL DD statement for the specified library would **not** satisfy the allocation request. A DDDEF entry **must** be used.

GIM69102S **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DDDEF ENTRY FOR** *ddname* **IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Add the required DDDEF to the zone named in the message. Then rerun the job if necessary. (See subsequent messages for details.)

Note: Adding a JCL DD statement for the specified library would **not** satisfy the allocation request. A DDDEF entry **must** be used.

GIM69102T **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DDDEF ENTRY FOR** *ddname* **IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Add the required DDDEF to the zone named in the message. Then rerun the job if necessary. (See subsequent messages for details.)

Note: Adding a JCL DD statement for the specified library would **not** satisfy the allocation request. A DDDEF entry **must** be used.

GIM69103W **SMP/E WAS UNABLE TO UPDATE MOD ENTRY** *modname* **IN ZONE** *zonename* **TO SHOW THAT LMOD** *loadmod1* **WAS RENAMED TO** *loadmod2* **BECAUSE THE CSI DATA SET CONTAINING ZONE** *zonename* **IS NOT AVAILABLE.**

Explanation:

modname	module name
zonename	cross-zone name
loadmod1	old load module name
loadmod2	new load module name

The indicated MOD entry in the cross-zone was not updated with the new load module name, because the CSI data set containing the cross-zone is not available.

System Action: Command processing continues.

Programmer Response: When the CSI data set containing the cross-zone is available, use the UCLIN command to update the LMOD subentry in the indicated MOD entry.

GIM69104E **MOD** *modname* **FROM SYSMOD** *sysmod* **WILL NOT BE INCLUDED IN LMOD** *loadmod* **BECAUSE A MODULE WITH THE SAME NAME FROM ZONE** *zonename* **IS ALREADY INCLUDED IN THIS LMOD.**

Explanation:

modname	module name
sysmod	ID of the SYSMOD containing the module
loadmod	load module name
zonename	cross-zone name

During APPLY or RESTORE processing, the indicated module could not be included in the indicated LMOD, because that LMOD already contains a MOD of the same name from the cross-zone.

System Action: Processing stops for the SYSMOD containing the module. If the SYSMOD is a function, command processing also stops.

Programmer Response: If the SYSMOD must be installed, take one of the following actions:

- Remove the cross-zone relationship by using the UCLIN command to delete the cross-zone information. Then link the load module outside of SMP/E to delete the cross-zone module from the load module.
- If you created the SYSMOD in question, you can edit the SYSMOD and delete the indicated module. Then rerun the APPLY command for this SYSMOD.

GIM69105S **LINK PROCESSING FAILED BECAUSE ZONE** *zonename* **RELATED TO FROMZONE** *fromzone* **IS NOT A DLIB ZONE.**

Explanation:

zonename	related zone for the FROMZONE
fromzone	FROMZONE name

The zone related to the FROMZONE must be a DLIB zone.

System Action: Command processing stops.

Programmer Response: Use the SMP/E

Administration dialogs or UCLIN to specify the correct related zone in the FROMZONE zone definition entry.

GIM69106I THE TIEDTO SUBENTRY FOR ZONE *zonename1* WILL BE DELETED FROM ZONE *zonename2* BECAUSE THERE IS NO LONGER A CROSS-ZONE RELATIONSHIP BETWEEN THE TWO ZONES.

Explanation:

zonename1	cross-zone name
zonename2	set-to zone name

The TIEDTO relationship between the two zones no longer exists. The TIEDTO subentry for the cross-zone will be deleted from the target zone.

System Action: Command processing continues.

Programmer Response: No action is required.

GIM69107I DYNAMIC ALLOCATION WAS SUCCESSFUL FOR *library* USING DDDEF *ddname* FROM ZONE *zonename* - *allocparms*.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	DDDEF name
zonename	zone name
allocparms	allocation parameters

This message is issued during LINK processing or the cross-zone phase of APPLY or RESTORE processing. SMP/E successfully allocated the indicated data set.

System Action: Processing continues.

Programmer Response: No action is required.

GIM69108I DYNAMIC DEALLOCATION WAS SUCCESSFUL FOR *library*, WHICH WAS ALLOCATED USING DDDEF *ddname* FROM ZONE *zonename*.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	DDDEF name
zonename	zone name

This message is issued during LINK command processing or the cross-zone phase of APPLY or RESTORE processing. SMP/E successfully freed the indicated library.

System Action: Processing continues.

Programmer Response: No action is required.

GIM69109I ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT FOR *ddname* IN ZONE *zonename* FAILED.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	DDDEF name
zonename	zone name

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Look at an earlier message to determine why the previous allocation failed. Fix the problem, then issue another SET command and rerun the failing command if necessary. (See subsequent messages for details.)

GIM69109W ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT FOR *ddname* IN ZONE *zonename* FAILED.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	DDDEF name
zonename	zone name

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Look at an earlier message to determine why the previous allocation failed. Fix the problem, then issue another SET command and rerun the failing command if necessary. (See subsequent messages for details.)

GIM69109E ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT FOR *ddname* IN ZONE *zonename* FAILED.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	DDDEF name
zonename	zone name

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Look at an earlier message to determine why the previous allocation failed. Fix the problem, then issue another SET command and rerun the failing command if necessary. (See subsequent messages for details.)

GIM69109S ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT FOR *ddname* IN ZONE *zonename* FAILED.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	DDDEF name
zonename	zone name

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Look at an earlier message to determine why the previous allocation failed. Fix the problem, then issue another SET command and rerun the failing command if necessary. (See subsequent messages for details.)

GIM69109T ALLOCATION WAS NOT ATTEMPTED FOR *library* BECAUSE A PREVIOUS ALLOCATION ATTEMPT FOR *ddname* IN ZONE *zonename* FAILED.

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	DDDEF name
zonename	zone name

Although the indicated library was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt failed for that library.

During processing, SMP/E keeps track of all dynamic allocation from one SET command to the next. When an allocation fails, SMP/E will not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Look at an earlier message to determine why the previous allocation failed. Fix the problem, then issue another SET command and rerun the failing command if necessary. (See subsequent messages for details.)

GIM69110I *svc99fn* FAILED FOR *library* USING DDDEF *ddname* FROM ZONE *zonename* - *errortext*.

Explanation:

svc99fn	SVC99 function that failed
library	SMP/E-generated ddname of the library being processed
ddname	DDDEF name
zonename	zone name
errortext	the text of the error message from the message processing facility of SVC 99 showing the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error and rerun the job.

GIM69110W *svc99fn* **FAILED FOR** *library* **USING DDDEF** *ddname* **FROM ZONE** *zonename* **-** *errortext***.**

Explanation:

svc99fn	SVC99 function that failed
library	SMP/E-generated ddname of the library being processed
ddname	DDDEF name
zonename	zone name
errortext	the text of the error message from the message processing facility of SVC 99 showing the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error and rerun the job.

GIM69110E *svc99fn* **FAILED FOR** *library* **USING DDDEF** *ddname* **FROM ZONE** *zonename* **-** *errortext***.**

Explanation:

svc99fn	SVC99 function that failed
library	SMP/E-generated ddname of the library being processed
ddname	DDDEF name
zonename	zone name
errortext	the text of the error message from the message processing facility of SVC 99 showing the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error and rerun the job.

GIM69110S *svc99fn* **FAILED FOR** *library* **USING DDDEF** *ddname* **FROM ZONE** *zonename* **-** *errortext***.**

Explanation:

svc99fn	SVC99 function that failed
library	SMP/E-generated ddname of the library being processed
ddname	DDDEF name
zonename	zone name
errortext	the text of the error message from the message processing facility of SVC 99 showing the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error and rerun the job.

GIM69110T *svc99fn* **FAILED FOR** *library* **USING DDDEF** *ddname* **FROM ZONE** *zonename* **-** *errortext***.**

Explanation:

svc99fn	SVC99 function that failed
library	SMP/E-generated ddname of the library being processed
ddname	DDDEF name
zonename	zone name
errortext	the text of the error message from the message processing facility of SVC 99 showing the reason for the failure

An error occurred during processing of an SVC99 (the dynamic allocation SVC) function for the indicated library.

Note: This message is issued for each buffer of information returned from the message processing facility of SVC 99.

System Action: A subsequent message identifies the system action.

Programmer Response: Fix the error and rerun the job.

GIM69111I **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT FOR** *ddname* **IN THE JCL AND NO DDDEF ENTRY FOR** *ddname* **IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing JCL DD statement or missing DDDEF entry
zonetype	TARGET or DLIB

zonename zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM69111W **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT FOR** *ddname* **IN THE JCL AND NO DDDEF ENTRY FOR** *ddname* **IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing JCL DD statement or missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM69111E **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT FOR** *ddname* **IN THE JCL AND NO DDDEF ENTRY FOR** *ddname* **IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing JCL DD statement or missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM69111S **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT FOR** *ddname* **IN THE JCL AND NO DDDEF ENTRY FOR** *ddname* **IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing JCL DD statement or missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM69111T **ALLOCATION FAILED FOR** *library* **BECAUSE THERE IS NO DD STATEMENT FOR** *ddname* **IN THE JCL AND NO DDDEF ENTRY FOR** *ddname* **IN** *zonetype* **ZONE** *zonename***.**

Explanation:

library	SMP/E-generated ddname used to allocate the library
ddname	name of missing JCL DD statement or missing DDDEF entry
zonetype	TARGET or DLIB
zonename	zone name

SMP/E was trying to allocate the indicated data set but could not find the required DD statement or DDDEF entry.

System Action: A subsequent message identifies the system action.

Programmer Response: Do **one** of the following:
- Add the required DDDEF entry to the specified zone.
- Include the DD statement in the JCL used to run SMP/E.

Then rerun the job.

GIM69112W CROSS-ZONE UPDATES WILL NOT BE MADE TO ZONE *zonename* BECAUSE IT IS NOT DEFINED AS A TARGET ZONE.

Explanation:

zonename	cross-zone name

A cross-zone subentry of a MOD or LMOD points to the indicated cross-zone. However, cross-zone updates will not be done, because the cross-zone is not a target zone.

System Action: Processing continues with the next cross-zone.

Programmer Response: Do **one** of the following:

- If the MOD entry or LMOD entry points to an incorrect cross-zone, use UCLIN to correct the XZLMOD subentry in the MOD entry or the XZMOD subentry in the LMOD.
- If the MOD entry or LMOD entry points to the correct cross-zone, the cross-zone is defined incorrectly, because the cross-zone must be defined as a target zone. To solve this problem, the specified cross-zone must be restored with a backup copy from tape.

Fix the error. Use the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to determine what updates need to be done for the zone named in this message. Use a combination of the following to complete the unfinished cross-zone updates identified:

- The LINK command
- UCLIN updates
- The linkage editor (outside of SMP/E)

GIM69113W OPTIONS(*ddname*) WAS SPECIFIED IN THE PARAMETER LIST ON STEP *stepname* BUT THE *ddname* DD STATEMENT WAS NOT FOUND IN THE *stepname* STEP.

Explanation:

ddname	ddname of the inline OPTIONS file
stepname	step name

During JCLIN processing of the link-edit utility parameters, SMP/E found that the parameter OPTIONS(*ddname*) was specified, indicating that an inline file containing link-edit parameters was present in the job step. However, SMP/E did not find this inline file in the step.

System Action: JCLIN processing continues.

Programmer Response: LIST or QUERY the LMOD entry to ensure that the appropriate link-edit attributes have been assigned to the LMOD. If the attributes are incorrect, do the following based on the command that was executed when the command was issued:

- JCLIN command — correct the JCLIN and rerun the JCLIN command, or use UCLIN to change the attributes in the LMOD entry.
- ACCEPT command — correct the inline JCLIN and rerun the ACCEPT command, or use UCLIN to change the attributes in the LMOD entry.
- APPLY command — correct the inline JCLIN and rerun the APPLY command, or use UCLIN to change the attributes in the LMOD entry and relink the load module.

GIM69114W MOD *modname* IN ZONE *zonename1* WILL NOT BE REPLACED IN LMOD *lmodname* IN ZONE *zonename2* BECAUSE THE CSI DATA SET CONTAINING ZONE *zonename2* IS NOT AVAILABLE.

Explanation:

modname	MOD name
zonename1	set-to zone name
lmodname	LMOD name
zonename2	cross-zone name

SMP/E has replaced the indicated module, which is contained in a cross-zone load module. The module was not replaced in the cross-zone load module because the data set containing the cross-zone is unavailable at this time.

System Action: Command processing continues with the next cross-zone.

Programmer Response: Use information from this message to determine which cross-zone update was not done and create an SMP/E LINK command to update the indicated load module.

GIM69115W MOD *modname* IN ZONE *zonename1* WILL NOT BE UPDATED IN LMOD *lmodname* IN ZONE *zonename2* BECAUSE THE CSI DATA SET CONTAINING ZONE *zonename2* IS NOT AVAILABLE.

Explanation:

modname	MOD name
zonename1	set-to zone name
lmodname	LMOD name
zonename2	cross-zone name

SMP/E has updated the indicated module, which is contained in a cross-zone load module. The module was not updated in the cross-zone load module because the data set containing the cross-zone is unavailable at this time.

System Action: Command processing continues with the next cross-zone.

Programmer Response: Use information from this message to determine which cross-zone update was not

done, and create an SMP/E LINK command to update the indicated load module.

GIM69116W MOD *modname* **IN ZONE** *zonename1* **WILL NOT BE DELETED FROM LMOD** *lmodname* **IN ZONE** *zonename2* **BECAUSE THE CSI DATA SET CONTAINING ZONE** *zonename2* **IS NOT AVAILABLE.**

Explanation:

modname	MOD name
zonename1	set-to zone name
lmodname	LMOD name
zonename2	cross-zone name

SMP/E has deleted the indicated module, which is contained in a cross-zone load module. The module was not deleted in the cross-zone load module because the data set containing the cross-zone is unavailable at this time.

System Action: Command processing continues with the next cross-zone.

Programmer Response: Use information from this message to determine which cross-zone update was not done, and create a link-edit job outside of SMP/E to remove the module from the cross-zone load module.

GIM69117W MOD *modname* **IN ZONE** *zonename1* **WILL NOT BE REPLACED IN LMOD** *lmodname* **IN ZONE** *zonename2* **BECAUSE OF A PREVIOUS ERROR.**

Explanation:

modname	MOD name
zonename1	set-to zone name
lmodname	LMOD name
zonename2	cross-zone name

SMP/E processing has replaced the indicated module, which is contained in a cross-zone load module. The module was not replaced in the cross-zone load module because of a previous error in SMP/E processing.

System Action: Command processing continues.

Programmer Response: Use information from this message to determine which cross-zone update was not done, and create an SMP/E LINK command to update the indicated load module.

GIM69118W ZAP FOR MOD *modname* **IN ZONE** *zonename1* **WILL NOT BE INCLUDED IN LMOD** *lmodname* **IN ZONE** *zonename2* **BECAUSE OF A PREVIOUS ERROR.**

Explanation:

modname	MOD name
zonename1	set-to zone name
lmodname	LMOD name
zonename2	cross-zone name

SMP/E processing has updated the indicated module, which is contained in a cross-zone load module. The module was not updated in the cross-zone load module because of a previous error in SMP/E processing.

System Action: Command processing continues.

Programmer Response: Use information from this message to determine which cross-zone update was not done, and create an SMP/E LINK command to update the indicated load module.

GIM69119W MOD *modname* **IN ZONE** *zonename1* **WILL NOT BE DELETED FROM LMOD** *lmodname* **IN ZONE** *zonename2* **BECAUSE OF A PREVIOUS ERROR.**

Explanation:

modname	MOD name
zonename1	set-to zone name
lmodname	LMOD name
zonename2	cross-zone name

SMP/E processing has deleted the indicated module, which is contained in a cross-zone load module. The module was not deleted in the cross-zone load module because of a previous error in SMP/E processing.

System Action: Command processing continues.

Programmer Response: Use information from this message to determine which cross-zone update was not done, and create a link-edit job outside of SMP/E to remove the module from the cross-zone load module.

GIM69120W SMP/E WAS UNABLE TO UPDATE MOD ENTRY *entname* **IN ZONE** *zonename* **TO SHOW THAT LMOD** *loadmod1* **WAS RENAMED TO** *loadmod2* **BECAUSE OF A PREVIOUS ERROR.**

Explanation:

entname	MOD name
zonename	cross-zone name
loadmod1	LMOD name
loadmod2	new LMOD name

SMP/E processing has renamed the indicated load module, which contains cross-zone modules. The cross-zone module entries were not updated with the new load module name because of a previous error in SMP/E processing.

System Action: Command processing continues.

Programmer Response: Use information from this message to determine which cross-zone update was not done, and create an SMP/E UCLIN command to update the module entry in the cross-zone.

GIM69121W **MOD** *modname* **IN ZONE** *zonename1* **WILL NOT BE UPDATED IN LMOD** *lmodname* **IN ZONE** *zonename2* **BECAUSE CROSS-ZONE UPDATES ARE BEING DEFERRED.**

Explanation:

modname	MOD name
zonename1	set-to zone name
lmodname	LMOD name
zonename2	cross-zone name

The indicated module has been changed by SMP/E and is contained in a cross-zone load module. However, the cross-zone load module was not updated with the new version of this module because cross-zone updates are being deferred.

System Action: Command processing continues.

Programmer Response: Use the LINK command to update the cross-zone load module.

GIM69122S **AN I/O ERROR OCCURRED WHEN SMP/E WAS TRYING TO LOCATE LMOD** *loadmod* **FROM ZONE** *zonename* **IN SYSLIB** *dataset***.**

Explanation:

loadmod	LMOD name
zonename	cross-zone name
dataset	ddname of the SYSLIB data set

SMP/E was unable to complete cross-zone update processing for the indicated load module because of an I/O error.

System Action: Command processing fails.

Programmer Response: Fix the I/O error. Use the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to determine which cross-zone work was not done and which action you may have to take to complete the cross-zone work.

GIM69123W **MOD** *modname* **IN ZONE** *zonename1* **DOES NOT REFER TO LMOD** *lmodname1* **IN ZONE** *zonename2***. MOD** *modname* **IN ZONE** *zonename1* **WILL NOT BE UPDATED TO SHOW THAT LMOD** *lmodname1* **WAS RENAMED TO** *lmodname2***.**

Explanation:

modname	MOD name
zonename1	cross-zone name
lmodname1	LMOD name
zonename2	set-to zone name
lmodname2	new LMOD name

The indicated module has been renamed by SMP/E and contains cross-zone modules. However, the cross-zone MOD entry does not indicate that it is contained in the load module. The cross-zone MOD entry will not be updated to show the new load module name.

System Action: Command processing continues.

Programmer Response: If the cross-zone module should be updated, use the UCLIN command to update the load module name in the MOD entry.

GIM69124I *command* **PROCESSING FOR THE SET-TO ZONE IS COMPLETE. THE HIGHEST RETURN CODE WAS** *rtncode***.**

Explanation:

command	an SMP/E command (APPLY or RESTORE)
rtncode	the highest return code for processing being done for the set-to zone

This message indicates the highest return code for processing being done in the set-to zone.

System Action: The return code determines the system action.

Programmer Response: See the return codes for this command to determine the success or failure of the command.

GIM69125I *command* **CROSS-ZONE PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS** *rtncode***.**

Explanation:

command	an SMP/E command (APPLY or RESTORE)
rtncode	the highest return code for the cross-zone phase of the command being processed

This message indicates the highest return code for the cross-zone phase of APPLY or RESTORE processing.

System Action: The return code determines the system action.

Programmer Response: See the return codes for this command to determine the success or failure of the command.

GIM69126S **MOD** *modname* **IN ZONE** *zonename1* **IS NEEDED FOR LINK PROCESSING BUT IS NOT INSTALLED IN A SYSTEM LIBRARY.**

Explanation:

modname	MOD name
zonename1	cross-zone name

SMP/E used the DLIB entry corresponding to the MOD entry DISTLIB value for the indicated module in order to

find a copy of the module. However, the module was not found in any of the SYSLIBs specified in the DLIB entry.

System Action: The LINK command is terminated.

Programmer Response: Check that the MOD entry for the module specifies the correct DISTLIB value and that the DLIB entry specifies the correct SYSLIBs.

GIM69127W LMOD *loadmod* **IS NEEDED TO COMPLETE CROSS-ZONE PROCESSING FOR MOD** *modname* **BUT IT DOES NOT EXIST IN ZONE** *zonename***.**

Explanation:

loadmod	LMOD name
modname	MOD name
zonename	set-to zone name

From the set-to zone MOD entries, SMP/E determined that the indicated LMOD contained a usable copy of MOD *modname*. However, SMP/E could not find an LMOD entry for *loadmod*.

System Action: Command processing continues.

Programmer Response: Subsequent messages (GIM69059W or GIM69060W) identify the actions you must take.

GIM69128I MOD *modname* **WILL BE DELETED FROM LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
loadmod	load module name
zonename	cross-zone name

Because the indicated module contains XZLMOD subentries, SMP/E has selected it as a candidate for cross-zone updates during RESTORE processing. If processing is successful, the module is deleted from the indicated load module. (During CHECK processing, this message indicates the cross-zone update processing that would occur.)

System Action: Command processing continues.

Programmer Response: Use the Element Summary report and messages issued during the cross-zone phase of RESTORE processing to determine if the cross-zone work identified by this message was done.

- If the cross-zone work was not done, you may need to complete it by using a combination of the LINK command, the UCLIN command, or the linkage editor outside SMP/E.
- If the cross-zone work was not done but the Element Summary report indicates that the module was **not** successfully deleted, no intervention is required to complete the cross-zone work. This work is done automatically once the module has been successfully deleted by a subsequent RESTORE job.
- If the cross-zone work was not done and the Element Summary report indicates that the module was successfully deleted, you may need to complete the cross-zone work identified by this message. Use messages issued during the cross-zone phase of command processing to determine what action should be taken.

GIM69129I MOD *modname* **WILL BE REPLACED IN LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
loadmod	load module name
zonename	cross-zone name

Because the indicated module contains XZLMOD subentries, SMP/E has selected it as a candidate for cross-zone updates during RESTORE processing. If processing is successful, the module is replaced in the indicated load module. (During CHECK processing, this message indicates the cross-zone update processing that would occur.)

System Action: Command processing continues.

Programmer Response: Use the Element Summary report and messages issued during the cross-zone phase of RESTORE processing to determine if the cross-zone work identified by this message was done. If the cross-zone work was not done, you may need to complete it by using a combination of the LINK command, the UCLIN command, or the linkage editor outside SMP/E.

- If the cross-zone work was not done but the Element Summary report indicates that the module was **not** successfully restored, no intervention is required to complete the cross-zone work. This work is done automatically once the module has been successfully restored.
- If the cross-zone work was not done and the Element Summary report indicates that the module was successfully restored, you may need to complete the cross-zone work identified by this message. Use messages issued during the cross-zone phase of command processing to determine what action should be taken.

GIM69130I MOD *modname* **WILL NOT BE DELETED FROM LMOD** *loadmod* **IN ZONE** *zonename***.**

Explanation:

modname	module name
loadmod	load module name
zonename	cross-zone name

The indicated module contains XZLMOD subentries. However, because the module is no longer a candidate

for RESTORE processing, it is no longer a candidate for cross-zone update processing. As a result, it will not be deleted from the indicated load module.

System Action: Command processing continues.

Programmer Response: No action is required. See the RESTORE output for more information about the module status.

GIM69131I MOD *modname* WILL NOT BE REPLACED IN LMOD *loadmod* IN ZONE *zonename*.

Explanation:

modname	module name
loadmod	load module name
zonename	cross-zone name

The indicated module contains XZLMOD subentries. However, because the module is no longer a candidate for RESTORE processing, it is no longer a candidate for cross-zone update processing. As a result, it is not replaced in the indicated load module.

System Action: Command processing continues.

Programmer Response: No action is required. See the RESTORE output for more information about the module status.

GIM69132W CROSS-ZONE UPDATES WILL NOT BE MADE TO ZONE *zonename* BECAUSE CROSS-ZONE PROCESSING FOR THIS ZONE IS DEFERRED.

Explanation:

zonename	cross-zone name

Cross-zone processing was not done for the indicated cross-zone because its TARGETZONE entry has an XZLINK value of DEFERRED.

System Action: Processing continues with the next cross-zone.

Programmer Response: Use the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to complete the cross-zone work for the indicated zone.

GIM69133W CROSS-ZONE UPDATES WILL NOT BE MADE TO ZONE *zonename* BECAUSE THE CSI DATA SET CONTAINING ZONE *zonename* IS NOT AVAILABLE.

Explanation:

zonename	cross-zone name

Cross-zone processing was not done for the indicated cross-zone because SMP/E could not obtain the data set containing the cross-zone.

System Action: Processing continues with the next cross-zone.

Programmer Response: Use the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to complete the cross-zone work for the indicated zone.

GIM69134W CROSS-ZONE UPDATES WILL NOT BE MADE TO ZONE *zonename* BECAUSE AN ERROR WAS ENCOUNTERED WHILE PROCESSING THE ZONE.

Explanation:

zonename	cross-zone name

Cross-zone processing was not done for the indicated cross-zone because SMP/E found an error while processing the cross-zone.

System Action: Processing continues with the next cross-zone.

Programmer Response: Fix the error and use the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to complete the cross-zone work for the indicated zone.

GIM69135W CROSS-ZONE PROCESSING WAS NOT COMPLETED FOR ZONE *zonename* BECAUSE OF A PREVIOUS ERROR.

Explanation:

zonename	cross-zone name

An error prevented SMP/E from completing cross-zone processing for the indicated zone.

System Action: Processing continues with the next cross-zone.

Programmer Response: Fix the error and use the Cross-Zone Summary report and the MOVE/RENAME/DELETE report to complete the cross-zone work for the indicated zone.

GIM69136W ASSEMBLY *asmname* IN SYSMOD *sysmod* WAS NOT INSTALLED IN ANY TARGET LIBRARY. *asmname* MAY HAVE TO BE UPDATED IN LMOD *loadmod* IN ZONE *zonename*.

Explanation:

asmname	name of assembly
sysmod	ID of SYSMOD causing the assembly
loadmod	name of cross-zone LMOD
zonename	cross-zone name

The assembly for *asmname* was not done, because SMP/E determined that this module was no longer applicable to the set-to zone (no load module from the set-to zone contains the module). However, this module

was previously included in the identified cross-zone load module and may have to be updated in it.

System Action: Processing continues.

Programmer Response: The module needs to be updated in the cross-zone load module if the module provides additional function that is essential for the operation of the cross-zone load module.

Determine if this module should be updated in the cross-zone load module. If so, do the following:

- If this message was issued during RESTORE processing, use the LINK command to link the module into the cross-zone load module. LINK uses the module from the distribution library, because no load modules in the target zone contain a usable copy of the module.
- If this message was issued during APPLY processing, do **one** of the following:
  - Accept the SYSMOD causing the assembly. The source is assembled by SMP/E and stored as a single-CSECT load module in the distribution library.

    Then use the LINK command to link the module into the cross-zone load module. LINK uses the module from the distribution library, because no load modules in the target zone contain a usable copy of the module.
  - Do the assembly outside of SMP/E and link-edit the object created into the cross-zone load module.
    - If SYSMOD *sysmod* supplied a source replacement or update, follow these steps to do the assembly:
      1. Obtain an updated copy of the source supplied by SYSMOD *sysmod*.

         Use the Element Summary report to determine the SYSLIB of source *asmname* from SYSMOD *sysmod*. This SYSLIB contains an updated copy of the source.
      2. Assemble the source.
      3. Link-edit the module into the cross-zone load module.
    - If SYSMOD *sysmod* supplied a macro replacement or update causing an assembly for *asmname*, follow these steps to do the assembly:
      1. Obtain a copy of the source. Choose the first match from the ASSEM entry, SRC entry, or member in the DISTSRC data set (in this order).
      2. Assemble the source. Include the SMPMTS at the beginning of the SYSLIB concatenation if the changed macro does not exist in any target library.
      3. Link-edit the module into the cross-zone load module.

GIM69138W LMOD *loadmod* WAS NOT DELETED FROM LIBRARY *library* BY SYSMOD *sysmod* BECAUSE *loadmod* DOES NOT EXIST IN LIBRARY *library*.

Explanation:

loadmod	name of LMOD
library	ddname of library
sysmod	ID of the SYSMOD

There was an entry for the indicated LMOD in the zone, but the LMOD does not exist in the specified target library. It may have been moved, renamed or deleted outside of SMP/E, or its entry may have been created using UCLIN or JCLIN.

System Action: Processing continues.

Programmer Response: None.

GIM69139W LOAD MODULE *loadmod* IN SYSLIB *syslib* DOES NOT INCLUDE MODULE *modname* IN ZONE *zonename* BECAUSE *modname* IS A CROSS-ZONE MODULE.

Explanation:

loadmod	load module name
syslib	ddname of library
modname	module name
zonename	zone name

The indicated load module is supposed to contain the indicated cross-zone module. However, because the load module needed to be completely rebuilt, the cross-zone module was not included in the load module.

System Action: Processing continues.

Programmer Response: After the load module has been successfully link-edited, run the LINK command to include the indicated cross-zone module.

GIM69140I ALLOCATION WAS NOT ATTEMPTED FOR A CALLLIBS CONCATENATION BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation: Although a CALLLIBS concatenation was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt for the concatenation failed. During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E does not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Ensure that DDDEF entries for all the CALLLIBS needed are correctly specified in

the zone. Then issue another SET command and rerun the failing command.

GIM69140W ALLOCATION WAS NOT ATTEMPTED FOR A CALLLIBS CONCATENATION BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation: Although a CALLLIBS concatenation was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt for the concatenation failed. During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E does not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Ensure that DDDEF entries for all the CALLLIBS needed are correctly specified in the zone. Then issue another SET command and rerun the failing command.

GIM69140E ALLOCATION WAS NOT ATTEMPTED FOR A CALLLIBS CONCATENATION BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation: Although a CALLLIBS concatenation was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt for the concatenation failed. During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E does not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Ensure that DDDEF entries for all the CALLLIBS needed are correctly specified in the zone. Then issue another SET command and rerun the failing command.

GIM69140S ALLOCATION WAS NOT ATTEMPTED FOR A CALLLIBS CONCATENATION BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation: Although a CALLLIBS concatenation was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt for the concatenation failed. During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E does not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Ensure that DDDEF entries for all the CALLLIBS needed are correctly specified in the zone. Then issue another SET command and rerun the failing command.

GIM69140T ALLOCATION WAS NOT ATTEMPTED FOR A CALLLIBS CONCATENATION BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.

Explanation: Although a CALLLIBS concatenation was required for command processing, SMP/E did not try to dynamically allocate it because a previous allocation attempt for the concatenation failed. During processing, SMP/E keeps track of all dynamic allocation attempts from one SET command to the next. When an allocation fails, SMP/E does not try that allocation again. The next SET command, however, resets the allocation history and lets SMP/E try the allocation again.

System Action: A subsequent message identifies the system action.

Programmer Response: Ensure that DDDEF entries for all the CALLLIBS needed are correctly specified in the zone. Then issue another SET command and rerun the failing command.

GIM69141W THE LINK-EDIT OF LMOD *loadmod* FROM ZONE *zonename* WAS NOT ATTEMPTED BECAUSE LMOD *loadmod* IS NOT IN THE SMPLTS.

Explanation:

loadmod	load module name
zonename	zone name

The indicated LMOD does not exist in the SMPLTS data set. SMP/E needs the SMPLTS version of the LMOD to do the link edit of this LMOD into its system library.

System Action: The system action depends on the processing being done when the message is issued:

- If it is issued during APPLY or RESTORE cross-zone processing, the cross-zone update processing for the LMOD is not done.
- If it is issued by LINK command processing, LINK processing stops for that LMOD.

Programmer Response: Correct the error identified in the associated SMP/E link-edit (GIM239*xx*) message.

GIM69141E THE LINK-EDIT OF LMOD *loadmod* FROM ZONE *zonename* WAS NOT ATTEMPTED BECAUSE LMOD *loadmod* IS NOT IN THE SMPLTS.

Explanation:

loadmod	load module name

zonename zone name

The indicated LMOD does not exist in the SMPLTS data set. SMP/E needs the SMPLTS version of the LMOD to do the link edit of this LMOD into its system library.

System Action: SYSMOD processing stops. This is indicated in a subsequent message.

Programmer Response: Correct the error identified in the associated SMP/E link-edit (GIM239*xx*) message.

GIM69142I OPEN PROCESSING FOR ZONE *zonename* IN DATA SET *dataset* FAILED BECAUSE LSR WAS SPECIFIED AND THE LARGEST BUFFER SIZE IN THE CURRENT POOL WAS EXCEEDED.

Explanation:
zonename name of the zone to be opened
dataset ddname of the CSI data set containing the zone to be opened

SMP/E was not able to open the indicated zone, because the CSI data set containing the zone takes advantage of local shared resources (LSR) and the control interval size of the data set exceeds the largest buffer size in the current pool.

System Action: SMP/E issues another OPEN for the CSI data set, but without specifying LSR.

Programmer Response: No action required.

GIM69143I SMP/E SUCCESSFULLY OPENED DATA SET *zonename* WITHOUT LSR TO COMPLETE COMMAND PROCESSING.

Explanation:
zonename name of the zone to be opened

SMP/E was previously not able to open the indicated CSI data set, because that data set takes advantage of local shared resources (LSR) and the control interval size of the data set exceeds the largest buffer size in the current pool.

SMP/E successfully issued another OPEN for the CSI data set, but without specifying LSR.

System Action: Processing continues.

Programmer Response: No action required.

GIM69149W CALLLIBS COULD NOT BE ALLOCATED FOR LMOD *loadmod* IN ZONE *zonename*.

Explanation:
loadmod load module name
zonename zone name

During processing for the LINK command, the CALLLIBS for the indicated LMOD could not be allocated. A previous message describes the specific allocation error.

System Action: SMP/E continues looking for additional errors.

Programmer Response: Correct the error identified in the earlier message.

GIM69149E CALLLIBS COULD NOT BE ALLOCATED FOR LMOD *loadmod* IN ZONE *zonename*.

Explanation:
loadmod load module name
zonename zone name

The CALLLIBS for the indicated LMOD could not be allocated. A previous message describes the specific allocation error.

System Action: SYSMOD processing stops.

Programmer Response: Correct the error identified in the earlier message.

GIM69150W ZAP PROCESSING WAS NOT DONE FOR LMOD *loadmod* IN THE SMPLTS BECAUSE THE LMOD DOES NOT EXIST IN THE SMPLTS.

Explanation:
loadmod load module name

Because the load module being zapped has CALLLIBS, the base version of the load module in the SMPLTS needed to be zapped as well. However, the load module does not exist in the SMPLTS.

System Action: Processing continues.

Programmer Response: None.

GIM69151I DYNAMIC DEALLOCATION WAS SUCCESSFUL FOR THE *library* CONCATENATION.

Explanation:
library ddname of concatenated library

SMP/E successfully freed the indicated library.

System Action: Processing continues.

Programmer Response: None.

GIM69152I PROGRAM *progname* MAY BE REQUIRED FOR SMP/E PROCESSING BUT IS NOT AVAILABLE.

Explanation:
progname utility program name

SMP/E could not find the indicated program, which it may need for processing.

System Action: Processing continues.

Programmer Response: None.

GIM69153E PATH MUST END WITH A SLASH ("/").

Explanation: The PATH name specified in the UCL statement did not end with the required delimiter, a slash ("/").

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM69153S THE PATH FROM_VALUE MUST END WITH A SLASH (″/″).

Explanation: The path name specified as the from_value on the CHANGE statement did not end with the required delimiter, a slash (″/″). The path name of the from_value must end in a slash when a full path name is specified.

System Action: For the CHANGE statement of the ZONEEDIT command, this CHANGE statement and remaining CHANGE statements, up to the ENDZEDIT command, are not processed.

Programmer Response: Fix the CHANGE statement and rerun the job.

GIM69154E PATH IS NOT ALLOWED WITH ANY OTHER DDDEF SUBENTRIES.

Explanation: One of the following errors occurred:

- Other operands were specified on a UCL statement to update an existing DDDEF entry containing the PATH subentry.
- The PATH operand was specified on a UCL statement to update an existing DDDEF entry containing other subentries.
- The PATH operand was specified along with other DDDEF subentry operands on the current UCL statement.

None of these combinations are allowed.

System Action: SMP/E does not process any of the changes for this UCL statement.

Programmer Response: Fix the UCL statement and rerun the job.

GIM69155E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod*. *sysmod* **INCLUDES** *elmtype elmname* **BUT THE HFSCOPY UTILITY** *progname* **WAS NOT AVAILABLE.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID
elmtype	element type
elmname	element name
progname	utility program name

SMP/E could not find the HFSCOPY utility that is needed to process the indicated element.

System Action: SYSMOD processing stops.

Programmer Response: Do **one** of the following:

- If the program is not available, add it to the JOBLIB, STEPLIB, LINKLIB, or LPALIB data set. Then rerun the job.
- If you specified an incorrect utility name, correct it and rerun the job.

GIM69156E *command* **PROCESSING FAILED FOR SYSMOD** *sysmod*. *sysmod* **INCLUDES** *elmtype elmname* **BUT THE HFSCOPY UTILITY** *progname* **WAS NOT DEFINED IN GIMUTTBL.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID
elmtype	element type
elmname	element name
progname	utility program name

System Action: SYSMOD processing stops.

Programmer Response: If you want to define the program for the indicated utility, add it to GIMUTTBL, then rerun the job. For more information, see the GIMUTTBL chapter of *SMP/E Reference*.

GIM69157T THE *service* **CALLABLE SERVICE IS NEEDED TO PROCESS FILE** *filename* **IN LIBRARY** *library* **BUT IS NOT AVAILABLE.**

Explanation:

service	callable service name
filename	file name
library	ddname of library being processed

SMP/E could not find the indicated callable service that is needed to process the file.

System Action: Command processing stops.

Programmer Response: Ensure that OS/390 or z/OS UNIX System Services is installed and available.

GIM69158I *progname* **WAS NOT ABLE TO SET THE EFFECTIVE UID TO 0. THIS IS NORMAL IF THE CURRENT UID IS NOT AUTHORIZED TO THE BPX.SUPERUSER FACILITY CLASS PROFILE. CALLABLE SERVICE BPX1SEU GAVE A RETURN CODE OF '***rtncode***'X WITH A REASON CODE OF '***rsncode***'X.**

Explanation:

progname	program name
rtncode	return code from BPX1SEU callable service in hexadecimal
rsncode	reason code from BPX1SEU callable service in hexadecimal

System Action: Processing continues.

Programmer Response: If the user of the running program is intended to be authorized to set the effective UID to 0, refer to *z/OS UNIX System Services Programming: Assembler Callable Services Reference* to determine the meaning of the return code and reason code given by the BPX1SEU callable service and correct the indicated problem. Otherwise, no action is required.

GIM69161T **FILE** *filename* **WAS NOT REMOVED FROM THE** *library* **LIBRARY. THE RETURN CODE FROM THE** *service* **SERVICE WAS** *rtncode* **AND THE REASON CODE WAS** *rsncode***.**

Explanation:

filename	file name
library	ddname of library being processed
service	callable service name
rtncode	return code
rsncode	reason code

SMP/E attempted to remove the file from the indicated library, but the callable service was unsuccessful.

System Action: Command processing stops.

Programmer Response: Look in *z/OS UNIX System Services Programming: Assembler Callable Services Reference* to determine the cause of the error. Correct the error, and rerun the job.

GIM69162I **FILE** *filename1* **WAS RENAMED TO** *filename2* **IN THE** *library* **LIBRARY.**

Explanation:

filename1	file name
filename2	new file name
library	ddname of library being processed

SMP/E renamed the specified file in the indicated library.

System Action: Processing continues.

Programmer Response: None.

GIM69163T **FILE** *filename1* **COULD NOT BE RENAMED TO** *filename2* **IN THE** *library* **LIBRARY. THE RETURN CODE FROM THE** *service* **SERVICE WAS** *rtncode* **AND THE REASON CODE WAS** *rsncode***.**

Explanation:

filename1	file name
filename2	new file name
library	ddname of library being processed
service	callable service name
rtncode	return code
rsncode	reason code

SMP/E attempted to rename the file in the indicated library, but the callable service was unsuccessful.

System Action: Command processing stops.

Programmer Response: Look in *z/OS UNIX System Services Programming: Assembler Callable Services Reference* to determine the cause of the error. Correct the error, and rerun the job.

GIM69164E **THE** *subtype* **VALUE ON THE** *stattype* **STATEMENT FOR** *enttype entname* **EXCEEDS THE** *maxlen* **CHARACTER MAXIMUM.**

Explanation:

subtype	subentry type
stattype	statement type (UCLIN or MCS)
enttype	entry type
entname	entry name
maxlen	maximum length allowed for the specified subentry type

The value specified for the indicated operand is longer than the maximum length allowed for the subentry or the specified value is missing a parenthesis.

System Action: Processing depends on type of statement:

- For a UCLIN statement, SMP/E does not process the UCLIN update for the entry.
- For an MCS statement, command processing stops for the SYSMOD containing the MCS statement.

Programmer Response: Specify a shorter name for the subentry or supply the missing parenthesis.

GIM69164S **THE PATH VALUE ON THE CHANGE STATEMENT EXCEEDS THE 255-CHARACTER MAXIMUM.**

Explanation: The value specified for the PATH operand is longer than the maximum length allowed for the subentry.

System Action: This CHANGE statement of the

ZONEEDIT statement and remaining CHANGE statements, up to the ENDZEDIT command, are not processed.

Programmer Response: Specify a shorter path name.

GIM69165I SMP/E COULD NOT OBTAIN ENOUGH STORAGE FOR ENHANCED PERFORMANCE PROCESSING. SMP/E WILL CONTINUE PROCESSING IN A DEGRADED MODE.

Explanation: SMP/E could not allocate enough storage for the internal tables that it uses to improve performance.

System Action: Processing continues.

Programmer Response: If you want better performance from SMP/E, you should increase the storage available above the 16M line.

Note: To enable the SMP/E job step to get the maximum space above the 16M line, you can specify REGION=0M. The JOB statement (or the EXEC statement, or both) can also include the REGION parameter to set the size of the region in which SMP/E runs. For details, see *z/OS MVS JCL User's Guide* or *z/OS MVS JCL Reference*.

GIM69166W THE SYSLIB ALLOCATION IN STEP *stepname* **OF JOB** *jobname* **SPECIFIED DUPLICATE DDNAME** *duplname***. THE DUPLICATE DDNAME IS IGNORED.**

Explanation:

stepname	step name or NONAME
jobname	job name or NONAME
duplname	low-level data set qualifier for the duplicate SYSLIB ddname

During JCLIN processing, SMP/E found a duplicate low-level data set qualifier in a SYSLIB concatenation. This low-level qualifier is used as the ddname for an LMOD's CALLLIBS subentry.

System Action: SMP/E saves only the first occurrence of the duplicate ddname in the LMOD's CALLLIBS concatenation.

Programmer Response: Verify the SYSLIB allocation, then correct the error and rerun the job if necessary.

GIM69167I HFSCOPY PROCESSING TO THE *library* **LIBRARY WAS SUCCESSFUL FOR** *enttype entname* **IN SYSMOD** *sysmod***. THE RETURN CODE WAS** *rtncode***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqnum***.**

Explanation:

library	ddname of library where element was installed
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
rtncode	return code
yy	year
ddd	Julian date
hh	military hour
mm	minutes
ss	seconds
seqnum	utility sequence number

The indicated element was successfully installed.

System Action: Processing continues.

Programmer Response: None.

GIM69167W HFSCOPY PROCESSING TO THE *library* **LIBRARY WAS SUCCESSFUL FOR** *enttype entname* **IN SYSMOD** *sysmod***. THE RETURN CODE WAS** *rtncode***. DATE** *yy.ddd* **- TIME** *hh:mm:ss* **- SEQUENCE NUMBER** *seqnum***.**

Explanation:

library	ddname of library where element was installed
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
rtncode	return code
yy	year
ddd	Julian date
hh	military hour
mm	minutes
ss	seconds
seqnum	utility sequence number

Because the return code from HFS copy utility processing did not exceed the maximum acceptable return code, processing was considered successful for the indicated element, which was part of SYSMOD *sysmod*. However, because the return code from HFS copy processing was not zero, processing may not have produced results that you consider acceptable. The utility sequence number matches the sequence number on the utility's SYSPRINT output.

System Action: Processing continues.

Programmer Response: Check the HFS copy SYSPRINT output to determine whether the results are what you consider acceptable.

- If the results are acceptable, no further action is necessary.
- If the results are not acceptable, you must take post-installation steps to achieve acceptable results.

GIM69168E HFSCOPY PROCESSING TO THE *library* LIBRARY FAILED FOR *enttype entname* IN SYSMOD *sysmod*. THE RETURN CODE (*rtncode*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *seqnum*.

Explanation:

library	ddname of library where element was to be installed
enttype	entry type
entname	entry name
sysmod	SYSMOD ID
rtncode	return code
yy	year
ddd	Julian date
hh	military hour
mm	minutes
ss	seconds
seqnum	utility sequence number

HFSCOPY processing failed for the indicated element.

System Action: SYSMOD processing stops because the return code was not 0, and **one** of the following is true:

- The return code is greater than the maximum return code specified in the UTILITY entry.
- The return code is greater than the default return code.

Programmer Response: Check the utility output in the print output data set to find the cause of the error. The sequence number may be used as an index into the print output data set. This number is incremented for each utility call in an SMP/E run. Fix the error and rerun the job.

GIM69169W THE GLOBAL ZONE IS NOT ALLOWED FOR THE ZONES OPERAND OF THE REPORT CALLLIBS COMMAND.

Explanation: The global zone is specified on the ZONES operand of the REPORT CALLLIBS command, or the global zone is contained in a ZONESET that is specified on the REPORT CALLLIBS command.

System Action: SMP/E ignores the global zone and continues processing.

Programmer Response: None.

GIM69170I ALTHOUGH THE *dataset* DATA SET WAS NOT ALLOCATED, PROCESSING WILL CONTINUE FOR THE *command* COMMAND. HOWEVER, NO OUTPUT WILL BE WRITTEN TO THE *dataset* DATA SET.

Explanation:

dataset	ddname of the data set for which allocation failed
command	command being processed at the time of failure

The data set could not be allocated because there was no DDDEF entry or JCL DD statement defining it.

System Action: Processing continues, but output is not generated to the *dataset* data set.

Programmer Response: If you want output to be generated to the *dataset* data set, define a DD statement or DDDEF entry for the *dataset* data set, and rerun the job.

GIM69171E AN ERROR OCCURRED DURING REPORT CALLLIBS GENERATION OF STEP *stepname* IN JOB *jobname*. THE LINK-EDIT PARAMETER STRING WAS TRUNCATED BECAUSE IT EXCEEDS THE 100-CHARACTER MAXIMUM.

Explanation:

stepname	name of the generated step with the erroneous parameter string
jobname	name of the generated job with the erroneous parameter string

Because the binder is not the link-edit utility being used, the length of the link-edit parameter string cannot exceed 100 characters. However, the parameters specified in the LMOD entry and linkage editor UTILITY entry, combined with CALL, exceed the 100-character maximum.

System Action: The parameter string is truncated after 100 characters.

Programmer Response: Edit the link-edit parameter string for generated step *stepname* in job *jobname* before executing the JCL to ensure that it is correct and valid.

Link-edit parameter information from the LMOD entry and the linkage editor UTILITY entry, as well as the link-edit CALL parameter, should be considered when editing the link-edit parameter string.

GIM69172E THE JOBCARD OPERAND IS MISSING FROM THE *command* COMMAND. A COMMENT WILL TAKE THE PLACE OF THE JOB STATEMENT IN THE GENERATED JCL.

Explanation:

command	command being processed at the time of failure and from which the JOBCARD operand is missing

JOBCARD is a required operand for this command when NOPUNCH is not specified and when one of the zones being processed is a target type zone.

System Action: SMP/E generates a JCL comment in

place of the jobcard in the generated JCL. Command processing continues, but the output job does not run unless you fix the error.

Programmer Response: Do **one** of the following:
- Add the missing operand and rerun the command.
- Edit the JCL output to add the required JCL statement in place of the comment.

GIM69174E THE SPECIFIED CHANGES WERE NOT PROCESSED. BECAUSE THE ENTRY CONTAINS A DATASET SUBENTRY, IT MUST ALSO CONTAIN A VALUE FOR THE INITIAL DISPOSITION.

Explanation: When a DDDEF entry contains a DATASET subentry, it must also specify the initial disposition of the data set (NEW, OLD, MOD, or SHR). When processing the specified changes, SMP/E noticed that the DDDEF entry contains a DATASET subentry but no initial disposition. This discrepancy may have been caused either by the UCL statement being processed or by changes made with an earlier level of SMP/E.

System Action: SMP/E ignores the UCL statement and does not change the entry. Processing continues with the next UCL statement.

Programmer Response: Correct the UCL statement so that if the updated version of the DDDEF entry will contain a DATASET subentry, it also contains a value for the initial disposition. Then rerun the statement.

GIM69175W THE BASE VERSION OF LOAD MODULE *loadmod* WAS DELETED FROM THE SMPLTS LIBRARY. THE EXECUTABLE VERSION OF THAT LOAD MODULE IN ITS TARGET LIBRARIES MAY INCLUDE MODULES FROM THE CALLLIBS LIBRARIES.

Explanation:

loadmod	load module name

SMP/E deleted the base version of the indicated load module from the SMPLTS, because it was built during the installation of a SYSMOD that is now being restored. The executable version of this load module in its target libraries may contain modules that were included by the automatic library call mechanism when the load module was link-edited during APPLY processing of the SYSMOD.

System Action: Processing continues.

Programmer Response: No action is required. Modules included by the automatic library call mechanism may continue to function in the load module if there are still external references to them. If these modules are no longer referenced in the load module, they become inactive code in the load module.

GIM69176T FILE *filename* COULD NOT BE ACCESSED IN THE *library* LIBRARY. USING ACCESS MODE *accmode* THE RETURN CODE FROM THE *service* SERVICE WAS *rtncode* AND THE REASON CODE WAS *rsncode*.

Explanation:

filename	file name
library	ddname of library being processed
accmode	access mode passed to the service
service	callable service name
rtncode	return code
rsncode	reason code

SMP/E attempted to test the accessibility of a file using the specified access mode in the indicated library, but the callable service was unsuccessful.

System Action: Command processing stops.

Programmer Response: Make sure you have the appropriate permission to access the file and directory.

Look in *z/OS UNIX System Services Programming: Assembler Callable Services Reference* to determine the cause of the error. Correct the error, and rerun the job.

GIM69177I LIBRARY *library* WAS NOT COMPRESSED BECAUSE IT IS A PATH.

Explanation:

library	ddname of library being processed

The indicated library was specified on the COMPRESS operand, but SMP/E does not compress any libraries that are paths in a hierarchical file system.

System Action: Processing continues.

Programmer Response: Remove the indicated library from the COMPRESS operand before rerunning the command.

GIM69178S *command* PROCESSING FAILED FOR SYSMOD *sysmod* BECAUSE THE *dataset* DATA SET NAME THAT WAS CONSTRUCTED EXCEEDS THE 44-CHARACTER MAXIMUM.

Explanation:

command	SMP/E command
sysmod	SYSMOD ID
dataset	ddname of data set

The operands specified for the indicated data set name caused the constructed data set name to exceed 44 characters. During RECEIVE processing, the RFPREFIX and RFDSNPFX operands may be combined to construct the RELFILE data set name, and sometimes the resulting data set name may be too long.

System Action: Processing stops.

Programmer Response: If the RFPREFIX operand is specified on the RECEIVE command, verify that the value is correct, and rerun the RECEIVE command.

GIM69179S *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE THE DATA SET NAME SPECIFIED FOR THE** *dataset1* **DATA SET IS THE SAME AS THE DATA SET NAME SPECIFIED FOR THE** *dataset2* **DATA SET. THIS IS NOT ALLOWED.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID
dataset1	description of the existing data set
dataset2	ddname of data set to be allocated

The data set name constructed for data set allocation is already specified for an existing data set.

System Action: Command processing stops.

Programmer Response: If this message was issued during RECEIVE processing and a DSPREFIX was specified for the SMPTLIB data set, one of the following is true:

- the DSPREFIX is the same as the RFPREFIX value specified on the RECEIVE command, or
- the FROMDS data set name matches the SMPTLIB data set name.

In either case, change the DSPREFIX value and rerun the RECEIVE command.

GIM69180S *command* **PROCESSING FAILED FOR SYSMOD** *sysmod* **BECAUSE SMP/E RECEIVED A RETURN CODE OF** *rtncode* **FROM THE** *macro* **MACRO.**

Explanation:

command	SMP/E command
sysmod	SYSMOD ID
rtncode	return code
macro	macro that failed

A return code other than zero was received from the indicated macro. If the indicated macro is the DEVTYPE macro, the return codes indicate the following:

04 **One** of the following occurred:
- The ddname specified could not be found.
- A TIOT address is invalid.
- The UCBLIST parameters are invalid.

08 The device information is unknown or not supported by the DEVTYPE macro.

System Action: Command processing stops.

Programmer Response: Report the problem to the IBM Support Center.

GIM69183I *svc99fun* **WAS SUCCESSFUL FOR** *dataset***.**

Explanation:

svc99fun	SVC 99 function that was successful
dataset	ddname of data set

SMP/E successfully performed the indicated function.

System Action: Processing continues.

Programmer Response: None.

GIM69184E *elmtype elmname* **COULD NOT BE DELETED FROM THE** *library* **LIBRARY BY SYSMOD** *sysmod* **BECAUSE** *library* **IS NOT ALLOCATED TO A PATH.**

Explanation:

elmtype	element type
elmname	element name
library	ddname of library being processed
sysmod	SYSMOD ID

SMP/E attempted to delete the specified element, but the library is allocated to a data set. The specified element is not allowed to reside in a data set. Therefore, the library must be allocated to a path in the hierarchical file system.

System Action: SYSMOD processing stops.

Programmer Response: Ensure that the DDDEF entry or DD statement for the specified library identifies a path in the hierarchical file system.

GIM69185E **AN INVALID ENTRY HAD BEEN ENCOUNTERED IN THE SMPSCDS DATA SET FOR SYSMOD** *sysmod***.**

Explanation:

sysmod	SYSMOD ID

Command processing was not done for the indicated SYSMOD because SMP/E found an invalid entry in the SMPSCDS data set. The problem in the entry caused an I/O error, which led to an abend.

System Action: Processing stops for the indicated SYSMOD. For the APPLY command, however, SMP/E fixes the entry so that subsequent APPLY commands do not encounter the same problem.

Programmer Response: For APPLY processing, rerun the APPLY command.

For LIST BACKUP processing, there is no way to fix the entry. Use the UCLIN command to delete the BACKUP entry from the SMPSCDS data set.

GIM69186E *elmtype elmname* **CANNOT BE PROCESSED IN LIBRARY** *library* **FOR SYSMOD** *sysmod* **BECAUSE THE EXECUTION PARAMETER STRING NEEDED TO INVOKE THE HFSCOPY UTILITY EXCEEDS THE MAXIMUM LENGTH OF 65,535 BYTES.**

Explanation:

elmtype	element type
elmname	element name
library	library ddname
sysmod	SYSMOD ID

SMP/E attempted to process the indicated element, but when SMP/E built the execution parameter string for invoking the hierarchical file system copy utility, the 65,535 byte limit for the string was exceeded.

System Action: Processing stops for the indicated SYSMOD.

Programmer Response: Contact the IBM Support Center.

GIM69199E **A LINK VALUE ON THE MCS FOR** *elmtype elmname* **IN SYSMOD** *sysmod* **EXCEEDS THE 64 CHARACTER MAXIMUM. z/OS** *level* **SMP/E (OR HIGHER) SUPPORTS LINK VALUES UP TO 1023 CHARACTERS LONG.**

Explanation:

elmtype	element type
elmname	element name
sysmod	SYSMOD ID
level	minimum version and release (*vv.rr*) of z/OS SMP/E required to process this MCS.

Note: SMP/E V3R1 is identical to z/OS V1R2 SMP/E.

The LINK value of the indicated modification control statement is longer than 64 characters. It cannot be used by the current level of SMP/E. However, the specified level of SMP/E can handle LINK values of up to 1023 characters.

System Action: Processing for the SYSMOD stops.

Programmer Response: Use the specified level of SMP/E, or higher, to process the SYSMOD.

GIM69200E **RETRANSFORMATION FAILED FOR** *elmtype elmname* **IN THE** *library* **LIBRARY FOR SYSMOD** *sysmod***. ABEND** *abncode* **OCCURRED WHILE PROCESSING** *library***. THE** *library* **LIBRARY RAN OUT OF SPACE.**

Explanation:

elmtype	element type
elmname	element name
library	ddname of the library
sysmod	SYSMOD ID
abncode	abend code

Retransformation processing failed for the indicated element.

System Action: SYSMOD processing stops for the indicated SYSMOD.

Programmer Response: Increase the size of the indicated library and rerun the job.

GIM69201I **SMPDATA1 DATA SET** *smpdata1* **IS OUT OF SPACE.**

Explanation:

smpdata1	data set name

This message is issued in either of the following cases:

- The indicated SMPDATA1 data set is full.
- There is not enough room on the DASD where the indicated SMPDATA1 data set resides.

System Action: The remaining library change records are written to the SMPDATA2 data set. Processing continues.

Programmer Response: Do **one** of the following:

- If the indicated SMPDATA1 data set is full:
  - Allocate a new larger SMPDATA1 data set and copy the contents of the old data set that is out of space into the new one.
  - Add the library change records from the SMPDATA2 data set into the newly allocated SMPDATA1 data set.
- If there is not enough room on the DASD, you can either:
  - Delete and compress data sets on that DASD, as appropriate, to free up space, or
  - Allocate the SMPDATA1 data set on a DASD that has sufficient space available.
  - Copy the contents of the old SMPDATA1 data set into the newly allocated SMPDATA1 data set.
  - Add the contents of the SMPDATA2 data set into the newly allocated SMPDATA1 data set.

GIM69201S **SMPDATA1 DATA SET** *smpdata1* **IS OUT OF SPACE.**

Explanation:

smpdata1	data set name

This message is issued in either of the following cases:

- The indicated SMPDATA1 data set is full.
- There is not enough room on the DASD where the indicated SMPDATA1 data set resides.

System Action: Since no SMPDATA2 data set was allocated for spill processing, any remaining library change records generated are lost. Processing ends.

Programmer Response: Do **one** of the following:

- If the indicated SMPDATA1 data set is full:
  - Allocate a new larger SMPDATA1 data set and copy the contents of the old data set that is out of space into the new one.
  - Allocate a SMPDATA2 data set and create a DDDEF for the SMPDATA2 data set name in the correct target zone. This data set will be used for future spill processing of library change records.
- If there is not enough room on the DASD, you can either:
  - Delete and compress data sets on that DASD as appropriate to free up space or,
  - Allocate the SMPDATA1 data set on a DASD that has sufficient space available.
  - Copy the contents of the old SMPDATA1 data set into the newly allocated SMPDATA1 data set.
  - Add a SMPDATA2 data set and create a DDDEF for the SMPDATA2 data set name in the correct target zone. This data set will be used for future spill processing of library change records.

GIM69202I LIBRARY CHANGE RECORDING WAS SWITCHED TO SMPDATA2 BECAUSE SMPDATA1 IS FULL.

Explanation: Because the SMPDATA1 data set is full, SMP/E is writing the remaining library change records to the SMPDATA2 data set.

System Action: SMP/E processing continues.

Programmer Response: Do **one** of the following:

- If the indicated SMPDATA1 data set is full:
  - Allocate a new larger SMPDATA1 data set and copy the contents of the old data set that is out of space into the new one.
  - Add the library change records from the SMPDATA2 data set into the newly allocated SMPDATA1 data set.
- If there is not enough room on the DASD, you can either:
  - Delete and compress data sets on that DASD as appropriate to free up space, or
  - Allocate the SMPDATA1 data set on a DASD that has sufficient space available.
  - Copy the contents of the old SMPDATA1 data set into the newly allocated SMPDATA1 data set.
  - Add the contents of the SMPDATA2 data set into the newly allocated SMPDATA1 data set.

GIM69203S SMPDATA2 DATA SET *smpdata2* IS OUT OF SPACE.

Explanation:

smpdata2	data set name

This message is issued in either of the following cases:

- The indicated SMPDATA2 data set is full.
- There is not enough room on the DASD where the indicated SMPDATA2 data set resides.

System Action: Since the SMPDATA2 data set was not large enough to hold the remaining spill records from library change recording, the remaining library change records generated are lost. Processing ends.

Programmer Response: Do **one** of the following:

- If the indicated SMPDATA2 data set is full:
  - Allocate a new larger SMPDATA2 data set and copy the contents of the old data set that is out of space into the new one.
- If there is not enough room on the DASD, you can either:
  - Delete and compress data sets on that DASD, as appropriate, to free up space, or
  - Allocate the SMPDATA2 data set on a DASD that has sufficient space available.
  - Copy the contents of the old SMPDATA2 data set into the newly allocated SMPDATA2 data set.

GIM69264S THE *attribute* VALUE EXCEEDS THE *maxlen* CHARACTER MAXIMUM.

Explanation:

attribute	attribute name
maxlen	maximum length allowed for the specified attribute value

The value specified for the indicated attribute is longer than the maximum length allowed for the attribute value.

System Action: Processing stops.

Programmer Response: Specify a shorter value for the attribute.

This message will be issued in the following situations:

- Archive name exceeds 65 characters
- File name exceeds 44 characters
- Firewall command exceeds 80 characters
- Host name exceeds 255 characters
- Package file name exceeds 266 characters
- Prefix value exceeds 26 characters
- User, password, or account value exceeds 80 characters.

GIM70700T UTILITY *progname* WAS NOT FOUND.

Explanation:

progname utility name

The copy utility specified on the INVOKE control card of the GIMIAP job could not be located on the system.

System Action: Processing of the job step is terminated.

Programmer Response: Do one of the following:

- Ensure that the name specified is correct.
- Make sure that the copy utility is installed on the system.

GIM70704S THE GIMIAP JOB STEP CONTAINS MORE THAN ONE INVOKE STATEMENT. ONLY ONE IS ALLOWED.

Explanation:

A job step within the copy job contained more than one INVOKE statement. Each job step can have only one INVOKE statement.

System Action: Processing of the job step is terminated.

Programmer Response: Ensure that the job step contains no more than 1 INVOKE statement and rerun the job.

GIM70705I GIMIAP PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS *rtncode*.

Explanation:

All data elements were processed. If the return code is greater than zero, there may have been problems with some of them.

rtncode return code

System Action: The return code determines the system action.

Programmer Response: If the return code is zero, no action is required. If the return code is greater than zero, find the message that applies to the data element with the problem. Correct the problem, if necessary.

GIM70707S THE *ctlcard1* CONTROL CARD IS MISSING. ONE IS EXPECTED BEFORE THE *ctlcard2* CONTROL CARD.

Explanation:

ctlcard1 name of GIMIAP control card
ctlcard2 name of GIMIAP control card

Either a COPY or INVOKE statement is missing.

System Action: Processing of the job step is terminated.

Programmer Response: Ensure that the job step contains an INVOKE statement and at least one copy statement. Then rerun the job.

GIM70708S GIMIAP COULD NOT OBTAIN ENOUGH STORAGE FOR PROCESSING.

Explanation:

GIMIAP could not GETMAIN the storage that it needed.

System Action: Processing of the job step is terminated.

Programmer Response: Increase the storage for the execution of this job. Rerun the job.

GIM70709E NO ELEMENTS WERE IDENTIFIED TO BE COPIED FROM LIBRARY *library1* TO LIBRARY *library2*.

Explanation:

library1 the FROMLIB value from the active COPY statement
library2 the TOLIB value from the active COPY statement

A COPY statement was encountered, but no SELECT statements follow it.

System Action: Processing continues.

Programmer Response: If elements were to have been copied from this library, add the SELECT statements and rerun the job step.

GIM70710E NO ELEMENTS WERE IDENTIFIED TO BE COPIED IN THIS JOB STEP.

Explanation:

An INVOKE statement existed with no COPY or SELECT statements.

System Action: Processing continues to the next job step.

Programmer Response: If elements were to have been copied in this job step, add the necessary COPY and SELECT statements and rerun the job step.

GIM70711T *library* DD STATEMENT WAS NOT FOUND.

Explanation:

library name of the library for which a DD statement was not found.

The JCL for this job step does not contain a DD statement for the specified library.

System Action: The job step terminates.

Programmer Response: Ensure that there is a proper DD statement for the library, then rerun the job step.

GIM70712E *elmtype* **ELEMENT** *elmname* **WAS NOT FOUND IN LIBRARY** *library***.**

Explanation:

elmtype	element type or blank
elmname	name of the element being installed
library	library name

The element being installed could not be found in the library indicated.

System Action: Processing continues.

Programmer Response: Ensure that correct library has been identified.

GIM70713E **AN ERROR WAS ENCOUNTERED BY SHELL SCRIPT** *shellscr* **WHILE INSTALLING** *elmtype elmname***.**

Explanation:

shellscr	shell script name
elmtype	element type
elmname	name of element being installed

Shell script processing failed for the indicated element.

System Action: Processing continues with the next selected element.

Programmer Response: Check the shell script output in the PRINT data set to determine the cause of the failure. Fix the error and rerun the job step.

Note: SYSPRINT is SMP/E's default print data set, and is used if no PRINT subentry was specified in the active UTILITY entry for the HFSCOPY utility.

GIM70714E **SHELL SCRIPT PROCESSING FOR** *elmtype elmname* **FAILED. SERVICE** *service* **ENDED WITH RETURN CODE** *rtncode* **AND REASON CODE** *rsncode***.**

Explanation:

elmtype	element type
elmname	element name
service	callable service name
rtncode	return code
rsncode	reason code

Shell script processing failed for the indicated element. SMP/E called the indicated service, but the service was unsuccessful.

System Action: Processing continues with the next selected element.

Programmer Response: See *z/OS UNIX System Services Programming: Assembler Callable Services Reference* for the cause of the error. Correct the error and rerun the job step.

GIM70715E **AN ERROR OCCURRED DURING A** *command* **OPERATION TO THE** *dataset* **DATA SET.**

Explanation:

command	OPEN or WRITE
dataset	ddname of print data set

An error occurred while opening or writing to the print data set. GIMIAP opens and writes shell script output to the print data set.

System Action: Processing continues with the next selected element.

Programmer Response: Ensure that a DD statement exists for print and the DCB information is correct. Fix the error and rerun the job step.

GIM70716E *member* **NOT REPLACED FROM INPUT DATA SET BECAUSE REPLACE WAS NOT SPECIFIED ON THE INVOKE CONTROL CARD.**

Explanation:

member	member name

The member specified already exists in the output data set. It is not replaced because REPLACE was not specified on the INVOKE control card, which is determined by the GENERATE command.

System Action: Processing continues. However, the member has not been replaced with the new element.

Programmer Response: Verify the output. If not correct, issue the GENERATE command with the REPLACE option to ensure members are replaced.

GIM70717I **GIMIAP PROCESSING TO THE** *library* **LIBRARY WAS SUCCESSFUL FOR** *enttype entname***.**

Explanation:

library	ddname of the library to which the element was to be installed.
enttype	entry type
entname	entry name

The indicated element was successfully processed.

System Action: Processing continues.

Programmer Response: None.

GIM70718E **GIMIAP PROCESSING TO THE** *library* **LIBRARY FAILED FOR** *enttype entname***.**

Explanation:

library	ddname of the library to which the element is to be installed.
enttype	entry type
entname	entry name

SMP/E processing failed for the indicated element.

System Action: Processing fails for the associated element and possibly related elements. Processing stops for the element.

Programmer Response:

Programmer Response: Look in SMPOUT for additional messages pertaining to this problem. Respond to the problem as described in those messages.

GIM70720S GIMIAP PROCESSING FROM THE *library1* **LIBRARY TO THE** *library2* **LIBRARY FAILED. THERE IS AN INCONSISTENCY IN THE DATA SET ATTRIBUTES. THE REASON CODE IS** *rsncode***.**

Explanation:

library1	ddname of the library from which the elements were to be copied/reformatted
library2	ddname of the library to which the elements were to be installed.
rsncode	reason for the failure

GIMIAP processing failed for the indicated libraries. The reason codes and their meanings are:

01 A data set with a record format of VS or VBS must be sequential.

02 The input and output data sets must have either a partitioned or sequential organization.

03 The record format of the input data set is not supported by SMP/E. The supported formats are F, FB, FS, FBS, FA, FBA, FM, FBM, V, VB, VA, VBA, VM, VBM, VS, and VBS.

04 A data set with record format VS must have a record length at least as large as that of the data set from which it is copying the element.

05 A data set with record format VBS must have a block size equal to that of the data set from which it is copying the element.

06 The record format of the output data set is not compatible with the record format of the input data set. Compatible record formats are:

Input	Output
F, FB	F, FB, FS, FBS, FA, FBA, FM, FBM, V, VB, VA, VBA, VM, VBM
FS, FBS	F, FB, FS, FBS, FA, FBA, FM, FBM, V, VB, VA, VBA, VM, VBM
FA, FBA	F, FB, FS, FBS, FA, FBA, V, VB, VA, VBA
FM, FBM	F, FB, FS, FBS, FM, FBM, V, VB, VM, VBM
V, VB	F, FB, FS, FBS, FA, FBA, FM, FBM, V, VB, VA, VBA, VM, VBM
VA, VBA	F, FB, FS, FBS, FA, FBA, V, VB, VA, VBA
VM, VBM	F, FB, FS, FBS, FM, FBM, V, VB, VM, VBM
VS	VS (sequential)
VBS	VBS (sequential)

07 The record length and block size of the library are not compatible. If the data set is fixed format, the block size must be evenly divisible by the record length. If the data set is variable format, the block size must be at least 4 larger than the record length.

08 The input data is not completely defined (that is, its data set organization, record format, or record length is zero).

09 The output data set organization is not defined.

System Action: Processing stops for the indicated libraries.

Programmer Response: Correct the inconsistent data set attribute and rerun the job.

GIM70721I GIMIAP PROCESSING TO THE *library* **LIBRARY WAS SUCCESSFUL FOR** *enttype entname***. BLANK DATA HAS BEEN TRUNCATED.**

Explanation:

library	ddname of the library to which the element was to be installed.
enttype	entry type
entname	entry name

The indicated element was successfully processed. However, blank data has been truncated during the installation of the element. You should check the output to verify that the results are acceptable.

System Action: Processing continues.

Programmer Response: None.

GIM70722E STOW PROCESSING TO THE *library* **LIBRARY FAILED FOR** *alias* **(ALIAS of** *elmtype elmname***).**

Explanation:

library	ddname of the library into which the alias was to be installed.
alias	alias name
elmtype	data element type
elmname	data element name

STOW processing failed for the indicated alias, leaving the installation of the data element incomplete.

System Action: Processing continues.

Programmer Response: Verify that there are no I/O errors on the indicated library, check the size of the directory for the indicated library, and try to rerun the job.

GIM70723E GIMIAP PROCESSING TO THE *library* LIBRARY FAILED FOR *enttype entname*. NON-BLANK DATA WOULD HAVE BEEN TRUNCATED.

Explanation:

library	ddname of the library to which the element was to be installed.
enttype	entry type
entname	entry name

The indicated element was not processed because non-blank data would have been truncated during the installation of the element.

System Action: Processing stops for the data element.

Programmer Response: Increase the record length of the output data set so that it matches the record length of the input data set. Then, rerun the job.

GIM70724S INVALID ELEMENT TYPE *elmtype* FOUND IN SELECT CONTROL CARD.

Explanation:

elmtype	element type

Based on the INVOKE control statement, GIMIAP is expected to process either data elements or hierarchical file system elements. If data elements were to be processed, then all SELECT control statements need to identify data elements. If hierarchical file system elements were to be processed, then all SELECT statements must identify hierarchical file system elements. The invalid element type did not follow this rule.

System Action: GIMIAP processing stops.

Programmer Response: Ensure that the SELECT statements specify either all data elements or all hierarchical file system elements, and that they match the utility that INVOKE specifies (COPY for data elements or HFSCOPY for hierarchical file system elements). After correcting the problem, rerun the job.

GIM70725S GIMIAP PROCESSING HAS FAILED FOR LIBRARY *library*. THE LIBRARY IS SEQUENTIAL, YET MORE THAN ONE DATA ELEMENT HAS BEEN SELECTED FOR IT.

Explanation:

library	ddname of the library to which the element was to be installed.

The COPY statement's TOLIB value identified a library that is a sequential data set. The associated SELECT statement or statements identified more than one data element to be copied to the library. A sequential data set can only hold one data element.

System Action: GIMIAP processing stops for all SELECTS associated with the COPY statement.

Programmer Response: Ensure that the library identified by the TOLIB value on the COPY statement is correct and should be sequential. If true, then select only one data element. Rerun the job.

GIM70726S GIMIAP PROCESSING HAS FAILED FOR LIBRARY *library*. IT IS NEITHER A PARTITIONED NOR A SEQUENTIAL DATA SET. THIS IS NOT ALLOWED FOR THE FROMLIB.

Explanation:

library	ddname of the library from which the element was to be installed.

The COPY statement's FROMLIB identified a library that is neither a sequential nor partitioned data set. The FROMLIB is a distribution library, which must always be sequential or partitioned.

System Action: GIMIAP processing stops.

Programmer Response: Ensure that the library identified by the FROMLIB in the COPY statement is correct. Rerun the job.

GIM70727S GIMIAP PROCESSING HAS FAILED FOR LIBRARY *library*. IT HAS A DATA SET ORGANIZATION OF *dsorg* THAT IS NOT COMPATIBLE WITH THE *elmtype* ELEMENT TYPE BEING PROCESSED.

Explanation:

library	ddname of the library to which the element was to be installed.
dsorg	data set organization of the library.
elmtype	the element type being processed.

The DSORG of the library is not compatible with the element type. For instance, if the element type is a data element, then the library cannot have a DSORG of HFS (hierarchical file system). Conversely, if the element type is HFS, the library cannot have a DSORG of PO

(partitioned) or PS (sequential). It is possible that the library has a DSORG of UNK (unknown), which always results in a failure.

System Action: GIMIAP processing stops.

Programmer Response: Ensure that the library identified in the COPY statement is correct. Rerun the job.

GIM70728S OPERAND *operand1* **ON CONTROL STATEMENT** *stmname1* **CANNOT BE USED WHEN OPERAND** *operand2* **IS PRESENT ON CONTROL STATEMENT** *stmname2***. THE OPERANDS ARE MUTUALLY EXCLUSIVE.**

Explanation:

operand1	operand name
stmname1	control statement name
operand2	operand name
stmname2	control statement name

The operand identified in the first statement cannot be used when the operand identified in the second control statement is present.

System Action: GIMIAP processing stops.

Programmer Response: Ensure that the operand present in the first statement identified is allowed when the operand in the second statement is used. Either change the operand in the second statement so that the operand in the first statement can be used, or remove the operand in the first statement that is not allowed. Rerun the job.

GIM70729E THE ELEMENT *elmname* **WAS ALREADY IDENTIFIED IN A PREVIOUS SELECT STATEMENT FOR THE SAME COPY STATEMENT. THIS IS NOT ALLOWED.**

Explanation:

elmname	element name

The element name appeared in a previous SELECT associated with the same COPY control statement. If this occurrence is also processed, the result may be incorrect data in the target library.

System Action: Processing stops for the element.

Programmer Response: Inspect the SELECT statement to ensure that the correct element is identified.

GIM70730I SHELL SCRIPT *shellscr* **PROCESSING FOR** *elmtype elmname* **WAS SUCCESSFUL.**

Explanation:

shellscr	shell script name
elmtype	element type
elmname	element name

Shell script processing completed successfully for the indicated element.

System Action: None.

Programmer Response: None.

Chapter 2. SMP/E Return Codes

SMP/E sets return codes according to conditions occurring during SMP/E command processing. Because many different conditions may occur in a single SMP/E job, the final return code that SMP/E lists is the return code associated with the most severe condition that occurred.

SMP/E writes messages to SMPOUT to describe the various situations and unusual conditions occurring during command processing. The severity of the condition determines the return code associated with the processing and the category the message falls into.

Severity Level	Return Code	Category	When Issued
I	00	Informational	• To show stages of SMP/E processing. • To accompany other messages as an explanation for errors or unusual conditions.
W	04	Warning	• Processing completed but SMP/E detected a possible error. Look at these messages to determine if SMP/E's action was appropriate.
E	08	Error	• SMP/E processing did not complete properly.At least one SYSMOD failed. For example, when APPLY processing of a SYSMOD stops,this is an error, and SMP/E issues a return code of at least 8.
S	12	Severe	• An entire SMP/E command failed. For example, when APPLY processing stops because of insufficient storage, SMP/E issues a return code of 12.
T	16	Terminating	• A situation occurred that forced SMP/E processing to stop.
T	20	Terminating	• An internal SMP/E error caused SMP/Eprocessing to stop. For example, there may be an interface problem between SMP/E modules or a data error on the SMPCSI data set.

The following sections list the return codes that may be issued for SMP/E commands, along with possible causes for the errors and suggestions for how to recover from these errors.

ACCEPT Return Codes

00 ACCEPT processing completed successfully.

04 ACCEPT processing completed with **possible** errors (warnings). Processing did not stop for any SYSMODs. These are some possible errors:

- ACCEPT called a system utility program to perform some work and the utility program issued a nonzero, but still acceptable, return code. Calls to one of the following utility programs could generate this return code:
  - Assembler (default: ASMBLR)
  - Copy, compress, retry utility (default: IEBCOPY)

© Copyright IBM Corp. 1986, 2002

– Update utility (default: IEBUPDTE)
– Superzap utility (default: IMASPZAP)
– Linkage editor (default: IEWL or IEWBLINK)

SMP/E sets the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD entries.

08 ACCEPT processing completed, but processing errors occurred. Processing stopped for at least one SYSMOD. These are some possible errors:

- ACCEPT called a system utility program to perform some work, and the utility program issued a nonzero and unacceptable return code. Calls to one of the following utility programs could generate this return code:
  – Assembler (default: ASMBLR)
  – Copy, compress, retry utility (default: IEBCOPY)
  – Update utility (default: IEBUPDTE)
  – Superzap utility (default: IMASPZAP)
  – Linkage editor (default: IEWL or IEWBLINK)

  SMP/E sets the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD entries.
- A SYSMOD was held for an error reason ID that was not resolved. Check the Causer SYSMOD Summary report and, if necessary, error messages in the SMPOUT output to determine the cause of the problem.
- SMP/E encountered an error while scanning superzap control statements. Check SMPOUT output for error messages to determine the cause of the problem.

  SMP/E sets the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD entries. However, the module was not updated unless an EXPAND linkage editor control statement was included in the modification. In this case, the module was link-edited to expand its size in the distribution library.
- A superzap VERIFY REJECT was encountered by program. Check SYSPRINT output for error messages to determine the cause of the problem.

  SMP/E sets the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD entries. However, the module was not updated unless an EXPAND linkage editor control statement was included in the modification. In this case, the module was link-edited to expand its size in the distribution library.
- A DD statement required by SMP/E was missing. ACCEPT did not process any SYSMOD that required the missing DD statement.
- The distribution zone entry for a SYSMOD specified on the SELECT operand indicates that the SYSMOD was superseded by another SYSMOD.
- A TXLIB or LKLIB member cannot be found. SMP/E sets the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD entries.
- A SYSMOD specified on the SELECT operand was not found on the SMPPTS or in the global zone.
- The PEMAX value being used was too small to process one or more SYSMOD entries or selected element entries being modified. If PEMAX was too small for the element entries, SMP/E may have set the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD

entries. Check SMPOUT output for error messages to determine which SYSMODs or elements were affected.

- SMP/E could not open a target or distribution library. SMP/E may have set the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD entries.
- A SYSMOD containing an element update (++MACUPD, ++SRCUPD, or ++ZAP) attempted to change the ownership of the element with the ++VER VERSION operand. The ownership can be changed only if the SYSMOD provides a replacement for the element.

12 ACCEPT processing stopped. These are some possible errors:

- A function SYSMOD was selected for processing. However, processing stopped for the function before distribution libraries were updated.
- No SYSMODs met ACCEPT specifications.
- GETMAIN failed during ACCEPT processing.
- SMP/E could not open or close a data set.
- The ACCEPT command contained a syntax error.
- A DD statement required by SMP/E was missing.
- A previous command issued an unacceptable return code.

16 A severe error caused SMP/E processing to stop. These are some possible errors:

- The copy utility was called to compress a data set but issued an unacceptable return code. No SYSMODs were accepted, but elements that would have been replaced by SYSMODs that would have been accepted may have been deleted from their distribution libraries.

  Note: The distribution libraries might be unusable. Examine the copy output to determine the status of the data set when the copy utility failed.
- A severe error occurred during an attempt to access an SMP/E data set.
- An error occurred while writing a message.

ACCEPT Error Recovery

After ACCEPT processing completes or abends, examine SMPOUT and SYSPRINT output to determine the relative success of the command. Check any reports that were produced.

For partially accepted SYSMODs, SMP/E sets the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD entries.

You must rerun ACCEPT for a SYSMOD that failed during a previous ACCEPT. After an ACCEPT fails, SMP/E does not allow any command other than ACCEPT to be run for that SYSMOD. If you remove the ERROR status indicator in the distribution zone SYSMOD entry and then try to restore other SYSMODs whose elements may have been updated or replaced in the distribution libraries by the SYSMOD in error, you should expect unpredictable results. The following processing takes place when you rerun ACCEPT:

- All linkage editor processing is repeated.
- All IEBCOPY processing is repeated.
- All macro and source module updating is repeated.
- All assemblies are repeated.

- All superzap processing is repeated. However, if any superzap processing completed through the superzap REPLACE stage, or if any superzap processing produced a superzap VERIFY REJECT in the previous ACCEPT, this rerun of ACCEPT will also fail. To correct this problem, do the following:
  1. Obtain the superzap control cards from the SMPPTS for the modules affected by the SYSMOD. To do this, use the IEBPTPCH utility to punch the SYSMOD.
  2. Reject the SYSMOD from the SMPPTS.
  3. Correct any superzap modification processed that caused a VERIFY REJECT.
  4. Receive and accept the SYSMOD as corrected.

APPLY Return Codes

00 APPLY processing completed successfully.

04 APPLY processing completed with **possible** errors (warnings). Processing did not stop for any SYSMODs. These are some possible errors:

- APPLY called a system utility program to perform some work and the utility program issued a nonzero, but still acceptable, return code. Calls to one of the following utility programs could generate this return code:
  – Assembler (default: ASMBLR)
  – Copy, compress, retry utility (default: IEBCOPY)
  – Update utility (default: IEBUPDTE)
  – Superzap utility (default: IMASPZAP)
  – Linkage editor (default: IEWL or IEWBLINK)

  SMP/E sets the APPLY and ERROR status indicators in the affected target zone SYSMOD entries. Because the target libraries might be unusable, correct the error and resubmit the job.
- APPLY attempted to reassemble a module because of a macro modification. However, no assembler input could be found in either the target zone or the distribution library specified in the DISTSRC or ASMLIB operand list. The module was not reassembled, but SMP/E sets the APPLY status indicator in the affected target zone SYSMOD entries.
- An error was encountered during the cross-zone phase of command processing.
- SMP/E processed a ++DELETE or ++RENAME MCS for an LMOD with a CALLLIBS subentry, but the LMOD was not in the SMPLTS.

08 APPLY processing completed, but processing errors occurred. Processing stopped for at least one SYSMOD. These are some possible errors:

- APPLY called a system utility program to perform some work, but the utility program issued a nonzero and unacceptable return code. Calls to one of the following utility programs could generate this return code:
  – Assembler (default: ASMBLR)
  – Copy, compress, retry utility (default: IEBCOPY)
  – Update utility (default: IEBUPDTE)
  – Superzap utility (default: IMASPZAP)
  – Linkage editor (default: IEWL or IEWBLINK)

  SMP/E sets the APPLY and ERROR status indicators in the affected target zone SYSMOD entries.

- A SYSMOD was held for an error reason ID that was not resolved. Check the Causer SYSMOD Summary report and, if necessary, error messages in the SMPOUT output to determine the cause of the problem.
- SMP/E encountered an error while scanning superzap control statements. Check SMPOUT output for error messages to determine the cause of the problem.

  SMP/E sets the APPLY and ERROR status indicators in the affected target zone SYSMOD entries. However, the module was not updated unless an EXPAND linkage editor control statement was included in the modification. In this case, the module was link-edited to expand its size and the load module was replaced in the target library.
- A superzap VERIFY REJECT was encountered by the superzap. Check SYSPRINT output for error messages to determine the cause of the problem.

  SMP/E sets the APPLY and ERROR status indicators in the affected target zone SYSMOD entries. However, the module was not updated unless an EXPAND linkage editor control statement was included in the modification. In this case, the module was link-edited to expand the size, and the load module was replaced in the target library.
- A DD statement required by SMP/E was missing. APPLY did not process any SYSMOD that required the missing DD statement.
- The target zone entry for a SYSMOD specified on the SELECT operand indicates that the SYSMOD was superseded by another SYSMOD.
- A TXLIB or LKLIB member cannot be found. SMP/E sets the APPLY and ERROR status indicators in the affected target zone SYSMOD entries.
- A SYSMOD specified in the SELECT operand list was not found on the SMPPTS.
- The PEMAX value being used was too small to process one or more SYSMOD entries or selected element entries being modified. If PEMAX was too small for the element entries, SMP/E may have set the APPLY and ERROR status indicators in the affected target zone SYSMOD entries. Check SMPOUT output for error messages to determine which SYSMODs or elements were affected.
- SMP/E could not open a target or distribution library. SMP/E may have set the APPLY and ERROR status indicators in the affected target zone SYSMOD entries.
- A SYSMOD containing an element update (++MACUPD, ++SRCUPD, or ++ZAP) attempted to change the ownership of the element with the ++VER VERSION operand. The ownership can be changed only if the SYSMOD provides a replacement for the element.

12 APPLY processing stopped. These are some possible errors:

- A function SYSMOD was selected for processing. However, processing stopped for the function before any target libraries were updated.
- No SYSMODs met APPLY specifications.
- GETMAIN failed during APPLY processing.
- SMP/E could not open or close a data set.
- The APPLY command contained a syntax error.
- A DD statement required by SMP/E was missing.
- A previous command issued an unacceptable return code.

16 A severe error caused SMP/E processing to stop. These are some possible errors:

- The copy utility was called to compress a data set but issued an unacceptable return code. No SYSMODs were applied, but elements that would have been replaced by SYSMODs that would have been applied may have been deleted from their target libraries.

  Note: The target libraries might be unusable. Examine the IEBCOPY output to determine the status of the data set when IEBCOPY failed.
- A severe error occurred during an attempt to access an SMP/E data set.
- An error occurred while writing a message.

APPLY Error Recovery

After APPLY processing completes or abends, examine SMPOUT and SYSPRINT output to determine the relative success of the command. Check any reports that were produced.

For any partially applied SYSMODs, SMP/E sets the APPLY and ERROR status indicators in the affected target zone SYSMOD entries.

You can rerun APPLY for a SYSMOD that has failed by correcting any conditions that caused SYSMOD processing to stop. If a SYSMOD failed APPLY processing but its inline JCLIN was successfully processed, you should rerun the APPLY command for that SYSMOD and specify the NOJCLIN operand with that SYSMOD ID. (NOJCLIN prevents the JCLIN from being processed again.) If you would rather not try to determine whether to specify NOJCLIN and for which SYSMODs, you may prefer to reapply everything that was processed in the failing run and always process the JCLIN.

The following processing takes place when you rerun APPLY:

- All linkage editor processing is repeated.
- All IEBCOPY processing is repeated.
- All macro and source updating is repeated.
- All assemblies are repeated.
- All superzap processing is repeated. However, if any superzap processing completed through the superzap REPLACE stage, or if any superzap processing produced a superzap VERIFY REJECT in the previous APPLY, this rerun of APPLY will fail. To correct this problem, do the following:
  1. Obtain the superzap control cards from the SMPPTS for the modules affected by the SYSMOD. To do this, use the IEBPTPCH utility to punch the SYSMOD.
  2. Reject the SYSMOD from the SMPPTS.
  3. Correct any superzap modification processed that caused a VERIFY REJECT.
  4. Receive and apply the SYSMOD as corrected.

CLEANUP Return Codes

00 CLEANUP processing completed successfully.

08 CLEANUP processing completed, but processing errors occurred. One

possibility is that there was no corresponding target zone entry for an entry in one of the data sets being cleaned up (SMPMTS, SMPSTS, or SMPSCDS).

12 CLEANUP processing stopped. One possibility is that SMP/E could not open one of the data sets it needed: the target zone, distribution zone, SMPMTS, SMPSTS, or SMPSCDS.

16 A severe error caused SMP/E processing to stop.

CLEANUP Error Recovery

If an error occurs during CLEANUP processing, follow the directions in the "Programmer Response" section of each message.

GENERATE Return Codes

00 GENERATE processing completed successfully.

04 GENERATE processing completed, but there are **potential** errors in the generated jobs. Refer to the messages for more information about these errors.

08 GENERATE processing completed, but SMP/E found potential errors when it analyzed entries in the target zone. These errors may cause problems when you run the generated jobs. Review each message for more information about the errors and follow the directions in the "Programmer Response" section of each message. These are some possible errors:

- SMP/E did not find the JOBCARD member in the specified data set.
- SMP/E did not find an ASSEM or SRC entry in the target zone.
- SMP/E could not assemble the identified module.
- The element entry is missing a subentry (FMID or DISTLIB).
- The LMOD entry had no MOD entries to show that the module was a part of that load module.
- The DDDEF entry defining the target library for a hierarchical file system element did not contain a PATH subentry.

12 GENERATE processing stopped. One possibility is that an operand was missing from the command.

16 An I/O error caused SMP/E processing to stop.

GENERATE Error Recovery

If an error occurs during GENERATE processing, follow the directions in the "Programmer Response" section of each message.

JCLIN Return Codes

00 JCLIN processing completed successfully.

04 JCLIN processing completed, but SMP/E reached an end-of-file statement in the JCLIN input data set before it reached the end of the JCLIN data.

08 JCLIN processing stopped because of a syntax error in the JCLIN input.

12 JCLIN processing stopped. These are some possible errors:

- The JCLIN command contained a syntax error.
- Not enough storage was available.

- The target zone ran out of space (*x*37 abend).
- The PEMAX value being used was too small to process one or more entries on the target zone.
- A DD statement required by SMP/E was missing.
- There is an error associated with the LIBRARYDD comment statement that should follow a SYSLIB or SYSLMOD DD statement specifying the PATH operand.
- A previous command issued an unacceptable return code.

16 A severe error caused SMP/E processing to stop.

JCLIN Error Recovery

If an error occurs in the JCLIN data set, examine SMPOUT output to determine the job, job step, and entry that caused the error. Correct the problem and rerun JCLIN.

If the error occurred in your user-specified JCLIN input data set, see the chapter on the JCLIN command in *SMP/E Commands* for more information.

If the SMPCSI ran out of space during JCLIN processing, try obtaining additional space for the SMPCSI. You can do **one** of the following:

- Allocate a new, larger CSI and copy the old, out-of-space data set into the new one.
- Use the Access Method Services (AMS) ALTER ADDVOLUMES command to add another volume to the CSI.
- Use the AMS IMPORT and EXPORT commands to move the old CSI to a new, larger CSI.
- Use the AMS REPRO command to copy the old CSI into a new, larger CSI.

After you try one of these methods, rerun JCLIN.

LINK Return Codes

00 LINK processing completed successfully.

04 LINK processing completed with possible errors:

- The linkage editor returned an acceptable, but nonzero, return code.
- A designated load module was not in its SYSLIB.
- A distribution zone copy of a module was used, but it was at a different service level from the target zone copy.

08 The linkage editor returned a return code greater than the threshold return code.

12 LINK processing stopped. These are some possible errors:

- The LINK command contained a syntax error.
- A previous command issued an unacceptable return code.
- A target library ran out of space (X37 ABEND).
- A DD statement required by SMP/E was missing.
- A CSI was not available.
- A required module could not be found or was not installed.
- A required load module did not exist (no LMOD entry).
- A required load module was a copied load module.

- A designated load module already contained a designated module by the same name from another zone.
- None of the load modules specified on the INTOLMOD operand are installed in a system library.

16 A severe error caused SMP/E processing to stop.

LINK Error Recovery

If an error occurs during LINK processing, follow the directions in the “Programmer Response” section of each message.

LIST Return Codes

00 LIST processing completed successfully.

04 LIST processing completed, but at least one requested item was not listed. These are some possible errors:

- An entry specified in the LIST command was not found on the data set being listed.
- An entry specified in the LIST command was found, but was not eligible as requested. For example, the SYSMOD specified in **LIST SYSMOD(UZ00004) FUNCTION.** was found, but it was not a function SYSMOD.
- The PEMAX value being used was too small to process a selected entry.
- A DD statement required by SMP/E was missing.

08 LIST processing stopped. One possibility is that the LIST command was specified without any operands.

12 LIST processing stopped. These are some possible errors:

- The LIST command contained a syntax error.
- Not enough storage was available.
- The date range specified on the LIST LOG command was not in the expected format.
- A DD statement required by SMP/E was missing.

16 A severe error caused SMP/E processing to stop.

LIST Error Recovery

If an error occurs during LIST processing, follow the directions in the “Programmer Response” section of each message.

LOG Return Codes

00 LOG processing completed successfully.

12 LOG processing stopped. These are some possible errors:

- The LOG command contained a syntax error.
- A DD statement required by SMP/E was missing.
- A previous command issued an unacceptable return code.

16 An I/O error caused SMP/E processing to stop.

RECEIVE Return Codes

00 RECEIVE processing completed successfully.

04 RECEIVE processing completed with **possible** errors. One possibility is that the RECEIVE exit routine took some action for at least one SYSMOD.

08 RECEIVE processing completed, but processing errors occurred. Processing stopped for at least one SYSMOD. These are some possible errors:

- A selected SYSMOD was not found on the SMPPTFIN data set.
- The PEMAX value being used was too small to process a SYSMOD.
- GETMAIN failed during SYSMOD processing.
- The element name specified on the ++MACUPD or ++SRCUPD MCS was different from the element name specified on the IEBUPDTE ./ CHANGE control statement.
- A SYSMOD in the SMPPTFIN data set contained a syntax error. Syntax errors include validation check errors.
- SMP/E reached an end-of-file statement in the SMPPTFIN data set while it was in the middle of processing of a SYSMOD.
- A return code from the RECEIVE exit routine required RECEIVE to stop processing a SYSMOD.
- Two MCS statements within a SYSMOD referred to the same element.
- The same SYSMOD ID was specified more than once on a single ++ASSIGN MCS statement.

12 RECEIVE processing stopped. These are some possible errors:

- GETMAIN failed during RECEIVE processing.
- A return code from the RECEIVE exit routine required RECEIVE processing to stop.
- The RECEIVE command contained a syntax error.
- An error occurred when SMP/E tried to write a SYSMOD or MCS entry to the SMPPTS.
- None of the SYSMODs specified in the SELECT operand list were found.
- The SMPPTFIN data set does not contain any SYSMODs.
- A DD statement required by SMP/E was missing.
- A previous command issued an unacceptable return code.

16 A severe error caused SMP/E processing to stop. These are some possible errors:

- An I/O error occurred.
- A return code from the RECEIVE exit routine required SMP/E processing to stop.

RECEIVE Error Recovery

If SMP/E issued a message indicating that the SYSMOD was successfully received, then the SYSMOD was completely stored, and the SYSMOD and MCS entries in the PTS have been created.

If you are unsure about the status of a SYSMOD, enter LIST SYSMOD for the global zone to obtain a listing of the SYSMODs on the PTS. If the SYSMOD entry contains the ERROR indicator, the SYSMOD was not successfully received and cannot be applied or accepted. If you are still unsure if the SYSMOD was successfully received, use the REJECT command to delete the SYSMOD. After correcting any conditions that might have caused problems during the previous RECEIVE job, rerun RECEIVE for the SYSMOD.

If the PTS ran out of space during RECEIVE processing, see Chapter 3, “Diagnosing SMP/E Problems” for information on handling the problem. Then rerun RECEIVE.

REJECT Return Codes

00 REJECT processing completed successfully.

04 REJECT processing completed with **possible** errors:

- The copy utility was called to compress a data set, and issued an acceptable, but nonzero, return code.
- A SYSMOD specified on the SELECT operand has already been accepted, and the appropriate BYPASS operand was not specified on the REJECT command. The SYSMOD was not rejected.

08 REJECT processing completed, but processing errors occurred. Processing stopped for at least one SYSMOD. A possible error is that the PEMAX value being used was too small to process a SYSMOD entry on the SMPCSI. That SYSMOD was not rejected.

12 REJECT processing stopped. These are some possible errors:

- No SYSMODs met REJECT specifications.
- GETMAIN failed during REJECT processing.
- SMP/E could not open or close a data set.
- The REJECT command contained a syntax error.
- A DD statement required by SMP/E was missing.
- A previous command issued an unacceptable return code.

16 A severe error caused SMP/E processing to stop. These are some possible errors:

- The copy utility was called to compress a data set but issued an unacceptable return code.
- A severe error occurred during an attempt to access an SMP/E data set.
- An error occurred while writing a message.

REJECT Error Recovery

If a failure occurs during REJECT processing, enter the REJECT command for those SYSMODs that were not successfully rejected. If a function SYSMOD is being rejected, check the GLOBALZONE entry to see if the FMID subentry for that SYSMOD ID was deleted. If it was not and should have been (that is, the SYSMOD was never applied or accepted), use the UCLIN command to delete the FMID subentry.

REPORT Return Codes

00 REPORT processing completed successfully. This return code is also issued if no applicable information was found. For example, it is issued for REPORT SOURCEID in these cases:

- In one of the specified zones or ZONESETs, no SOURCEIDs were found in any of the SYSMOD entries.
- In all of the specified zones or ZONESETs, no SOURCEIDs were found in any of the SYSMOD entries.
- No SYSMOD entries were found in any of the specified zones or ZONESETs.

08 REPORT processing completed, but processing errors occurred. One

possibility is that SMP/E could not find an FMID or FMIDSET that matched the value specified on the FORFMID operand.

12 The REPORT command stopped. These are some possible errors:

- The ZONESET entry is not defined in the global zone (REPORT CROSSZONE, REPORT ERRSYSMODS, REPORT SOURCEID only).
- There is no ZONEINDEX subentry for one or more of the following:
  - The zone specified on the INZONE or COMPAREDTO operand (REPORT SYSMODS only)
  - One of the zones in the ZONESET (REPORT CROSSZONE, REPORT ERRSYSMODS, REPORT SOURCEID only)
  - A zone specified on the ZONES operand (REPORT ERRSYSMODS, REPORT SOURCEID only)
- One of the zone names in the FORZONE operand is not included in the ZONESET (REPORT CROSSZONE only).
- The zones in the ZONESET are not all the same type (target or distribution) and neither the TARGETZONE operand nor the DLIBZONE operand was specified (REPORT CROSSZONE only).
- Output is to be written to SMPPUNCH (NOPUNCH was not specified), but the DD statement or DDDEF entry needed for the SMPPUNCH data set was missing.
- The date range specified by the BEGINDATE and ENDDATE operands is not correct. For example, the BEGINDATE value might have been later than the ENDDATE value (REPORT ERRSYSMODS only).
- The zones specified on the INZONE or COMPAREDTO operands have no matching SREL (REPORT SYSMODS only).
- The global zone was specified on the INZONE or COMPAREDTO operand (REPORT SYSMODS only).
- The same zone was specified on both the INZONE and COMPAREDTO operands (REPORT SYSMODS only).
- The zone specified on the INZONE or COMPAREDTO operand contains no SYSMOD entries (REPORT SYSMODS only).

REPORT Error Recovery

If an error occurs during REPORT processing, follow the directions in the “Programmer Response” section of each message.

RESTORE Return Codes

00 RESTORE processing completed successfully.

04 RESTORE processing completed with **possible** errors:

- RESTORE called a system utility program to perform some work and the utility program issued a nonzero, but acceptable, return code. Calls to one of the following utility programs could generate this return code:
  - Assembler (default: ASMBLR)
  - Copy, compress, retry utility (default: IEBCOPY)
  - Update utility (default: IEBUPDTE)
  - Superzap utility (default: IMASPZAP)
  - Linkage editor (default: IEWL or IEWBLINK)

  SMP/E restores the affected SYSMOD and sets the RESTORE indicator in the associated target zone SYSMOD entry.

- Assembly input for a selected module was not found on the target zone.
- An error was encountered during the cross-zone phase of command processing.
- SMP/E processed a ++RENAME MCS for an LMOD with a CALLLIBS subentry, but the LMOD was not in the SMPLTS.

08 RESTORE processing completed, but processing errors occurred. Processing stopped for at least one SYSMOD. These are some possible errors:

- RESTORE called a system utility program to perform some work and the utility program issued a nonzero, unacceptable return code. Calls to one of the following utility programs could generate this return code:
  - Assembler (default: ASMBLR)
  - Copy, compress, retry utility (default: IEBCOPY)
  - Update utility (default: IEBUPDTE)
  - Superzap utility (default: IMASPZAP)
  - Linkage editor (default: IEWL or IEWBLINK)

  SMP/E sets the RESTORE and ERROR status indicators in the affected target zone SYSMOD entries.
- A DD statement required by SMP/E was missing. RESTORE did not process any SYSMOD that requires the missing DD statement.
- A SYSMOD selected for RESTORE processing had never been applied. However, SMP/E had created a SYSMOD entry for it when it was superseded by another SYSMOD. (This is called a “superseded-only” SYSMOD entry.) Neither the superseded SYSMOD nor the superseding SYSMOD was restored.
- RESTORE processing requires an element entry that cannot be found on the target zone. RESTORE processing stopped for all affected SYSMODs.
- The PEMAX value being used was too small to process a selected SYSMOD or element entry.
- A SYSMOD selected for RESTORE had already been accepted. That SYSMOD was not restored.
- SMP/E could not open a required data set. None of the affected SYSMODs were restored.

12 RESTORE processing stopped. These are some possible errors:

- No SYSMODs met RESTORE specifications.
- GETMAIN failed during RESTORE processing.
- SMP/E could not open a data set.
- The RESTORE command contained a syntax error.
- A DD statement required by SMP/E was missing.
- A previous command issued an unacceptable return code.

16 A severe error caused SMP/E processing to stop. These are some possible errors:

- The copy utility was called to compress a data set but issued an unacceptable return code. No SYSMODs were restored. However, modules that would have been replaced if the SYSMODs had been restored were deleted from the appropriate target libraries.

  Note: The target libraries might be unusable; that is, the system or some of its components might not run.

- A severe error occurred while deleting members from a target library before compression of that library.

  Note: The target libraries might be unusable; that is, the system or some of its components might not run.
- The SMPSCDS contains an entry that has not been converted to the format required for IBM SMP/E for z/OS and OS/390, V3R1.
- An error occurred while writing a message.
- A severe error occurred during an attempt to access an SMP/E data set.

RESTORE Error Recovery

After the RESTORE command completes, examine SMPOUT and SYSPRINT output to determine the relative success of the command. Check any reports that were produced.

For any partially restored SYSMODs, SMP/E sets the RESTORE and ERROR status indicators in the affected target zone SYSMOD entries.

You should rerun RESTORE for a SYSMOD that failed during previous RESTORE processing. The following processing takes place when you rerun RESTORE:
- All linkage editor processing is repeated.
- All IEBCOPY processing is repeated.
- All assemblies are repeated.
- All IEBUPDTE processing is repeated.

If any library runs out of space during RESTORE processing, see Chapter 3, “Diagnosing SMP/E Problems” and *SMP/E Commands* for information on how to handle the problem. Then rerun RESTORE.

SET Return Codes

00 SET processing completed successfully.

08 SET processing completed, but processing errors occurred.

12 SET processing stopped. These are some possible errors:
- SMP/E could not open the zone.
- You specified a zone that does not exist.

16 An I/O error occurred when SMP/E tried to open a zone. This caused SMP/E processing to stop.

SET Error Recovery

If an error occurs when SMP/E is processing a SET statement, correct the data (zone name, OPTIONS entry, or data in the zone definition entry). Follow the directions in the “Programmer Response” section of each message.

UCLIN Return Codes

00 UCLIN processing completed successfully.

04 UCLIN processing completed, but with unexpected results.
- SMP/E reached an end-of-file statement in SMPCNTL before it processed an ENDUCL command.
- No UCL statement followed the UCLIN command.

08 UCLIN processing completed, but processing errors occurred. These are some possible errors:
- At least one UCL statement contained a syntax error.
- At least one UCL statement does not meet conditions for update.

12 UCLIN processing stopped.
- The UCLIN command contained a syntax error.
- A previous command issued an unacceptable return code.
- A DD statement required by SMP/E was missing.

16 A severe error caused SMP/E processing to stop.

UCLIN Error Recovery

If a failure occurs when processing a UCL statement, follow the directions in the "Programmer Response" section of each message.

ZONECOPY Return Codes

00 ZONECOPY processing completed successfully.

04 ZONECOPY processing completed, but with unexpected results. The input zone contained cross-zone subentries.

08 A cross-zone subentry from the input zone refers to a zone with the same name as the receiving zone.

12 ZONECOPY processing stopped. These are some possible errors:
- The name of the zone in the SET BOUNDARY command and the name of the receiving zone are different.
- The input and receiving zone have the same name.
- The input or receiving zone is GLOBAL.
- The combination of input and receiving zone types is invalid. Valid combinations are:
  - A distribution zone into a distribution zone
  - A distribution zone into a target zone
  - A target zone into a target zone
- The input and receiving zones exist in the same CSI.
- SMP/E could not open the input zone or the receiving zone.
- The receiving zone already exists.

16 An I/O error caused SMP/E processing to stop.

ZONECOPY Error Recovery

If an error occurs during ZONECOPY processing, follow the directions in the "Programmer Response" section of each message.

ZONEDELETE Return Codes

00 ZONEDELETE processing completed successfully.

04 ZONEDELETE processing completed, but SMP/E could not acquire the global zone to delete the zone index. The deleted zone contained cross-zone subentries.

12 An I/O error caused ZONEDELETE processing to stop.

16 A severe error caused SMP/E processing to stop.

ZONEDELETE Error Recovery

If SMP/E could not delete the zone index from the global zone, use UCLIN to delete the zone index. Follow the directions in the “Programmer Response” section of each message.

ZONEEDIT Return Codes

00 ZONEEDIT processing completed successfully.

04 ZONEEDIT processing completed with **possible** errors (warnings). Processing did not stop for any SYSMODs.

08 ZONEEDIT processing completed, but processing errors occurred. These are some possible errors:

- A changed data set name would have been more than 44 characters.
- SMP/E reached an end-of-file statement in SMPCNTL before it processed an ENDZONEEDIT command.

12 ZONEEDIT processing stopped. These are some possible errors:

- The entry type, field name, or field value was not valid.
- The entry does not exist in the specified zone.
- SMP/E could not open the specified zone.
- The same field name was specified more than once on a single CHANGE statement.
- The ZONEVALUE specified for XZENTRIES matched the name of the set-to zone.
- A conditional change was specified for XZENTRIES.
- The ZONEVALUE specified for XZENTRIES would have created a duplicate subentry.

16 A severe error caused SMP/E processing to stop.

ZONEEDIT Error Recovery

If an error occurs during ZONEEDIT processing, follow the directions in the “Programmer Response” section of each message.

ZONEEXPORT Return Codes

00 ZONEEXPORT processing completed successfully.

04 ZONEEXPORT processing completed, but the global zone was not available for update processing. The exported zone contained cross-zone subentries.

12 ZONEEXPORT processing stopped. These are some possible errors:

- PURGE was specified when the input zone was the global zone.
- The SET BOUNDARY zone name and the input zone name were not the same.
- SMP/E could not open the zone.
- SMP/E opened the zone but could not position to the first record.
- The zone was not the global zone, and the first record in the zone was not for the zone definition entry.
- SMP/E could not open the output data set.
- The OUTFILE ddname is the same as the zone name specified on the SET command.

16 An I/O error caused SMP/E processing to stop.

ZONEEXPORT Error Recovery

If an error occurs during ZONEEXPORT processing, follow the directions in the "Programmer Response" section of each message.

ZONEIMPORT Return Codes

00 ZONEIMPORT processing completed successfully.

04 ZONEIMPORT processing completed, but with unexpected results.

- The receiving zone type is target, and the input zone type is DLIB. (SMP/E changes the type in the zone definition entry to target).
- The receiving zone name and the name from the zone definition entry are different. (SMP/E inserts the receiving zone name to the zone definition entry).

08 A cross-zone subentry from the imported zone refers to a zone with the same name as the receiving zone.

12 ZONEIMPORT processing stopped. These are some possible errors:

- The set-to zone and receiving zone names are different.
- The receiving zone already exists.
- SMP/E could not open the input data set.
- SMP/E opened the data set, but the zone definition entry is not valid for the ZONEIMPORT command.
- The receiving zone type is not allowed for the zone type being imported. You can either import a zone into the same type of zone, or you can import a distribution zone into a target zone.
- The name of the zone that SMP/E is importing and the name from the zone definition entry are different.
- SMP/E could not open the receiving zone.
- The receiving zone is GLOBAL, and SMP/E already defined a zone in the CSI data set.
- SMP/E reached an end-of-file statement in the input data set before it got to the record with a zone encode value of X'FFFF'.
- The INFILE ddname is the same as the zone name specified on the SET command.

ZONEIMPORT Error Recovery

If an error occurs during ZONEIMPORT processing, follow the directions in the "Programmer Response" section of each message.

ZONEMERGE Return Codes

00 ZONEMERGE processing completed successfully.

04 ZONEMERGE processing completed, but with unexpected results. The receiving zone contained cross-zone subentries.

08 A cross-zone subentry from the merged zone refers to a zone with the same name as the receiving zone.

12 ZONEMERGE processing stopped, but with unexpected results.

- SMP/E met an I/O error.

- The job was out of storage.

16 A severe error caused SMP/E processing to stop.

ZONEMERGE Error Recovery

If the message shows that the job ran out of storage, resubmit the job using a larger REGION value. In addition, follow the directions in the “Programmer Response” section of each message.

ZONERENAME Return Codes

00 ZONERENAME processing completed successfully.

04 ZONERENAME processing completed, but with unexpected results. The renamed zone contained cross-zone subentries.

12 ZONERENAME processing stopped, but with unexpected results.

- There is no ZONEINDEX subentry for the old zone name.
- There is a ZONEINDEX subentry for the old zone name, but there is no zone definition entry.
- There is already a ZONEINDEX entry for the new zone name.
- The new zone already exists in the SMPCSI.
- SMP/E could not open the data set that contains the zone to be renamed.
- SMP/E could not rename the old zone name to the new zone name.
- The old zone name was the same as the new zone name.
- The ZONERENAME command specified mutually exclusive operands.
- **GLOBAL** was specified as either the new zone name or the old zone name.
- The TOTYPE operand was specified to change the renamed zone from a distribution zone to a target zone. However, the old zone is already defined as a target zone in either the ZONEINDEX subentry or the zone definition entry.
- A cross-zone subentry from the zone being renamed refers to the new zone name.

16 A severe error caused SMP/E processing to stop.

ZONERENAME Error Recovery

If an error occurs during ZONERENAME processing, follow the directions in the “Programmer Response” section of each message.

Chapter 3. Diagnosing SMP/E Problems

This chapter discusses diagnosis.

Is SMP/E the Problem?

IBM SMP/E for z/OS and OS/390, V3R1 works with a variety of other programs. It calls system utilities and depends on virtual storage access method (VSAM) support for the consolidated software inventory data set (SMPCSI). The SMP/E dialogs run under time-sharing option (TSO) and Interactive System Productivity Facility (ISPF). As a result, what may appear to be an SMP/E problem may actually be caused by one of these other products, or there may be a problem with the installation procedure you are following or with a system modification (SYSMOD) you are trying to install.

Before you report the problem to IBM, try to determine whether SMP/E is the cause. Check the items in the following lists to see if something other than SMP/E might be the source of the problem. If your problem matches one of those items, follow the suggested steps. Otherwise, go to "Handling SMP/E Problems without Calling IBM" on page 290 to see if you have an SMP/E problem that you can correct.

Installation Problems

To avoid installation problems:

- For all SMP/E processing, allocate a real data set for the log data set (SMPLOG), not a dummy data set. This helps make sure the log data is not lost.
- Get all the necessary service. Some of the methods available for obtaining service and preventive maintenance include:
  - Enhanced Service Offering (ESO), which provides monthly PER closed PTFs
  - CBPDO, which in addition to the monthly PER closed PTFs, includes COR closed reach ahead fixes for HIPERs and PEs on a weekly basis. For more information on CBPDOs, see *z/OS and z/OS.e Planning for Installation*.
  - SystemPac's Selective Follow On Service tapes, which provide HIPERs and PTFs resolving PEs
  - Service Update Facility (SUF), which can be used to order and receive corrective and recommended preventive maintenance. SUF uses your CSI as input to tailor the service to your environment. For more information on SUF, visit the SUF web site at http://www.ibm.com/s390/suf/.
  - PSP (Preventive Service Planning) Buckets, which contain the HIPERs and PEs in the Service Recommendation list. The PSP buckets are available by contacting the IBM Support Center or electronically through ServiceLink.
  - RefreshPac Offering
  - Enhanced HOLDDATA, which is an SMP/E installable file containing ++HOLDs for HIPERs and PEs used to identify HIPERs and PE fixes that are not installed on a system. After the file is received, the SMP/E REPORT ERRSYSMODS command can be run to identify the HIPERs and resolving PEs not installed. The resulting report includes the HIPER symptom flags. For more information on Enhanced HOLDDATA, visit the Enhanced HOLDDATA web site at http://service.boulder.ibm.com/390holddata.html.

For more information, visit http://www.s390.ibm.com/marketing/psos390maint.html

© Copyright IBM Corp. 1986, 2002

Is SMP/E the Problem?

- Follow the directions in the program directory for the product you are installing. If you have an installation package, such as the Custom-Built Product Delivery Offering (CBPDO), follow the directions in the installation materials that come with it, such as the related installation materials (RIMs) for the CBPDO. An installation problem may be due to the installation procedure itself rather than SMP/E. In such cases, report the product or the installation package as the cause of the problem.

If you have a problem during installation:

- If there is no error in SMP/E processing but you need help or additional information about using SMP/E in installing a product, contact your systems programmer.
- If you are having difficulty installing a function SYSMOD, check the PSP file for additional information about installing the function. If this information is not enough, or if the PSP file states that the function is not constructed correctly, the problem may be in the function itself and not SMP/E. In such cases, report the function as the cause of the problem.
- Always check the Element Summary Report for uninstalled elements, even if a SYSMOD appears to have been installed without errors.

Messages and Return Codes

Look up all SMP/E warning messages and error return codes in Chapter 1, “SMP/E Messages” and Chapter 2, “SMP/E Return Codes” and follow the recommended procedures.

Performance Problems

If you have a performance problem:

- Check to see whether any programs that do a lot of I/O operations, such as Information Management System (IMS), were running when you noticed the degraded performance.
- Check to see whether you have installed any engineering changes or service to SMP/E or other products that may have degraded system performance.

For information about collecting documentation and reporting an SMP/E performance problem, see Collecting Documentation for SMP/E Problems.

Utility Problems

If you have a problem when running a utility program under SMP/E:

- Check the utility output in the SYSPRINT data set and SMP/E messages in SMPOUT for indications of problems during utility processing. SMP/E writes a time, date, and sequence number stamp on the completion messages for copy, link-edit, superzap, and update processing. Since the same sequence number is included in the utility SYSPRINT output, it can be used as an index to more easily find the desired information in the SYSPRINT data set. Assembler messages have a time and date stamp you can use to find assembler output.

  Note: These completion messages appear only after APPLY, ACCEPT, and RESTORE processing.
- Check to see whether the associated UTILITY entry refers to the correct program and return code. Note that the RETURN CODE value in the LMOD entry overrides (for that load module) the value in the UTILITY entry. Check to see whether this LMOD RC value is appropriate for this utility program.

- If you still are not sure whether SMP/E or the utility caused the error, run the job outside SMP/E, if possible, to see if the same problem occurs. Use the same libraries and either recreate the same job or run a similar utility job.

SMPCSI Problems

If you have a problem with the SMPCSI, do the following to make sure your CSI is at the right level:

- Initialize your CSI with the GIMZPOOL record shipped with a supported release of SMP/E. A CSI to be used with IBM SMP/E for z/OS and OS/390, V3R1 must be initialized with the appropriate GIMZPOOL record.
- Run IBM SMP/E for z/OS and OS/390, V3R1 against a CSI created by a supported release of SMP/E.
- Check to see whether the necessary VSAM service has been installed.
- Check to see whether there are any user modifications to VSAM, SMP/E, or the catalog manager that might have caused the problem.
- Check whether message GIM27901S has been issued. If it has, you may want to use the VPLFUNCT specified in this message on the DEBUG DUMPRPL command to request a dump of the request parameter list (RPL). For more information on the DEBUG DUMPRPL command, see *SMP/E Commands*.
- Check to see whether the system has encountered VSAM problems outside of SMP/E processing.
- If VSAM is having trouble accessing or storing the index part of the SMPCSI, you may want to use the following procedure to fix the index part of the data set:
  1. Make a backup copy of the current level of the SMPCSI on tape.
  2. Enter the following command for the SMPCSI to obtain the key information:

     LISTCAT CLUSTER ENTRY('*name*')
  3. Use the access method services (AMS) REPRO command to copy the data part of the SMPCSI to a sequential data set.
  4. Sort the data by key using the SORT utility and the key information from the LISTCAT output. Eliminate any duplicate records.
  5. Delete the SMPCSI and allocate a new one. Do not prime it with GIMZPOOL.
  6. Use the REPRO command to copy the sorted data into the new SMPCSI.

Exception SYSMOD Problems

Sometimes SMP/E cannot install a SYSMOD because one of the SYSMOD's requisites is being held. You must handle that exception SYSMOD before SMP/E can install the other SYSMOD.

- If the SYSMOD is held for a SYSTEM reason ID, list the associated HOLDDATA entry to see what action must be taken to install the SYSMOD. If the action was taken before the PTF was installed, you can bypass the HOLD to install the PTF. Otherwise, perform the action specified in the HOLDDATA and then bypass the HOLD to install the PTF.
- If the SYSMOD is held for an ERROR reason ID, which is associated with a program error PTF (PE-PTF), check the ++HOLD comment and the authorized program analysis report (APAR) that describes the problem to see whether installing the SYSMOD will cause any problems.

  If possible, install the APAR or the fixing PTF along with the PE-PTF. This automatically releases the ERROR HOLD. If the fixing PTF is unavailable and you want to install the PE-PTF anyway, you can bypass the HOLD.

Notes:

1. Generally, when doing preventive service, you do not need to research PE-PTFs. These PTFs remain held until the fixing PTF is installed. When you install the CBPDO tape or ESO that contains the fixing PTF, SMP/E automatically installs the fixing PTF along with the PE-PTF.
2. When you are doing corrective service, if the PTF you need to install to fix your specific problem is a PE-PTF, research the APAR that describes the PE-PTF to see if installing the PE-PTF will introduce an acceptable problem to your system.
3. You may not always have to correct the problem before you install a PE-PTF. For example, a PE-PTF that contains a syntax error in a comment in the cover letter is generally safe to install.

For more information on processing exception SYSMODs, see the *SMP/E User's Guide*.

Handling SMP/E Problems without Calling IBM

This chapter describes some problems that you may be able to correct without calling IBM. For example, you may be able to:

- Analyze error conditions indicated by SMP/E messages, return codes, and LIST output
- Resolve problems with the SMP/E region size
- Resolve *x*37 abends for data sets that run out of space
- Resolve VSAM access problems
- Handle SMP/E abends
- Prevent SMP/E dialog problems

If the problem you are experiencing falls into one of these categories, read the appropriate section in this chapter and follow the suggested procedure. If your problem is not described here, or if the suggested procedure does not correct the problem, see Reporting SMP/E Problems to IBM, which discusses reporting a problem to IBM.

Analyzing Errors

Many suspected SMP/E problems can be corrected by responding to SMP/E error messages. Therefore, when you get an SMP/E error message, refer to Chapter 1, "SMP/E Messages" to find its explanation and the recommended response. If you receive a message (such as GIM35302E) containing an MVS error code, refer also to *z/OS MVS Programming: Authorized Assembler Services Guide*, SA22-7608, to find an explanation of the error code and the recommended response.

If the recommended response does not correct the problem, follow these steps to determine the cause of the problem and the correct recovery methods:

1. Examine the return codes in each job step and subsequently the return codes for each command.

   Starting with the final code (the one returned by the failing job step), trace back through the job log to find the SMP/E command return codes that caused the job step return code. Remember, a single return code can be the product of multiple errors.

   The job step return code issued for SMP/E is the highest return code generated by all SMP/E commands in that step. The job step return codes are:

Return Code	Meaning
00	SMP/E processing ran successfully and without errors. The severity code in associated SMP/E messages is I.
04	SMP/E processing ran, but warning messages were issued. The severity code in associated SMP/E messages is W.
08	SMP/E processing ran, but processing errors occurred and processing stopped for at least one SYSMOD. The severity code in associated SMP/E messages is E. Check the reports produced to see which SYSMODs were not successfully processed.
12	SMP/E processing stopped for at least one SMP/E command. The severity code in associated SMP/E messages is S.
16	SMP/E processing stopped because of a severe error. The severity code in associated SMP/E messages is T.
20	SMP/E processing stopped because of an internal SMP/E error. For example, there may be an interface problem between SMP/E modules or a data error on the SMPCSI data set. The severity code in associated SMP/E messages is T.

For specific return codes for each of the SMP/E commands, see Chapter 2, “SMP/E Return Codes”.

2. As you trace back through the return codes in SMPOUT, check for error and warning messages issued with the return codes. Use the information supplied by the messages to help you interpret the meaning of the return codes.
3. If an APPLY, ACCEPT, or RESTORE command completes with a return code greater than 4, first check the Causer SYSMOD Summary Report in SMPRPT to determine the root failures.
4. Check to see whether you coded the RC operand on any SMP/E commands.

   The RC operand enables you to specify the maximum acceptable return codes from specified SMP/E commands in order to bypass normal SMP/E return code processing. If a specified command returns a code higher than the maximum specified in the RC operand, the command that contains the RC operand is not processed and issues a return code of 12.

   For example, if you specify RC(RECEIVE=04) on the APPLY command, and the previous RECEIVE command returns a code of 08, APPLY processing is not performed and the APPLY return code is 12.

   For further information about the RC operand, refer to the discussion of it under the description in each command in *SMP/E Commands*.
5. Check the SYSPRINT data set for information about the success or failure of the system utility programs invoked by SMP/E commands.
6. Use the LIST command to check the contents of the SMPLOG data set. If you need to check SYSMOD status but do not have the reports and messages that include it, use the LIST command as follows:
   - Enter **LIST LOG** to display the contents of the SMPLOG data set. This log is cumulative and should be examined for the impact of prior SMP/E runs on the current problem.

     For more details on the LIST command, see *SMP/E Commands*.
   - Enter **LIST SYSMOD ERROR** for the desired target zone to obtain the status of any SYSMODs that were applied or restored but may be in error. Check whether the ERROR indicator is set for APPLY or RESTORE processing.

- Enter **LIST SYSMOD ERROR** for the desired distribution zone to obtain the status of any SYSMODs that were accepted but may be in error.
- Enter **LIST SYSMOD** for the global zone to obtain the status of SYSMODs that were received.

Resolving Problems with the SMP/E Region Size

If an SMP/E message indicates that there is not enough storage for processing, do one or more of the following:

- Check if you have allocated enough storage for SMP/E. If necessary, allocate a larger region size for SMP/E, or specify **REGION=0M** to enable the SMP/E job step to get the maximum space above 16 megabytes, and run the job step again.
- Make sure you are using the OPTIONS entry containing the desired PEMAX value. For example, if you are doing an ACCEPT, the DLIBZONE entry on the SET command must name the OPTIONS entry you want to use.
- Decrease the number of SYSMODs being processed. For example, run the command again and limit the SYSMODs to a particular type, function modification identifier (FMID), or SOURCEID.
- Remove a command operand that uses extra storage (such as XREF on the LIST command), and process the command again.
- Decrease the PEMAX value in the OPTIONS entry being used. Remember, however, that you may not be able to process SYSMODs that depend on the larger PEMAX size.

Resolving Space Problems (x37 Abends)

This section suggests ways to:

- Prevent *x*37 abends
- Recover from *x*37 abends

Preventing x37 Abends

You can prevent data sets from running out of space by doing the following:

- Periodically list the volume tables of contents (VTOCs) of the target library and distribution library volumes, especially after doing a system generation or compressing the data sets. If a data set has a small amount of free space, reallocate a larger data set and copy the old data set into it. This will help prevent future space problems.

  For more information about the initial allocation of system and SMP/E data sets, see the program directory shipped with SMP/E.
- Make sure the OPTIONS entry being used contains a RETRYDDN list reflecting the data sets you may need to compress. When this list is specified, if any data set runs out of space during processing, SMP/E tries to recover from the error by compressing the data set. If retry processing does not reclaim sufficient space and input to the utility was batched (copy or link-edit utility only), SMP/E debatches the input and retries the utility for each member separately.

  For detailed information on the RETRY operand, see the explanation of ACCEPT, APPLY, LINK, and RESTORE commands in the *SMP/E Commands* manual. For guidelines on how to set up the desired retry processing, see the *SMP/E User's Guide*.
- Compress the affected data sets by specifying the COMPRESS operand on the APPLY, ACCEPT, REJECT, or RESTORE command. The affected data sets will be compressed even if they have enough space for SMP/E processing.

  If you specify COMPRESS(ALL) whenever you install a product, you can avoid space problems.

For detailed information on the COMPRESS operand, see the explanation of ACCEPT, APPLY, REJECT, and RESTORE commands in the *SMP/E Commands* manual.

- If you use JCLIN to define your own modules assembled with your own macros to SMP/E, and the SYSIN data set to be assembled is large, you may want to include the assembler COPY statement as part of the assembly SYSIN. This enables you to obtain large amounts of data from SYSLIB at assembly time and reduces the size of the assembly data stored in the target zone.

  Note: If you use this approach, SMP/E checks the COPY statement, but not the copied data. Therefore, it does not determine whether any modules need to be reassembled because of changes in macros being processed.

  Another possibility is to supply the assembly code as a ++SRC element to eliminate the storing of the assembly data in the target zone. This has the side effect of not requiring an assembler JCLIN step.

For examples of adding new load modules and module entries to the target or distribution zone, see the *SMP/E Commands* manual.

Recovering from x37 Abends

You can often recover from *x*37 abends by using the following methods:

- If an SMP/E command fails because there is not enough space, check if the COMPRESS operand is allowed and was specified for that command. If the COMPRESS operand is valid, and if it was not specified the last time, rerun the SMP/E command with COMPRESS.
- To obtain additional space in the SMPLOG data set, use one of the following methods:
  - Allocate a new SMPLOG data set and create a backup copy of the old SMPLOG data set, retaining it according to your usual recovery procedures.
  - Create a backup copy of the old SMPLOG, and retain it according to your usual recovery procedures. The next time you run SMP/E commands, specify DISP=OLD for SMPLOG. This overlays the contents of the old SMPLOG that you saved.

    You **must** specify DISP=MOD every time you run SMP/E, or you will continue to overlay the SMPLOG every time SMP/E commands are processed.

  Note: You can also allocate an alternative SMPLOG data set, SMPLOGA, to be used when the primary SMPLOG data set is full. See the *SMP/E Reference* manual for more information.
- To obtain additional space for the target libraries or the distribution libraries, allocate a new, larger data set and copy the old data set that is out of space into the new data set.
- To obtain additional space for the SMPCSI data set, you can do one of the following:
  - Allocate a new, larger CSI and copy the old data set that is out of space into the new one. (You can use the AMS REPRO command to copy the old CSI into a new, larger CSI.)
  - Use the AMS ALTER ADDVOLUMES command to add another volume to the CSI.
  - Use the AMS IMPORT and EXPORT commands to move the old CSI to a new, larger CSI.

Note: If the job fails again after you have taken steps to get more space, the reason could be that SMP/E updated or created CSI entries before processing stopped for the original abend. If this was the case, do one of the following:

- Restore your CSI with a previously made backup, or
- Check which element entries (including LMOD entries) were affected by the processing. The Element Summary report from your first attempt to run the job will help you with this step. Then use UCLIN to undo changes to updated entries and to delete any entries that were created and rerun the job.

- To obtain additional space in the SMPMTS, SMPPTS, SMPSCDS, or SMPSTS data set, use one of the following methods:
  - Run the CLEANUP command to delete unnecessary entries from SMPMTS, SMPSTS, or SMPSCDS. You may need to do this when you apply SYSMODs after accepting them, or when you accept SYSMODs that were applied to a group of target libraries built from the same distribution library.
  - Run the ACCEPT command or the REJECT command for any SYSMODs that have not been accepted. On the next ACCEPT or REJECT command you process, specify the COMPRESS operand with a value of SMPMTS, SMPPTS, or SMPSTS. For ACCEPT processing, you must specify PURGE in the OPTIONS entry that is in effect.
  - If no SYSMODs are candidates for an ACCEPT or REJECT command, allocate a new, larger data set, and copy the old data set that is out of space into the new data set.

Resolving VSAM Access Problems

If SMP/E has difficulty accessing a VSAM data set, it issues message GIM27901S to tell you that a VSAM error has occurred. The message contains a VPLFUNCT value, which may be specified on the DEBUG DUMPRPL command to request a dump of the RPL control block. This dump will be written to the SMPDEBUG data set. (For more information on this command, see “When VSAM Errors Occur during Attempts to Access a CSI Data Set” on page 307 and the DEBUG command in *SMP/E Commands*.) You may also see many other messages, depending on the particular VSAM problem. Here are the steps you should follow to correct these problems:

1. Look up any messages you received in Chapter 1, “SMP/E Messages” to see if there is any information that can help you correct the problem. Sometimes, the information in the messages may not be definitive enough to correct the problem. However, save any information you can find; it will be important if you call the IBM Support Center.
2. Use the AMS EXAMINE command to analyze errors in the index and data portion of the cluster.
3. If you obtained an RPL dump, review its contents.

   The RPL fields that you should check first are in the feedback word at offset X'0C' (the RPL function, the RPL return code, and the error code), as follows:

   X'00' ID word
   X'04' Pointer to placeholder
   X'08' ECB information
   X'0C' Feedback word
   X'0C' **Function** <===
   X'0D' Feedback area
   X'0D' **RPL return code** <===
   X'0E' RPL condition

X'0F' **Error code** <===

For an explanation of the RPL return and error codes, see *z/OS DFSMS Macro Instructions for Data Sets*.

4. If none of these steps correct the problem, call the IBM Support Center. They will help you request a dump of the RPL control block to obtain further information.

Handling SMP/E Abends

When SMP/E abends, the SMP/E extended subtask abend exit (ESTAE) routine receives control and does the following processing:

- Issues message GIM43201T to inform you that ESTAE processing is in effect.
- Performs completion processing for processed or partially processed SYSMODs, and issues completion messages.
- Marks SYSMODs that were in process when the abend occurred, but were not completed, with the ERROR status.
- Produces reports that would have been produced by the command being processed had the abend not occurred.
- Passes control to the supervisor for termination processing with no attempt to retry processing.

If you cannot determine the cause of the abend from the reports, the dump, or the LIST output, you can use the VSAM VERIFY command against the SMPCSI cluster to close the cluster and see if you receive any further VSAM error messages. This also helps prevent future problems with the SMPCSI. Then resubmit the job to see if it runs without abending.

If you still cannot correct the abend, contact the IBM Support Center. Reporting SMP/E Problems to IBM, describes the information you should have available when you make the call.

Preventing Problems with the SMP/E Dialogs

You can avoid some problems with the SMP/E dialogs by following these suggestions:

- If you plan to browse or edit JCL created by the dialogs, either allow ISPF to allocate the ISPCTL1 and ISPCTL2 data sets, or allocate them yourself as real (non-virtual I/O) and cataloged.
- If you plan to use OPTIONS entries, use the Administration dialog to define them in the global zone before you specify them on other processing panels. You cannot dynamically create an OPTIONS entry in other SMP/E dialogs (such as the SYSMOD Management dialog) by entering its name on a dialog panel.

Reporting SMP/E Problems to IBM

When you report a problem to IBM, you should provide information about your system and the problem you are experiencing. The service representative will use this information to see if your problem is already known to IBM and if a fix is available. You can expect a prompter problem resolution when you make this information readily available when you call.

Information for All Failure Types

No matter what type of problem you are reporting, you should be ready to provide the IBM Support Center with information about it.

Note: To make sure that all the information you collect is for the same level of your system, you should obtain it by running a single job step. Otherwise, there could be discrepancies. For example, some information might reflect the state your system was in before certain SYSMODs were installed, and some might reflect the state it was in after they were installed.

Be prepared to give the following information to the IBM Support Center:

- Information about your system
  - Details such as the following:

Information about Your System

Product name	IBM SMP/E for z/OS and OS/390.
Component ID	5655–G44.
Release	Version 3 Release 1.
CBPDO level	The ID of the most recent CBPDO tape you have installed.
PUT*xxxx* level	The identifier of the latest service level (PUT*xxxx*) you have installed.
SMP/E service level	See the report headings, output data sets, message GIM20502I or SMP/E dialog options panel for the SMP/E service level, which will appear as SMP/E *nn.nn*. For example, SMP/E 31.*nn* is IBM SMP/E for z/OS and OS/390, V3R1 level *nn*.

  - Whether you recently installed products or service
  - Whether you have made any modifications to the SMP/E code
  - Whether you bypassed any SMP/E service
- Information about the problem:
  - The command being processed and all the operands specified
  - A list of messages issued, including the message ID and the complete message text, as found in SMPRPT, SMPOUT, SMPLOG and (possibly) SYSPRINT.
  - The job entry subsystem (JES) job log and allocation messages for related JES messages
  - Any reports that were produced
  - LIST output from before and after command processing
  - A description of what you think the problem is
  - Whether you ran the job in error more than once with the same results
  - Whether you have recently experienced any other unusual problems
  - Failure keyword and associated documentation, as described in the next sections

Types of Failure Keywords

Failure keywords describe typical problems and can be used to search a database to determine whether your problem is known to IBM.

To determine the failure keyword for your SMP/E problem, read the following list and try to match the symptoms with one or more of those listed for a failure type. If you find a match, follow the procedure for that failure type, as described later in this chapter.

If you do not find a match, report the problem to the IBM Support Center with the information listed at the beginning of this chapter.

- **Abend (ABEND):** SMP/E or an SMP/E subtask has ended abnormally. For example:
  - Message GIM43201T indicates that SMP/E has abended. SMP/E stops running.
  - Message GIM4400*x* indicates that an SMP/E subtask has abended. SMP/E may keep running, but the SYSMOD it was processing fails.

  For information about reporting abends, see “Reporting Abends” on page 298.
- **Loop or Wait (LOOP, WAIT):** SMP/E seems to be looping uncontrollably or to have been suspended unexpectedly. For example, the same group of messages appears over and over.

  Note: SMP/E can appear to loop for a long time, depending on what it is doing and what system it is running on. This does not necessarily mean there is a problem.

  For information about reporting loops and waits, see “Reporting Loops and Waits” on page 300.
- **Message (MSG):** An SMP/E message indicates an error. For example:
  - An SMP/E message is the only symptom of a problem.
  - An SMP/E message is wrong, missing, or incorrectly formatted.
  - An SMP/E message does not adequately explain why it was issued.

  For information about reporting message problems, see “Reporting Message Problems” on page 301.
- **Incorrect Output (INCORROUT):** SMP/E output is missing or contains unexpected information. For example, the results of APPLY processing are different from the results of APPLY CHECK processing for the same SYSMODs.

  For information about reporting incorrect output, see “Reporting Incorrect Output” on page 301.
- **Performance (PERFM):** SMP/E command processing takes an unexpectedly long time.

  For information about reporting performance problems, see “Reporting Performance Problems” on page 301.
- **Documentation (DOC):** A programming problem appears to be caused by incorrect or missing information in an SMP/E manual. For example:
  - Documented descriptions of SMP/E processing do not match actual SMP/E processing.
  - Essential information for installing or using SMP/E is missing.
  - SMP/E information is so vague that it prevents the effective use of SMP/E.
  - Information in one SMP/E manual disagrees with information in another.

Reporting SMP/E Problems to IBM

For information about reporting documentation problems, see “Reporting Documentation Problems” on page 302.

Note: Report a documentation problem only when it causes a programming error. For suggestions, comments, or questions about the documentation, use the Readers’ Comment Form at the back of the publication.

- **Dialogs (DIALOG):** An SMP/E dialog does not work as expected.

  For information about reporting dialog problems, see “Reporting Dialog Problems” on page 303.

Reporting Abends

Recommended Documentation

Abend dump output containing the:
- Abend code
- Program status word (PSW) in effect at the time of the abend
- Name (ID) of the module that the PSW points to
- Address of the failing instruction
- Program interrupt code
- Save area trace
- Contents of the general registers at the time of the abend

Follow this procedure to gather the recommended documentation and report the problem.

Note: For help in locating items in the abend dump, see *z/OS MVS Diagnosis: Tools and Service Aids*, GA22-7589.

1. Obtain the dump output.

   SMP/E produces various types of dumps, depending on the type of abend that has occurred and the dump data sets you have defined. For example:
   - If you have defined SMPSNAP, a dump is taken before any cleanup operations are attempted. The dump reflects the state of SMP/E storage at the time of the abend.
   - If you have defined SYSABEND or SYSUDUMP, a dump is taken by the ESTAE routine after the cleanup operations.
   - If you have defined SMPSNAP and either SYSABEND or SYSUDUMP, dumps are taken before and after cleanup.

   See “Collecting Documentation for SMP/E Problems” on page 306 for additional information about abend dumps.
2. Find the abend code.

   The SMP/E abend messages show the abend code for the failure:
   - GIM43201T is issued when SMP/E abends.
   - GIM4400*x* is issued when the SMP/E subtask that interfaces with utilities abends.
3. Find the program interrupt code in the supervisor request block (SVRB).
4. Find the save area trace.
5. For a hex 0C1 or hex 0C4 abend:
   a. Find the PSW at the time of the abend.

The location of the PSW in dump output depends on the type of dump taken. Generally, it is on the top of the first page of the dump. You can also find the PSW in the recovery termination manager 2 work area (RTM2WA).

Note: For a SNAP dump, you do **not** want the PSW at the time the dump was taken (the first PSW), but rather the next one, which should be the load module PSW.

Make sure the PSW is for the last SMP/E module that was in control, not for an ISPF or TSO module.

b. Find the failing address.

The PSW may contain the address of either the failing instruction or the next instruction that was to be processed when the abend occurred (this depends on the type of abend).

c. Find the ID of the module that contains the address in the PSW. Then find its entry point address and the displacement within the module of the failing instruction.

You can use the save area trace in a formatted dump to find the module ID, compile date, and entry point address.

If you do not have a save area trace, follow these steps to find the information in the dump:

1) To find the module ID, scan the dump output to find the address given in the PSW. Starting at the failing address, scan in descending address order along the right-hand side of the listing until you see an SMP/E module ID. The module ID will be printed in EBCDIC.
2) To find the entry point address of the module, continue scanning the listing in descending address order and look for the 47F0F0*nn* instruction. The address of this instruction is the module's entry point address.
3) To find the displacement of the failing instruction, subtract the entry point address from the address in the PSW.

d. Find the general registers.

You can find the general registers being used at the time of the abend in the save area trace, in RTM2WA, or in the abend SVRB register save area.

e. Determine which register has incorrect information.

Often the failing instruction refers to a general register with an incorrect address, or it points to an incorrect location (such as low-address storage). Use the principles-of-operation manual, the program interruption code from the PSW, and the general registers used in the failing instruction to determine (if possible) the register that contains or points to incorrect data.

You are now ready to report the problem. Go to step 7 on page 300.

6. For other abend codes, see *z/OS MVS System Codes*, SA22-7626.

A complete list of abend codes can be found in *z/OS MVS System Codes*. Each code has an explanation of the documentation required, and problem-determination steps to follow. For example, abends may occur during the processing of supervisor call (SVC) instructions. Parameter lists and register contents passed to SVC routines are documented in *z/OS MVS Diagnosis: Reference*. These manuals may suggest that you get additional information, such as a module name, a return code, a register containing incorrect information, or the name of a system control block containing incorrect parameters.

After making a complete check of these sources, you are ready to report the problem. Continue with the next step.

7. Report the problem to the IBM Support Center.

   After asking for your account name, license number, and other customer identification, the service representative will ask for a brief description of the problem.

Reporting Loops and Waits

Recommended Documentation

- Dumps of the failure
- System generalized trace facility (GTF) trace output
- SMPLOG data

Follow this procedure to gather the recommended documentation and report the problem.

Attention: SMP/E can appear to loop for a long time, depending on what it is doing and what system it is running on. This does not necessarily indicate a problem. Do not cancel an SMP/E job unless you are sure it is looping or in a wait state. Canceling a job could damage the SMPCSI or the other data sets SMP/E uses. If the job seems to be looping or waiting too long, call your IBM Support Center to see if you should cancel it.

1. If you have an online I/O checker, such as resource measurement facility (RMF), use it to see if:
   - No I/O is being performed.
   - The same number of execute channel programs (EXCPs) is used continuously.
2. Check the SMPLOG to see if:
   - No messages are being printed.
   - The same message or group of messages is printed over and over.
3. Slip-trace the branch and link register commands (BALRs) in the system GTF trace, and look for repeated BALRs or repeated strings of BALRs between SMP/E modules.
4. If you have a backup copy of SMP/E, its data sets, and your system data sets, try to recreate the problem. Then, take several dumps of the failure to trace a consistent PSW instruction address or range of addresses. To take the dumps, use a SYSABEND DD card and cancel the job that contains the problem.
5. Report the problem to the IBM Support Center.

   After asking for your account name, license number, and other customer identification, the service representative will ask for a brief description of the problem.

Reporting Message Problems

Recommended Documentation

- Message number
- Issuing module and offset where issued
- Sequence of events leading to the problem
- SMPLOG output with the complete message text
- JES job log
- JES allocation messages

Follow this procedure to gather the recommended documentation and report the problem:

1. Look up the message in Chapter 1, “SMP/E Messages” and follow the recommended procedures.
2. Identify the issuing module.

   Rerun the job, if possible, and place the DEBUG MSGMODID command before the command that is causing the problem. This will prefix each SMP/E message with the name of the module that issued the message and the offset where the message was issued. See “Tracing SMP/E Messages” on page 309 for more information about the DEBUG MSGMODID command.
3. Check the JES job log for any messages associated with the SMP/E job.
4. Report the problem to the IBM Support Center.

   After asking for your account name, license number, and other customer identification, the service representative will ask for a brief description of the problem.

Reporting Incorrect Output

Recommended Documentation

- Documentation that led you to expect certain output
- Output that shows a discrepancy

Follow this procedure to gather the recommended documentation and report the problem:

1. Save a copy of the documentation that led you to expect certain output, such as output from APPLY CHECK or ACCEPT CHECK.
2. Save a copy of the output that shows any discrepancies.
3. Report the problem to the IBM Support Center.

   After asking for your account name, license number, and other customer identification, the service representative will ask for a brief description of the problem.

Reporting Performance Problems

Recommended Documentation

- Record of the unexpected performance
- Record of performance under another level of SMP/E

Reporting SMP/E Problems to IBM

Follow this procedure to gather the recommended documentation and report the problem.

Note: You should measure and discuss performance in terms of EXCP counts and CPU time rather than wall-clock time, which can be affected by conditions outside the control of SMP/E.

1. Check to see whether any programs that do a lot of I/O operations, such as Information Management System (IMS), were running when you noticed the degraded performance.
2. Check to see whether you installed any engineering changes or service to SMP/E or other products that may have degraded system performance.
3. If the problem does not seem to be caused by another program or an engineering change, rerun the job under your current level of SMP/E.
4. Run the same job under a previous PTF level or release of SMP/E to demonstrate a difference in performance. Use the same environment, including the same operating system and DASD configuration.
5. If there is a difference in performance, report the problem to the IBM Support Center.

   After asking for your account name, license number, and other customer identification, the service representative will ask for a brief description of the problem.

Reporting Documentation Problems

Note: Report a documentation problem only when it causes a programming problem. For suggestions, comments, or questions about the documentation, use the Readers' Comment Form at the back of the publication.

Recommended Documentation

- Identification of the incorrect information
- Form number of the publication

Follow this procedure to gather the recommended documentation and report the problem:

1. Identify the incorrect information.

   Note the name of the command, operand, or procedure that is incorrectly explained in the documentation (for example, ACCEPT FORFMID).
2. Record the form number.

   Note the form number of the SMP/E publication that contains the error. Omit the dashes, but include the revision level. For example, you would report the form number of this publication as GA22177002.
3. Report the problem to the IBM Support Center.

   After asking for your account name, license number, and other customer identification, the service representative will ask for a brief description of the problem.

Reporting Dialog Problems

Recommended Documentation
- Job to be accomplished
- ISPF log data set
- Sequence of dialog panels
- Abend dump (if an abend occurred)
- Copy of the panel in error
- ISPF version and service level

Follow this procedure to gather the recommended documentation and report the problem:

1. Describe the job you intended to do with the dialogs.
2. Record the sequence of dialog panels. Enter **panelid** on the command line to have the ID of each panel appear in the upper left corner of the screen. Recreate the problem, and record the ID of each panel you go through and the data you enter on each panel. One way to get this information is to print each screen as you go through the dialog. This writes a copy of each screen to your LIST data set. You can then print that data set to obtain a record of the panels and what you entered. For an example of what to report, see the panel sequence in “Sample Dialog Problem”.
3. Print the panel on which the error occurred. Note the panel ID, and print that member from the appropriate data set. The name for the target library for the SMP/E dialog panels depends on which feature of SMP/E you are using:
   - English feature–GIM.SGIMPENU
   - Japanese feature–GIM.SGIMPJPN
4. Print the ISPF log data set. This log contains any dialog error messages that were issued. If you usually delete this data set, change the default to save the log. Then, recreate the problem and print the log.
5. If an abend occurred (such as 0C4), obtain an abend dump. Either add a SYSABEND DD statement to the logon procedure for the TSO session, or allocate a SYSABEND DD statement. Then recreate the problem. When the abend message appears, press ENTER to take the abend dump.
6. If possible, try to run an SMP/E batch job using the LIST command to access the same data.
7. Report the problem to the IBM Support Center.

   After asking for your account name, license number, and other customer identification, the service representative will ask for a brief description of the problem.

Sample Dialog Problem

The following figures are a sample sequence of SMP/E dialog panels. Following the figures is the information you would report if a problem were to occur when you were using the panels.

When you invoke ISPF, there are modes of processing you can specify to obtain additional information that can help you diagnose dialog problems. For example, you could specify TEST when you call ISPF. If an error occurs while you are using a dialog panel, you can request the help panel. This panel will display the ID of the help panel, the ID of the preceding dialog or help panel, and the ID of the error message that was issued.

For more information on the ISPF TEST mode, see *z/OS ISPF Dialog Developer's Guide and Reference*.

For the following figures, assume that you had wanted to display a particular SYSMOD entry in the global zone, but you did not get a panel showing the SYSMOD entry. When you recreate the problem, you would go through the following screens:

```
GIM@PRIM                   SMP/E PRIMARY OPTION MENU            SMP/E 31.nn
===> 3

    1  ADMINISTRATION     - Administer the SMPCSI contents
    2  SYSMOD MANAGEMENT  - Receive SYSMODs and HOLDDATA
                            and install SYSMODs
    3  QUERY              - Display SMPCSI information
    4  COMMAND GENERATION - Generate SMP/E commands
    5  RECEIVE            - Receive SYSMODs, HOLDDATA and
                            support information
    6  MIGRATION ASSISTANT- Generate Planning and Migration Reports
    D  DESCRIBE           - An overview of the dialogs
    T  TUTORIAL           - Details on using the dialogs

Specify the name of the CSI that contains the global zone:
   SMPCSI DATA SET   ===> 'DESCTEST.VSAM.CSI'
(Leave blank for a list of SMPCSI data set names.)

Specify YES to have DD statements for SYSOUT and temporary
data sets generated. Specify NO, to use DDDEFs.
   Generate DD statements   ===> NO

5694-A01(C) COPYRIGHT IBM CORP 1982, 2002
LICENSED MATERIAL - PROGRAM PROPERTY OF IBM
```

Figure 1. SMP/E Primary Option Menu

```
GIMQUPO                        QUERY SELECTION MENU
===> 1

   1  CSI QUERY           - Display SMPCSI entries
   2  CROSS-ZONE QUERY    - Display status of an entry in
                            all zones
   3  SOURCEID QUERY      - Display SOURCEIDs for specified zone


   D  DESCRIBE            - Overview of using QUERY

   T  TUTORIAL            - Information on using QUERY

To return to the SMP/E primary option menu, enter  END .


5694-A01(C) COPYRIGHT IBM CORP 1982, 2002
```

Figure 2. Query Selection Menu

```
GIMQU1PO                        CSI QUERY
===>

Specify the zone, entry type, and name to be queried:

      ZONE NAME      ===> global       Name of the zone to be queried.
                                       To display a list of all zones,
                                       leave blank

      ENTRY TYPE     ===> sysmod       Entry type to be queried.
                                       To display a list of all valid
                                       entry types, leave ENTRY TYPE
                                       and ENTRY NAME blank

      ENTRY NAME     ===>              Entry name to be queried.
                                       To display a list of all entries
                                       for the selected zone and entry
                                       type, leave blank


To return to the Query selection menu, enter  END .
```

Figure 3. CSI Query Panel

```
GIMQUSEA                CSI QUERY - SELECT ENTRY
===>                                                        SCROLL ===>

 Select one entry to query from global zone global  :

 S   NAME         ACTION
 s   F000000
     P000001
     P000002
```

Figure 4. CSI Query – Select Entry Panel

This is the information you would report for this panel sequence:

- **Job to be accomplished:**

  Display SYSMOD entry F000000 in the global zone.
- **Sequence of dialog panels and data entered:**

Panel	Data Entered
GIM@PRIM	Command line: **3** to select QUERY SMPCSI DATA SET: **'DESCTEST.VSAM.CSI'** Generate DD statements: left as **NO**
GIMQUPO	Command line: **1** to select CSI QUERY
GIMQU1PO	ZONE NAME: **global** ENTRY TYPE: **sysmod**
GIMQUSEA	Line command: **s** to select SYSMOD entry F000000.

- **Problem:**

  Panel GIMQIT26 was not displayed after SYSMOD entry F000000 was selected from the global zone.

Collecting Documentation for SMP/E Problems

This chapter describes SMP/E dumps, commands, and reports that provide helpful diagnostic information.

Obtaining Dumps When Abends Occur

If an abend occurs while you are processing a command, define an SMPSNAP data set and rerun the command to get a dump. A diagnostic dump of SMP/E storage is automatically written to the SMPSNAP data set when:

- An unexpected VSAM error occurs (dump ID 083).

  SMP/E issues message GIM443*xx* or GIM442*xx* (or both) to describe the condition that caused the dump.
- A nonrecoverable SMP/E error occurs (dump ID 084).

  This dump is made upon entry to the SMP/E ESTAE routine for nonrecoverable abends. In addition to the SMPSNAP data set, you may also define a SYSABEND or SYSUDUMP data set.
  - If SMPSNAP was defined, a dump is taken before any cleanup operations are attempted. This dump reflects the state of SMP/E storage at the time of the abend.
  - If SYSABEND or SYSUDUMP was defined, a dump is taken after the cleanup operations by the ESTAE routine.
  - If SMPSNAP and either SYSABEND or SYSUDUMP were defined, dumps are taken before and after cleanup.

If an error occurs while SMP/E is trying to access a VSAM data set, message GIM27901S informs you that a VSAM error has occurred. The message contains a VPLFUNCT value, which may be specified on the DEBUG DUMPRPL command to request a dump of the VSAM RPL control block. You may also see many other messages; look these up in Chapter 1, “SMP/E Messages” to see if you can correct the problem.

If you cannot correct the problem after checking the messages you received from running the job, call the IBM Support Center. They will help you obtain a dump of the RPL control block, which will provide them with further information.

Using the DEBUG Command

The DEBUG command helps you to collect diagnostic information. If you are experiencing a problem with an SMP/E command, the IBM Support Center may ask you to use the SMP/E DEBUG command to gather more information about the problem. The DEBUG command is used with other SMP/E commands to:

- Obtain dumps of SMP/E storage
- Obtain dumps of VSAM RPL control blocks
- Trace the source of SMP/E messages

These uses of the DEBUG command are described later in this chapter.

Obtaining DEBUG Dumps

Since most SMP/E problems are not serious enough to cause abends, you do not always have an abend dump to help you find the source of the problem. To obtain this information, you can use the DEBUG command to request dumps of SMP/E control blocks, storage, and work areas. There are several types of dumps you can get, depending on whether you want SMP/E to take the dump:

- When specific messages are issued
- When VSAM errors occur during an attempt to access a CSI data set
- At specific points in SMP/E processing

When Specific Messages Are Issued: SMP/E storage and work areas can be dumped whenever specific SMP/E messages are issued. To obtain this dump, use the DEBUG DUMPMSG and DEBUG DUMPOFF DUMPMSG commands with the command where the problem is occurring. For the dump output, you must define the SMPSNAP data set with either a DD statement or a DDDEF entry.

- To request a dump of SMP/E storage and work areas, enter:

  DEBUG DUMPMSG(*msg-id***)[,(***msg-id***)...]**

  msg-id
  is the first 8 characters of an SMP/E message ID (such as GIM62801). You must specify at least one message ID.
- To stop these dumps, enter:

  DEBUG DUMPOFF DUMPMSG.

  DUMPOFF DUMPMSG
  stops dumps for all messages that were specified on a previous DEBUG DUMPMSG command. You cannot stop dumps for a particular message ID.

  Note: DEBUG DUMPOFF without the DUMPMSG operand stops **all** DEBUG dumps, including those requested with DEBUG DUMPMSG and those requested with DEBUG DUMPON and DEBUG DUMPRPL, which are also described in this chapter. Use this command only if you want to stop **all** DEBUG dumping.

For more information on the DEBUG command, see the *SMP/E Commands* manual.

When VSAM Errors Occur during Attempts to Access a CSI Data Set: VSAM RPL control blocks can be dumped when an error occurs during an attempt to access a CSI data set. This RPL dump can be used to analyze the cause of the error.

Note: Because VPLFUNCT values refer to types of internal SMP/E processing, they can be used only in conjunction with the IBM Support Center and are not described in this publication.

To obtain this dump, use the DEBUG DUMPRPL and DEBUG DUMPOFF commands. For the dump output, you must define the SMPDEBUG data set with either a DD statement or a DDDEF entry.

- To request a dump of the VSAM RPL control block, enter the following command after the SET command and before the command experiencing the error:

  DEBUG DUMPRPL(*vplfunct***[,***vplfunct***]...) .**

  vplfunct
  is a VPLFUNCT value supplied by IBM.

  Note: You can combine VPLFUNCT values and dump points on the same DEBUG DUMPOFF command.
- To stop specific dumps, enter the following command after the command experiencing the error:

  DEBUG DUMPOFF(*vplfunct***[,***vplfunct***]...) .**

vplfunct
is a VPLFUNCT value specified on a previous DEBUG DUMPRPL command.

- To stop all DEBUG dumps, enter the following command after the command experiencing the error:

  DEBUG DUMPOFF .

  Note: This stops **all** DEBUG dumps, including those requested with DEBUG DUMPMSG, which is described in "When Specific Messages Are Issued" on page 307. Use this command only if you want to stop **all** DEBUG dumping.

If you specify DUMPON or DUMPRPL and DUMPOFF on the same DEBUG command, these operands are processed in the order in which they occur. If you specify the same dump ID on both operands, the last specification is used.

The DEBUG commands cause an RPL dump to be written to the SMPDEBUG data set.

The RPL fields that you should check first are in the feedback word at offset X'0C' (the RPL function, the RPL return code, and the error code), as follows:

```
X'00'   ID word
X'04'   Pointer to placeholder
X'08'   ECB information
X'0C'   Feedback word
        X'0C'   Function <===
        X'0D'   Feedback area
                X'0D'   RPL return code <===
                X'0E'   RPL condition
                X'0F'   Error code <===
```

For an explanation of the RPL return and error codes, see *z/OS DFSMS Macro Instructions for Data Sets*.

For more information on the DEBUG command, see *SMP/E Commands*.

At a Specific Point in SMP/E Processing: SMP/E control blocks and data areas can be dumped and formatted for various dump points within SMP/E.

Note: Because the dump points correspond to specific areas of SMP/E code, they can be used only in conjunction with the IBM Support Center and are not described in this publication.

To obtain this dump, use the DEBUG DUMPON and DEBUG DUMPOFF commands with the command in which the problem is occurring. You can request a formatted dump and give it a title, or you can request a SNAP dump without a title. For a formatted dump, you must define the SMPDEBUG data set with either a DD statement or a DDDEF entry. For a SNAP dump, you must define the SMPSNAP data set with either a DD statement or a DDDEF entry.

There are various ways to start and stop these dumps, depending on which dump you are interested in:

- To request a formatted dump, enter:

  DEBUG DUMPON((*dump-id***[,***dump-title* **])[,(***dump-id***[,** *dump-title***])...]).**

dump-id
is the name of the dump point provided by IBM. You must specify at least one dump point.

dump-title
is an optional header page title that you can give a formatted dump. The dump title may have up to 100 characters. If it contains parentheses, right and left parentheses must be in matched pairs.

- To request a SNAP dump, enter:

  DEBUG DUMPON((*dump-id***)[,(***dump-id***)]... SNAP) .**

  dump-id
  is the name of the dump point provided by IBM. You must specify at least one dump point.

  Note: You cannot specify a dump title for a SNAP dump.

- To stop specific dumps, enter:

  DEBUG DUMPOFF(*dump-id***[,***dump-id***]...) .**

  dump-id
  is the name of a dump point that was specified on a previous DEBUG DUMPON command.

  Note: You can combine dump points and VPLFUNCT values on the same DEBUG DUMPOFF command.

- To stop all DEBUG dumps, enter:

  DEBUG DUMPOFF .

  Note: This stops **all** DEBUG dumps, including those requested with DEBUG DUMPMSG, which is also described in this chapter. Use this command only if you want to stop **all** DEBUG dumping.

If you specify DUMPON or DUMPRPL and DUMPOFF on the same DEBUG command, these operands are processed in the order in which they occur. If you specify the same dump ID on both operands, the last specification is used.

For more information on the DEBUG command, see *SMP/E Commands*.

Tracing SMP/E Messages

By use of the DEBUG command, you can determine which module issued a particular message and the offset in that module in which the message was issued.

- To trace the source of SMP/E messages, enter:

  DEBUG MSGMODID(ON) .

  This prefixes all SMP/E messages with the following string:

  `@module+X'offset'`

  module
  is the name of the SMP/E module (without the GIM prefix) that issued the message.

  offset
  is the hexadecimal offset into the module where the message was issued.

- To stop SMP/E message tracing, enter:

  DEBUG MSGMODID(OFF) .

  Messages are no longer prefixed with the name and offset of the module.

Appendix. Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS enable users to:

- Use assistive technologies such as screen-readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size

Using assistive technologies

Assistive technology products, such as screen-readers, function with the user interfaces found in z/OS. Consult the assistive technology documentation for specific information when using it to access z/OS interfaces.

Keyboard navigation of the user interface

Users can access z/OS user interfaces using TSO/E or ISPF. Refer to *z/OS TSO/E Primer*, *z/OS TSO/E User's Guide*, and *z/OS ISPF User's Guide Volume I* for information about accessing TSO/E and ISPF interfaces. These guides describe how to use TSO/E and ISPF, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

© Copyright IBM Corp. 1986, 2002

Notices

This information was developed for products and services offered in the USA.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described 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:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

© Copyright IBM Corp. 1986, 2002

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States, other countries, or both:

BookManager
CICS
DATABASE 2
DB2
DFSMS
DFSMS/MVS
DFSMSdfp
DFSMSdss
IBM
IBMLink
IMS
OS/2
OS/390
RACF
RETAIN
SP
System/390
SystemPac
VTAM
z/OS

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Glossary

This glossary defines technical terms and abbreviations used in SMP/E documentation. If you do not find the term you are looking for, refer to the index of the appropriate SMP/E manual or view *IBM Glossary of Computing Terms*, located at:
http://www.ibm.com/networking/nsg/nsgmain.htm

Sequence of Entries: For clarity and consistency of style, this glossary arranges the entries alphabetically on a letter-by-letter basis. In other words, only the letters of the alphabet are used to determine sequence; special characters and spaces between words are ignored.

Organization of Entries: Each entry consists of a single-word or multiple-word term or the abbreviation or acronym for a term, followed by a commentary. A commentary includes one or more items (definitions or references) and is organized as follows:

1. An item number, if the commentary contains two or more items.
2. A usage label, indicating the area of application of the term, for example, “In programming,” or “In SMP/E.” Absence of a usage label implies that the term is generally applicable to SMP/E, to IBM, or to data processing.
3. A descriptive phrase, stating the basic meaning of the term. The descriptive phrase is assumed to be preceded by “the term is defined as ...” The part of speech being defined is indicated by the opening words of the descriptive phrase: “To ...” indicates a verb, and “Pertaining to ...” indicates a modifier. Any other wording indicates a noun or noun phrase.
4. Annotative sentences, providing additional or explanatory information.
5. References, directing the reader to other entries or items in the dictionary.

References: The following cross-references are used in this glossary:

Contrast with. This refers to a term that has an opposed or substantively different meaning.
Synonym for. This indicates that the term has the same meaning as a preferred term, which is defined in its proper place in the glossary.
Synonymous with. This is a backward reference from a defined term to all other terms that have the same meaning.
See. This refers you to multiple-word terms that have the same last word.
See also. This refers the reader to related terms that have a related, but not synonymous, meaning.
Deprecated term for or **Deprecated abbreviation for**. This indicates that the term or abbreviation should not be used. It refers to a preferred term, which is defined in its proper place in the glossary.

Selection of Terms: A term is a word or group of words to be defined. In this glossary, the singular form of the noun and the infinitive form of the verb are the terms most often selected to be defined. If the term may be abbreviated, the abbreviation is given in parentheses immediately following the term. The abbreviation is also defined in its proper place in the glossary.

A

ACCEPT. The SMP/E command used to install SYSMODs in the distribution libraries.

accept. In SMP/E, to install SYSMODs in the distribution libraries. This is done with the ACCEPT command.

accepted SYSMOD. A SYSMOD that has been successfully installed by the SMP/E ACCEPT command. Accepted SYSMODs do not have the ERROR flag set and are found as SYSMOD entries in the distribution zone.

Access method services (AMS). The system utility program used to support VSAM data sets.

AMS. Access method services.

APAR. Authorized program analysis report.

APAR fix. A temporary correction of a defect in an IBM system control program or licensed program that affects a specific user. An APAR fix is usually replaced later by a permanent correction called a PTF. APAR fixes are identified to SMP/E by the ++APAR statement.

applied SYSMOD. A SYSMOD that has been successfully processed by the SMP/E APPLY command. Applied SYSMODs do not have the ERROR flag set and are found as SYSMOD entries in the target zone.

© Copyright IBM Corp. 1986, 2002

APPLY. The SMP/E command used to install SYSMODs in the target libraries.

apply. In SMP/E, to install SYSMODs in the target libraries. This is done with the APPLY command.

archive. An archive is a network transportable file containing two files; a file attribute file (FAF) and and the data file that the FAF describes.

ASSEM entry. An SMP/E entry containing assembler statements that can be assembled to create an object module.

authorized program analysis report (APAR). A report of a problem caused by a suspected defect in a current unaltered release of a program. The correction is called an APAR fix.

B

BACKUP entries. A collection of SMP/E target zone entries that are copied into the SMPSCDS data set during APPLY processing before they are updated by inline JCLIN, a ++MOVE MCS, or a ++RENAME MCS, or before they are deleted by an element MCS with the DELETE operand.

BACKUP entries consist of:

- A SYSMOD entry indicating what entries have been added, deleted, or updated
- ASSEM entries for updated target zone ASSEM entries
- LMOD entries for updated target zone LMOD entries
- MAC entries for updated or deleted target zone MAC entries
- MOD entries for updated or deleted target zone MOD entries
- SRC entries for updated or deleted target zone SRC entries
- Data element entries for deleted target zone data element entries
- DLIB entries for updated target zone DLIB entries

BALR. Branch and link register commands.

base function. A SYSMOD defining elements of the base system or other products that were not previously present in the target libraries. Base functions are identified to SMP/E by the ++FUNCTION statement. SMP/E is an example of a base function.

base level system. The level of the target system modules, macros, source, and DLIBs created by system generation, to which function and service modifications are applicable.

base version of a load module. Some load modules include modules both explicitly (through INCLUDE statements) and implicitly (through a SYSLIB allocation). The base version of such a load module includes only the explicitly-defined modules for the load module. It is maintained by SMP/E in the SMPLTS data set. The base version of a load module is used with the SYSLIB allocation as input to the link-edit utility in order to build the load module in its target libraries.

binder. A program that processes the output of language translators and compilers into an executable program (load module). It is part of the DFSMS/MVS element of z/OS.

bypass. In SMP/E, to circumvent errors that would otherwise cause SYSMOD processing to fail. This is done by using the BYPASS operand on an SMP/E command.

C

causer SYSMOD. A SYSMOD identified by root cause analysis to be at the base of errors that caused other SYSMODs to fail. See *root cause analysis*.

CBPDO. MVS Custom-Built Product Delivery Offering.

CICS. Customer Information Control System.

CLEANUP. The SMP/E command used to delete entries from the SMPMTS, SMPSTS, and SMPSCDS data sets after a SYSMOD has been accepted into the related distribution zone.

CNTL. See *SMPCNTL*.

coexisting functions. Functions that meet these requirements: (1) they are for the same system or subsystem and have the same SREL value, (2) they do not delete or supersede each other and are not negative prerequisites, and (3) if they are base functions, they are for different products. See also *conditionally coexisting functions* and *unconditionally coexisting functions*.

comment statements. Special control statements that are coded as JCL comments and which are used to convey information to SMP/E during JCLIN processing.

conditional requisites. Requisites defined by an ++IF statement. These are requisites that must be installed if the functions specified on the ++IF statements are installed.

conditionally coexisting functions. Functions that coexist but do not have to be in the same zone.

consolidated software inventory. See *SMPCSI*.

corequisite SYSMODs. SYSMODs each of which can be installed properly only if the other is present. Corequisites are defined by the REQ operand on the ++VER statement.

corrective service. Any SYSMOD used to selectively fix a system problem. Generally, corrective service refers to APAR fixes.

cross-zone. (1) A target zone other than the set-to zone that defines a load module containing modules from set-to zone. Also called a *TIEDTO zone.* The modules were added to the load module through the SMP/E LINK command. The relationship between the cross-zone and the set-to zone is established through the TIEDTO subentry in their zone definition entries. See also *set-to zone* and *TIEDTO relationship*. (2) Pertaining to relationships between zones, especially as a result of conditional requisites (++IF statements) or LINK processing. See also *cross-zone requisite, cross-zone load module,* and *cross-zone module*.

cross-zone load module. A load module containing modules from a different zone as a result of LINK processing.

cross-zone module. A module included in a load module from a different zone as a result of LINK processing.

cross-zone requisite. A conditional requisite that must be installed in one zone because of another SYSMOD that is installed in a different zone. The REPORT command can be used to check information saved from ++IF statements and determine where any cross-zone requisites should be installed.

CSI. Consolidated software inventory data set. See *SMPCSI*.

D

data element. An element that is not a macro, module, or source—for example, a dialog panel or sample code.

DDDEF entry. An SMP/E entry containing the information SMP/E needs in order to dynamically allocate a particular data set.

DEBUG. The SMP/E command used to obtain additional information for problem determination—for example, to trace messages or take dumps.

debug. In SMP/E, to obtain additional information for problem determination—for example, to trace messages or take dumps. This is done with the DEBUG command.

definition side deck. A file in the hierarchical file system, a member of a partitioned data set, or a sequential data set that contains binder IMPORT control statements.

deleted function. In SMP/E, a function that was removed from the system when another function was installed. This is indicated by the DELBY subentry in the SYSMOD entry for the deleted function. See also *explicitly deleted function* and *implicitly deleted function*.

deleting function. A function that removes other functions from the system. This is done by specifying them on the DELETE operand of the ++VER statement.

dependent function. A function that introduces new elements or redefines elements of the base level system or other products. A dependent function cannot exist without a base function. Dependent functions are identified to SMP/E by the ++FUNCTION statement.

dialog. The interactive support provided by SMP/E through ISPF. Instead of entering specific commands and operands, you can use panels to specify the desired processing.

distribution library (DLIB). A library that contains the master copy of all the elements in a system. A distribution library can be used to create or back up a target library.

distribution zone. In SMP/E, a group of records in a CSI data set that describes the SYSMODs and elements in a distribution library.

DLIB. Distribution library.

DLIB entry. An SMP/E entry describing a distribution library that has been totally copied into a target library.

DLIBZONE entry. An SMP/E entry containing information used by SMP/E to process a specific distribution zone and the associated distribution libraries.

DLL. Dynamic link library

E

EC. Engineering change.

element. In SMP/E, part of a product, such as a macro, module, dialog panel, or sample code.

element MCS. An MCS used to replace or update an element.

element selection. The process by which SMP/E chooses the appropriate changes for an element affected by several SYSMODs being installed at the same time.

entry. In SMP/E, a collection of records in a CSI data set. An entry can be created, updated, or deleted by use of UCL statements.

environment. The functions (FMIDs) installed on a particular system or subsystem (SREL).

ERROR indicator. In SMP/E, an indicator in a target or distribution zone SYSMOD entry that shows that

SYSMOD processing failed. The ERROR indicator is set before SMP/E updates any libraries and is reset if processing is successful. If processing fails, it remains set to show that an error occurred.

ESO. Expanded service options.

exception SYSMOD. A SYSMOD that is in error or that requires special processing before it can be installed. ++HOLD and ++RELEASE statements identify exception SYSMODs.

EXCP. Execute channel programs.

expanded service options (ESO). A tape that includes preventive service PTFs. Where available, it replaces PUTs as the vehicle for delivering preventive service. An ESO contains PTFs and ++ASSIGN statements assigning source IDs for the PTFs. In the United States, this tape is available from the IBM Support Center and can be ordered either by subscription or as needed.

explicitly deleted function. A function deleted because it was specified on the DELETE operand of a ++VER statement in another SYSMOD.

exported zone. A zone copied into a sequential data set by use of the SMP/E ZONEEXPORT command.

external HOLDDATA. ++HOLD statements contained in SMPHOLD. Contrast with *internal HOLDDATA*.

F

FAF. file attribute file.

FE. Field engineering.

feature. See *dependent function*.

file attribute file. A file attribute file (FAF) is a file containing control statements that describe the attributes of the file contained in an SMP/E network transportable archive. The FAF is contained within the archive information about how the file was created.

File Transfer Protocol. File Transfer Protocol (FTP) is a protocol that defines the interactions necessary between a client and server to facilitate the exchange of binary and textual data files.

firewall. A firewall is an intermediate server that functions to isolate a secure network from an insecure network.

FTP. File Transfer Protocol.

FMID. Function modification identifier.

FMIDSET. A group of FMIDs to be used in processing an SMP/E command—for example, to indicate that SYSMODs applicable to certain functions should be installed.

FMIDSET entry. An SMP/E entry defining an FMIDSET.

function. In SMP/E, a product (such as a system component or licensed program) that can be installed in a user's system if desired. Functions are identified to SMP/E by the ++FUNCTION statement. Each function must have a unique FMID.

function modification identifier (FMID). The SYSMOD ID of a function SYSMOD. It identifies the function that currently owns a given element.

functionally higher SYSMOD. A SYSMOD that uses the function contained in an earlier SYSMOD (called the *functionally lower SYSMOD*) and contains additional functions as well.

functionally lower SYSMOD. A SYSMOD whose function is also contained in a later SYSMOD (called the *functionally higher SYSMOD*).

G

GENASM. A subentry in the MAC entry that lists the ASSEM or SRC entries that must be assembled if the macro is replaced or updated.

GENERATE. The SMP/E command used to create a job stream that builds a set of target libraries from a set of distribution libraries.

generate. In SMP/E, to create a job stream that builds a set of target libraries from a set of distribution libraries. This is done with the GENERATE command.

GIMUNZIP. An SMP/E service routine used to extract files contained in network transportable packages that were built using GIMZIP.

GIMZIP. An SMP/E service routine used to produce network transportable packages.

global zone. A group of records in a CSI data set used to record information about SYSMODs received for a particular system. The global zone also contains information that (1) enables SMP/E to access target and distribution zones in that system, and (2) enables you to tailor aspects of SMP/E processing.

GLOBALZONE entry. An SMP/E entry containing information that SMP/E uses to process the global zone, the associated target and distribution zones, and SMPPTS.

GTF. Generalized trace facility.

H

hashing. An operation that uses a one-way (irreversible) function on data, usually to reduce the length of the data and to provide a verifiable authentication value (checksum) for the hashed data.

header MCS. An ++APAR, ++FUNCTION, ++PTF, or ++USERMOD statement. The header MCS indicates the type of SYSMOD.

HFS. Hierarchical file system.

hierarchical file system element. An element that has a hierarchical file system as its “target library”.

hierarchy. In SMP/E, the top-down structure of function and service SYSMODs, in which each SYSMOD is dependent on the one above it.

higher functional level. An element version that contains all the functions of all other relevant versions of that element.

HOLDDATA. In SMP/E, MCSs used to indicate that certain SYSMODs contain errors or require special processing before they can be installed. ++HOLD and ++RELEASE statements are used to define HOLDDATA. SYSMODs affected by HOLDDATA are called *exception SYSMODs*.

HOLDDATA entry. An SMP/E entry containing ++HOLD statements that either were received from SMPHOLD (external HOLDDATA) or were within a SYSMOD that was received (internal HOLDDATA).

I

ICSF. Integrated Cryptographic Service Facility.

IFREQ. A conditional requisite. Conditional requisites are specified on the REQ operand of the ++IF statement.

IMASPZAP. The system utility program used to install superzaps, which are changes for modules, load modules, or CSECTs within modules.

implicitly deleted function. A function deleted because of its dependency on an explicitly deleted function that is specified on the DELETE operand of the ++VER statement.

imported zone. A zone copied from a sequential data set into another zone by use of the SMP/E ZONEIMPORT command.

IMS. Information Management System.

IMSGEN. IMS generation.

indicator. See *subentry indicator*.

in effect. Having control over SMP/E processing. For example, an OPTIONS entry is in effect if (1) it is specified on the SET command or (2) it is defined as the default OPTIONS entry for the set-to zone.

inline data. Information (such as utility control statements or code for an element) that is packaged directly after the associated MCS, rather than in a separate file or data set.

inline JCLIN. The JCL statements associated with a ++JCLIN statement. Inline JCLIN may immediately follow the ++JCLIN statement, or it may be in the RELFILE or TXLIB data set pointed to by the ++JCLIN statement. Inline JCLIN is used to update the target zone when a SYSMOD is applied, or the distribution zone when a SYSMOD is accepted. Contrast with *JCLIN input*.

inner macro. A macro invoked by another macro. In particular, inner macros are those that SMP/E does not detect during JCLIN processing of assembler job steps.

install. In SMP/E, to apply a SYSMOD to the target libraries or to accept a SYSMOD into the distribution libraries.

internal HOLDDATA. ++HOLD statements contained within a SYSMOD. Contrast with *external HOLDDATA*.

I/O. Input or output.

IOGEN. Input/output device generation.

IPL. Initial program load.

ISMD. IBM Software Manufacturing and Delivery (formerly called *PID*).

ISPF. Interactive System Productivity Facility.

ISPF/PDF. Interactive System Productivity Facility/Program Development Facility.

IVP. Installation verification procedure.

J

JCL. Job control language.

JCLIN. (1) The SMP/E command used to process data from SMPJCLIN. (2) The ++JCLIN statement, which is associated with JCLIN data that is included in a SYSMOD. (3) SMPJCLIN. See *SMPJCLIN*.

See also *inline JCLIN* and *JCLIN data*.

JCLIN data. The JCL statements associated with the ++JCLIN statement or saved in the SMPJCLIN data set. They are used by SMP/E to update the target zone when the SYSMOD is applied. Optionally, SMP/E can use JCLIN data to update the distribution zone when the SYSMOD is accepted.

JCLIN input. The JCL statements contained in SMPJCLIN and used as input for the JCLIN command. Contrast with *inline JCLIN*.

job control language (JCL). A problem-oriented language designed to express statements in a job that are used to identify the job or describe its requirements to an operating system.

L

licensed program. A program that performs a function for the user, and usually interacts with and relies upon the system control program or some other IBM-provided control program. Generally, a licensed program is a software package that can be ordered from the program libraries, such as IBM Software Distribution (ISMD). IMS and CICS are examples of licensed programs.

link library (LKLIB). A data set containing link-edited object modules.

LIST. The SMP/E command used to display entries in SMP/E data sets.

list. In SMP/E, to display entries in SMP/E data sets. This is done with the LIST command.

LKLIB. Link library.

LMOD. In SMP/E, an abbreviation for load module.

LMOD entry. An SMP/E entry containing all the information needed to replace or update a given load module.

load module. A computer program in a form suitable for loading into main storage for execution. It is usually the output of a link-edit utility.

LOG. (1) The SMP/E command used to write user-supplied information to the SMPLOG data set. (2) The SMPLOG data set. See *SMPLOG*.

lower functional level. An element version that is contained in a later element version.

M

MAC. The SMP/E entry or MCS that describes a macro.

macro. An instruction in a source language that is to be replaced by a defined sequence of instructions in the same source language.

MACUPD. The SMP/E MCS used to update a macro.

mass-mode processing. In SMP/E, processing that includes all eligible SYSMODs, regardless of whether they were individually selected.

master CSI. The CSI data set that contains the global zone.

MCS. Modification control statement.

MCS entry. An SMP/E entry containing a copy of a SYSMOD exactly as it was received from SMPPTFIN. MCS entries are in SMPPTS, which is used to store SYSMODs.

MOD. The SMP/E entry or MCS that describes an object module or a single-module load module.

MODID. Modification identifier.

modification. In SMP/E, an alteration or correction to a system control program, licensed program, or user program. Synonymous with *system modification* (SYSMOD).

modification control statement (MCS). An SMP/E control statement used to package a SYSMOD. MCSs describe the elements of a program and the relationships that program has with other programs that may be installed on the same system.

modification identifier (MODID). A list of SYSMOD IDs, including the last SYSMOD that totally replaced the element (RMID), any subsequent partial updates to the element (UMIDs), and the function that owns the element (FMID). MODIDs are contained in element entries.

modification level. A distribution of all temporary fixes that have been issued since the previous modification level. A change in modification level does not add new functions or change the programming support category of the release to which it applies. Contrast with *release* and *version*.

Note: Whenever a new release of a program is shipped, the modification level is set to 0. When the release is reshipped with the accumulated services changes incorporated, the modification level is incremented by 1.

module. Synonym for *object module* or *single-module load module*.

MTS. Macro temporary storage data set. See *SMPMTS*.

MTSMAC entry. An SMP/E entry that is a copy of a macro that resides only in a distribution library but is needed temporarily during APPLY processing. MTSMAC entries are in the SMPMTS data set.

MVS. Multiple Virtual Storage.

MVS Custom-Built Product Delivery Offering (CBPDO). A software delivery offering used to add products or service to an existing MVS, NCP, CICS, or IMS system.

N

NCP. Network Control Program.

negative prerequisite (NPRE). In SMP/E, a function that is mutually exclusive with another function. It is defined by the NPRE operand on the ++VER statement.

NPRE. Negative prerequisite.

O

object deck. Object module input to the link-edit utility that is placed in the input stream, in card format.

object module. A module that is the output from a language translator (such as a compiler or an assembler). An object module is in relocatable format with machine code that is not executable. Before an object module can be executed, it must be processed by the link-edit utility.

When an object module is link-edited, a load module is created. Several modules can be link-edited together to create one load module (for example, as part of SMP/E APPLY processing), or an object module can be link-edited by itself to create a single-module load module (for example, to prepare the module for shipment in RELFILE format or in an LKLIB data set or as part of SMP/E ACCEPT processing).

operating system. In SMP/E, the system updated by APPLY and RESTORE processing. It consists of the target libraries. Also called the target system.

OPTIONS entry. An SMP/E entry defining processing options that are to be used by SMP/E.

P

package attribute file. A package attribute file (PAF) is a file that contains control statements describing the contents of a network transportable package.

packaging. Adding the appropriate MCS statements to elements to create a SYSMOD, then putting the SYSMOD in the proper format on the distribution medium, such as a tape or direct access data sets.

PAF. package attribute file.

partitioned data set extended (PDSE). A system-managed data set containing an indexed directory and members that are similar to the directory and members of partitioned data sets. A PDSE can be used instead of a partitioned data set.

PE. See *program error PTF*.

PE-PTF. See *program error PTF*.

PID. The former name for ISD.

PRE. Prerequisite.

prerequisite (PRE). In SMP/E, a SYSMOD that must be installed before or along with another SYSMOD in order for that other SYSMOD to be successfully installed. It is defined by the PRE operand on the ++VER statement.

preventive service. (1) The mass installation of PTFs to avoid rediscoveries of the APARs fixed by those PTFs. (2) The SYSMODs delivered on the program update tape.

preventive service planning (PSP). Installation recommendations and HOLDDATA for a product or a service level. PSP information can be obtained through the IBM Support Center.

product. Generally, a software package, such as a licensed program or a user application. A product can contain one or more functions and can consist of one or more versions and releases.

product version. All the releases for a given version of a product.

program error PTF (PE-PTF). A PTF that has been found to contain an error. A PE-PTF is identified on a ++HOLD ERROR statement, along with the APAR that first reported the error.

program object. An executable program stored in a PDSE program library. It is similar to a load module, but has fewer restrictions. For SMP/E purposes, program objects are referred to as load modules.

program packaging. See *packaging*.

program product. The former term for licensed program.

program temporary fix (PTF). A temporary solution or bypass of a problem that may affect all users and that was diagnosed as the result of a defect in a current unaltered release of the program. In the absence of a new release of a system or component that incorporates the correction, the fix is not temporary but is the permanent and official correction mechanism. New elements can also be defined in a PTF. PTFs are identified to SMP/E by the ++PTF statement.

program update tape (PUT). The former vehicle for preventive service. See *expanded service options*.

PSP. Preventive service planning.

PSW. Program status word.

PTF. Program temporary fix.

PTS. PTF temporary store data set. See *SMPPTS*.

PTFIN. PTF input. See *SMPPTFIN*.

PUT. See *expanded service options*.

R

RACF. Resource Access Control Facility.

RECEIVE. The SMP/E command used to read in SYSMODs and other data from SMPPTFIN and SMPHOLD.

receive. In SMP/E, to read SYSMODs and other data from SMPPTFIN and SMPHOLD and store them on the global zone for subsequent SMP/E processing. This is done with the RECEIVE command.

regressed SYSMOD. A SYSMOD one or more of whose elements are modified by subsequent SYSMODs that are not related to it.

regressing SYSMOD. A SYSMOD that causes regression of previous modifications when it is installed.

regression. In SMP/E, the condition that occurs when an element is changed by a SYSMOD that is not related to SYSMODs that previously modified the element.

REJECT. The SMP/E command used to remove SYSMODs from the global zone and SMPPTS.

reject. In SMP/E, to remove SYSMODs from the global zone and SMPPTS and delete any related SMPTLIB data sets. This is done with the REJECT command.

related installation materials (RIMs). In IBM custom-built offerings, task-oriented documentation, jobs, sample exit routines, procedures, parameters, and examples developed by IBM.

related SYSMOD. A SYSMOD associated with other SYSMODs by the FMID, PRE, REQ, or SUP operands.

related zone. The zone named in the RELATED subentry of a TARGETZONE or DLIBZONE entry. For a target zone, the related zone is generally the distribution zone for the libraries used to create the target libraries. For a distribution zone, the related zone is generally the target zone for the libraries built from the distribution libraries.

relative file (RELFILE) format. A SYSMOD packaging method where elements and JCLIN data are in separate relative files from the MCSs. When SYSMODs are packaged in relative file format there is a file of MCSs for one or more SYSMODs, and one or more relative files containing unloaded source-code data sets and unloaded link-edited data sets containing executable modules. The relative files can be either unloaded files in IEBCOPY format, or they can be partitioned data sets. Relative file format is the typical method used for packaging function SYSMODs.

relative files (RELFILEs). Unloaded files containing modification text and JCL input data associated with a SYSMOD. These files are used to package a SYSMOD in relative file format.

release. A distribution of a new product or new function and APAR fixes for an existing product. Contrast with *modification level* and *version*.

replacement modification identifier (RMID). The SYSMOD ID of the last SYSMOD that completely replaced a given element.

REPORT. The SMP/E command used to obtain information about SYSMODs that have been installed. These are the types of REPORT commands:

- REPORT CALLLIBS: Identifies load modules that need to be relinked because implicitly-included modules in a particular library have been updated.
- REPORT CROSSZONE: Lists conditional requisites that must be installed in certain zones because of SYSMODs installed in other zones.
- REPORT ERRSYSMODS: Determines whether any SYSMODs already installed are now exception SYSMODs.
- REPORT SOURCEID: Lists the source IDs associated with SYSMODs in the specified zones.
- REPORT SYSMODS: Compares the SYSMODs installed in two target or distribution zones.

requisite. A SYSMOD that must be installed before or at the same time as the SYSMOD being processed. There are several types of requisites:

- Prerequisites, which are specified by the PRE operand on the SYSMOD's ++VER statement
- Corequisites, which are specified by the REQ operand on the SYSMOD's ++VER statement
- Conditional requisites, which are specified by the REQ operand on the SYSMOD's associated ++IF statement

requisite chain. A sequence of SYSMODs that are directly or indirectly identified as requisites for a given SYSMOD, (A SYSMOD may identify other SYSMODs as requisites, which in turn may have requisites of their own. The requisite chain extends from the initial SYSMOD, through the hierarchy of requisites, until no more SYSMODs are found that have requisites.) See *requisite*.

requisite set. The set of all SYSMODs on the requisite chain for a particular SYS

RESETRC. The SMP/E command used to set the return codes for the previous commands to zero, so that SMP/E can process the current command.

RESTORE. The SMP/E command used to remove applied SYSMODs from the target libraries.

restore. In SMP/E, to remove applied SYSMODs from the target libraries by use of the RESTORE command.

restore group. All the SYSMODs that have a direct or indirect relationship with a SYSMOD being restored by use of the GROUP operand.

RIM. Related installation material.

RMID. Replacement modification identifier.

RMF. Resource measurement facility.

root cause analysis. Processing done by SMP/E for the ACCEPT, APPLY, and RESTORE commands to identify causer SYSMODs (SYSMODs whose failure has led to the failure of other SYSMODs). The types of errors SMP/E analyzes to determine causer SYSMODs include the following:

- Held SYSMODs
- Missing requisite SYSMODs
- Utility program failures: copy, update, assembler, link, zap
- Out-of-space conditions: *x*37 abends
- Missing DD statements and other allocation errors
- ID errors (a SYSMOD does not supersede or specify as a prerequisite an RMID or a UMID)
- JCLIN failures (syntax errors)

RPL. Request parameter list.

RTM2WA. Recovery termination manager 2 work area.

S

SCDS. Save control data set. See *SMPSCDS*.

SCP. System control program.

select-mode processing. In SMP/E, processing that includes individually selected SYSMODs.

service. PTFs and APAR fixes.

service level. The FMID, RMID, and UMID values for an element. The service level identifies the owner of the element, the last SYSMOD to replace the element, and all the SYSMODs that have updated the element since it was last replaced.

service order relationship. A relationship among service SYSMODs that is determined by the PRE and SUP operands, and the type of SYSMOD.

service SYSMOD. Any SYSMOD identified by an ++APAR or ++PTF statement.

service update. The integration of available service into the current release of a function. Since this is not a new release of the function, it does not change the function's FMID.

SET. The SMP/E command used to indicate the zone to be processed.

set. In SMP/E, to indicate which zone should be processed by the subsequent commands. This is done with the SET command.

set-to zone. The zone that was specified on the previous SET command and that is currently being processed. Contrast with *cross-zone*.

SHA-1. Secure Hash Algorithm 1.

side deck. See *definition side deck*.

single-module load module. A load module created by link-editing a single object module by itself—for example, to prepare the module for shipment in RELFILE format or in an LKLIB data set or as part of SMP/E ACCEPT processing.

SMPCNTL. The SMP/E data set or file in the hierarchical file system that contains the SMP/E commands to be processed.

SMPCSI. The SMP/E data set that contains information about the structure of a user's system as well as information needed to install the operating system on a user's system. The SMPCSI DD statement refers specifically to the CSI that contains the global zone. This is also called the *master CSI*.

SMPDEBUG. The SMP/E data set or file in the hierarchical file system that contains a dump requested by the DEBUG command. Depending on the operands specified, it may contain (1) a dump of SMP/E control blocks and storage areas associated with the specified dump points or (2) a dump of the VSAM RPL control block for the specified SMP/E function.

SMP/E. A program product, or an element of OS/390 or z/OS, used to install software and software changes on z/OS systems. SMP/E consolidates installation data, allows more flexibility in selecting changes to be installed, provides a dialog interface, and supports dynamic allocation of data sets.

SMP/E commands. Commands defining the processing to be done by SMP/E, such as RECEIVE.

SMP/E entry. An entry in an SMP/E data set—for example, a MOD entry in a CSI data set.

SMPHOLD. SMPHOLD is the source for HOLDDATA (++HOLD and ++RELEASE statements) to be processed by the RECEIVE command. SMPHOLD may be a tape file, a data set, or one or more files in the hierarchical file system.

SMPJCLIN. The SMP/E data set or file in the hierarchical file system that contains a job stream of assembly, link-edit, and copy job steps. This data is typically the stage 1 output from the most recent full or

partial system generation. However, it may also be other data in a similar format, such as the output of the GENERATE command. This job stream is used as input to the JCLIN command to update or create entries in a target zone.

SMPLIST. The SMP/E data set or file in the hierarchical file system that contains the output of all LIST commands.

SMPLOG. The SMP/E data set that contains time-stamped records of SMP/E processing. The records in this data set can be written automatically by SMP/E or added by the user through the LOG command.

SMPLOGA. A secondary log data set for SMP/E processing. If SMPLOGA is defined, it is automatically used when the SMPLOG data set is full.

SMPLTS. The SMP/E data set used as a target load module library to maintain the base version of a load module that specifies a SYSLIB allocation in order to implicitly include modules.

SMPMTS. The SMP/E data set used as a target library for macros that exist only in a distribution library, such as macros in SYS1.AMODGEN. The SMPMTS enables the current version of these macros to be used for assemblies during APPLY processing.

SMPNTS. The SMPNTS is a directory structure and associated files contained in the hierarchical file system used for temporary storage of network transported packages that were received during SMP/E RECEIVE processing.

SMPOBJ. The SMP/E data set used for source-maintained products. SMPOBJ contains preassembled modules that can be used to avoid reassembling those modules. These modules must be in load module format—that is, in the same format as modules residing in the distribution library.

SMPOUT. The SMP/E data set or file in the hierarchical file system that contains messages issued during SMP/E processing. It may also contain a dump of the VSAM RPL, if a dump was taken. In addition, it may contain LIST output and reports if SMPLIST and SMPRPT are not defined.

SMPPARM. The data set that contains members to define parameters, such as macros, assembler operation codes, GIMDDALC control statements, and exit routines.

SMPPTFIN. SMPPTFIN is the source of SYSMODs and ++ASSIGN statements to be processed by the RECEIVE command. SMPPTFIN may be a tape file, a data set, or one or more files in the hierarchical file system.

SMPPTS. The SMP/E data set that contains SYSMODs received from SMPPTFIN. SMPPTS is the source of SYSMODs that are installed in the target and distribution libraries.

SMPPTS spill data sets. Optional SMP/E data sets that can be used to store SYSMODs when the SMPPTS data set becomes full.

SMPPUNCH. The SMP/E data set or file in the hierarchical file system that contains output from various SMP/E commands. This output generally consists of commands or control statements.

- GENERATE: A job stream for building target libraries
- REPORT: Commands for installing or listing SYSMODs
- UNLOAD: UCLIN statements for recreating the entries that were unloaded

SMPRPT. The SMP/E data set or file in the hierarchical file system that contains the reports produced during SMP/E processing.

SMPSCDS. The SMP/E data set that contains backup copies of target zone entries that are created during APPLY processing. These backup copies are made before the entries are (1) changed by inline JCLIN, a ++MOVE MCS, or a ++RENAME MCS, or (2) deleted by an element MCS with the DELETE operand. The backup copies are used during RESTORE processing to return the entries to the way they were before APPLY processing.

SMPSNAP. The SMP/E data set that is used for snap dump output. When a severe error such as an abend or severe VSAM return code occurs, SMP/E requests a snap dump of its storage before doing any error recovery. In addition, the DEBUG command can request a snap dump of SMP/E storage when specified messages are issued, or can request a snap dump of control blocks and storage areas associated with a specified dump point.

SMPSTS. The SMP/E data set used as a target library for source that exists only in a distribution library. The SMPSTS enables the current version of this source to be used for assemblies during APPLY processing.

SMPTLIB. The SMP/E data sets used as temporary storage for relative files loaded from SMPPTFIN during RECEIVE processing. The SMPTLIB data sets are deleted when the associated SYSMOD is deleted by REJECT, RESTORE, or ACCEPT processing.

SMPWRK1. The SMP/E data set used as temporary storage for macro updates and replacements that will be processed by an update or copy utility program. SMP/E places the input in SMPWRK1 during APPLY and ACCEPT processing before calling the utility.

SMPWRK2. The SMP/E data set used as temporary storage for source updates and source replacements

that will be processed by an update or copy utility program. SMP/E places the input in SMPWRK2 during APPLY and ACCEPT processing before calling the utility.

SMPWRK3. The SMP/E data set used as temporary storage for object modules supplied by a SYSMOD, object modules created by assemblies, and zap utility input following ++ZAP statements.

SMPWRK4. The SMP/E data set used as temporary storage for zap utility or link-edit utility input that contains EXPAND control statements.

SMPWRK6. The SMP/E data set used during ACCEPT and APPLY processing as temporary storage for inline replacements for data elements. SMP/E places the input in this data set so that it can be directly accessed and installed by the copy utility or SMP/E.

source. The source statements that constitute the input to a language translator for a particular translation.

source ID. A 1- to 8-character identifier that indicates how a SYSMOD was obtained—for example, from a particular service level in an ESO. A source ID is associated with a specific SYSMOD by the RECEIVE command or the ++ASSIGN statement.

SOURCEID. The operand used to refer to a source ID.

source module. The source statements that constitute the input to a language translator, such as a compiler or an assembler, for a particular translation.

SRC. The SMP/E entry or MCS statement that describes a source.

SRCUPD. The MCS used to update a source.

SREL. System release identifier.

Storage Management Subsystem (SMS). A DFSMS/MVS facility used to automate and centralize the management of storage. Using SMS, a storage administrator describes data allocation characteristics, performance and availability goals, backup and retention requirements, and storage requirements to the system through data class, storage class, management class, storage group, and ACS routine definitions.

STS. Source temporary store data set. See *SMPSTS*.

STSSRC entry. An SMP/E entry that is a copy of source that resides only in a distribution library but is needed temporarily during APPLY processing. STSSRC entries are in the SMPSTS data set.

stub entry. An element entry or LMOD entry that does not contain the basic information SMP/E requires in order to process the element or load module (such as FMID, RMID, or library names), but does contain other information, such as subentries describing cross-zone relationships.

stub load module. A load module that does not contain the modules needed to perform its basic functions, but does contain other modules, such as cross-zone modules.

subentry. A field in an SMP/E entry. Each subentry has associated with it a type and a value. The same subentry type may occur several times in a single entry, each time with a different value. For example, the modules supplied by a PTF are saved as MOD subentries in the PTF's SYSMOD entry. Some subentries occur only once within an entry, such as the FMID subentry in a target zone MOD entry.

subentry indicator. A subentry that does not have a data value associated with it. An example of an indicator is the ERROR indicator in the SYSMOD entry. An indicator is either on or off.

subentry list. Multiple occurrences of the same subentry type in an entry, each with a different value. For example, the modules supplied by a PTF are saved as names in the MOD subentry list within the SYSMOD entry for that PTF.

SUP. Supersede.

superseded-only SYSMOD. A SYSMOD that has not been installed, but that has been superseded by another SYSMOD that has been installed.

superseded SYSMOD. In SMP/E, a SYSMOD that is contained in or replaced by the SYSMOD or requisite set of SYSMODs currently being processed. This is indicated by the SUPBY subentry in the SYSMOD entry for the superseded SYSMOD. A superseded SYSMOD is functionally lower than the SYSMOD that superseded it.

superseding SYSMOD. In SMP/E, a SYSMOD that contains all the functions in another SYSMOD and is recognized as the equivalent of that other SYSMOD. The superseding SYSMOD uses SUP operand on its ++VER statement to specify the superseded SYSMOD.

superzap. A generic term for the process performed by IMASPZAP. It can also refer to the module updates processed by IMASPZAP.

SVC. Supervised call.

SVRB. Supervisor request block.

SYSGEN. System generation.

SYSLIB. (1) A subentry used to identify the target library in which an element is installed. (2) A concatenation of macro libraries to be used by the assembler. (3) A set of routines used by the link-edit utility to resolve unresolved external references.

SYSMOD. System modification.

SYSMOD entry. An SMP/E entry containing information about a SYSMOD that has been received into SMPPTS, accepted into the distribution libraries, or applied to the target libraries.

SYSMOD ID. System modification identifier.

SYSMOD packaging. See *packaging*.

SYSMOD selection. The process of determining which SYSMODs are eligible to be processed.

SYSPRINT. The data set that contains output from the utilities called by SMP/E.

SYSPUNCH. The temporary data set containing object modules assembled by running the job stream produced by system generation or the GENERATE command. These modules are not installed in the distribution libraries at ACCEPT time.

system control program (SCP). IBM-supplied programming that is fundamental to the operation and maintenance of the system. It serves as an interface with licensed programs and user programs and is available without additional charge.

system generation (SYSGEN). The process of selecting optional parts of an operating system and of creating a particular operating system tailored to the requirements of a data processing installation.

system modification (SYSMOD). The input data to SMP/E that defines the introduction, replacement, or update of elements in the operating system and associated distribution libraries to be installed under the control of SMP/E. A system modification is defined by a set of MCS.

system modification identifier (SYSMOD ID). The name that SMP/E associates with a system modification. It is specified on the ++APAR, ++FUNCTION, ++PTF, or ++USERMOD statement.

system release identifier (SREL). A 4-byte value representing the system or subsystem, such as Z038 for MVS-based products.

SYSUT1, SYSUT2, SYSUT3. Scratch data sets for SMP/E and the utilities it calls.

SYSUT4. A data set that is used instead of the SYSIN data sets when certain utilities are called.

T

target library. A library containing the executable code that makes up a system.

target system. The system updated during APPLY and RESTORE processing, also referred to as the operating system. See also target libraries.

target zone. In SMP/E, a group of records in a CSI data set that describes the SYSMODs, elements, and load modules in a target library.

TARGETZONE entry. An SMP/E entry containing information used by SMP/E to process a specific target zone and the associated target libraries.

TCP/IP. Transmission Control Protocol/Internet Protocol (TCP/IP) is a hardware independent communication protocol used between physically separated computers. It was designed to facilitate communication between computers located on different physical networks.

temporary data set. A work data set (SMPWRK1–SMPWRK6) or utility data set (SYSUT1–SYSUT4). Temporary data sets are allocated when processing for an SMP/E command begins, and deleted when processing is finished.

text library (TXLIB). A data set containing JCLIN input or replacements for macros, source, or object modules that have not been link-edited. It is used when the JCLIN or elements are provided in partitioned data sets rather than inline or in relative files.

TGTLIB. Target library.

TIEDTO relationship. A cross-zone relationship between two target zones created when the LINK command updates a load module in one of the zones to include modules from the other zone. This relationship is established through the TIEDTO subentry in the zone definition entries for each of the zones.

TIEDTO zone. See *cross-zone*.

TLIB. Temporary library. See *SMPTLIB*.

transformed data. Data processed by the GIMDTS service routine so that it can be packaged inline in fixed-block 80 records.

TSO. Time-sharing option.

TXLIB. Text library.

U

UCL. Update control language.

UCL statement. An SMP/E control statement used to define or change information in an SMP/E data set entry. UCL statements are coded between the UCLIN and ENDUCL commands. The UCL statement specifies the action to be taken (ADD, REP, or DEL), the entry to be modified, and any indicators and subentries to be changed.

UCLIN. The SMP/E command used to mark the beginning of UCL statements, which are used to make changes to entries in SMP/E data sets.

UMID. Update modification identifier.

unconditionally coexisting functions. Functions that coexist and must be in the same zone.

UNLOAD. The SMP/E command used to copy data out of SMP/E data set entries in the form of UCL statements.

unload. In SMP/E, to copy data out of SMP/E data set entries in the form of UCL statements, by use of the UNLOAD command.

update. In SMP/E, to change an existing element without replacing it.

update modification identifier (UMID). The SYSMOD ID of a SYSMOD that updated the last replacement of a given element.

user modification (USERMOD). A change constructed by a user to modify an existing function, add to an existing function, or add a user-defined function. USERMODs are identified to SMP/E by the ++USERMOD statement.

USERMOD. User modification.

UTILITY entry. An SMP/E entry containing information used by SMP/E to invoke a particular system utility program.

V

VERSION. An operand on the ++VER or element statement. VERSION specifies one or more SYSMODs containing elements that are functionally lower than elements in the SYSMOD that specifies the operand. The VERSION operand is also used to change ownership of elements.

version. A separate licensed program that is based on an existing licensed program and that usually has significant new code or new functions. Contrast with *release* and *modification level*.

versioned element. An element that is part of more than one function—for example, one that is part of a base function and a dependent function.

VSAM. Virtual Storage Access Method.

VTOC. Volume table of contents.

Z

ZAP. (1) The SMP/E MCS used to package an update for an object module. (2) The superzap control statement used to update an object module. (3) A shortened name for the superzap utility, which is used to install these updates. See *IMASPZAP*.

zone. A partition in a CSI data set.

ZONECOPY. The SMP/E command used to copy a zone from one CSI data set to another.

ZONEDELETE. The SMP/E command used to delete a zone from a CSI data set.

ZONEEDIT. The SMP/E command used to change the values for a subentry in all the DDDEF or UTILITY entries in a given zone.

ZONEEXPORT. The SMP/E command used to copy a zone into a sequential data set.

ZONEIMPORT. The SMP/E command used to load an exported zone from a sequential data set into another zone.

ZONEMERGE. The SMP/E command used to copy one zone into another, or to merge two zones into one.

ZONERENAME. The SMP/E command used to change the name of a zone.

ZONESET. A group of zones to be used when processing an SMP/E command. For example, it may define the zones that the REPORT command is to check for cross-zone requisites. A ZONESET may also define a group of zones to be checked or ignored by the REJECT command.

ZONESET entry. An SMP/E entry defining a ZONESET.

z/OS UNIX System Services (z/OS UNIX). The set of functions provided by the Shell and Utilities, kernel, debugger, file system, C/C++ Run-Time Library, Language Environment, and other elements of the z/OS operating system that allow users to write and run application programs that conform to UNIX standards.

Index

A

abend
  dumps
    dialogs 303
    SMP/E error 306
    VSAM error 306
  ESTAE processing 295, 306
  problems 298
  x37 abends 292
ABEND keyword 297
ACCEPT command
  return codes 269
accessibility 311
adding space to data sets 292
AMS utility
  EXAMINE command 294
  REPRO command 293
APPLY command
  return codes 272

C

CBPDO (Custom-Built Product Delivery Offering) 287
CLEANUP command 294
  return codes 274
compressing data sets 292, 293, 294
CSI problems 289
Custom-Built Product Delivery Offering (CBPDO) 287

D

DEBUG command
  dumps
    dump points 308
    messages 307
  message tracing 309
  SMP/E processing 306
DIALOG keyword 298
dialog problems
  documentation 303
  preventing 295
  reporting procedure 303
  sample of reporting 303
disability 311
DOC keyword 297
documentation problems 297, 302
dumping data with the DEBUG command
  SMP/E storage and control blocks 307
  VSAM RPL control blocks 307
dumps
  abend
    dialogs 303
    SMP/E error 306
    VSAM error 306
  DEBUG
    dump points 308
    messages 307
dumps *(continued)*
  SMPDEBUG 308
  SMPSNAP
    abend problems 298, 306
    using the DEBUG command 307, 308
  SYSABEND
    abend problems 298, 306
    dialog problems 303
    loop problems 300
    wait problems 300
  SYSUDUMP 298, 306

E

Element Summary report 288
Enhanced HOLDDATA 287
Enhanced Service Offering (ESO) 287
error messages
  problems with 301
  SMP/E command processing 1
  symptoms of problems 288, 290
  tracing 309
ESO (Enhanced Service Offering) 287
exception SYSMODs 289

F

failure keywords 297

G

GENERATE command
  return codes 275

I

incorrect output problems 301
INCORROUT keyword 297
installation problems
  ERROR HOLDs 289
  function SYSMOD 288
  SYSTEM HOLDs 289

J

JCLIN command
  return codes 275
job step return codes 290

K

keyboard 311
keywords for failure types 297

© Copyright IBM Corp. 1986, 2002

L

LINK command
  return codes 276
link-edit jobs, problems with 288
LIST command
  return codes 277
LIST output
  SYSMOD status 291, 292
  using XREF 292
LOG command
  return codes 277
log data set
  alternate log data set 293
  check contents of 291
  increasing size 293
LOG data set
  installation 287
loop
  keyword 297
  problems 300

M

messages
  explanations 2
  problems with 301
  SMP/E command processing 1
  symptoms of problems 288, 290
  tracing 309
MSG keyword 297
MSGMODID 309

N

notices 313

O

output incorrect, problems with 301

P

PERFM keyword 297
performance problems
  overview 288
  reporting 301
preventing shortages of storage
  assembler COPY statement 293
  COMPRESS operand 292
  JCLIN 293
  RETRY operand 292
  x37 abends 292
preventive service planning (PSP) bucket 287
PSP (preventive service planning) 287

R

RC operand 291
RECEIVE command
  return codes 277
recovering from storage shortages
  abend ESTAE processing 295
  backup SMPLOG 293
  CLEANUP command 294
  COMPRESS 293
  library space 293
  SMPCSI space 293
  SMPMTS space 294
  SMPPTS space 294
  SMPSCDS space 294
  SMPSTS space 294
  x37 abends 293
RefreshPac 287
region size for SMP/E 292
REJECT command
  return codes 279
REPORT CALLLIBS command
  return codes 279
REPORT CROSSZONE command
  return codes 279
REPORT ERRSYSMODS command
  return codes 279
REPORT SOURCEID command
  return codes 279
REPORT SYSMODS command
  return codes 279
reporting problems
  abends 298
  checklist 296
  dialog problems 303
  documentation problems 302
  incorrect output 301
  keywords 297
  loop problems 300
  message problems 301
  performance problems 301
  wait problems 300
RESTORE command
  return codes 280
return codes
  associated messages 291
  contained in job steps 290
  explanations 269
  job step return codes 290
  RC operand 291
  severity codes 290
  severity levels 1
  SMP/E commands 269, 291
  symptoms of problems 288
RPL control block, dumping 289, 294, 306
running out of space in data sets used by SMP/E 292

S

Service Update Facility (SUF) 287
SET command
  return codes 282
severity codes 290
severity level 1
shortage
  data set space 292

shortage *(continued)*
SMP/E storage 292
shortcut keys 311
SMP/E commands
messages 1
SMP/E dialog problems
documentation 303
preventing 295
reporting procedure 303
sample of reporting 303
SMP/E messages
explanations of 1
problems with 301
symptoms of problems 290
tracing 309
SMP/E performance, problems with 288
SMP/E return codes 290
explanations of 269
SMP/E storage, dumping 307
SMPCSI problems 289
SMPDEBUG, using 308
SMPLOG
alternative SMPLOG 293
check contents of 291
increasing size 293
installation 287
SMPMTS, compressing 294
SMPPTS, compressing 294
SMPSCDS, compressing 294
SMPSNAP dumps
abend problems 298, 306
using the DEBUG command 307, 308
SMPSTS, compressing 294
space problems
SMP/E region size 292
SMPMTS 294
SMPPTS 294
SMPSCDS 294
SMPSTS 294
storage problems
SMP/E region size 292
x37 abends 292
SUF (Service Update Facility) 287
SYSABEND dumps
abend problems 298, 306
dialog problems 303
loop problems 300
wait problems 300
SYSMOD status 291, 292
SYSPRINT utility data 291
SystemPac Selective Follow On Service 287
SYSUDUMP dumps 298, 306

U

UCLIN command
return codes 282
utility programs
problems with 288
SYSPRINT output 291

V

VSAM problems 289, 294, 306
VSAM RPL control blocks, dumping 289, 294, 306

W

WAIT keyword 297
wait problems 300

X

x37 abends
preventing 292
recovering from 293

Z

ZONECOPY command
return codes 283
ZONEDELETE command
return codes 283
ZONEEDIT command
return codes 284
ZONEEXPORT command
return codes 284
ZONEIMPORT command
return codes 285
ZONEMERGE command
return codes 285
ZONERENAME command
return codes 286

Readers' Comments — We'd Like to Hear from You

IBM SMP/E for z/OS and OS/390
Messages, Codes, and Diagnosis

Publication No. GA22-7770-02

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

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. _____

Readers' Comments — We'd Like to Hear from You
GA22-7770-02

IBM

Cut or Fold
Along Line

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

IBM Corporation
Department 55JA, Mail Station P384
2455 South Road
Poughkeepsie, NY
12601-5400

Fold and Tape **Please do not staple** Fold and Tape

GA22-7770-02

Cut or Fold
Along Line

Program Number: 5655-G44

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GA22-7770-02