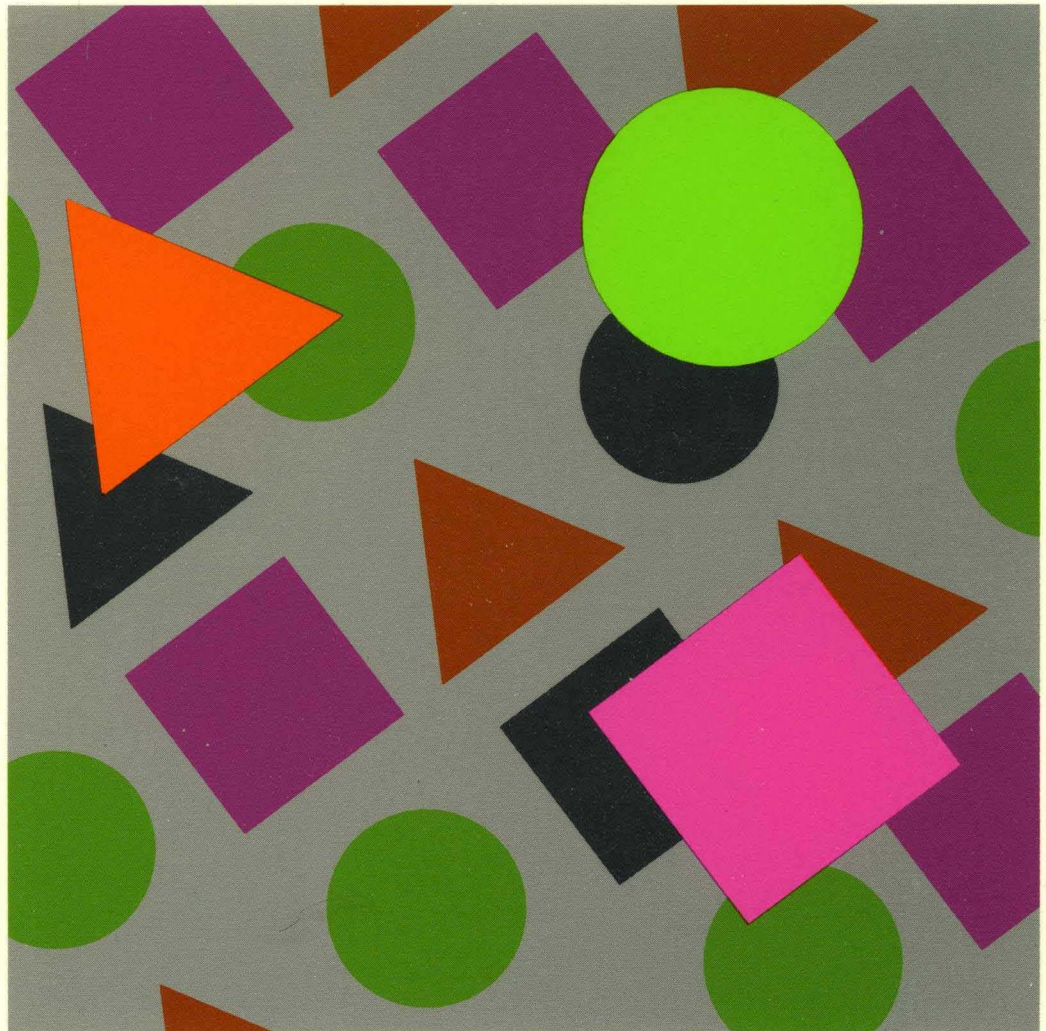


System Modification Program Extended

SC28-1108-09

**Messages and Codes**

Release 6





# System Modification Program Extended

SC28-1108-09

## **Messages and Codes**

Release 6

---

**Note!**

Before using this information and the product it supports, be sure to read the general information under "Notices" on page v.

**Tenth Edition (April 1992)**

This book replaces the previous edition, SC28-1108-08, 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 System Modification Program Extended (SMP/E) Release 6 for OS/VS2 (MVS) and to all subsequent releases unless otherwise indicated in new editions.

Consult the latest edition of the *IBM System/370, 30xx, 4300, and 9370 Processors Bibliography*, GC20-0001, for current information on this product.

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

A form for reader's comments is provided at the back of this publication. If the form has been removed, address your comments to:

IBM Corporation  
Information Development  
Department 52QA/MS 911  
Neighborhood Road  
Kingston NY 12401-0000  
USA

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 1983, 1992. All rights reserved.

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>Notices</b> . . . . .	v
Trademarks . . . . .	v
<b>Preface</b> . . . . .	vii
When to Use This Publication . . . . .	vii
How to Use This Publication . . . . .	vii
The SMP/E Library . . . . .	vii
Related Publications . . . . .	ix
Notations Used in This Publication . . . . .	xii
<b>Summary of Changes for SMP/E Release 6</b> . . . . .	xiii
April 1992 (SC28-1108-09) . . . . .	xiii
August 1991 (SC28-1108-08) . . . . .	xiii
Message Changes for SC28-1108-09 . . . . .	xv
Message Changes for SC28-1108-08 . . . . .	xvi
<b>Chapter 1. SMP/E Return Codes</b> . . . . .	1
ACCEPT Return Codes . . . . .	1
ACCEPT Error Recovery . . . . .	3
APPLY Return Codes . . . . .	4
APPLY Error Recovery . . . . .	5
CLEANUP Return Codes . . . . .	6
CLEANUP Error Recovery . . . . .	6
CONVERT Return Codes . . . . .	6
CONVERT Error Recovery . . . . .	7
GENERATE Return Codes . . . . .	7
GENERATE Error Recovery . . . . .	8
JCLIN Return Codes . . . . .	8
JCLIN Error Recovery . . . . .	8
LIST Return Codes . . . . .	9
LIST Error Recovery . . . . .	9
LOG Return Codes . . . . .	9
RECEIVE Return Codes . . . . .	9
RECEIVE Error Recovery . . . . .	10
REJECT Return Codes . . . . .	11
REJECT Error Recovery . . . . .	11
REPORT Return Codes . . . . .	11
REPORT Error Recovery . . . . .	12
RESTORE Return Codes . . . . .	12
RESTORE Error Recovery . . . . .	14
SET Return Codes . . . . .	14
SET Error Recovery . . . . .	14
UCLIN Return Codes . . . . .	15
UCLIN Error Recovery . . . . .	15
ZONECOPY Return Codes . . . . .	15
ZONECOPY Error Recovery . . . . .	15
ZONEDELETE Return Codes . . . . .	16
ZONEDELETE Error Recovery . . . . .	16
ZONEEDIT Return Codes . . . . .	16
ZONEEDIT Error Recovery . . . . .	16

---

ZONEEXPORT Return Codes . . . . .	16
ZONEEXPORT Error Recovery . . . . .	17
ZONEIMPORT Return Codes . . . . .	17
ZONEIMPORT Error Recovery . . . . .	17
ZONEMERGE Return Codes . . . . .	18
ZONEMERGE Error Recovery . . . . .	18
ZONERENAME Return Codes . . . . .	18
ZONERENAME Error Recovery . . . . .	18
<b>Chapter 2. SMP/E Messages . . . . .</b>	<b>19</b>
Message Format . . . . .	19
Diagnostic Messages . . . . .	20
<b>Chapter 3. CBIPO Dialogs—Batch Messages . . . . .</b>	<b>223</b>
Message Format . . . . .	223
Diagnostic Messages . . . . .	224
<b>Glossary . . . . .</b>	<b>247</b>

---

## Notices

### Notice to OS/VS1 Users

Do **not** use this publication for your OS/VS1 system. SMP/E Release 6 does not run under OS/VS1. Use the latest SMP/E Release 2 edition of this publication instead. The order number for the Release 2 edition is GT28-1108.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights or other legally protected rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, programs, or services, except those expressly designated by IBM, are the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Commercial Relations, IBM Corporation, Purchase, NY 10577.

---

## Trademarks

The following terms, denoted by an asterisk (\*) in this publication, are trademarks of the IBM Corporation in the United States or other countries:

BookManager  
IBM  
MVS/ESA  
MVS/SP  
MVS/XA  
System/370



---

## Preface

---

### When to Use This Publication

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.

---

### How to Use This Publication

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

- **Chapter 1, “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 2, “SMP/E Messages”** This chapter lists the SMP/E diagnostic messages, along with the system action and your required response. Messages appear alphanumerically.
- **Chapter 3, “CBIPO Dialogs—Batch Messages”** This chapter lists the batch messages for CBIPO dialogs. Messages appear alphanumerically.

The pages have running heads to help you find messages more quickly. The running head changes every time the first digit of the message number changes. For example, if you were looking for message GIM35901, you would look in the section with the heading GIM3xxxx. If you were looking for message GIM51401, you would look in the section with the heading GIM5xxxx.

---

### The SMP/E Library

Table 1 lists the SMP/E Release 6 publications and briefly describes each one.

<i>Table 1 (Page 1 of 2). Publications for SMP/E Release 6</i>	
<b>Title</b>	<b>Description</b>
<i>MVS Software Manufacturing Offerings General Information, GC23-0351</i> Online book: GIM9MST	Provides a summary of SMP/E, CBIPO, and CBPDO.
<i>SMP/E Program Directory for Installation Planning (English Feature), GC23-0130</i> Online book: GIM1MST	Explains how to plan for installing SMP/E Release 6 using SMP/E Release 5 or higher.
<i>SMP/E Program Directory for Installation Planning (Japanese Feature), GC23-0469</i> Online book: GIMJMST	
<i>SMP/E User's Guide, SC28-1302</i> Online book: GIM8MST	Describes how to use SMP/E to install programs and service.



<i>Table 1 (Page 2 of 2). Publications for SMP/E Release 6</i>	
<b>Title</b>	<b>Description</b>
<i>SMP/E Messages and Codes</i> , SC28-1108 Online book: GIM4MST	Explains SMP/E messages and return codes and the actions to take for each message and code.
<i>SMP/E Reference</i> , SC28-1107 Online book: GIM6MST	Explains SMP/E commands and processing in detail.
<i>SMP/E Reference Summary</i> , SX22-0006 Online book: GIM7MST	Reviews the SMP/E commands in a convenient form.
<i>SMP/E Program Packaging Guide</i> , SC23-0221 Online book: GIM5MST	Explains how to package programs for installation by SMP/E.
<i>SMP/E CBIPO Dialogs User's Guide</i> , SC23-0538 Online book: GIMIMST	Explains how to use the CBIPO dialogs to install, reinstall, and redistribute CBIPO orders.
<i>SMP/E Diagnosis Guide</i> , LY27-8047 (no online book)	Explains how to handle suspected SMP/E problems.

**Notes:**

1. You can order hardcopy versions of individual SMP/E Release 6 publications or use a Bill of Forms number to order the complete set of unlicensed publications. (Because *SMP/E Diagnosis Guide* is a licensed publication, it must be ordered separately and is only available if you are licensed for SMP/E Release 6.)
  - For the English feature of SMP/E, use **SBOF-1587**.
  - For the Japanese feature of SMP/E, use **SBOF-3161**.
2. Online versions of the SMP/E Release 6 books (except for *SMP/E Diagnosis Guide*) are provided with the product.
3. You can order binders and inserts for the SMP/E library. (Inserts are the slip-in covers and spine for a binder.) Here are the order numbers:
  - For *SMP/E Reference*:
    - Binder plus inserts: **SBOF-2136**
    - Inserts only: **SX23-0442**
  - For the rest of the SMP/E library:
    - Binder plus inserts: **SBOF-2137**
    - Inserts only: **SX23-0443**.

**Note:** You will probably need two binders for the rest of the library.

## Related Publications

Product	Publications
All Products	<ul style="list-style-type: none"> <li>• <i>Programming System General Information Manual (PSGIM)</i>, G229-2228</li> </ul>
CBIPO and CBPDO	<ul style="list-style-type: none"> <li>• <i>MVS Software Manufacturing Offerings General Information</i>, GC23-0351</li> <li>• <i>MVS Custom-Built Offerings Planning and Installation</i>, SC23-0352</li> </ul>
ISPF Version 2 Release 3	<ul style="list-style-type: none"> <li>• <i>ISPF and PDF General Information</i>, GC34-4116</li> <li>• <i>ISPF Dialog Management Guide</i>, SC34-4112</li> <li>• <i>ISPF Dialog Management Services Examples</i>, SC34-4113</li> </ul>
MVS/ESA* (MVS/SP* Version 3)	<ul style="list-style-type: none"> <li>• <i>MVS/ESA Library User's Guide</i>, GC28-1563</li> <li>• <i>MVS/ESA Linkage Editor and Loader User's Guide</i>, SC26-4510</li> <li>• <i>MVS/ESA Integrated Catalog Administration: Access Method Services Reference</i>, SC26-4500</li> <li>• <i>MVS/ESA VSAM Catalog Administration: Access Method Services Reference</i>, SC26-4501</li> <li>• <i>MVS/ESA Catalog Administration Guide</i>, SC26-4502</li> <li>• <i>MVS/ESA Data Administration Guide</i>, SC26-4505</li> <li>• <i>MVS/ESA Data Administration: Utilities</i>, SC26-4516</li> <li>• <i>MVS/ESA Data Administration: Macro Instruction Reference</i>, SC26-4517</li> <li>• <i>MVS/ESA VSAM Administration Guide</i>, SC26-4518</li> <li>• <i>MVS/ESA Message Library: System Codes</i>, GC28-1815</li> <li>• <i>MVS/ESA Diagnosis: System Reference</i>, LY28-1011</li> <li>• <i>MVS/ESA JCL Reference</i>, GC28-1829</li> <li>• <i>MVS/ESA JCL User's Guide</i>, GC28-1830</li> <li>• <i>MVS/ESA Basics of Problem Determination</i>, GC28-1839</li> <li>• <i>MVS/ESA Service Aids</i>, GC28-1844</li> </ul>

MVS/ESA and MVS/SP are trademarks of the IBM Corporation.

Product	Publications
MVS/XA*	<ul style="list-style-type: none"> <li>• <i>MVS/XA System Macros and Facilities</i>, Vol. 1, GC28-1150</li> <li>• <i>MVS/XA Linkage Editor and Loader User's Guide</i>, GC26-4143</li> <li>• <i>MVS/XA System Data Administration</i>, GC26-4010</li> <li>• <i>MVS/XA Data Administration Guide</i>, GC26-4013</li> <li>• <i>MVS/XA Data Administration Guide: Macro Instruction Reference</i>, GC26-4014</li> <li>• <i>MVS/XA Data Administration: Utilities</i>, GC26-4150</li> <li>• <i>MVS/XA VSAM Administration Guide</i>, GC26-4015</li> <li>• <i>MVS/XA VSAM Administration: Macro Instruction Reference</i>, GC26-4152</li> <li>• <i>MVS/XA Catalog Administration Guide</i>, GC26-4138</li> <li>• <i>MVS/XA ICF Administration: Access Method Services Reference</i>, GC26-4135</li> <li>• <i>MVS/XA VSAM Catalog Administration: Access Method Services Reference</i>, GC26-4136</li> <li>• <i>MVS/XA ICF Administration: Access Method Services Reference</i>, GC26-4135</li> <li>• <i>MVS/XA Access Method Services Reference Summary for Integrated Catalog Facility</i>, GX26-3739</li> <li>• <i>MVS/XA DADSM and Common VTOC Access Facility Diagnosis Guide</i>, SY26-3896</li> <li>• <i>MVS/XA DADSM Diagnosis Reference</i>, SY26-3904</li> <li>• <i>Assembler H Version 2 Application Programming: Guide</i>, SC26-4036</li> <li>• <i>Assembler H Version 2 Application Programming: Language Reference</i>, GC26-4037</li> <li>• <i>MVS/XA Debugging Handbook</i>, Vol. 1, LC28-1164</li> <li>• <i>MVS/XA Debugging Handbook</i>, Vol. 2, LC28-1165</li> <li>• <i>MVS/XA Debugging Handbook</i>, Vol. 3, LC28-1166</li> <li>• <i>MVS/XA Debugging Handbook</i>, Vol. 4, LC28-1167</li> <li>• <i>MVS/XA Debugging Handbook</i>, Vol. 5, LC28-1168</li> <li>• <i>MVS/XA Debugging Handbook</i>, Vol. 6, LC28-1169</li> <li>• <i>MVS/XA Diagnostic Techniques</i>, LY28-1199</li> <li>• <i>MVS/XA Message Library: System Messages</i>, Vol. 1, GC28-1376</li> <li>• <i>MVS/XA Message Library: System Messages</i>, Vol. 2, GC28-1377</li> <li>• <i>MVS/XA System Codes</i>, GC28-1157</li> <li>• <i>System 370-XA Principles of Operation</i>, SA22-7085</li> <li>• <i>MVS/XA JCL User's Guide</i>, GC28-1351</li> <li>• <i>MVS/XA JCL Reference</i>, GC28-1352</li> <li>• <i>MVS/XA Service Aids</i>, GC28-1159</li> </ul>

MVS/XA is a trademark of the IBM Corporation.

Product	Publications
MVS/370	<ul style="list-style-type: none"> <li>• <i>MVS/370 Linkage Editor and Loader</i>, GC26-4061</li> <li>• <i>MVS/370 Data Facilities Product Utilities Data Administration</i>, GC26-4065</li> <li>• <i>MVS/370 Data Facilities Product System Data Administration</i>, GC26-4056</li> <li>• <i>MVS/370 Data Facilities Product Data Administration Guide</i>, GC26-4058</li> <li>• <i>MVS/370 Data Facilities Product Data Administration Macro Instruction Reference</i>, GC26-4057</li> <li>• <i>MVS/370 VSAM User's Guide</i>, GC26-4066</li> <li>• <i>MVS/370 VSAM Reference</i>, GC26-4074</li> <li>• <i>MVS/370 Catalog Administration Guide</i>, GC26-4053</li> <li>• <i>MVS/370 ICF Catalog Administration: Access Method Services Reference</i>, GC26-4153</li> <li>• <i>MVS/370 DADSM and Common VTOC Access Facility Diagnosis Guide</i>, LY27-9509</li> <li>• <i>MVS/370 DADSM Diagnosis Reference</i>, LY27-9510</li> <li>• <i>MVS/370 Access Method Services Reference Summary for the Integrated Catalog Facility</i>, GX26-3745</li> <li>• <i>MVS/370 VSAM Catalog Administration: Access Method Services Reference</i>, GC26-4154</li> <li>• <i>OS/VS, DOS/VSE and VM/370 Assembler Language</i>, GC33-4010</li> <li>• <i>MVS/370 JCL User's Guide</i>, GC28-1349</li> <li>• <i>MVS/370 JCL Reference</i>, GC28-1350</li> <li>• <i>OS/VS2 MVS System Programming Library: Service Aids</i>, GC28-0674</li> <li>• <i>OS/VS2 System Programming Library: Debugging Handbook</i>, Vol. 1, GC28-1047</li> <li>• <i>OS/VS2 System Programming Library: Debugging Handbook</i>, Vol. 2, GC28-1048</li> <li>• <i>OS/VS2 System Programming Library: Debugging Handbook</i>, Vol. 3, GC28-1049</li> <li>• <i>MVS System Product MVS Diagnostic Techniques</i>, SY28-1133</li> <li>• <i>MVS/370 Message Library: System Messages</i>, Vol. 1, GC28-1374</li> <li>• <i>MVS/370 Message Library: System Messages</i>, Vol. 2, GC28-1375</li> <li>• <i>OS/VS2 Message Library: VS2 System Codes</i>, GC38-1008</li> <li>• <i>IBM System/370 Principles of Operation</i>, GA22-7000</li> </ul>
MVS Release 3.8	<ul style="list-style-type: none"> <li>• <i>OS/VS Linkage Editor and Loader</i>, GC26-3813</li> <li>• <i>OS/VS MVS Utilities</i>, GC26-3902</li> <li>• <i>OS/VS Virtual Storage Access Method Programmer's Guide</i>, GC26-3838</li> <li>• <i>OS/VS2 Access Method Services</i>, GC26-3841</li> <li>• <i>OS/VS2 System Programming Library: Data Management</i>, GC26-3830</li> <li>• <i>OS/VS2 MVS Data Management Services Guide</i>, GC26-3875</li> <li>• <i>OS/VS, DOS/VSE and VM/370 Assembler Language</i>, GC33-4010</li> <li>• <i>OS/VS2 MVS JCL</i>, GC28-0692</li> <li>• <i>OS/VS2 System Programming Library: Service Aids</i>, GC28-0674</li> <li>• <i>OS/VS2 System Programming Library: Debugging Handbook</i>, Vol. 1, GC28-1047</li> <li>• <i>OS/VS2 System Programming Library: Debugging Handbook</i>, Vol. 2, GC28-1048</li> <li>• <i>OS/VS2 System Programming Library: Debugging Handbook</i>, Vol. 3, GC28-1049</li> <li>• <i>OS/VS2 System Programming Library: MVS Diagnostic Techniques</i>, GC28-0725</li> <li>• <i>OS/VS Message Library: VS2 System Messages</i>, GC38-1002</li> <li>• <i>OS/VS Message Library: VS2 System Codes</i>, GC38-1008</li> <li>• <i>OS/VS2 DADSM Logic</i>, SY26-3828</li> <li>• <i>IBM System/370 Principles of Operation</i>, GA22-7000</li> </ul>

---

## Notations Used in This Publication

- Abbreviations used in this publication are:
  - DLIB** Distribution library
  - PTF** Program temporary fix
  - SMP/E** System Modification Program Extended
  - SMP4** System Modification Program Release 4
- Messages appear in all uppercase letters. Symbols are listed below:
  - ' Apostrophe
  - \* Asterisk
  - Blank (Not coded)
  - , Comma
  - = Equal sign
  - ( ) Parentheses
  - . Period
- *Italic* character strings represent variables. For example, *aaaaaaaa* is a variable. However, the message that SMP/E generates will have a specific value (for example, MODA).

---

## Summary of Changes for SMP/E Release 6

---

### April 1992 (SC28-1108-09)

---

#### Change for Root Cause SPE

---

- **Root Cause Analysis**

The changes for root cause analysis are now available in a small programming enhancement (SPE), PTF UR90259 for FMID HMP1600 and, if the Japanese feature is installed, PTF UR90260 for FMID JMP1611. Information about root cause analysis is clearly marked in all the SMP/E manuals to indicate that it applies only to the root cause SPE.

To simplify the process of determining the errors that caused SYSMODs to fail, SMP/E provides a new report called the Causer SYSMOD Summary report. This report summarizes the messages describing the errors that actually caused the SYSMODs to fail and that must be fixed in order to resolve the problem. The report filters out subsequent messages that are generated because of these initial errors.

The SYSMOD Status report has also been improved:

- For each SYSMOD that failed, the report indicates the ID of the SYSMOD that caused the failure.
- A new statistics section indicates the number of SYSMODs that have been applied, accepted, or restored.

A new chapter has been added to the *SMP/E User's Guide* to explain how to use these reports.

Also, messages have been added, reworded, and deleted to provide better information for determining the cause of SYSMOD failure.

---

#### End of Change for Root Cause SPE

---

- **Corrections**

This book contains technical corrections and clarifications.

Vertical bars to the left of the text mark all these changes.

---

### August 1991 (SC28-1108-08)

- **CBIPO Dialogs**

SMP/E now includes dialogs that can be used to install CBIPO packages. These dialogs can be used instead of the existing batch installation process.

With the CBIPO dialogs you can:

- Install a system or subsystem from a CBIPO or redistribution tape.

- Reinstall existing user application programs and user data while installing a new system or subsystem from a CBIPO or redistribution tape.
- Redistribute a system or subsystem by either:
  - Copying the complete system or subsystem to DASD at the same location. This is called *local redistribution*.
  - Copying the complete system or subsystem to tape, transporting it, and installing the copied system or subsystem from the tape onto DASD at either the same location or at a different location. This is called *remote redistribution*.

For both types of redistribution, the data set, DASD, and catalog configuration of the copied system or subsystem need not be the same as the configuration of the original system or subsystem.

- Process I/O definitions for an MVS system that was installed using the CBIPO dialogs:
  - Process an MVSCP deck
  - Include existing IODFs (I/O definition files) for use during IPL and to be copied during reinstall or redistribution processing.
- Connect a previously installed subsystem to another MVS system.

- **Online Books**

Online books will now be provided for SMP/E. These online books can be viewed using IBM® BookManager® READ Release 2.

- **New Packaging for SMP/E**

SMP/E is now packaged using data elements. As a result, you can now install SMP/E only with SMP/E Release 5 or higher. (To install the online books you need SMP/E Release 5.1 or higher.) You cannot install this release of SMP/E using SMP4 or a release of SMP/E before Release 5.

- **SYSMOD Status Report Accuracy**

The accuracy of the SYSMOD Status report has been improved when listing unsatisfied requisite system modification (SYSMOD) and hold conditions. SMP/E has been enhanced to search beyond the initial point of SYSMOD termination and report on additional requisite SYSMOD and hold conditions that are unresolved for the failing SYSMOD. These conditions, identified with a dash (-) in the SYSMOD Status report, are also identified with additional GIM359xx messages. This enhancement can improve system programmer productivity by reducing the number of times a system programmer has to run a mass APPLY CHECK or ACCEPT CHECK in order to resolve all conditions required to install the desired SYSMODs.

- **REPLACE Operand on GENERATE Command**

The GENERATE command now provides a new operand, REPLACE, to create copy and link-edit jobs that replace existing members and load modules.

Change for Root Cause SPE

• **Root Cause Analysis**

To simplify the process of determining the errors that caused SYSMODs to fail, SMP/E will provide a new report called the Causer SYSMOD Summary report. This report summarizes the messages describing the errors that actually caused the SYSMODs to fail and that must be fixed in order to resolve the problem. The report filters out subsequent messages that are generated because of these initial errors.

The SYSMOD Status report has also been improved:

- For each SYSMOD that failed, the report indicates the ID of the SYSMOD that caused this failure.
- A new statistics section summarizes the number of SYSMODs that were successfully processed and the number of SYSMODs that were not successful.

In addition, messages have been added, reworded, and deleted to provide better information for determining the cause of SYSMOD failure.

The changes for root cause analysis will be available later in a small programming enhancement (SPE). (See the announcement letter for details.) Information about root cause analysis is, however, currently included in the SMP/E manuals. It is clearly marked to indicate that it only applies to the root cause SPE.

End of Change for Root Cause SPE

**Message Changes for SC28-1108-09**

The following messages have been added:

GIM35306      GIM69142      GIM69143

The following messages have been changed:

GIM31901      GIM43401      GIM62701  
 GIM31902      GIM50801      GIM62702  
 GIM31903      GIM54902      GIM65001  
 GIM31904      GIM57901      GIM65002  
 GIM35302

The following messages have been deleted:

GIM28001      GIM35303



**Message Changes for SC28-1108-08**

The following messages have been added:

GIM00002	GIM35942	GIM35946
GIM22602	GIM35943	GIM35947
GIM35940	GIM35944	GIM35948
GIM35941	GIM35945	

The following messages have been changed:

GIM24505	GIM35920	GIM35927
GIM30213	GIM35922	GIM35928
GIM30224	GIM35924	GIM37401
GIM30225	GIM35926	GIM50701
GIM35303		

The following messages have been deleted:

GIM30205	GIM30212	GIM35925
GIM30207	GIM35921	GIM35929
GIM30210	GIM35923	GIM43013
GIM30211		

---

## Chapter 1. SMP/E Return Codes

SMP/E sets return codes according to conditions that occur 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 the SMPOUT data set to describe the various situations and unusual conditions that occur during command processing. These messages fall into one of the following categories, according to the severity of the condition.

Category	Return Code	Severity Level	When Issued
Informational	00	I	<ul style="list-style-type: none"><li>To show stages of SMP/E processing.</li><li>To accompany other messages as an explanation for unusual conditions.</li></ul>
Warning	04	W	<ul style="list-style-type: none"><li>SMP/E detected a possible error.</li></ul> Look at these messages to determine if SMP/E's action was appropriate.
Error	08	E	<ul style="list-style-type: none"><li>SMP/E processing did not complete properly.</li></ul> For example, when APPLY processing of a SYSMOD stops, this is an error, and SMP/E issues a return code of at least 8.
Severe	12	S	<ul style="list-style-type: none"><li>An entire SMP/E command failed.</li></ul> For example, when APPLY processing stops because of insufficient storage, SMP/E issues a return code of 12.
Terminating	16	T	<ul style="list-style-type: none"><li>A situation occurred that forced SMP/E processing to stop.</li></ul>
Terminating	20	T	<ul style="list-style-type: none"><li>An internal SMP/E error caused SMP/E processing to stop.</li></ul> 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)
    - Update utility (default: IEBUPDTE)
    - Superzap utility (default: IMASPZAP)
    - Linkage editor (default: IEWL)

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, re:ry utility (default: IEBCOPY)
  - Update utility (default: IEBUPDTE)
  - Superzap utility (default: IMASPZAP)
  - Linkage editor (default: IEWL)

SMP/E sets the ACCEPT and ERROR status indicators in the affected distribution zone SYSMOD entries.

- SMP/E encountered an error while scanning IMASPZAP 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.

- An IMASPZAP 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.

**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 SMPDOUT 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 IMASPZAP processing is repeated. However, if any IMASPZAP processing completed through the IMASPZAP REPLACE stage, or if any IMASPZAP processing produced an IMASPZAP VERIFY REJECT in the previous ACCEPT, this rerun of ACCEPT will also fail. To correct this problem, do the following:
  1. Obtain the IMASPZAP control cards from the SMPPTS for the modules affected by the SYSMOD. To do this, use the IEBTPCH utility to punch the SYSMOD.
  2. Reject the SYSMOD from the SMPPTS.
  3. Correct any IMASPZAP 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)
- 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.
- 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)
- SMP/E sets the APPLY and ERROR status indicators in the affected target zone SYSMOD entries.
- SMP/E encountered an error while scanning IMASPZAP 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.
- An IMASPZAP VERIFY REJECT was encountered by IMASPZAP. 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.
- 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.) 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 IMASPZAP processing is repeated. However, if any IMASPZAP processing completed through the IMASPZAP REPLACE stage, or if any IMASPZAP processing produced an IMASPZAP VERIFY REJECT in the previous APPLY, this rerun of APPLY will fail. To correct this problem, do the following:
  1. Obtain the IMASPZAP control cards from the SMPPTS for the modules affected by the SYSMOD. To do this, use the IEBTPCH utility to punch the SYSMOD.
  2. Reject the SYSMOD from the SMPPTS.
  3. Correct any IMASPZAP 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.

---

### CONVERT Return Codes

- 00** CONVERT processing completed successfully.
- 04** CONVERT processing completed, but with unexpected results.
  - SMP4 conversion—SMP/E may have only partially converted the specified entry because of an error that occurred when it was converting that entry.
  - SMP/E conversion:
    - SMP/E could not recognize an encode value.
    - SMP/E found duplicate encode values, and only one of them was defined in a ZONEINDEX subentry.

- 
- 12** CONVERT processing stopped. These are some possible errors:
- The CONVERT command did not specify any data set type operands.
  - The zone type specified on the SET command is not correct for the type of data set specified on the CONVERT operand.
  - An error occurred while SMP/E was reading an entry to be converted or storing a converted entry.
  - SMP/E conversion:
    - There is no GIMZPOOL record in the OLDCSI or NEWCSI data set.
    - The NEWCSI data set is not empty.
    - The SMPCSI DD statement and NEWCSI operand point to the wrong data sets for the types of zones being converted. (There may have been an attempt to reconfigure the CSI data sets during conversion.)
    - The global zone was not converted before the target and distribution zones.
    - The global zone does not contain a ZONEINDEX subentry for a zone that is being converted.
- 16** A severe error caused SMP/E processing to stop.

## CONVERT Error Recovery

If an error occurs during CONVERT 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.
- 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 (x37 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.
  - 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 Reference* 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 the above methods, rerun JCLIN.

---

## 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 *SMP/E Diagnosis Guide* 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)

---

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.

**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)

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 a nonacceptable 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.

## Return Codes

---

- 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 SMP/E Release 6.
- 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 *SMP/E Reference* and *SMP/E Diagnosis Guide* 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 the SMP\_CNTL data set 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.
- 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.
- 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 the SMP\_CNTL data set 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.
- 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.
- 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).

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.
- 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.
- 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.
- 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 2. 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 `GIMnnnnnv yy text`. WTO, WTP, and WTOR messages are in the format `GIMnnnv yy text`.

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

I	Informational message	(Return Code = 00)
W	Warning message	(Return Code = 04)
E	Error message	(Return Code = 08)
S	Severe error message	(Return Code = 12)
T	Terminating error message	(Return Code = 16 or 20).

SMP/E sets the severity code 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 code 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

**GIM000v MESSAGE MODULE *aaaaaaaa* COULD NOT BE LOADED. SMP/E PROCESSING IS STOPPED. ABEND CODE: *bbbb* REASON CODE: *cc***

**Explanation:**

- *aaaaaaaa*—name of the SMP/E message module
- *bbbb*—system abend code
- *cc*—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.

Check *OS/VS Message Library: VS2 System Codes*, or the corresponding manual for your operating system, for an explanation of the system abend and reason codes.

**GIM00001v MODULE GIMKXENU COULD NOT BE LOADED. SMP/E PROCESSING IS STOPPED. ABEND CODE: *aaaa* REASON CODE: *bb*.**

**Explanation:**

- *aaaa*—system abend code
- *bb*—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.

Check *OS/VS Message Library: VS2 System Codes*, or the corresponding manual for your operating system, for an explanation of the system abend and reason codes.

---

Message for Root Cause SPE

---

**GIM00002v 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.

---

End of Message for Root Cause SPE

---

**GIM20101v** *aaaaaa* PROCESSING FAILED FOR THE *bbbbbbb* LIBRARY.**Explanation:**

- *aaaaaaa*—OPEN or CLOSE
- *bbbbbbb*—ddname of the library being processed.

**System Action:** Command processing stops. 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 nameFix the problem and rerun the job.
- If CLOSE failed, rerun the job. If CLOSE continues to fail, SMP/E needs data set maintenance.

**GIM20201v** 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.

**GIM20301v** THERE IS A SYNTAX ERROR IN THE COMMAND AT COLUMN *aa*.**Explanation:**

- *aa*—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.

**System Action:** Command processing stops.

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

**GIM20302v** THERE IS A SYNTAX ERROR IN THE MCS AT COLUMN *aa*.**Explanation:**

- *aa*—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.

**System Action:** SYSMOD processing stops.

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

**GIM20303v** THERE IS A SYNTAX ERROR IN THE UCL STATEMENT AT COLUMN *aa*.**Explanation:**

- *aa*—column number.

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

**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.

**GIM20304v 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.

**GIM20305v 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.

**GIM20306v 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.

**GIM20307v THERE IS A SYNTAX ERROR IN THE EXEC PARM STATEMENT AT COLUMN *aa*.**

**Explanation:**

- *aa*—column number.

SMP/E found a syntax error at the indicated column on the EXEC statement for GIMSMP.

**System Action:** SMP/E processing stops.

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

**GIM20308v THERE IS A SYNTAX ERROR IN THE ZONEEDIT CHANGE STATEMENT AT COLUMN *aa*.**

**Explanation:**

- *aa*—column number.

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

**System Action:** Command processing stops.

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

**GIM20309v THERE IS A SYNTAX ERROR IN PARMLIB MEMBER *aaaaaaaa* AT COLUMN *bb*.**

**Explanation:**

- *aaaaaaaa*—name of a member in SYS1.PARMLIB
- *bb*—column number.

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

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

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

**GIM20310v** THERE IS A DOUBLE-BYTE CHARACTER SYNTAX ERROR IN THE *aaaaaaaa* INPUT.

**Explanation:**

- *aaaaaaaa*—name of the 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.

**GIM20311v** THERE IS A SYNTAX ERROR IN THE CONTROL STATEMENT AT COLUMN *aa*.

**Explanation:**

- *aa*—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.

**System Action:** Command processing stops.

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

**GIM20501v** *aaaaaaaa* PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS *bb*.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bb*—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	-	nnnnnn	SYSMODS NOT REJECTED	-	nnnnnn
FMIDS DELETED	-	nnnnnn	FMIDS NOT DELETED	-	nnnnnn
HOLDDATA DELETED	-	nnnnnn			

**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 DELETEDFMID 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 DELETEDFMID operand that were not deleted.

**HOLDDATA DELETED**

is the number of external HOLDDATA entries that were deleted.

**System Action:** The return code determines the system action.

**Programmer Response:** See the return codes for that command in Chapter 1, "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.

**GIM20502v** GIMSMP PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS *aa*. GIMSMP IS AT LEVEL *bb.bb*.

**Explanation:**

- *aa*—return code for the SMP/E job step
- *bb.bb*—the level of SMP/E, in terms of version, release, and service level. For example, *GIMSMP LVL 16.nn* is SMP/E Release 6 service level 03.

SMP/E processing completed with the indicated return code. The *bb.bb* 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.

**GIM20503v** UNKNOWN PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS *aa*.

**Explanation:**

- *aa*—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.

**GIM20601v** *aaaaaaa* PROCESSING STOPPED BECAUSE OF THE RETURN CODE ISSUED BY THE EXIT ROUTINE.

**Explanation:**

- *aaaaaaa*—one of the following: an SMP/E command, the word *SYSMOD*, or the word *SMP/E*.

Because of the return code from an installation exit routine, one of the following occurred:

- *SYSMOD* processing stopped.
- Processing stopped for the current command.
- 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.

**GIM20701v** *aaaaaaa* PROCESSING STOPPED BECAUSE THE EXIT ROUTINE ISSUED AN UNDEFINED RETURN CODE (*bb*).

**Explanation:**

- *aaaaaaa*—one of the following: an SMP/E command or the word *SMP/E*
- *bb*—return code issued by the exit routine.

The installation exit routine issued an undefined return code.

**System Action:** Based on the exit routine SMP/E called, either the current command stops or SMP/E stops.

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

**GIM21401v** *aaaaaaaa* ENTRY *bbbbbbbb* COULD NOT BE STORED IN THE *ccccccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—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.

**GIM21402v** *aaaaaaaa* ENTRY *bbbbbbbb* COULD NOT BE STORED IN THE *ccccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—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.

**GIM21601v** UPDATE PROCESSING FOR SYSMOD *aaaaaaaa* FAILED FOR MEMBER *bbbbbbbb* IN THE *ccccccc* LIBRARY. THE RETURN CODE WAS *dd*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.

**Explanation:**

- *aaaaaaaa*—SYSMOD ID
- *bbbbbbbb*—element name
- *ccccccc*—ddname of the library being processed
- *dd*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

Update utility processing failed for the indicated element, which was part of SYSMOD *aaaaaaaa*. 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.
- 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.

**GIM21602v UPDATE PROCESSING FOR SYSMOD *aaaaaaa* WAS SUCCESSFUL FOR MEMBER *bbbbbbbb* IN THE *ccccccc* LIBRARY. THE RETURN CODE WAS *dd*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbbb*—element name
- *ccccccc*—ddname of the library being processed
- *dd*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

Update utility processing was successful for the indicated element, which was part of SYSMOD *aaaaaaa*. 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 SMP/OUT.

**Programmer Response:** None.

**GIM21801v *aaaaaaa* ENTRY *bbbbbbbb* WAS STORED IN THE *ccccccc* LIBRARY.**

**Explanation:**

- *aaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—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.

**GIM21802v *aaaaaaa* ENTRY *bbbbbbbb* WAS STORED IN THE *ccccccc* ZONE.**

**Explanation:**

- *aaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—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.

**GIM21803v MEMBER *aaaaaaa* WAS STORED IN THE *bbbbbbb* LIBRARY.**

**Explanation:**

- *aaaaaaa*—member name
- *bbbbbbb*—ddname of the library being processed.

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

**System Action:** SYSMOD processing continues.

**Programmer Response:** None.

**GIM21901v** *aaaaaaaa* ENTRY *bbbbbbbb* IN THE *ccccccc* LIBRARY WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—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:** Depending on the return code, processing stops for the entry, 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.

**GIM21902v** *aaaaaaaa* ENTRY *bbbbbbbb* IN THE *ccccccc* ZONE WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—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.

**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.

**GIM21903v** ENTRY *aaaaaaaa* IN THE *bbbbbbb* ZONE WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.

**Explanation:**

- *aaaaaaaa*—entry name
- *bbbbbbb*—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.

**GIM21904v** ENTRY *aaaaaaaa* IN THE *bbbbbbb* LIBRARY WAS NOT PROCESSED BECAUSE THE PEMAX VALUE IN THE CURRENT OPTIONS ENTRY WAS EXCEEDED.

**Explanation:**

- *aaaaaaaa*—entry name
- *bbbbbbb*—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.

**GIM22401v** *aaaaaaaa bbbbbbbb* WAS DELETED FROM THE *ccccccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbb*—element name
- *ccccccc*—ddname of the library being processed.

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

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM22402v** *aaaaaaaa bbbbbbbb* (ALIAS OF *ccccccc ddddddd*) WAS DELETED FROM THE *eeeeeee* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type of the alias
- *bbbbbbb*—entry name of the alias
- *ccccccc*—entry type of the element
- *ddddddd*—entry name of the element
- *eeeeeee*—ddname of the library being processed.

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

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM22403v MEMBER *aaaaaaaa* WAS DELETED FROM THE *bbbbbbb* LIBRARY.**

**Explanation:**

- *aaaaaaaa*—member name
- *bbbbbbb*—ddname of the library being processed.

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

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM22404v *aaaaaaaa* ENTRY *bbbbbbb* WAS DELETED FROM THE *ccccccc* ZONE.**

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbb*—entry name
- *ccccccc*—name of the zone being processed.

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

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM22601v *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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 stops for the SYSMOD or the command.

**Programmer Response:** Do the following:

1. Check previous messages to determine the cause of error.
2. Correct the error and rerun the job.

**GIM22602v *aaaaaaaa* PROCESSING WAS INCOMPLETE FOR SYSMOD *bbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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.

**GIM22701v *aaaaaaaa* PROCESSING WAS SUCCESSFUL FOR SYSMOD *bbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.

Command processing was successful for SYSMOD *bbbbbbb*.

**System Action:** None.

**Programmer Response:** None.

**GIM22801v IEANUC01 IS NOT IN THE NUCLEUS LIBRARY.**

**Explanation:** SMP/E did not find the nucleus, IEANUC01, on the nucleus library.

**System Action:** Processing for all SYSMODs that affect IEANUC01 stops.

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

- Create IEANUC01 if it was missing.
- Specify a different data set name on the NUCLEUS DD statement if it is incorrect.

Fix the problem and rerun the job.

**GIM22901v 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.

**GIM23001v IEHIOSUP PROCESSING IS COMPLETE. THE RETURN CODE WAS *aa*.**

**Explanation:**

- *aa*—return code.

SMP/E ran the IEHIOSUP program to update the TTR entries in the transfer control tables of the SVCLIB.

**System Action:** If the return code is nonzero, processing stops.

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

- If the return code is zero, no action is necessary.
- If the return code is nonzero, look at SYSPRINT output to find why IEHIOSUP did not work. Refer to *OS/VS1 Utilities*, GC26-3901.

**GIM23101v THERE IS A SYNTAX ERROR IN A ZAP CONTROL STATEMENT FOR MODULE *aaaaaaaa* IN SYSMOD *bbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbb*—SYSMOD ID.

SMP/E detected a syntax error in the IMASPZAP 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.

**GIM23401v PROGRAM *aaaaaaaa* WAS REQUIRED FOR SMP/E PROCESSING BUT WAS NOT AVAILABLE.**

**Explanation:**

- *aaaaaaaa*—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 not available, add it to the JOBLIB, STEPLIB, or LINKLIB data set. Then rerun the job.
- If you specified an incorrect utility name, correct it and rerun the job.

**GIM23402v** A BLDL FAILED FOR PROGRAM *aaaaaaaa* THAT IS REQUIRED FOR SMP/E PROCESSING. THE RETURN CODE WAS *bb*.

**Explanation:**

- *aaaaaaaa*—program name
- *bb*—return code.

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

**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.

**GIM23701v** ZAP VER PROCESSING FOR SYSMOD *aaaaaaaa* FAILED FOR MODULE *bbbbbbbb* IN LMOD *cccccccc* IN THE *ddddddd* LIBRARY. THE RETURN CODE (*ee*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.

**Explanation:**

- *aaaaaaaa*—SYSMOD ID
- *bbbbbbbb*—module name
- *cccccccc*—load module name
- *ddddddd*—ddname of the library being processed
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

ZAP utility processing failed for the indicated module, which was part of SYSMOD *aaaaaaaa*. 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.
3. Correct the error and rerun the job.

**GIM23702v** ZAP VER PROCESSING FOR SYSMOD *aaaaaaaa* WAS SUCCESSFUL FOR MODULE *bbbbbbbb* IN LMOD *cccccccc* IN THE *ddddddd* LIBRARY. THE RETURN CODE WAS *ee*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.

**Explanation:**

- *aaaaaaaa*—SYSMOD ID
- *bbbbbbbb*—module name
- *cccccccc*—load module name
- *ddddddd*—ddname of the library being processed
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.



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

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM23703v** ZAP REP PROCESSING FOR SYSMOD *aaaaaaa* FAILED FOR MODULE *bbbbbbb* IN LMOD *ccccccc* IN THE *ddddddd* LIBRARY. THE RETURN CODE (*ee*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—module name
- *ccccccc*—load module name
- *ddddddd*—ddname of the library being processed
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

ZAP utility processing failed for the indicated module, which was part of SYSMOD *aaaaaaa*. 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.
3. Correct the error and rerun the job.

**GIM23704v** ZAP REP PROCESSING FOR SYSMOD *aaaaaaa* WAS SUCCESSFUL FOR MODULE *bbbbbbb* IN LMOD *ccccccc* IN THE *ddddddd* LIBRARY. THE RETURN CODE WAS *ee*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—module name
- *ccccccc*—load module name
- *ddddddd*—ddname of the library being processed
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

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

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM23801v COPY PROCESSING TO THE *aaaaaaa* LIBRARY FAILED FOR MODULE *bbbbbbbb* IN LMOD *ccccccc* IN SYSMOD *ddddddd*. THE RETURN CODE (*ee*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.**

**Explanation:**

- *aaaaaaa*—ddname of the library into which the element was to be copied
- *bbbbbbbb*—module name
- *ccccccc*—load module name
- *ddddddd*—SYSMOD ID
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

Copy processing failed for the indicated module, which was part of SYSMOD *aaaaaaa*. 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.

Correct the error and rerun the job.

**GIM23802v COPY PROCESSING TO THE *aaaaaaa* LIBRARY WAS SUCCESSFUL FOR MODULE *bbbbbbbb* IN LMOD *ccccccc* IN SYSMOD *ddddddd*. THE RETURN CODE WAS *ee*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.**

**Explanation:**

- *aaaaaaa*—ddname of the library into which the element was to be copied
- *bbbbbbbb*—module name
- *ccccccc*—load module name
- *ddddddd*—SYSMOD ID
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

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

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM23901v LINK-EDIT PROCESSING FOR SYSMOD *aaaaaaa* FAILED FOR MODULE *bbbbbbbb* IN LMOD *cccccccc* IN THE *ddddddd* LIBRARY. THE RETURN CODE (*ee*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbbb*—module name
- *cccccccc*—load module name
- *ddddddd*—ddname of the library being processed
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

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

**Note:** An exception to this is if SMP/E is deleting CSECTs. In this case, if the return code was originally 12, SMP/E changes it to 0.

**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 SYSPRINT output to find the error. The sequence number is incremented for each utility call in an SMP/E run.
3. Correct the error and rerun the job.

**GIM23902v LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING FAILED FOR MODULE *aaaaaaa* IN LMOD *bbbbbbbb* IN THE *ccccccc* LIBRARY. THE RETURN CODE (*dd*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.**

**Explanation:**

- *aaaaaaa*—module name
- *bbbbbbbb*—load module name
- *ccccccc*—ddname of the library being processed
- *dd*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—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 if:

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

**Note:** An exception to this is if SMP/E is deleting CSECTs. In this case, if the return code was originally 12, SMP/E changes it to 0.

Message GIM23901 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 SYSPRINT output to find the error. The sequence number is incremented for each utility call in an SMP/E run.
3. Fix the error and rerun the job.

**GIM23903v LINK-EDIT PROCESSING FOR SYSMOD *aaaaaaa* WAS SUCCESSFUL FOR MODULE *bbbbbbbb* IN LMOD *ccccccc* IN THE *ddddddd* LIBRARY. THE RETURN CODE WAS *ee*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbbb*—module name
- *ccccccc*—load module name
- *ddddddd*—ddname of the library being processed
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—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 *aaaaaaa*. However, if 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 GIM23903 messages.

**System Action:** Processing continues.

**Programmer Response:** If the link-edit return code was zero, no action is needed. Otherwise, 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 non-zero 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 GIM23902.

**GIM23904v LINK-EDIT PROCESSING FOR LOAD MODULE BUILD PROCESSING WAS SUCCESSFUL FOR MODULE *aaaaaaaa* IN LMOD *bbbbbbbb* IN THE *cccccc* LIBRARY. THE RETURN CODE WAS *dd*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.**

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbbb*—load module name
- *cccccc*—ddname of the library being processed
- *dd*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—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 GIM23903 names the SYSMOD containing the modules that caused load module build processing to be done.

**Programmer Response:** None.

**GIM24001v ASSEMBLER PROCESSING FOR SYSMOD *aaaaaaaa* FAILED FOR MODULE *bbbbbbbb* IN THE *cccccc* LIBRARY. THE RETURN CODE (*dd*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss*.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID
- *bbbbbbbb*—module name
- *cccccc*—ddname of the library being processed
- *dd*—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 *aaaaaaaa*.

**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.

**Note:** 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.

3. Correct the error and rerun the job.

**GIM24002v ASSEMBLER PROCESSING FOR LOAD MODULE BUILD PROCESSING FAILED FOR MODULE *aaaaaaaa* IN THE *bbbbbbb* LIBRARY. THE RETURN CODE (*cc*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss*.**

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbb*—ddname of the library being processed
- *cc*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds.

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

**System Action:** The load module is link-edited without the module 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.

**Note:** 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.

3. Correct the error. Then do the assembly and relink the affected load modules outside of SMP/E. Message GIM67301 or GIM67302 identifies these load modules and their target libraries.

**GIM24003v ASSEMBLER PROCESSING FOR SYSMOD *aaaaaaaa* WAS SUCCESSFUL FOR MODULE *bbbbbbb* IN THE *cccccc* LIBRARY. THE RETURN CODE WAS *dd*. DATE *yy.ddd* - TIME *hh:mm:ss*.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID
- *bbbbbbb*—module name
- *cccccc*—ddname of the library being processed
- *dd*—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 *aaaaaaaa*.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM24004v ASSEMBLER PROCESSING FOR LOAD MODULE BUILD PROCESSING WAS SUCCESSFUL FOR MODULE *aaaaaaaa* IN THE *bbbbbbb* LIBRARY. THE RETURN CODE WAS *cc*. DATE *yy.ddd* - TIME *hh:mm:ss*.**

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbb*—ddname of the library being processed
- *cc*—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.

**GIM24501v RESTORE PROCESSING FAILED FOR SYSMOD *aaaaaaaa* BECAUSE OTHER SYSMODS WERE DELETED WHEN *aaaaaaaa* WAS APPLIED.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

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

**System Action:** SYSMOD processing stops. This message will be followed by GIM37001, 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.

**GIM24502v RESTORE PROCESSING FAILED FOR SYSMOD *aaaaaaaa* BECAUSE *aaaaaaaa* WAS PREVIOUSLY ACCEPTED.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SMP/E cannot restore SYSMOD *aaaaaaaa* because SYSMOD *aaaaaaaa* 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 GIM37001, 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.

**GIM24503v RESTORE PROCESSING FAILED FOR SYSMOD *aaaaaaa* BECAUSE *aaaaaaa* WAS PREVIOUSLY DELETED.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E cannot restore SYSMOD *aaaaaaa* 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 GIM37001, 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.

**GIM24504v RESTORE PROCESSING FAILED FOR SYSMOD *aaaaaaa* BECAUSE *aaaaaaa* HAS NOT BEEN APPLIED.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E cannot restore SYSMOD *aaaaaaa* because SYSMOD *aaaaaaa* 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.

**GIM24505v RESTORE PROCESSING FAILED FOR SYSMOD *aaaaaaa* BECAUSE *aaaaaaa* WAS PREVIOUSLY SUPERSEDED BY SYSMOD *bbbbbbb*.****Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—superseding SYSMOD ID.

SMP/E cannot restore SYSMOD *aaaaaaa* because it has never been applied. Instead, SYSMOD *bbbbbbb*, which supersedes SYSMOD *aaaaaaa*, 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 *bbbbbbb* should be restored, add it to the SELECT operand.

Then rerun the RESTORE command.

**GIM24506v SYSMOD *aaaaaaa* SELECTED FOR RESTORE PROCESSING IS SUPERSEDED.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* 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.

**GIM24601v** *aaaaaaaa* ENTRY *bbbbbbbb* IS NEEDED FOR PROCESSING BUT IS NOT IN THE *cccccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *cccccc*—ddname of the library being processed.

Element *bbbbbbbb* 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.

**GIM24602v** *aaaaaaaa* ENTRY *bbbbbbbb* IS NEEDED TO PROCESS SYSMOD *cccccc* BUT IS NOT IN THE *dddddd* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *cccccc*—SYSMOD ID
- *dddddd*—ddname of the library being processed.

Element *bbbbbbbb* 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 type was ASSEM, 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.

**GIM24603v** *aaaaaaaa* ENTRY *bbbbbbbb* IS NEEDED FOR LOAD MODULE BUILD PROCESSING BUT IS NOT IN THE *cccccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *cccccc*—ddname of the library being processed.

Element *bbbbbbbb* is needed for load module build 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 type was ASSEM, message GIM67301 or GIM67302 will name the affected load modules and their system libraries.

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

If the assembly failed, do the assembly and relink the affected load modules outside of SMP/E.

**GIM24605v** SYSMOD *aaaaaaaa* WAS SELECTED FOR REJECT PROCESSING BUT IT WAS NOT FOUND IN THE SMPPTS DATA SET.

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaaa* 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.

**GIM24606v** *aaaaaaaa* ENTRY *bbbbbbbb* IS NEEDED FOR PROCESSING BUT IS NOT IN THE *ccccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—name of the zone being processed.

Element *bbbbbbbb* 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.

**GIM24607v** THE *aaaaaaaa* ENTRY IS NEEDED TO PROCESS SYSMOD *bbbbbbb* BUT IS NOT IN THE *ccccccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—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.

**GIM24701v** SMP/E COULD NOT OBTAIN LINK-EDIT PARAMETERS FOR LOAD MODULE *aaaaaaaa* IN THE *bbbbbbb* LIBRARY FOR SYSMOD *ccccccc*. DEFAULTS WERE USED.

**Explanation:**

- *aaaaaaaa*—load module name
- *bbbbbbb*—ddname of the library being processed
- *ccccccc*—SYSMOD ID.

The BLDL command issued by SMP/E to get link-edit parameters for this load module failed.

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

**System Action:**

- For ACCEPT processing, SMP/E uses the following default link-edit parameters: RENT, REUS, and REFR.
- For APPLY processing, SMP/E uses the parameters in the LMOD entry. (If the SYSGEN job stream was used as input for the JCLIN command to update the target zone, the link-edit parameters were saved in the LMOD entry for the load module.)

**Programmer Response:**

- If you are accepting the SYSMOD, no response is necessary.
- If you are applying the SYSMOD, check whether the ddname of the library is incorrect.

**GIM24801v NO SYSMODS SATISFIED THE OPERANDS SPECIFIED ON THE *aaaaaaaa* COMMAND.****Explanation:**

- *aaaaaaaa*—an SMP/E command.

The operands that you specify on a command determine which SYMODs 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.
  - The levels of the elements in the distribution libraries need other SYSMODs to be restored along with UZ00001.
  - 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 Reference*.

**System Action:** For commands other than **RECEIVE**, command processing fails. Processing continues with the next command.

For **RECEIVE**, 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.

**GIM24901v SYSMOD *aaaaaaaa* FAILED BECAUSE THE SAME MODULE (*bbbbbbbb*) WAS SPECIFIED ON DIFFERENT TYPES OF ZAP NAME STATEMENTS.****Explanation:**

- *aaaaaaaa*—SYSMOD ID
- *bbbbbbbb*—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.

**GIM25101v** WHEN THE *aaaaaaaa* OPERAND IS SPECIFIED, THE *bbbbbbbb* COMMAND MUST SPECIFY AT LEAST ONE *cccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—operand
- *bbbbbbbb*—SMP/E command (REJECT or REPORT)
- *cccccc*—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.

**GIM25201v** 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.

**GIM25301v** *aaaaaaaa* ENTRY *bbbbbbbb* WAS NOT DELETED BECAUSE IT DOES NOT EXIST.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name.

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.

**GIM25401v** *aaaaaaaa* IS NOT DEFINED AS A ZONE OR A ZONESET IN THE GLOBAL ZONE.

**Explanation:**

- *aaaaaaaa*—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.

**GIM25501v ENTRY aaaaaaaa WAS UPDATED BY UCLIN.**

**Explanation:**

- aaaaaaaa—entry name.

UCLIN processing for the entry was successful.

**System Action:** Processing continues with the next UCL statement.

**Programmer Response:** None.

**GIM25601v 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.

**GIM25701v THE SPECIFIED CHANGES WERE NOT PROCESSED BECAUSE THEY WOULD LEAVE THE ENTRY INCOMPLETE. aaaaaaaa IS REQUIRED.**

**Explanation:**

- aaaaaaaa—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 aaaaaaaa.

**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.

**GIM25801v UCLIN PROCESSING FAILED BECAUSE SMP/E REACHED THE END-OF-FILE WITHOUT FINDING THE ENDUCL COMMAND.**

**ZONEEDIT PROCESSING FAILED BECAUSE SMP/E REACHED THE END-OF-FILE WITHOUT FINDING THE ENDZONEEDIT COMMAND.**

**Explanation:** Before SMP/E found the ENDUCL or ENDZONEEDIT statement, an end-of-file occurred on the SMPCTL data set.

**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.

**GIM25901v SMP/E COULD NOT ADD aaaaaaa SUBENTRY bbbbbbbb TO cccccc ENTRY ddddddd BECAUSE THE NUMBER OF ELEMENTS IN THE ENTRY EXCEEDED THE CURRENT PEMAX VALUE.**

**Explanation:**

- aaaaaaaa—subentry type
- bbbbbbbb—subentry name
- dddddddd—entry type
- cccccc—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.

**GIM25902v SMP/E COULD NOT ADD *aaaaaaaa* SUBENTRY *bbbbbbbb* TO *ccccccc* ENTRY *ddddddd* IN THE SMPSCDS DATA SET BECAUSE THE NUMBER OF ELEMENTS IN THE ENTRY EXCEEDED THE CURRENT PEMAX VALUE.**

**Explanation:**

- *aaaaaaaa*—subentry type
- *bbbbbbbb*—subentry name
- *ddddddd*—entry type
- *ccccccc*—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.

**GIM25903v SMP/E COULD NOT ADD ++ASSIGN SOURCEID *aaaaaaaa* TO SYSMOD ENTRY *bbbbbbb* BECAUSE THE NUMBER OF ELEMENTS IN THE ENTRY EXCEEDED THE CURRENT PEMAX VALUE.**

**Explanation:**

- *aaaaaaaa*—SOURCEID value
- *bbbbbbb*—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.

**GIM26001v** THE *aaaaaaaa* OPERAND WAS NOT PROCESSED BECAUSE IT IS NOT ALLOWED FOR THE SMPTLIB DATA SET.

**Explanation:**

- *aaaaaaaa*—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*.

**GIM26002v** UCLIN PROCESSING FAILED BECAUSE THE *aaaaaaaa* OPERAND IS NOT ALLOWED FOR THE SMPTLIB DATA SET.

**Explanation:**

- *aaaaaaaa*—DATASET or SYSOUT.

The SMPTLIB DDDEF entry cannot contain a data set name or SYSOUT class.

**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 or SYSOUT value and rerun the UCL statement.

For more information about SMPTLIB data sets, see *SMP/E Reference*.

**GIM26101v** THE *aaaaaaaa* OPERAND VALUE SPECIFIED ON THE UCL DEL STATEMENT DOES NOT MATCH THE EXISTING VALUE (*bbbbbbbb*) IN THE ENTRY.

**Explanation:**

- *aaaaaaaa*—value specified on the UCL DEL statement
- *bbbbbbbb*—subentry value in the existing entry.

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.

**GIM26201v** A PREVIOUS ERROR FORCED JCLIN SCANNING TO FAIL.

**Explanation:** An error explained in a previous message caused the JCLIN command to stop scanning the SMPJCLIN data set.

**System Action:** JCLIN processing stops.

**Programmer Response:** Fix the previous error and rerun the job.

**GIM26301v** AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. THE MOD OPERAND WAS NOT SPECIFIED ON THE EXEC STATEMENT FOR AN ASSEMBLER STEP.

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—job name or NONAME.

SMP/E did not find the MOD operand on the EXEC statement for assembler processing. 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.

---

**GIM26302v AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. AN UNKNOWN ERROR OCCURRED FOR A MODULE IN A LINK-EDIT STEP.**

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26303v AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. THERE IS A SYNTAX ERROR IN THE COPY CONTROL STATEMENT.**

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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 the utilities manual for your operating system for the correct syntax of copy control statements. Fix the error and rerun the job.

**GIM26304v AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. SMP/E DID NOT RECOGNIZE THE CHARACTERS ON THE MOD OPERAND IN A LINK-EDIT STEP.**

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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 the linkage editor and loader manual for your operating system for the correct syntax of linkage editor control statements. Fix the error and rerun the job.



**GIM26305v AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. THE MACRO NAME MUST BE FROM 1 TO 8 CHARACTERS LONG.**

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26306v AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. AN I/O ERROR OCCURRED IN THE LAST LINE PROCESSED BY SMP/E.**

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26307v AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. THERE IS A SYNTAX ERROR IN A LINK-EDIT CONTROL STATEMENT.**

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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 the linkage editor and loader manual for your operating system for the correct syntax of link-edit control statements. Fix the error and rerun the job.

**GIM26308v** AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. SMP/E RAN OUT OF STORAGE WHILE PROCESSING AN ASSEMBLER STEP.

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26309v** AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. THE MODULE NAME WAS NOT SPECIFIED ON A SYSLMOD DD STATEMENT OR ON A NAME STATEMENT.

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26310v** AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. THE PGM OPERAND WAS SPECIFIED ON THE EXEC STATEMENT, BUT NO SYSLMOD DD STATEMENT WAS FOUND.

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26311v** AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. SMP/E RAN OUT OF STORAGE WHILE PROCESSING A LINK-EDIT STEP.

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26312v** AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. THE DSNAME OPERAND WAS NOT SPECIFIED ON THE EXEC STATEMENT FOR A LINK-EDIT STEP.

**Explanation:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26313v** AN ERROR OCCURRED DURING JCLIN PROCESSING OF STEP *aaaaaaaa* IN JOB *bbbbbbbb*. 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:**

- *aaaaaaaa*—step name or NONAME
- *bbbbbbbb*—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.

**GIM26401v** THE *aaaaaaaa* DDDEF ENTRY CANNOT HAVE MORE THAN ONE VOLUME SERIAL NUMBER.

**Explanation:**

- *aaaaaaaa*—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.

**GIM26501v THE *aaaaaaaa* SUBENTRY WAS NOT CHANGED.****Explanation:**

- *aaaaaaaa*—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.

**GIM26601v 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.

**GIM26602v 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.

**GIM26603v 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.

**GIM26701v DIRECTORY SPACE FOR THE *aaaaaaaa* LIBRARY WAS EXCEEDED DURING AN ATTEMPT TO STORE *bbbbbbbb* ENTRY *ccccccc*.****Explanation:**

- *aaaaaaaa*—ddname of the library being processed
- *bbbbbbbb*—entry type
- *ccccccc*—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.

**GIM26801v AN I/O ERROR OCCURRED DURING AN ATTEMPT TO STORE *aaaaaaaa* *bbbbbbbb* ON THE *ccccccc* LIBRARY.****Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—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.

**GIM26901v AN I/O ERROR OCCURRED DURING BLDL PROCESSING FOR *aaaaaaaa* *bbbbbbbb* ON THE *cccccc* LIBRARY.**

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *cccccc*—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.

**GIM26902v AN I/O ERROR OCCURRED DURING BLDL PROCESSING FOR *aaaaaaaa* *bbbbbbbb* ON THE *cccccc* ZONE.**

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *cccccc*—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.

**GIM27001v THE *aaaaaaa* LIBRARY WAS NOT PROTECTED BECAUSE RACF IS NOT ACTIVE OR NOT INSTALLED.**

**Explanation:**

- *aaaaaaa*—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.

**GIM27002v DATA SET *aaaaaaaa* WAS NOT PROTECTED BECAUSE RACF IS NOT ACTIVE OR NOT INSTALLED.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM27101v SMPTLIB aaaaaaaaa WAS NOT PROTECTED BECAUSE THE USER IS NOT DEFINED TO RACF.****Explanation:**

- *aaaaaaaa*—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.

**GIM27201v THE aaaaaaa OPERAND IS NOT ALLOWED IN REJECT MASS MODE.****Explanation:**

- *aaaaaaaa*—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 Reference* 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.

**GIM27301v THE ELEMENT OR JCLIN DATA (TEXT OR OBJECT) IS MISSING.****Explanation:**

During RECEIVE 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 SMPJCLIN, RELFILE, 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 TXLIB, LKLIB, or RELFILE data set.
- Add JCL to the JCLIN input data set.

Then rerun the RECEIVE command.

**GIM27401v** I/O ERROR - *jobname, stepname, address, device, lib, action, description, access\_method*

I/O ERROR - *jobname, stepname, address, device, lib, action, description, rbn, access\_method*

I/O ERROR - *jobname, stepname, address, device, lib, action, description, track\_address, block\_number, access\_method*

**Explanation:** An I/O error occurred when SMP/E was processing the indicated data set. The message corresponds to the SYNADAF information listed in the data management services guide or data administration guide for your operating system.

**System Action:** The messages that follow this message show SMP/E's action.

**Programmer Response:** Fix the error and rerun the job.

**GIM27501v** YOU MAY WANT TO REJECT FUNCTION *aaaaaaa* BECAUSE IT NO LONGER APPLIES TO THE GLOBAL ZONE.

**Explanation:**

- *aaaaaaa*—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(*aaaaaaa*) 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.

**GIM27601v** *aaaaaaa* AND *bbbbbbbb* ARE MUTUALLY EXCLUSIVE OPERANDS.

**Explanation:**

- *aaaaaaa*—first operand
- *bbbbbbbb*—second operand.

The indicated operands are not allowed on the same command, MCS, or UCL statement.

**System Action:**

- If the operands were specified on a command or UCL statement, processing stops for that command or UCL statement.
- If the operands were specified on an MCS statement, SYSMOD processing stops.

**Programmer Response:** Fix the error and rerun the job.

**GIM27602v** A SPECIFIED OPERAND IS NOT ALLOWED FOR THE *aaaaaaaaa* DATA SET.

**Explanation:**

- *aaaaaaaaa*—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 Reference* to see which operands are allowed for the data set being processed.

Fix the error and rerun the job.

**GIM27701v** *aaaaaaaa* ENTRY *bbbbbbbb* WAS ADDED INSTEAD OF REPLACED BECAUSE IT DID NOT EXIST.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name.

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.

**GIM27801v** 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.

**GIM27901v** A VSAM ERROR OCCURRED DURING AN ATTEMPT TO ACCESS AN SMPCSI DATA SET – VPLFUNCT = *aaaaaa*.

**Explanation:**

- *aaaaaa*—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.

*SMP/E Diagnosis Guide* contains more information about how to handle VSAM problems.



**GIM28002v SMPTLIB *aaaaaaaa* WAS NOT DELETED BECAUSE VOLUME INFORMATION WAS NOT SPECIFIED.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM28101v SYSMOD *aaaaaaa* CONTAINS MORE THAN ONE MCS FOR ELEMENT *bbbbbbbb*. ONLY ONE IS ALLOWED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbbb*—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.

**GIM28201v ZONE *aaaaaaaa* WAS NOT PROCESSED BECAUSE IT IS A DLIB ZONE AND TARGETZONE WAS SPECIFIED.**

**ZONE *aaaaaaaa* WAS NOT PROCESSED BECAUSE IT IS A TARGET ZONE AND DLIBZONE WAS SPECIFIED.**

**Explanation:**

- *aaaaaaaa*—zone name.

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.

**GIM28301v THE *aaaaaaa* DD STATEMENT IS MISSING FOR LOAD MODULE *bbbbbbbb*. THE SYSMOD IS *ccccccc*.**

**Explanation:**

- *aaaaaaa*—ddname of the required library
- *bbbbbbbb*—load module name
- *ccccccc*—SYSMOD ID.

The DD statement needed to process the load module is missing or specifies the wrong ddname.

**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 ddname.

Then rerun the job.

**GIM28302v THE *aaaaaaa* DD STATEMENT IS MISSING FOR COMPRESS PROCESSING.****Explanation:**

- *aaaaaaa*—ddname of the required library.

The DD statement needed for compress processing is missing or specifies the wrong ddname.

**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 ddname.

Then rerun the job.

**GIM28303v THE *aaaaaaa* DD STATEMENT IS MISSING FOR SYSMOD *bbbbbbb*.****Explanation:**

- *aaaaaaa*—ddname of the required library
- *bbbbbbb*—SYSMOD ID.

The DD statement needed to process the SYSMOD is missing or specifies the wrong ddname.

**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 ddname.

Then rerun the job.

**GIM28304v THE SMPLOGA DD STATEMENT IS MISSING.**

**Explanation:** The SMPLOGA DD statement is missing or specifies the wrong ddname.

**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 ddname.

Then rerun the job.

**GIM28305v 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.

**GIM28306v THE *aaaaaaa* DD STATEMENT IS MISSING.****Explanation:**

- *aaaaaaa*—ddname of the required library.

The DD statement needed for SMP/E processing is missing or specifies the wrong ddname.

**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 ddname.

Then rerun the job.

**GIM28307v** THE *aaaaaaa* DD STATEMENT IS MISSING FOR ELEMENT *bbbbbbb*. IT IS REQUIRED FOR SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaa*—ddname of the required library
- *bbbbbbb*—element name
- *ccccccc*—SYSMOD ID.

The DD statement needed to process the element is missing or specifies the wrong ddname.

**System Action:** Subsequent messages in SMP/OUT 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 ddname.

Then rerun the job.

**GIM28401v** THE LOAD MODULE ON THE ZAP NAME STATEMENT FOR MODULE *aaaaaaaa* IN SYSMOD *bbbbbbb* IS NOT AN LMOD SUBENTRY IN THE MOD ENTRY FOR *aaaaaaaa*.

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbb*—SYSMOD ID.

The load module name specified on an IMASPZAP 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 IMASPZAP 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.

**GIM28501v** COMPRESS PROCESSING FAILED FOR THE *aaaaaaa* LIBRARY BECAUSE IT IS AN UNMOVABLE DATA SET.

**Explanation:**

- *aaaaaaa*—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.

**GIM28701v** 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 SMP/OUT data to find what command caused SMP/E to bypass the current command. Fix any errors and decide whether to rerun the job.

**GIM28801v TARGET ZONE *aaaaaaa* WAS NOT UPDATED BECAUSE JCLIN PROCESSING FAILED.**

**Explanation:** 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.

**GIM29201v 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.

**GIM30201v *aaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. ALL SYSMODS THAT WOULD HAVE DELETED *bbbbbbb* HAVE FAILED.**

**Explanation:**

- *aaaaaaa*—SMP/E command
- *bbbbbbb*—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.

**GIM30202v *aaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. ALL SYSMODS THAT WOULD HAVE SUPERSEDED *bbbbbbb* HAVE FAILED.**

**Explanation:**

- *aaaaaaa*—SMP/E command
- *bbbbbbb*—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 one of the following:

- 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.

Fix the error and rerun the job.

**GIM30203v *aaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING FOR AN ELEMENT DELETED BY *bbbbbbb* FAILED.**

**Explanation:**

- *aaaaaaa*—SMP/E command
- *bbbbbbb*—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:** Do one of the following:

- 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.

Fix the error and rerun the job.

**GIM30204v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. REQUIRED SYSMODS WERE EXCLUDED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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:** Do one of the following:

- 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.

Fix the error and rerun the job.

**GIM30205v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. THE SYSMOD SPECIFIED ON THE ++VER FMID OPERAND FAILED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.

SMP/E could not process the indicated SYSMOD because the owning function failed.

**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 elements of the SYSMOD.
- Use the APPLY/RESTORE/ACCEPT Element Summary report to find the associated SYSMOD that caused the failure.

Fix the error and rerun the job.

**GIM30206v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. HOLD REASON IDS WERE NOT RESOLVED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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 GIM359xx, 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:** Do one of the following:

- 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.

Fix the error and rerun the job.

**GIM30207v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. THE SYSMOD SPECIFIED ON THE ++IF REQ OPERAND FAILED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.

SMP/E could not process the indicated SYSMOD because a conditional requisite failed.

**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 SMP/OUT 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.

Fix the error and rerun the job.

**GIM30208v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. INLINE JCLIN PROCESSING FAILED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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 TXLIB or RELFILE data set.)

**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 SMP/OUT 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.

Fix the error and rerun the job.

**GIM30209v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. REQUIRED SYSMODS FAILED OR WERE MISSING.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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 GIM359xx, 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 SMP/OUT 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.

Fix the error and rerun the job.

**GIM30210v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. A SYSMOD SPECIFIED ON THE +-+VER PRE OPERAND FAILED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.

SMP/E could not process the indicated SYSMOD because a prerequisite SYSMOD failed.

**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 elements of the SYSMOD.
- Use the APPLY/RESTORE/ACCEPT Element Summary report to find the associated SYSMOD that caused the failure.

Fix the error and rerun the job.

**GIM30211v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. A SYSMOD CONTAINING HIGHER LEVELS OF COMMON ELEMENTS FAILED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.

The indicated SYSMOD had a version of an element that SMP/E did not select for processing because another SYSMOD was processed that had a higher level of the element. However, that other SYSMOD failed. The SMP/E reports and messages will help you determine which element caused the problem.

**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 elements of the SYSMOD.
- Use the APPLY/RESTORE/ACCEPT Element Summary report to find the associated SYSMOD that caused the failure.

Fix the error and rerun the job.

**GIM30212v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. A SYSMOD SPECIFIED ON THE +-+VER REQ OPERAND FAILED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.

SMP/E did not process the indicated SYSMOD because a corequisite SYSMOD failed.

**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 elements of the SYSMOD.
- Use the APPLY/RESTORE/ACCEPT Element Summary report to find the associated SYSMOD that caused the failure.

Fix the error and rerun the job.

**GIM30213v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. SYSMOD *ccccccc*, WHICH IS SUPERSEDED BY SYSMOD *bbbbbbb*, MUST ALSO BE SPECIFIED ON THE RESTORE COMMAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.
- *ccccccc*—superseded SYSMOD ID.

SMP/E did not process SYSMOD *bbbbbbb* because it supersedes SYSMOD *ccccccc* and SYSMOD *ccccccc* 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:** Do one of the following:

- 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 SYSMOD *ccccccc* on the RESTORE command.

Fix the error and rerun the job.

**GIM30213v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. THE SYSMOD THAT *bbbbbbb* SUPERSEDES MUST ALSO BE SPECIFIED ON THE RESTORE COMMAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.

SMP/E did not process the indicated SYSMOD because a SYSMOD that it supersedes was not specified on the RESTORE command. Message GIM359xx names the superseded SYSMOD.

**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 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.

**GIM30214v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. AN ERROR OCCURRED WHEN SMP/E TRIED TO UPDATE THE SMPCSI DATA SET.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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:** Do one of the following:

- 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.

**GIM30215v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. A SYSTEM ABEND OCCURRED.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID.

SMP/E did not process the indicated SYSMOD because a system abend occurred. Message GIM43201 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:** Do one of the following:

- 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.

**GIM30216v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. SYSTEM UTILITY PROCESSING FAILED FOR AN ELEMENT IN *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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:** Do one of the following:

- 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.

**GIM30217v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. RETRANSFORMATION PROCESSING FAILED FOR AN ELEMENT IN *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—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:** Do one of the following:

- Examine SMPOUT for messages relating to elements in 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.

**GIM30218v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING IS INCOMPLETE FOR SYSMOD *ccccccc*, WHICH WAS SPECIFIED ON THE ++VER PRE OPERAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—related SYSMOD ID.

SMP/E could not process SYSMOD *bbbbbbb* because processing is incomplete for prerequisite SYSMOD *ccccccc*.

**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 *ccccccc*. Fix the error and rerun the job.

**GIM30219v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING FAILED FOR SYSMOD *ccccccc*, WHICH WAS SPECIFIED ON THE ++VER PRE OPERAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—related SYSMOD ID.

SMP/E could not process SYSMOD *bbbbbbb* because processing failed for prerequisite SYSMOD *ccccccc*.

**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 *ccccccc*. Fix the error and rerun the job.

**GIM30220v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING IS INCOMPLETE FOR SYSMOD *ccccccc*, WHICH WAS SPECIFIED ON THE ++VER REQ OPERAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—related SYSMOD ID.

SMP/E could not process SYSMOD *bbbbbbb* because processing is incomplete for requisite SYSMOD *ccccccc*.

**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 cccccc. Fix the error and rerun the job.

**GIM30221v** aaaaaaaa PROCESSING FAILED FOR SYSMOD bbbbbbb. PROCESSING FAILED FOR SYSMOD cccccc, WHICH WAS SPECIFIED ON THE ++VER REQ OPERAND.

**Explanation:**

- aaaaaaaa—SMP/E command
- bbbbbbb—SYSMOD ID
- cccccc—related SYSMOD ID.

SMP/E could not process SYSMOD bbbbbbb because processing failed for requisite SYSMOD cccccc.

**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 cccccc. Fix the error and rerun the job.

**GIM30222v** aaaaaaaa PROCESSING FAILED FOR SYSMOD bbbbbbb. PROCESSING IS INCOMPLETE FOR SYSMOD cccccc, WHICH WAS SPECIFIED ON THE ++IF REQ OPERAND.

**Explanation:**

- aaaaaaaa—SMP/E command
- bbbbbbb—SYSMOD ID
- cccccc—related SYSMOD ID.

SMP/E could not process SYSMOD bbbbbbb because processing is incomplete for conditional requisite SYSMOD cccccc.

**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 cccccc. Fix the error and rerun the job.

**GIM30223v** aaaaaaaa PROCESSING FAILED FOR SYSMOD bbbbbbb. PROCESSING FAILED FOR SYSMOD cccccc, WHICH WAS SPECIFIED ON THE ++IF REQ OPERAND.

**Explanation:**

- aaaaaaaa—SMP/E command
- bbbbbbb—SYSMOD ID
- cccccc—related SYSMOD ID.

SMP/E could not process SYSMOD bbbbbbb because processing failed for conditional requisite SYSMOD cccccc.

**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 cccccc. Fix the error and rerun the job.

**GIM30224v** aaaaaaaa PROCESSING FAILED FOR SYSMOD bbbbbbb. PROCESSING IS INCOMPLETE FOR SYSMOD cccccc, WHICH WAS SPECIFIED ON THE ++VER FMID OPERAND.

**Explanation:**

- aaaaaaaa—SMP/E command
- bbbbbbb—SYSMOD ID
- cccccc—related SYSMOD ID.

SMP/E could not process SYSMOD bbbbbbb because processing is incomplete for function SYSMOD cccccc.

**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 *cccccc*. Fix the error and rerun the job.

**GIM30224v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING IS INCOMPLETE FOR SYSMOD *cccccc*, WHICH WAS SPECIFIED ON THE ++IF FMID OPERAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID
- *cccccc*—related SYSMOD ID.

SMP/E could not process SYSMOD *bbbbbbb* because processing is incomplete for SYSMOD *cccccc*, where *cccccc* is the function for which a conditional requisite was specified.

**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 *cccccc*. Fix the error and rerun the job.

**GIM30225v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING FAILED FOR SYSMOD *cccccc*, WHICH WAS SPECIFIED ON THE ++VER FMID OPERAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID
- *cccccc*—related SYSMOD ID.

SMP/E could not process SYSMOD *bbbbbbb* because processing is incomplete for function SYSMOD *cccccc*.

**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 *cccccc*. Fix the error and rerun the job.

**GIM30225v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING FAILED FOR SYSMOD *cccccc*, WHICH WAS SPECIFIED ON THE ++IF FMID OPERAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID
- *cccccc*—related SYSMOD ID.

SMP/E could not process SYSMOD *bbbbbbb* because processing failed for SYSMOD *cccccc*, where *cccccc* is the function for which a conditional requisite was specified.

**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 *cccccc*. Fix the error and rerun the job.

**GIM30226v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING IS INCOMPLETE FOR SYSMOD *cccccc*, WHICH CONTAINS HIGHER LEVELS OF COMMON ELEMENTS.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID
- *cccccc*—related SYSMOD ID.

SMP/E could not process SYSMOD *bbbbbbb* because processing is incomplete for SYSMOD *cccccc*, where *cccccc* contains higher levels of elements that also exist in SYSMOD *bbbbbbb*.

**System Action:** SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

**Programmer Response:** Examine SMP/OUT for messages relating to SYSMOD *ccccccc*. Fix the error and rerun the job.

**GIM30227v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb*. PROCESSING FAILED FOR SYSMOD *ccccccc*, WHICH CONTAINS HIGHER LEVELS OF COMMON ELEMENTS.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—related SYSMOD ID.

SMP/E could not process SYSMOD *bbbbbbb* because processing failed for SYSMOD *ccccccc*, where *ccccccc* contains higher levels of elements that also exist in SYSMOD *bbbbbbb*.

**System Action:** SYSMOD processing stops. SMP/E does not attempt to process any other elements in the SYSMOD.

**Programmer Response:** Examine SMP/OUT for messages relating to SYSMOD *ccccccc*. Fix the error and rerun the job.

**GIM30301v** COMPRESS PROCESSING FAILED FOR THE *aaaaaaaa* LIBRARY. THE RETURN CODE WAS *bb*.

**Explanation:**

- *aaaaaaaa*—ddname of the library
- *bb*—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.

**GIM30302v** COMPRESS PROCESSING WAS SUCCESSFUL FOR THE *aaaaaaaa* LIBRARY. THE RETURN CODE WAS *bb*.

**Explanation:**

- *aaaaaaaa*—ddname of the library
- *bb*—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 either zero or is not greater than one of the following:

- The maximum return code specified in the UTILITY entry
- The default return code.

**Programmer Response:** None.

**GIM31901v** SYSMOD *aaaaaaaa* DOES NOT SPECIFY *bbbbbbb* ON THE PRE OR SUP OPERAND. *bbbbbbb* IS THE RMID FOR *ccccccc dddddddd* THAT IS CURRENTLY INSTALLED.

**Explanation:**

- *aaaaaaaa*—ID of the SYSMOD being installed
- *bbbbbbb*—ID of the SYSMOD specified as an RMID
- *ccccccc*—element type
- *ddddddd*—element name.

This message describes the ID check reported by message GIM38201 or GIM38202. SMP/E issues GIM31901 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 *aaaaaaa* 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 GIM38201 or GIM38202 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 *aaaaaaa* to see if it was already installed.
- Reject SYSMOD *aaaaaaa* 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 *aaaaaaa* to establish the correct relationship.

**GIM31902v SYSMOD *aaaaaaa* DOES NOT SPECIFY *bbbbbbb* ON THE PRE OR SUP OPERAND. *bbbbbbb* IS A UMID FOR *ccccccc ddddddd* THAT IS CURRENTLY INSTALLED.**

**Explanation:**

- *aaaaaaa*—ID of the SYSMOD being installed
- *bbbbbbb*—ID of the SYSMOD specified as a UMID
- *ccccccc*—element type
- *ddddddd*—element name.

This message describes the ID check reported by message GIM38201 or GIM38202. SMP/E issues GIM31902 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 *aaaaaaa* 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 GIM38201 or GIM38202 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 *aaaaaaa* to see if it was already installed.
- Reject SYSMOD *aaaaaaa* and change the PRE and SUP operands that were specified. This is only necessary if SYSMOD *aaaaaaa* replaces the indicated element. In that case, it must specify a relationship with all the UMIDs for the element being replaced. If SYSMOD *aaaaaaa* 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 *aaaaaaa* to establish the correct relationship.

**GIM31903v SYSMOD *aaaaaaa* DOES NOT SPECIFY *bbbbbbb* ON THE PRE OR SUP OPERAND. *bbbbbbb* IS THE RMID FOR *ccccccc ddddddd* THAT IS BEING INSTALLED.**

**Explanation:**

- *aaaaaaa*—ID of the SYSMOD being installed
- *bbbbbbb*—ID of the SYSMOD specified as an RMID
- *ccccccc*—element type
- *ddddddd*—element name.

This message describes the ID check reported by message GIM38201 or GIM38202. SMP/E issues GIM31903 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 *aaaaaaa* 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. Look at the preceding GIM38201 or GIM38202 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 *aaaaaaa* to see if it was already installed.
- Reject SYSMOD *aaaaaaa* 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 *aaaaaaa* to establish the correct relationship.

**GIM31904v SYSMOD *aaaaaaa* DOES NOT SPECIFY *bbbbbbb* ON THE PRE OR SUP OPERAND. *bbbbbbb* IS A UMID FOR *cccccccc dddddddd* THAT IS BEING INSTALLED.**

**Explanation:**

- *aaaaaaa*—ID of the SYSMOD being installed
- *bbbbbbb*—ID of the SYSMOD specified as a UMID
- *cccccccc*—element type
- *ddddddd*—element name.

This message describes the ID check reported by message GIM38201 or GIM38202. SMP/E issues GIM31904 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 *aaaaaaa* 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. Look at the preceding GIM38201 or GIM38202 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 *aaaaaaa* to see if it was already installed.
- Reject SYSMOD *aaaaaaa* and change the PRE and SUP operands that were specified. This is only necessary if SYSMOD *aaaaaaa* replaces the indicated element. In that case, it must specify a relationship with all the UMIDs for the element being replaced. If SYSMOD *aaaaaaa* 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 *aaaaaaa* to establish the correct relationship.

**GIM32401v SYSMOD *aaaaaaa* REGRESSES *bbbbbbb ccccccc*, WHICH WAS PREVIOUSLY REPLACED BY SYSMOD *ddddddd*.**

**Explanation:**

- *aaaaaaa*—ID of the SYSMOD being processed
- *bbbbbbb*—element type
- *ccccccc*—element name
- *ddddddd*—ID of the SYSMOD that was previously processed.

SYSMOD *aaaaaaa* should have superseded SYSMOD *ddddddd* or specified it as a prerequisite because SYSMOD *ddddddd* has already been processed. **One of the following is true:**

- SYSMOD *ddddddd* has already been applied.
- SYSMOD *ddddddd* has already been accepted.
- SYSMOD *ddddddd* is being installed concurrently with SYSMOD *aaaaaaa*.

Because the relationship between the SYSMODs is not defined, SMP/E considers SYSMOD *ddddddd* to be regressed by SYSMOD *aaaaaaa*.

**System Action:** SYSMOD processing continues.

**Programmer Response:** None.

**GIM32402v** SYSMOD *aaaaaaa* REGRESSES *bbbbbbb* *ccccccc*, WHICH WAS PREVIOUSLY UPDATED BY SYSMOD *ddddddd*.

**Explanation:**

- *aaaaaaa*—ID of the SYSMOD being processed
- *bbbbbbb*—element type
- *ccccccc*—element name
- *ddddddd*—ID of the SYSMOD that was previously processed.

SYSMOD *aaaaaaa* should have superseded SYSMOD *ddddddd* or specified it as a prerequisite because SYSMOD *ddddddd* had already been processed. **One of the following is true:**

- SYSMOD *ddddddd* has already been applied.
- SYSMOD *ddddddd* has already been accepted.
- SYSMOD *ddddddd* is being installed concurrently with SYSMOD *aaaaaaa*.

Because the relationship between the SYSMODs is not defined, SMP/E considers SYSMOD *ddddddd* to be regressed by SYSMOD *aaaaaaa*.

**System Action:** SYSMOD processing continues.

**Programmer Response:** None.

**GIM32501v** SYSMOD *aaaaaaa* SUPERSEDES SYSMOD *bbbbbbb* BUT DOES NOT CONTAIN ELEMENT *ccccccc*.

**Explanation:**

- *aaaaaaa*—ID of the superseding SYSMOD
- *bbbbbbb*—ID of the superseded SYSMOD
- *ccccccc*—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.

**GIM32701v** INLINE DATA IS NOT ALLOWED AFTER AN MCS WHEN EITHER THE LKLIB, TXLIB, RELFILE, OR DELETE 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
  - LKLIB
  - RELFILE
  - TXLIB
- The ++JCLIN MCS specifies one of the following operands:
  - 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 LKLIB, TXLIB, RELFILE, or DELETE operands.

Fix the error and rerun the job.

**GIM33801v THE ++aaaaaaa MCS IS NOT ALLOWED IN FUNCTION SYSMODS.**

**Explanation:**

- *aaaaaaa*—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.

**GIM33901v THE aaaa OPERAND IS ONLY ALLOWED IN FUNCTION SYSMODS.**

**Explanation:**

- *aaaa*—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.

**GIM34001v 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.

**GIM34201v 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.

**GIM34202v MORE THAN ONE aaaaaaa OPERAND WAS SPECIFIED ON AN MCS. ONLY ONE IS ALLOWED.**

**Explanation:**

- *aaaaaaa*—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.

**GIM34301v 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.

**GIM34501v THE MCS CONTAINS AN ERROR.**

**Explanation:** During RECEIVE processing, SMP/E found an MCS that contained an error.

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

**Programmer Response:** Check the following:

1. The MCS associated with the message:
  - Is its syntax correct?
  - Is it in the right order with other MCS statements in the SYSMOD?
2. The input that SMP/E is processing:
  - Is the input in uppercase letters?
3. The MCS type:
  - Is it is on the same line as the ++?

Rebuild the SYSMOD and run the job again.

**GIM34601v 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.

**GIM34701v 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.

**GIM34801v 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.

**GIM34901v 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.

**GIM35001v 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.

**GIM35101v SYSMOD *aaaaaaaa* FAILED BECAUSE OF AN ERROR THAT OCCURRED WHILE RELFILES WERE BEING LOADED. THE RETURN CODE FROM THE COPY UTILITY WAS *bb*.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID
- *bb*—return code from copy processing.

An error occurred while the copy utility was loading relative files for the indicated SYSMOD.

**System Action:** SYSMOD processing stops.

**Programmer Response:** Look at the SYSPRINT output for the copy utility to determine the problem.

**GIM35201v SMPTLIB *aaaaaaaaaa* WAS ALLOCATED AND CATALOGED ON VOLUME *bbbbbb*.**

**Explanation:**

- *aaaaaaaaaa*—SMPTLIB data set name
- *bbbbbb*—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.

**GIM35202v SMP/E USED SMPTLIB *aaaaaaaaaa*, WHICH ALREADY EXISTS ON VOLUME *bbbbbb*.**

**Explanation:**

- *aaaaaaaaaa*—SMPTLIB data set name
- *bbbbbb*—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.

**GIM35301v SMPTLIB *aaaaaaaa* WAS NOT ALLOCATED ON VOLUME *bbbbbb* BECAUSE IT IS NOT A PARTITIONED DATA SET.**

**Explanation:**

- *aaaaaaaa*—data set name
- *bbbbbb*—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.

**GIM35302v SMPTLIB *aaaaaaaa* WAS NOT ALLOCATED ON VOLUME *bbbbbb* BECAUSE ERROR CODE *cccc* WAS ISSUED BY DYNAMIC ALLOCATION.**

**Explanation:**

- *aaaaaaaa*—data set name
- *bbbbbb*—volume serial number
- *cccc*—error code issued by dynamic allocation.

During RECEIVE processing, SMP/E tried to allocate the SMPTLIB data set on the indicated volume in order 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, 47B0	DADSM error
47A8—47AC	RACF 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 the system macros and facilities manual for your operating system for more information about the error code.

**GIM35304v SMPTLIB *aaaaaaaa* WAS NOT ALLOCATED BECAUSE NO VOLUMES WERE SPECIFIED.**

**Explanation:**

- *aaaaaaaa*—data set name.

During RECEIVE processing, SMP/E tried to allocate the SMPTLIB data set on the indicated volume in order to load relative files for a SYSMOD. However, no volumes were specified in the SMPTLIB DDDEF entry or on the SMPTLIB DD statement.

**System Action:** Command processing stops.

**Programmer Response:** Specify the desired volumes in the SMPTLIB DDDEF entry or on the SMPTLIB DD statement.

**GIM35305v SMPTLIB aaaaaaaaa WAS NOT ALLOCATED BECAUSE IT IS DEFINED AS A DUMMY DATA SET.**

**Explanation:**

- aaaaaaaaa—data set name.

During RECEIVE processing, SMP/E tried to allocate the SMPTLIB data set on the indicated volume in order 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.

**GIM35306v SMPTLIB aaaaaaaaa WAS NOT ALLOCATED BECAUSE ERROR CODE bbbb WAS ISSUED BY DYNAMIC ALLOCATION.**

**Explanation:**

- aaaaaaaaa—data set name
- bbbb—error code issued by dynamic allocation.

During RECEIVE processing, SMP/E tried to allocate the SMPTLIB data set in order to load relative files for a SYSMOD. However, an error occurred during dynamic allocation.

Error codes from dynamic allocation fall into the following ranges:

<b>Codes</b>	<b>Meaning</b>
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, 47B0	DADSM error
47A8—47AC	RACF 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 the system macros and facilities manual for your operating system for more information about the error code.

**GIM35307v SMPTLIB aaaaaaaaa FOR bbbbbbb WAS NOT ALLOCATED BECAUSE THE DATA SET WAS NOT FOUND.**

**Explanation:**

- aaaaaaaaa—data set name
- bbbbbbb—SYSMOD ID.

Allocation failed for SMPTLIB bbbbbbb 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 relfile data set was allocated at RECEIVE time. If the relfile data set has been deleted, receive the SYSMOD again.

**GIM35401v SMPTLIB aaaaaaaaa WAS DELETED AND UNCATALOGED FROM VOLUME bbbbbb.**

**Explanation:**

- aaaaaaaaa—data set name
- bbbbbb—volume serial number.

SMP/E deleted and uncataloged the SMPTLIB data set from the indicated volume.

**System Action:** SYSMOD processing continues.

**Programmer Response:** None.

**GIM35501v AN ERROR OCCURRED DURING PROCESSING OF THE aaaaaaa ENTRY FOR SYSMOD bbbbbb IN THE cccccc LIBRARY.**

**Explanation:**

- aaaaaaa—entry type
- bbbbbb—SYSMOD ID
- cccccc—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.

**GIM35601v aaaaaaa SUBENTRY bbbbbb WAS NOT ADDED BECAUSE IT ALREADY EXISTS.**

**Explanation:**

- aaaaaaa—subentry type
- bbbbbb—subentry name (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.

**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.

**GIM35701v aaaaaaa SUBENTRY bbbbbb WAS NOT DELETED BECAUSE IT DOES NOT EXIST.**

**Explanation:**

- aaaaaaa—subentry type
- bbbbbb—subentry 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 in this UCL statement.

**Programmer Response:** Rerun the UCL statement without the specified subentry.

**GIM35801v** *aaaaaaaa* SUBENTRY *bbbbbbbb* WAS ADDED INSTEAD OF REPLACED BECAUSE IT DID NOT EXIST.

**Explanation:**

- *aaaaaaaa*—subentry type
- *bbbbbbbb*—subentry name.

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.

**GIM35901v** ERROR HOLD *aaaaaaaa* WAS NOT RESOLVED.

**Explanation:**

- *aaaaaaaa*—error hold reason ID.

SMP/E could not process the SYSMOD named in GIM30206 because it was held for the indicated error reason ID (APAR), which was not resolved.

**System Action:** See message GIM30206.

**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 GIM30206 along with or after the resolving SYSMOD.
- Rerun the ACCEPT or APPLY command to install the SYSMOD named in GIM30206, but specify **BYPASS(HOLDERR(*aaaaaaaa*))**.

**GIM35902v** SYSTEM HOLD *aaaaaaaa* WAS NOT RESOLVED.

**Explanation:**

- *aaaaaaaa*—system hold reason ID.

One of the following occurred:

- GIM30206 was also issued.

SMP/E could not process the SYSMOD named in GIM30206 because it was held for the indicated system reason ID, which was not resolved.

- A GIM302xx message other than GIM30206 was also issued.

The previous GIM302xx (for example, GIM30209) indicates why APPLY or ACCEPT processing failed for the indicated SYSMOD. Message GIM35902 is issued along with other GIM359xx messages, even if you specified **BYPASS(HOLDSYSTEM)**.

**System Action:** See the description of the associated GIM302xx message that was issued.

**Programmer Response:**

- GIM30206 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 GIM30206, but specify **BYPASS(HOLDSYS(*aaaaaaaa*))**.

- A GIM302xx message other than GIM30206 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, keeping **BYPASS(HOLDSYS(*aaaaaaaa*))**.

**GIM35903v USER HOLD *aaaaaaaa* WAS NOT RESOLVED.****Explanation:**

- *aaaaaaaa*—user hold reason ID.

SMP/E could not process the SYSMOD named in GIM30206 because it was held for the indicated user reason ID, which was not resolved.

**System Action:** See message GIM30206.

**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 GIM30206, but specify **BYPASS(HOLDUSER(*aaaaaaaa*))**.

**GIM35904v PREREQUISITE SYSMOD *aaaaaaaa* WAS MISSING.****Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaaa* is a prerequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaaa* 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 *aaaaaaaa* was missing.)

**Note:** For RESTORE processing, this message is only issued for a prerequisite if the SYSMOD named in GIM302xx failed. It is not issued for all the prerequisites in the chain.

**System Action:** See message GIM302xx.

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

- If SYSMOD *aaaaaaaa* is no longer in the global zone, receive it again so that it is available to be installed. Then rerun the job.
- If SYSMOD *aaaaaaaa* is available but was not specified on the SELECT operand, include it on the SELECT operand and rerun the job.

**GIM35905v PREREQUISITE SYSMOD *aaaaaaaa* WAS EXCLUDED.****Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaaa* is a prerequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaaa* was specified on the EXCLUDE operand. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *aaaaaaaa* was excluded.)

**Note:** For RESTORE processing, this message is only issued for a prerequisite if the SYSMOD named in GIM302xx failed. It is not issued for all the prerequisites in the chain.

**System Action:** See message GIM302xx.

**Programmer Response:** Remove SYSMOD *aaaaaaaa* from the EXCLUDE operand and rerun the job.

**GIM35906v PREREQUISITE SYSMOD *aaaaaaaa* WAS HELD.****Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaaa* is a prerequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because there is a ++HOLD MCS for SYSMOD *aaaaaaaa* that was not resolved. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *aaaaaaaa* was held.)

**Note:** For RESTORE processing, this message is only issued for a prerequisite if the SYSMOD named in GIM302xx failed. It is not issued for all the prerequisites in the chain.

**System Action:** See message GIM302xx.



**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.

**GIM35907v PREREQUISITE SYSMOD *aaaaaaaa* FAILED.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaaa* is a prerequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaaa* 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 *aaaaaaaa* failed.)

**Note:** For RESTORE processing, this message is only issued for a prerequisite if the SYSMOD named in GIM302xx failed. It is not issued for all the prerequisites in the chain.

**System Action:** See message GIM302xx.

**Programmer Response:** Look at the preceding messages in SMPOUT to determine why the SYSMOD failed. Fix the error and rerun the job.

**GIM35908v COREQUISITE SYSMOD *aaaaaaaa* WAS MISSING.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaaa* is a corequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaaa* 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 *aaaaaaaa* was missing.)

**System Action:** See message GIM302xx.

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

- If SYSMOD *aaaaaaaa* is no longer in the global zone, receive it again so that it is available to be installed. Then rerun the job.
- If SYSMOD *aaaaaaaa* is available but was not specified on the SELECT operand, include it on the SELECT operand and rerun the job.

**GIM35909v COREQUISITE SYSMOD *aaaaaaaa* WAS EXCLUDED.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaaa* is a corequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaaa* was specified on the EXCLUDE operand. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *aaaaaaaa* was excluded.)

**System Action:** See message GIM302xx.

**Programmer Response:** Remove SYSMOD *aaaaaaaa* from the EXCLUDE operand and rerun the job.

**GIM35910v COREQUISITE SYSMOD *aaaaaaa* WAS HELD.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is a corequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because there is a ++HOLD MCS for SYSMOD *aaaaaaa* that was not resolved. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *aaaaaaa* was held.)

**System Action:** See message GIM302xx.

**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.

**GIM35911v COREQUISITE SYSMOD *aaaaaaa* FAILED.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is a corequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaa* 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 *aaaaaaa* failed.)

**System Action:** See message GIM302xx.

**Programmer Response:** Look at the preceding messages in SMPOUT to determine why the SYSMOD failed. Fix the error and rerun the job.

**GIM35912v CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS MISSING.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaa* 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 *aaaaaaa* was missing.) The SYSMOD named in GIM302xx contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

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

- If SYSMOD *aaaaaaa* is no longer in the global zone, receive it again so that it is available to be installed. Then rerun the job.
- If SYSMOD *aaaaaaa* is available but was not specified on the SELECT operand, include it on the SELECT operand and rerun the job.

**GIM35913v CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS EXCLUDED.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaa* was specified on the EXCLUDE operand. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *aaaaaaa* was excluded.) The SYSMOD named in GIM302xx contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

**Programmer Response:** Remove SYSMOD *aaaaaaa* from the EXCLUDE operand and rerun the job.

**GIM35914v CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS HELD.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because there is a ++HOLD MCS for SYSMOD *aaaaaaa* that was not resolved. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *aaaaaaa* was held.) The SYSMOD named in GIM302xx contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

**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.

**GIM35915v CONDITIONAL REQUISITE SYSMOD *aaaaaaa* FAILED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaa* 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 *aaaaaaa* failed.) The SYSMOD named in GIM302xx contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

**Programmer Response:** Look at the preceding messages in SMPOUT to determine why the SYSMOD failed. Fix the error and rerun the job.

**GIM35916v CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* CONTAINED THE ++IF STATEMENT.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID of the requisite
- *bbbbbbb*—ID of the SYSMOD that contained the ++IF MCS.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaa* 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 *aaaaaaa* was missing.) SYSMOD *bbbbbbb* contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

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

- If SYSMOD *aaaaaaa* is no longer in the global zone, receive it again so that it is available to be installed. Then rerun the job.
- If SYSMOD *aaaaaaa* is available but was not specified on the SELECT operand, include it on the SELECT operand and rerun the job.

**GIM35917v** **CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS EXCLUDED. SYSMOD *bbbbbbb* CONTAINED THE ++IF STATEMENT.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID of the requisite
- *bbbbbbb*—ID of the SYSMOD that contained the ++IF MCS.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaa* was specified on the EXCLUDE operand. SYSMOD *bbbbbbb* 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 *aaaaaaa* was excluded.)

**System Action:** See message GIM302xx.

**Programmer Response:** Remove SYSMOD *aaaaaaa* from the EXCLUDE operand and rerun the job.

**GIM35918v** **CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS HELD. SYSMOD *bbbbbbb* CONTAINED THE ++IF STATEMENT.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID of the requisite
- *bbbbbbb*—ID of the SYSMOD that contained the ++IF MCS.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because there is a ++HOLD MCS for SYSMOD *aaaaaaa* that was not resolved. (Because superseding SYSMODs may resolve requisites, this message may be issued if a SYSMOD that superseded SYSMOD *aaaaaaa* was held.) SYSMOD *bbbbbbb* contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

**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.

**GIM35919v** **CONDITIONAL REQUISITE SYSMOD *aaaaaaa* FAILED. SYSMOD *bbbbbbb* CONTAINED THE ++IF STATEMENT.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID of the requisite
- *bbbbbbb*—ID of the SYSMOD that contained the ++IF MCS.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx because SYSMOD *aaaaaaa* 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 *aaaaaaa* failed.) SYSMOD *bbbbbbb* contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

**Programmer Response:** Look at the preceding messages in SMP/OUT to determine why the SYSMOD failed. Fix the error and rerun the job.

**GIM35920v** **SYSMOD *aaaaaaa*, WHICH SUPERSEDES THE INDICATED SYSMOD, FAILED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* supersedes the SYSMOD in GIM30209. Therefore, SYSMOD *aaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35920v SYSMOD *aaaaaaa* SUPERSEDES THE INDICATED SYSMOD.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* supersedes the SYSMOD in GIM30209. Therefore, SYSMOD *aaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35921v SYSMOD *aaaaaaa* IS A PREREQUISITE FOR THE INDICATED SYSMOD.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* is a prerequisite for the SYSMOD in GIM30209. Therefore, SYSMOD *aaaaaaa* must either be accepted, or it must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Accept SYSMOD *aaaaaaa*, if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209.
- Restore SYSMOD *aaaaaaa* along with the SYSMOD named in GIM30209. There are two ways to do this:
  - Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
  - Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

After this, you can reapply SYSMOD *aaaaaaa* if you want those changes in the target libraries.

**GIM35922v SYSMOD *aaaaaaa*, WHICH SPECIFIES THE INDICATED SYSMOD AS A PREREQUISITE, FAILED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a prerequisite. Therefore, SYSMOD *aaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35922v SYSMOD *aaaaaaa* SPECIFIES THE INDICATED SYSMOD AS A PREREQUISITE.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a prerequisite. Therefore, SYSMOD *aaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35923v SYSMOD *aaaaaaa* IS A COREQUISITE FOR THE INDICATED SYSMOD.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* is a corequisite for the SYSMOD in GIM30209. Therefore, SYSMOD *aaaaaaa* must either be accepted, or it must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Accept SYSMOD *aaaaaaa*, if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209.
- Restore SYSMOD *aaaaaaa* along with the SYSMOD named in GIM30209. There are two ways to do this:
  - Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
  - Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

After this, you can reapply SYSMOD *aaaaaaa* if you want those changes in the target libraries.

**GIM35924v SYSMOD *aaaaaaa*, WHICH SPECIFIES THE INDICATED SYSMOD AS A COREQUISITE, FAILED.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a corequisite. Therefore, SYSMOD *aaaaaaa* must be restored along with or before the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.
- Restore SYSMOD *aaaaaaa*, then restore the SYSMOD in GIM30209.

**GIM35924v SYSMOD *aaaaaaa* SPECIFIES THE INDICATED SYSMOD AS A COREQUISITE.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a corequisite. Therefore, SYSMOD *aaaaaaa* must be restored along with or before the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.
- Restore SYSMOD *aaaaaaa*, then restore the SYSMOD in GIM30209.

**GIM35925v SYSMOD *aaaaaaa* IS A CONDITIONAL REQUISITE FOR THE INDICATED SYSMOD.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD in GIM30209. Therefore, SYSMOD *aaaaaaa* must either be accepted, or it must be restored along with the SYSMOD in GIM30209. The SYSMOD named in GIM30209 contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM30209.

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

- Accept SYSMOD *aaaaaaa*, if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209.
- Restore SYSMOD *aaaaaaa* along with the SYSMOD named in GIM30209. There are two ways to do this:
  - Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
  - Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

After this, you can reapply SYSMOD *aaaaaaa* if you want those changes in the target libraries.

**GIM35926v SYSMOD *aaaaaaa*, WHICH SPECIFIES THE INDICATED SYSMOD AS A CONDITIONAL REQUISITE, FAILED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a conditional requisite. Therefore, SYSMOD *aaaaaaa* must be restored along with or before the SYSMOD in GIM30209. SYSMOD *aaaaaaa* contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.
- Restore SYSMOD *aaaaaaa*, then restore the SYSMOD in GIM30209.

**GIM35926v SYSMOD *aaaaaaa* SPECIFIES THE INDICATED SYSMOD AS A CONDITIONAL REQUISITE.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a conditional requisite. Therefore, SYSMOD *aaaaaaa* must be restored along with or before the SYSMOD in GIM30209. SYSMOD *aaaaaaa* contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.
- Restore SYSMOD *aaaaaaa*, then restore the SYSMOD in GIM30209.

**GIM35927v SYSMOD *aaaaaaa*, WHICH SPECIFIES THE INDICATED SYSMOD AS AN FMID, FAILED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as an FMID. Therefore, SYSMOD *aaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35927v SYSMOD *aaaaaaa* SPECIFIES THE INDICATED SYSMOD AS AN FMID.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as an FMID. Therefore, SYSMOD *aaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35928v SYSMOD *aaaaaaa*, WHICH IS AN FMID FOR THE INDICATED SYSMOD, FAILED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is the FMID that owns the SYSMOD in GIMxxx.

- If this message was issued for APPLY or ACCEPT processing, it indicates that SYSMOD *aaaaaaa* must be applied or accepted along with or before the SYSMOD in GIMxxx.
- If this message was issued for RESTORE processing, SMP/E tried to restore the SYSMOD named in GIM30209. Therefore, SYSMOD *aaaaaaa* must either be accepted, or it must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

**Programmer Response:**

If this message was issued for APPLY or ACCEPT processing, check the following:

- Did SMP/E successfully install SYSMOD *aaaaaaa* in the target libraries?
- Did SMP/E select SYSMOD *aaaaaaa* for installation during the APPLY or ACCEPT step for the SYSMOD named in message GIM30209?

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 *aaaaaaa* fails, the requisite SYSMOD might also fail.



If this message was issued for RESTORE processing, do **one** of the following:

- Accept SYSMOD *aaaaaaa* if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209.
- Restore SYSMOD *aaaaaaa* along with the SYSMOD named in GIM30209. There are two ways to do this:
  - Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
  - Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

After this, you can reapply SYSMOD *aaaaaaa* if you want those changes in the target libraries.

**GIM35928v SYSMOD *aaaaaaa* IS AN FMID FOR THE INDICATED SYSMOD.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is the FMID that owns the SYSMOD in GIMxxx.

- If this message was issued for APPLY or ACCEPT processing, it indicates that SYSMOD *aaaaaaa* must be applied or accepted along with or before the SYSMOD in GIMxxx.
- If this message was issued for RESTORE processing, SMP/E tried to restore the SYSMOD named in GIM30209. Therefore, SYSMOD *aaaaaaa* must either be accepted, or it must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

**Programmer Response:** If this message was issued for APPLY or ACCEPT processing, check the following:

- Did SMP/E successfully install SYSMOD *aaaaaaa* in the target libraries?
- Did SMP/E select SYSMOD *aaaaaaa* for installation during the APPLY or ACCEPT step for the SYSMOD named in message GIM30209?

If SMP/E is installing the requisite SYSMOD during the current APPLY or ACCEPT step, look at the preceding messages in SMP/OUT to find why the requisite SYSMOD failed.

**Note:** If a SYSMOD that supersedes SYSMOD *aaaaaaa* fails, the requisite SYSMOD might also fail.

If this message was issued for RESTORE processing, do **one** of the following:

- Accept SYSMOD *aaaaaaa*, if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209.
- Restore SYSMOD *aaaaaaa* along with the SYSMOD named in GIM30209. There are two ways to do this:
  - Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
  - Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

After this, you can reapply SYSMOD *aaaaaaa* if you want those changes in the target libraries.

**GIM35929v SYSMOD *aaaaaaa* IS SUPERSEDED BY THE INDICATED SYSMOD.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209 or GIM30213, but SYSMOD *aaaaaaa* is superseded by the SYSMOD in GIM30209 or GIM30213. Therefore, SYSMOD *aaaaaaa* must either be accepted, or it must be restored along with the SYSMOD in GIM30209 or GIM30213.

**System Action:** See message GIM30209 or GIM30213.

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

- Accept SYSMOD *aaaaaaa*, if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209 or GIM30213.
- Restore SYSMOD *aaaaaaa* along with the SYSMOD named in GIM30209 or GIM30213. There are two ways to do this:
  - Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209 or GIM30213.
  - Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209 or GIM30213.

After this, you can reapply SYSMOD *aaaaaaa* if you want those changes in the target libraries.

**GIM35930v PREREQUISITE SYSMOD *aaaaaaa* WAS BYPASSED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

The SYSMOD named in GIM42001 named SYSMOD *aaaaaaa* as a prerequisite. However, **BYPASS(PRE)** was specified, so SMP/E did not make sure the requisite was satisfied.

**System Action:** See message GIM42001.

**Programmer Response:** None.

**GIM35931v COREQUISITE SYSMOD *aaaaaaa* WAS BYPASSED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

The SYSMOD named in GIM42001 named SYSMOD *aaaaaaa* as a corequisite. However, **BYPASS(REQ)** was specified, so SMP/E did not make sure the requisite was satisfied.

**System Action:** See message GIM42001.

**Programmer Response:** None.

**GIM35932v CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS BYPASSED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

The SYSMOD named in GIM42001 named SYSMOD *aaaaaaa* as a conditional requisite. However, **BYPASS(IFREQ)** was specified, so SMP/E did not make sure the requisite was satisfied. The SYSMOD named in GIM42001 contains the **++IF** MCS that defines the requisite.

**System Action:** See message GIM42001.

**Programmer Response:** None.

**GIM35933v CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS BYPASSED. SYSMOD *bbbbbbb* CONTAINED THE **++IF** STATEMENT.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of the SYSMOD that contained the **++IF** MCS.

The SYSMOD named in GIM42001 named SYSMOD *aaaaaaa* as a conditional requisite. However, **BYPASS(IFREQ)** was specified, so SMP/E did not make sure the requisite was satisfied. SYSMOD *bbbbbbb* contains the **++IF** MCS that defines the requisite.

**System Action:** See message GIM42001.

**Programmer Response:** None.

**GIM35934v ERROR HOLD aaaaaaa WAS BYPASSED.**

**Explanation:**

- aaaaaaa—error reason ID.

The SYSMOD named in GIM42001 was held for the indicated error reason ID. However, **BYPASS(HOLDERR(aaaaaaa))** 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 GIM42001.

**Programmer Response:** None.

**GIM35935v SYSTEM HOLD aaaaaaa WAS BYPASSED.**

**Explanation:**

- aaaaaaa—system reason ID.

The SYSMOD named in GIM42001 was held for the indicated system reason ID. However, **BYPASS(HOLDSYS(aaaaaaa))** 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 GIM42001.

**Programmer Response:** None.

**GIM35936v USER HOLD aaaaaaa WAS BYPASSED.**

**Explanation:**

- aaaaaaa—user reason ID.

The SYSMOD named in GIM42001 was held for the indicated user reason ID. However, **BYPASS(HOLDUSER(aaaaaaa))** 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 GIM42001.

**Programmer Response:** None.

**GIM35937v ERROR HOLD aaaaaaa WAS BYPASSED BECAUSE CLASS bbbbbbbb WAS BYPASSED.**

**Explanation:**

- aaaaaaa—error reason ID
- bbbbbbbb—hold class.

The SYSMOD named in GIM42001 was held for the indicated error reason ID and the associated class. However, **BYPASS(HOLDCLASS(aaaaaaa))** 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 GIM42001.

**Programmer Response:** None.

**GIM35938v SYSTEM HOLD aaaaaaa WAS BYPASSED BECAUSE CLASS bbbbbbbb WAS BYPASSED.**

**Explanation:**

- aaaaaaa—system reason ID
- bbbbbbbb—hold class.

The SYSMOD named in GIM42001 was held for the indicated system reason ID and the associated class. However, **BYPASS(HOLDCLASS(aaaaaaa))** 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 GIM42001.

**Programmer Response:** None.

**GIM35939v USER HOLD *aaaaaaaa* WAS BYPASSED BECAUSE CLASS *bbbbbbbb* WAS BYPASSED.**

**Explanation:**

- *aaaaaaaa*—user reason ID
- *bbbbbbbb*—hold class.

The SYSMOD named in GIM42001 was held for the indicated user reason ID and the associated class. However, **BYPASS(HOLDCLASS(*aaaaaaaa*))** 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 GIM42001.

**Programmer Response:** None.

**GIM35940v SYSMOD *aaaaaaaa*, WHICH WOULD HAVE RESOLVED HOLD *bbbbbbbb* FOR SYSMOD *ccccccc*, FAILED.**

**Explanation:**

- *aaaaaaaa*—superseding SYSMOD ID
- *bbbbbbbb*—HOLD ID
- *ccccccc*—failing SYSMOD ID.

During GROUPEXTEND processing, the SYSMOD that supersedes a hold has failed. This message follows GIM30206I.

**System Action:** SYSMOD processing stops.

**Programmer Response:** Fix the error and rerun the job.

**GIM35941v SYSMOD *aaaaaaaa*, WHICH WOULD HAVE DELETED SYSMOD *bbbbbbbb*, FAILED.**

**Explanation:**

- *aaaaaaaa*—deleting SYSMOD ID
- *bbbbbbbb*—SYSMOD ID.

This message follows GIM30201E and names the SYSMODs that delete the SYSMOD in GIM30201E. SMP/E could not process SYSMOD *bbbbbbbb* 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.

**GIM35942v SYSMOD *aaaaaaaa*, WHICH WOULD HAVE SUPERSEDED SYSMOD *bbbbbbbb*, FAILED.**

**Explanation:**

- *aaaaaaaa*—superseding SYSMOD ID
- *bbbbbbbb*—SYSMOD ID.

This message follows GIM30202E and names the superseding SYSMODs. SMP/E could not process SYSMOD *bbbbbbbb* 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.

**GIM35943v SYSMOD *aaaaaaaa*, WHICH SUPERSEDES THE INDICATED SYSMOD, WAS MISSING.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaaa* supersedes the SYSMOD in GIM30209. Therefore, SYSMOD *aaaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35944v SYSMOD *aaaaaaa*, WHICH SPECIFIES THE INDICATED SYSMOD AS A PREREQUISITE, WAS MISSING.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a prerequisite. Therefore, SYSMOD *aaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35945v SYSMOD *aaaaaaa*, WHICH SPECIFIES THE INDICATED SYSMOD AS A COREQUISITE, WAS MISSING.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a corequisite. Therefore, SYSMOD *aaaaaaa* must be restored along with or before the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.
- Restore SYSMOD *aaaaaaa*, then restore the SYSMOD in GIM30209.

**GIM35946v SYSMOD *aaaaaaa*, WHICH SPECIFIES THE INDICATED SYSMOD AS A CONDITIONAL REQUISITE, WAS MISSING.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as a conditional requisite. Therefore, SYSMOD *aaaaaaa* must be restored along with or before the SYSMOD in GIM30209. SYSMOD *aaaaaaa* contains the ++ IF MCS that defines the requisite.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.
- Restore SYSMOD *aaaaaaa*, then restore the SYSMOD in GIM30209.

**GIM35947v SYSMOD *aaaaaaa*, WHICH SPECIFIES THE INDICATED SYSMOD AS AN FMID, WAS MISSING.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E tried to restore the SYSMOD named in GIM30209, but SYSMOD *aaaaaaa* specifies the SYSMOD in GIM30209 as an FMID. Therefore, SYSMOD *aaaaaaa* must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
- Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

**GIM35948v SYSMOD *aaaaaaa*, WHICH IS AN FMID FOR THE INDICATED SYSMOD, WAS MISSING.****Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* is the FMID that owns the SYSMOD in GIMxxx.

- If this message was issued for APPLY or ACCEPT processing, it indicates that SYSMOD *aaaaaaa* must be applied or accepted along with or before the SYSMOD GIMxxx.
- If this message was issued for RESTORE processing, SMP/E tried to restore the SYSMOD named in GIM30209. Therefore, SYSMOD *aaaaaaa* must either be accepted, or it must be restored along with the SYSMOD in GIM30209.

**System Action:** See message GIM30209.

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

- For APPLY or ACCEPT processing, check the following:
  - Did SMP/E successfully install SYSMOD *aaaaaaa* in the target libraries?
  - Did SMP/E select SYSMOD *aaaaaaa* for installation during the APPLY or ACCEPT step for the SYSMOD named in message GIM30209?

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 *aaaaaaa* fails, the requisite SYSMOD might also fail.

- For RESTORE processing, do one of the following:
  - Accept SYSMOD *aaaaaaa*, if you want those changes in the distribution libraries. Then restore the SYSMOD named in GIM30209.
  - Restore SYSMOD *aaaaaaa* along with the SYSMOD named in GIM30209. There are two ways to do this:
    1. Specify SYSMOD *aaaaaaa* on the SELECT operand of the RESTORE command along with the SYSMOD in GIM30209.
    2. Specify the GROUP operand on the RESTORE command for the SYSMOD in GIM30209.

After this, you can reapply SYSMOD *aaaaaaa* if you want those changes in the target libraries.

**GIM35949v ERROR HOLD *aaaaaaa* WAS NOT RESOLVED. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT HAS FAILED.**

**Explanation:**

- *aaaaaaa*—error hold reason ID
- *bbbbbbb*—ID of a SYSMOD that supersedes the reason ID.

SMP/E could not process the SYSMOD named in GIM30206, because that SYSMOD was held for the indicated error reason ID (APAR), which was not resolved. SYSMOD *bbbbbbb* would have resolved the error reason ID, but has failed processing.

**System Action:** See message GIM30206.

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

- Resolve the errors that caused SYSMOD *bbbbbbb* to fail, and rerun the APPLY or ACCEPT command.
- Rerun the APPLY or ACCEPT command to install the SYSMOD named in GIM30206, but specify **BYPASS(HOLDERR(*aaaaaaa*))**.

**GIM35950v ERROR HOLD *aaaaaaa* WAS NOT RESOLVED. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT WAS EXCLUDED.**

**Explanation:**

- *aaaaaaa*—error hold reason ID
- *bbbbbbb*—ID of a SYSMOD that supersedes the reason ID.

SMP/E could not process the SYSMOD named in GIM30206, because that SYSMOD was held for the indicated error reason ID (APAR), which was not resolved. SYSMOD *bbbbbbb* would have resolved the error reason ID, but was excluded from processing.

**System Action:** See message GIM30206.

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

- Remove SYSMOD *bbbbbbb* from the EXCLUDE operand, and rerun the command.
- Rerun the APPLY or ACCEPT command to install the SYSMOD named in GIM30206, but specify **BYPASS(HOLDERR(*aaaaaaa*))**.

**GIM35951v PREREQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT HAS FAILED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of a SYSMOD that supersedes *aaaaaaa*.

SYSMOD *aaaaaaa* is a prerequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx, because SYSMOD *aaaaaaa* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *bbbbbbb* would have superseded SYSMOD *aaaaaaa*, but has failed processing.

**System Action:** See message GIM302xx.

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

- Resolve the errors that caused SYSMOD *bbbbbbb* to fail, and rerun the APPLY or ACCEPT command.
- Receive SYSMOD *aaaaaaa*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

**GIM35952v PREREQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT WAS EXCLUDED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of a SYSMOD that supersedes *aaaaaaa*.

SYSMOD *aaaaaaa* is a prerequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx, because SYSMOD *aaaaaaa* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *bbbbbbb* would have superseded SYSMOD *aaaaaaa*, but was excluded from processing.

**System Action:** See message GIM302xx.

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

- Remove SYSMOD *bbbbbbb* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *aaaaaaa*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

**GIM35953v COREQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT HAS FAILED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of a SYSMOD that supersedes *aaaaaaa*.

SYSMOD *aaaaaaa* is a corequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx, because SYSMOD *aaaaaaa* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *bbbbbbb* would have superseded SYSMOD *aaaaaaa*, but has failed processing.

**System Action:** See message GIM302xx.

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

- Resolve the errors that caused SYSMOD *bbbbbbb* to fail, and rerun the APPLY or ACCEPT command.
- Receive SYSMOD *aaaaaaa*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

**GIM35954v COREQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT WAS EXCLUDED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of a SYSMOD that supersedes *aaaaaaa*.

SYSMOD *aaaaaaa* is a corequisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx, because SYSMOD *aaaaaaa* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *bbbbbbb* would have superseded SYSMOD *aaaaaaa*, but was excluded from processing.

**System Action:** See message GIM302xx.

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

- Remove SYSMOD *bbbbbbb* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *aaaaaaa*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.



**GIM35955v** CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT HAS FAILED.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of a SYSMOD that supersedes *aaaaaaa*.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx, because SYSMOD *aaaaaaa* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *bbbbbbb* would have superseded SYSMOD *aaaaaaa*, but has failed processing.

**System Action:** See message GIM302xx.

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

- Resolve the errors that caused SYSMOD *bbbbbbb* to fail, and rerun the APPLY or ACCEPT command.
- Receive SYSMOD *aaaaaaa*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

**GIM35956v** CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT WAS EXCLUDED.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of a SYSMOD that supersedes *aaaaaaa*.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx, because SYSMOD *aaaaaaa* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *bbbbbbb* would have superseded SYSMOD *aaaaaaa*, but was excluded from processing.

**System Action:** See message GIM302xx.

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

- Remove SYSMOD *bbbbbbb* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *aaaaaaa*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

**GIM35957v** CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT HAS FAILED. SYSMOD *ccccccc* CONTAINED THE ++IF STATEMENT.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of a SYSMOD that supersedes *aaaaaaa*
- *ccccccc*—ID of the SYSMOD that contained the ++IF MCS.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx, because SYSMOD *aaaaaaa* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *bbbbbbb* would have superseded SYSMOD *aaaaaaa*, but has failed processing. SYSMOD *ccccccc* contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

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

- Resolve the errors that caused SYSMOD *bbbbbbb* to fail, and rerun the APPLY or ACCEPT command.
- Receive SYSMOD *aaaaaaa*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

**GIM35958v** **CONDITIONAL REQUISITE SYSMOD *aaaaaaa* WAS MISSING. SYSMOD *bbbbbbb* WOULD HAVE SUPERSEDED *aaaaaaa*, BUT WAS EXCLUDED. SYSMOD *ccccccc* CONTAINED THE ++IF STATEMENT.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—ID of a SYSMOD that supersedes *aaaaaaa*
- *ccccccc*—ID of the SYSMOD that contained the ++IF MCS.

SYSMOD *aaaaaaa* is a conditional requisite for the SYSMOD named in GIM302xx. However, SMP/E could not process the SYSMOD named in GIM302xx, because SYSMOD *aaaaaaa* either is not available (it is not in the GLOBAL zone), or is available but was not selected for processing. SYSMOD *bbbbbbb* would have superseded SYSMOD *aaaaaaa*, but was excluded from processing. SYSMOD *ccccccc* contains the ++IF MCS that defines the requisite.

**System Action:** See message GIM302xx.

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

- Remove SYSMOD *bbbbbbb* from the EXCLUDE operand, and rerun the job.
- Receive SYSMOD *aaaaaaa*, making it available to select for processing; or, if it is already available, select it for processing. Then rerun the job.

**GIM36201v** **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.

**GIM36202v** **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.

**GIM36203v** **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.

**GIM36204v** **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.

**GIM36301v** ***aaaaaaa* IS NOT ALLOWED AS *bbbbbbb* AND *ccccccc*.**

**Explanation:**

- *aaaaaaa*—operand value
- *bbbbbbb*—first operand
- *ccccccc*—second operand.

The SYSMOD is not built correctly because the same value was specified for the indicated operands.

**System Action:** SYSMOD processing stops.

**Programmer Response:** Fix the problem and rerun the job.

**GIM36302v 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.

**GIM36303v 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.

**GIM36304v THE ++VER aaaaaaa OPERAND IS NOT ALLOWED IN A SERVICE SYSMOD.**

**Explanation:**

- aaaaaaa—DELETE or NPRES.

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.

**GIM36305v 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.

**GIM36306v 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.

**GIM36307v 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.

**GIM36401v** *aaaaaaa* PROCESSING FAILED FOR *bbbbbbb* AND *ccccccc* BECAUSE SYSMOD *bbbbbbb* DELETES SYSMOD *ccccccc*. ONLY ONE OF THESE SYSMODS CAN BE PROCESSED AT A TIME.

**Explanation:**

- *aaaaaaa*—an SMP/E command (APPLY or ACCEPT)
- *bbbbbbb*—ID of the deleting SYSMOD
- *ccccccc*—ID of the deleted SYSMOD.

The two indicated function SYSMODs are eligible for processing. However, SYSMOD *bbbbbbb* is supposed to delete SYSMOD *ccccccc*. 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.

**GIM36901v** SYSMOD *aaaaaaa* WAS SPECIFIED ON THE SELECT OPERAND BUT IS NOT IN THE SMPPTFIN DATA SET.

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SYSMOD *aaaaaaa* 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 *aaaaaaa* and define a SMPPTFIN DD statement that points to the tape or data set that contains the SYSMOD. Then rerun the job.

**GIM37001v** *aaaaaaa* PROCESSING STOPPED BECAUSE FUNCTION SYSMOD *bbbbbbb* FAILED.

**Explanation:**

- *aaaaaaa*—an SMP/E command
- *bbbbbbb*—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.

**GIM37201v** *aaaaaaa* PROCESSING FAILED FOR *bbbbbbb* AND *ccccccc* BECAUSE *bbbbbbb* AND *ccccccc* ARE MUTUALLY EXCLUSIVE SYSMODS. ONLY ONE OF THESE SYSMODS CAN BE PROCESSED AT A TIME.

**Explanation:**

- *aaaaaaa*—an SMP/E command (APPLY or ACCEPT)
- *bbbbbbb*—ID of one SYSMOD
- *ccccccc*—ID of the other SYSMOD.

The two indicated function SYSMODs are eligible for processing. However, they specify each other as ++VER NPRES 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.

**GIM37301v** SYSMOD *aaaaaaa* HAS MORE THAN ONE APPLICABLE ++VER MCS. ONLY ONE IS ALLOWED.

**Explanation:**

- *aaaaaaa*—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 *aaaaaaa*.

**System Action:** SYSMOD processing stops.

**Programmer Response:** Change the SYSMOD so that it only contains one applicable ++VER MCS. Then rerun the job.

**GIM37401v** PROCESSING FAILED FOR SYSMOD *aaaaaaa*. *aaaaaaa* REQUIRES SYSMOD *bbbbbbb*. HOWEVER, SYSMOD *bbbbbbb* SPECIFIES *aaaaaaa* ON ITS ++VER *cccc* OPERAND.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—second SYSMOD ID
- *cccc*—operand on ++VER (FMID or PRE).

Processing for SYSMOD *bbbbbbb* stopped because SMP/E could not determine which SYSMOD to process first: SYSMOD *bbbbbbb* or SYSMOD *aaaaaaa*. 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.

**GIM37401v** PROCESSING FAILED FOR SYSMOD *aaaaaaa*. *aaaaaaa* REQUIRES ANOTHER SYSMOD. HOWEVER, THAT OTHER SYSMOD SPECIFIES *aaaaaaa* ON ITS ++VER *bbbb* OPERAND.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbb*—++VER operand (FMID or PRE).

This message follows message GIM22601. Processing for the SYSMOD in GIM22601 stopped because SMP/E could not determine which SYSMOD to process first: the one in GIM22601 or the one in GIM37401. 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.

**GIM37701v SMPTLIB *aaaaaaaa* WAS NOT DELETED BECAUSE IT IS DEFINED AS A DUMMY DATA SET.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM37702v SMPTLIB *aaaaaaaa* WAS NOT DELETED FROM VOLUME *bbbbbb* BECAUSE THE DATA SET WAS NOT FOUND.**

**Explanation:**

- *aaaaaaaa*—data set name
- *bbbbbb*—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.

**GIM37703v SMPTLIB *aaaaaaaa* WAS NOT DELETED FROM VOLUME *bbbbbb* BECAUSE ERROR CODE *cccc* WAS ISSUED BY DYNAMIC ALLOCATION.**

**Explanation:**

- *aaaaaaaa*—data set name
- *bbbbbb*—volume serial number
- *cccc*—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, 47B0	DADSM error
47A8–47AC	RACF error
5704–571C	CATALOG error

Codes	Meaning
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 the system macros and facilities manual for your operating system for more information about the error code.

**GIM37704v SMPTLIB aaaaaaaaa WAS NOT DELETED BECAUSE NO VOLUMES WERE SPECIFIED.**

**Explanation:**

- aaaaaaaaa—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.

**GIM37801v ++aaaaaaaa AND ++bbbbbbb FOR ELEMENT ccccccc ARE NOT ALLOWED IN THE SAME SYSMOD.**

**Explanation:**

- aaaaaaa—first element MCS
- bbbbbbbb—second element MCS
- ccccccc—element name.

The indicated combination of MCS statements is not allowed for the same element in the same SYSMOD. The following table shows the combinations that are allowed.

MCS	MOD	ZAP	SRC	SRCUPD	MAC	MACUPD	Data element
MOD			Yes	Yes	Yes		Yes
ZAP					Yes	Yes	
SRC	Yes				Yes	Yes	Yes
SRCUPD	Yes				Yes	Yes	
MAC	Yes	Yes	Yes	Yes			Yes
MACUPD	Yes	Yes	Yes	Yes			
Data element	Yes		Yes		Yes		Yes

**System Action:** The SYSMOD is not received.

**Programmer Response:** Fix the problem and rerun RECEIVE.

**GIM37901v aaaaaaaaa PROCESSING FAILED FOR SYSMOD bbbbbbb BECAUSE IT WAS PREVIOUSLY SUPERSEDED.**

**Explanation:**

- aaaaaaaaa—APPLY or ACCEPT
- bbbbbbb—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 bbbbbbb. 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.

**GIM37902v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb* BECAUSE IT WAS PREVIOUSLY DELETED.

**Explanation:**

- *aaaaaaaa*—APPLY or ACCEPT
- *bbbbbbb*—SYSMOD ID.

SMP/E cannot apply or accept the indicated SYSMOD because it has already applied or accepted a SYSMOD that deletes SYSMOD *bbbbbbb*. 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 *bbbbbbb*.

- If you want the deleting SYSMOD on your system, do not try to reinstall SYSMOD *bbbbbbb*.
- If you want SYSMOD *bbbbbbb* back on your system, you must remove the deleting SYSMOD from the system, delete the SYSMOD entry for SYSMOD *bbbbbbb*, then reinstall SYSMOD *bbbbbbb*.

**GIM37903v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb* BECAUSE IT HAS NO APPLICABLE ++VER MCS.

**Explanation:**

- *aaaaaaaa*—APPLY or ACCEPT
- *bbbbbbb*—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.

**GIM37904v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb* BECAUSE OF AN ERROR DURING A PREVIOUS ATTEMPT TO RESTORE *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—APPLY or ACCEPT
- *bbbbbbb*—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.



**GIM38001v** *aaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb* BECAUSE THE SYSMOD ENTRY FOR *bbbbbbb* IN THE *ccccccc* ZONE WAS IN ERROR.

**Explanation:**

- *aaaaaaa*—APPLY or ACCEPT
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—target zone name.

SMP/E cannot apply or accept the indicated SYSMOD because the associated target zone SYSMOD entry is in error.

**System Action:** SYSMOD processing stops.

**Programmer Response:** Apply the SYSMOD again to replace the SYSMOD entry that is in error. Then rerun the APPLY or ACCEPT job.

**GIM38101v** *aaaaaaa* PROCESSING STOPPED. A ++*bbbbbbb* MCS IN SYSMOD *ccccccc* APPLIES TO ELEMENT *ddddddd*, WHICH WAS DELETED BY ANOTHER SYSMOD THAT IS BEING PROCESSED.

**Explanation:**

- *aaaaaaa*—APPLY or ACCEPT
- *bbbbbbb*—element update MCS
- *ccccccc*—SYSMOD ID
- *ddddddd*—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.

**GIM38201v** THERE IS A MODID ERROR FOR *aaaaaaa* ENTRY *bbbbbbb* IN SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaa*—element type
- *bbbbbbb*—element name
- *ccccccc*—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 (GIM319xx for APPLY or ACCEPT, or GIM430xx for RESTORE).

The severity of this message depends on the BYPASS options specified on the APPLY or ACCEPT command.

**System Action:**

SYSMOD processing stops if all of the following conditions are 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.

SYSMOD processing does not stop if any of the following conditions are 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 GIM319xx 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 GIM382xx. 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 GIM382xx, followed by GIM319xx. 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 GIM382xx. 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 GIM319xx 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 GIM319xx messages. Rework and reinstall these modifications.

**GIM38202v** THERE IS A MODID ERROR FOR ASSEMBLY *aaaaaaaa* FOR *bbbbbbbb* ENTRY *ccccccc* IN SYSMOD *ddddddd*.

**Explanation:**

- *aaaaaaaa*—assembly name
- *bbbbbbbb*—element type
- *ccccccc*—element name
- *ddddddd*—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 (GIM319xx 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 if all of the following conditions are 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.

SYSMOD processing does not stop if either of the following conditions is 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 GIM319xx 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 GIM382xx. 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 GIM382xx, followed by GIM319xx. 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 GIM382xx. SMP/E selects the indicated element, and SYSMOD processing continues.

**Programmer Response:** Carefully check the messages that follow this message in SMP/E. 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 GIM319xx 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 GIM319xx messages. Rework and reinstall these modifications.

**GIM38401v** *aaaaaaaa* PROCESSING FAILED FOR *bbbbbbb* AND *ccccccc*. SYSMOD *bbbbbbb* DELETES OTHER SYSMODS, BUT IT IS SUPERSEDED BY SYSMOD *ccccccc*. ONLY ONE OF THESE SYSMODS CAN BE PROCESSED AT A TIME.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbbb*—ID of the deleting SYSMOD
- *ccccccc*—ID of the superseding SYSMOD.

SMP/E is concurrently processing two function SYSMODs. SYSMOD *aaaaaaaa* deletes other SYSMODs, but it is superseded by SYSMOD *bbbbbbb*. SMP/E cannot determine which function to process.

**System Action:** Message GIM370xx 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.

**GIM38501v SYSMOD *aaaaaaa* WAS NOT RECEIVED BECAUSE THE RETURN CODE FROM THE EXIT ROUTINE IS 8 OR HIGHER.**

**Explanation:**

- *aaaaaaa*—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.

**GIM38601v SYSMOD *aaaaaaa* WAS NOT RECEIVED BECAUSE IT HAS ALREADY BEEN RECEIVED.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E has already successfully received SYSMOD *aaaaaaa* 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.

**GIM38701v SYSMOD *aaaaaaa* WAS NOT RECEIVED BECAUSE IT WAS NOT SPECIFIED ON THE SELECT OPERAND.**

**Explanation:**

- *aaaaaaa*—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.

**GIM38801v SYSMOD *aaaaaaa* WAS NOT RECEIVED BECAUSE IT WAS SPECIFIED ON THE EXCLUDE OPERAND.**

**Explanation:**

- *aaaaaaa*—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.

**GIM38901v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb* BECAUSE IT HAS NO APPLICABLE **++VER** MCS.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbbb*—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.

**GIM39001v** SYSMOD *aaaaaaaa* WAS NOT RECEIVED EVEN THOUGH IT WAS SPECIFIED ON THE SELECT OPERAND.

**Explanation:**

- *aaaaaaaa*—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.

**GIM39101v** SYSMOD *aaaaaaaa* WAS NOT RECEIVED BECAUSE AN ERROR OCCURRED DURING RELFILE PROCESSING.

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

An error occurred when the relative files for the indicated SYSMOD were being loaded into the SMPTLIB data sets. 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.

The data set name for a relative file must be in the form *sysmod-id.Fnnnn*, where:

*sysmod-id*

is the SYSMOD ID of the SYSMOD to which that file belongs.

*Fnnnn*

is the letter *F* followed by the number specified on the RELFILE operand of the corresponding MCS statement in the SYSMOD.

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.Fnnnn*, where:

*dsprefix*

is an optional data set prefix.

- SMP/E could not allocate an SMPTLIB data set.
- The return code from the copy utility was not 0. (See the copy SYSPRINT output to find the cause.)

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 the RELFILE tape using the correct data set names.

Then receive the SYSMOD again.

**GIM39401v RELFILES WERE LOADED FOR SYSMOD *aaaaaaa*.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

The copy utility successfully loaded relative files for the indicated SYSMOD into SMPTLIB data sets.

**System Action:** SYSMOD processing continues. SMP/E indicates that the indicated SYSMOD has been received successfully.

**Programmer Response:** None.

**GIM39402v RELFILES WERE LOADED FOR SYSMOD *aaaaaaa*. THE RETURN CODE FROM THE COPY UTILITY WAS *bb*.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bb*—maximum return code from the copy utility.

The copy utility successfully loaded relative files for the indicated SYSMOD into SMPTLIB data sets. 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.

**GIM39501v THE ELEMENTS IN SYSMOD *aaaaaaa* ARE IN RELFILES.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

Elements for the indicated SYSMOD are packaged in relative files.

**System Action:** If message GIM22601 was not issued, the elements are loaded from the relative files into SMPTLIB data sets.

**Programmer Response:** Look for the following pair of messages:

- GIM39401 RELFILES WERE LOADED FOR SYSMOD *aaaaaaa*
- GIM22701 RECEIVE PROCESSING WAS SUCCESSFUL FOR SYSMOD *aaaaaaa*

These messages will appear after all the MCS statements for all the SYSMODs listed in SMPDOUT.

**GIM39601v SYSMOD *aaaaaaa* WAS NOT RECEIVED BECAUSE IT HAS NO ++VER MCS.**

**Explanation:**

- *aaaaaaa*—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.

**GIM39701v SYSMOD aaaaaaa HAS NO ELEMENTS.**

**Explanation:**

- aaaaaaa—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.

**GIM39801v SYSMOD aaaaaaa WAS NOT RECEIVED BECAUSE IT HAS A SYNTAX OR CONSTRUCTION ERROR.**

**Explanation:**

- aaaaaaa—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.

**GIM399v 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.

**GIM40001v SYSMOD aaaaaaa HAS ALREADY BEEN FOUND IN THE SMPPTFIN DATA SET THAT IS CURRENTLY BEING PROCESSED.**

**Explanation:**

- aaaaaaa—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 program update tape (PUT) for each product. Consequently, multiple copies of the same PTF may be found on that PUT. During RECEIVE processing, SMP/E finds the duplicated copies of the PTF and issues this message. However, no real error occurred.

**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 a PUT 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.

---

**GIM40101v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbb* EVEN THOUGH IT WAS SPECIFIED ON THE SELECT OPERAND. *bbbbbbb* IS NOT IN THE *cccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbbb*—SYSMOD ID
- *cccccc*—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.

**GIM40201v** THE *aaaaaaaa* OPERAND IS NOT ALLOWED FOR SUPERSEDED ONLY SYSMODS.

**Explanation:**

- *aaaaaaaa*—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.

**GIM40202v** THE *aaaaaaaa* OPERAND IS NOT ALLOWED FOR DELETED SYSMODS.

**Explanation:**

- *aaaaaaaa*—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.

**GIM40203v** 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.

**GIM40204v** 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.

**GIM40301v** THE *aaaaaaa* LIBRARY IS NOT AVAILABLE FOR EXCLUSIVE USE BECAUSE IT IS ALREADY BEING USED.

**Explanation:**

- *aaaaaaa*—ddname of the library.

SMP/E tried to obtain exclusive use of the indicated library, but it was, already being used and was, therefore, not available.

**System Action:** Processing stops for the SYSMOD that needs the data set.

**Programmer Response:** When the data set is available for exclusive use, rerun the job for the affected SYSMOD.

**GIM40401v** SMP/E DID NOT RESTORE *aaaaaaa bbbbbbbb* FOR SYSMOD *ccccccc* BECAUSE *aaaaaaa bbbbbbbb* IS NOT IN THE *ddddddd* LIBRARY.

**Explanation:**

- *aaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—SYSMOD ID
- *ddddddd*—ddname of the library (either SMPSCDS or the target zone name).

During RESTORE processing for SYSMOD *ccccccc*, 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.

**GIM40501v** THE DISTLIB SPECIFIED FOR *aaaaaaa bbbbbbbb* IN SYSMOD *ccccccc* DOES NOT MATCH THE DISTLIB VALUE IN THE *aaaaaaa* ENTRY FOR *bbbbbbbb*.

**Explanation:**

- *aaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—SYSMOD ID.

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. This is an error because the indicated SYSMOD is not a function.

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.

**GIM40502v** THE DISTLIB FOR *aaaaaaaa bbbbbbbb* WAS CHANGED BY SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccc*—SYSMOD ID.

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. The indicated SYSMOD is a function, so the DISTLIB value in the element entry was changed.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM40901v** COPY PROCESSING TO THE *aaaaaaa* LIBRARY FAILED FOR *bbbbbbbb ccccccc* IN SYSMOD *ddddddd*. THE RETURN CODE (*ee*) EXCEEDED THE ALLOWABLE VALUE. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.

**Explanation:**

- *aaaaaaa*—ddname of the library to which the element was to be copied
- *bbbbbbbb*—entry type
- *ccccccc*—entry name
- *ddddddd*—SYSMOD ID
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

Copy processing failed for the indicated element.

Multiple elements are copied from one input data set to one output data set during a single invocation of the copy utility. If this invocation fails, SMP/E assumes that processing failed for all the elements being copied and sets the error indicator for the 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 error. The sequence number may be used as an index into the SYSPRINT output. This number is incremented for each utility call in an SMP/E run. Fix the error and rerun the job.

**GIM40902v COPY PROCESSING TO THE *aaaaaaa* LIBRARY WAS SUCCESSFUL FOR *bbbbbbb* *ccccccc* IN SYSMOD *ddddddd*. THE RETURN CODE WAS *ee*. DATE *yy.ddd* - TIME *hh:mm:ss* - SEQUENCE NUMBER *nnnnnn*.**

**Explanation:**

- *aaaaaaa*—ddname of the library to which the element was to be copied
- *bbbbbbb*—entry type
- *ccccccc*—entry name
- *ddddddd*—SYSMOD ID
- *ee*—return code
- *yy*—year
- *ddd*—Julian day
- *hh*—military hour
- *mm*—minutes
- *ss*—seconds
- *nnnnnn*—utility sequence number.

The indicated element was successfully copied.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM41001v *aaaaaaa* PROCESSING FAILED FOR *bbbbbbb* BECAUSE BOTH *bbbbbbb* AND *ccccccc* CONTAIN ZAP UPDATES FOR MODULE *ddddddd*. ONLY ONE OF THESE SYSMODS CAN BE PROCESSED AT A TIME.**

**Explanation:**

- *aaaaaaa*—APPLY or ACCEPT
- *bbbbbbb*—first SYSMOD ID
- *ccccccc*—second SYSMOD ID
- *ddddddd*—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.

**GIM41101v SMP/E DID NOT RESTORE *aaaaaaa* ENTRY *bbbbbbb* FOR SYSMOD *ccccccc* BECAUSE *aaaaaaa* *bbbbbbb* WAS CHANGED BY JCLIN.**

**SMP/E DID NOT RESTORE *aaaaaaa* ENTRY *bbbbbbb* FOR SYSMOD *ccccccc* BECAUSE *aaaaaaa* *bbbbbbb* WAS CHANGED BY *ddddddd*.**

**SMP/E DID NOT RESTORE *aaaaaaa* ENTRY *bbbbbbb* FOR SYSMOD *ccccccc* BECAUSE *aaaaaaa* *bbbbbbb* WAS CHANGED BY UCLIN.**

**Explanation:**

- *aaaaaaa*—entry type
- *bbbbbbb*—entry name
- *ccccccc*—ID of the SYSMOD being restored
- *ddddddd*—ID of the SYSMOD that updated the element.

When SMP/E was restoring SYSMOD *ccccccc*, 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 *ccccccc*. Because these changes are not in the BACKUP entries, they would be lost if SYSMOD *ccccccc* 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.

**GIM41201v 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 GIM41301. 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.

**GIM41301v SYSMOD=aaaaaaa FMID=bbbbbbb PRE=ccccccc.**

**Explanation:**

- aaaaaaa—SYSMOD ID
- bbbbbbbb—FMID associated with the SYSMOD
- ccccccc—list of prerequisites for the SYSMOD.

This message follows message GIM41201 and lists the SYSMODs for which SMP/E could not determine a processing order.

**System Action:** Command processing stops.

**Programmer Response:** See message GIM41201.

**GIM41501v THE ++aaaaaaa MCS FOR ELEMENT bbbbbbbb IN SYSMOD ccccccc WAS NOT PROCESSED BECAUSE bbbbbbbb IS NOT IN THE ddddd ZONE.**

**Explanation:**

- aaaaaaa—element update MCS
- bbbbbbbb—element name
- ccccccc—SYSMOD ID
- ddddd—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.

**GIM41601v DALIAS aaaaaaa FOR MOD bbbbbbbb HAS BEEN FOUND AS A MOD ENTRY WITH THE SAME DISTLIB.**

**Explanation:**

- aaaaaaa—DALIAS value on the ++MOD MCS
- bbbbbbbb—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 GIM22601 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.

**GIM41801v INLINE JCLIN PROCESSING FAILED FOR SYSMOD *aaaaaaa*.**

**Explanation:**

- *aaaaaaa*—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.

**GIM41802v INLINE JCLIN PROCESSING WAS SUCCESSFUL FOR SYSMOD *aaaaaaa*.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID.

SMP/E successfully processed inline JCLIN for the indicated SYSMOD.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM41901v THE GLOBAL ZONE SYSMOD ENTRY FOR *aaaaaaa* INDICATES THAT IT CONTAINS A ++*bbbbbbbb* MCS. HOWEVER, THERE IS NO ++*bbbbbbbb* MCS FOR *aaaaaaa* IN THE SMPPTS DATA SET.**

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbbb*—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.

**GIM42001v THE FOLLOWING CONDITIONS FOR SYSMOD *aaaaaaa* WERE NOT SATISFIED, BUT WERE IGNORED BECAUSE THE BYPASS OPERAND WAS SPECIFIED. PROCESSING CONTINUES.**

**Explanation:**

- *aaaaaaa*—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 GIM359xx follows this message and names the requisites or holds that were not satisfied.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM42201v 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.

**GIM42301v THE SPECIFIED DATE IS NOT IN THE EXPECTED RANGE.**

**Explanation:** 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*).

**System Action:** Command processing stops.

**Programmer Response:** Correct the date as follows:

- 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.

Then rerun the command.

**GIM42401v THE FOLLOWING PARAMETERS WERE SPECIFIED ON THE EXEC STATEMENT FOR GIMSMP: *aaaa...aaaa*.**

**Explanation:**

- *aaaa...aaaa*—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.

**GIM42601v SYSMOD *aaaaaaa* WAS NOT APPLIED BECAUSE IT DELETES SYSMOD *bbbbbbb*, WHICH WAS ALREADY APPLIED. *aaaaaaa* CAN ONLY BE APPLIED IF IT IS SPECIFIED ON THE SELECT OPERAND.**

**Explanation:**

- *aaaaaaa*—ID of the deleting SYSMOD
- *bbbbbbb*—ID of the SYSMOD being deleted.

SYSMOD *aaaaaaa* deletes another SYSMOD (SYSMOD *bbbbbbb*). Therefore, it will be processed only if it is specified on the SELECT operand.

**System Action:** SYSMOD processing stops. Message GIM37001 follows this message and shows that command processing also fails.

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

- If you want to apply SYSMOD *aaaaaaa*, specify it on the SELECT operand.
- If you do not want to apply SYSMOD *aaaaaaa*, do one of the following:
  - Use the REJECT command to remove SYSMOD *aaaaaaa* from the global zone.
  - Specify SYSMOD *aaaaaaa* on the EXCLUDE operand to exclude it from processing.

Then rerun the APPLY command.

**GIM42602v SYSMOD *aaaaaaa* WAS NOT ACCEPTED BECAUSE IT DELETES SYSMOD *bbbbbbb*, WHICH WAS ALREADY ACCEPTED. *aaaaaaa* CAN ONLY BE ACCEPTED IF IT IS SPECIFIED ON THE SELECT OPERAND.**

**Explanation:**

- *aaaaaaa*—ID of the deleting SYSMOD
- *bbbbbbb*—ID of the SYSMOD being deleted.

SYSMOD *aaaaaaa* deletes another SYSMOD (SYSMOD *bbbbbbb*). Therefore, it will be processed only if it is specified on the SELECT operand.

**System Action:** SYSMOD processing stops. Message GIM37001 follows this message and shows that command processing also fails.

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

- If you want to accept SYSMOD *aaaaaaa*, specify it on the SELECT operand.
- If you do not want to accept SYSMOD *aaaaaaa*, do one of the following:
  - Use the REJECT command to remove SYSMOD *aaaaaaa* from the global zone.
  - Specify SYSMOD *aaaaaaa* on the EXCLUDE operand to exclude it from processing.

Then rerun the ACCEPT command.

**GIM42701v SYSMOD *aaaaaaa* WAS NOT APPLIED BECAUSE IT IS MUTUALLY EXCLUSIVE (NPRE) WITH FUNCTION SYSMOD *bbbbbbb*, WHICH WAS ALREADY APPLIED.**

**Explanation:**

- *aaaaaaa*—ID of the SYSMOD being installed
- *bbbbbbb*—ID of the SYSMOD that is already installed.

Mutually exclusive SYSMODs cannot be installed on the same system. SYSMOD *aaaaaaa* was not applied because a mutually exclusive SYSMOD, SYSMOD *bbbbbbb*, had already been applied.

**System Action:** SYSMOD processing stops. Message GIM37001 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 *aaaaaaa*, do one of the following:
  - Use the REJECT command to remove SYSMOD *aaaaaaa* from the global zone.
  - Specify SYSMOD *aaaaaaa* on the EXCLUDE operand to exclude it from processing.
- If you want to process SYSMOD *aaaaaaa*, use the RESTORE command to remove SYSMOD *bbbbbbb* from the target library and target zone.

Then rerun the APPLY command.

**GIM42702v SYSMOD *aaaaaaa* WAS NOT ACCEPTED BECAUSE IT IS MUTUALLY EXCLUSIVE (NPRE) WITH FUNCTION SYSMOD *bbbbbbb*, WHICH WAS ALREADY ACCEPTED.**

**Explanation:**

- *aaaaaaa*—ID of the SYSMOD being installed
- *bbbbbbb*—ID of the SYSMOD that is already installed.

Mutually exclusive SYSMODs cannot be installed on the same system. SYSMOD *aaaaaaa* was not accepted because a mutually exclusive SYSMOD, SYSMOD *bbbbbbb*, had already been accepted.

**System Action:** SYSMOD processing stops. Message GIM37001 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 *aaaaaaa*, do one of the following:
  - Use the REJECT command to remove SYSMOD *aaaaaaa* from the global zone.
  - Specify SYSMOD *aaaaaaa* on the EXCLUDE operand to exclude it from processing.
- If you want to process SYSMOD *aaaaaaa*, use UCLIN to delete SYSMOD *bbbbbbb*.

Then rerun the ACCEPT command.

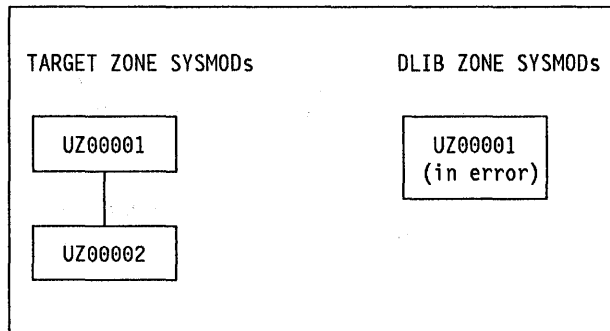
**GIM42901v** SYSMOD *aaaaaaa* WAS NOT RESTORED BECAUSE *bbbbbbb cccccc* IS IN BOTH SYSMOD *aaaaaaa* AND SYSMOD *dddddd*, BUT *dddddd* IS ACCEPTED IN ERROR.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—element type
- *ccccc*—element name
- *dddddd*—SYSMOD ID.

SYSMOD *aaaaaaa*, which is being restored, has an element in common with SYSMOD *dddddd*. However, SMP/E cannot restore the element to the level of SYSMOD *dddddd* because SYSMOD *dddddd* was accepted in error.

For example, assume you have installed the PTFs indicated below:



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 *dddddd* so that it is accepted without errors. Then rerun the RESTORE command for SYSMOD *aaaaaaa*.

**GIM42902v** SYSMOD *aaaaaaa* WAS NOT RESTORED BECAUSE *bbbbbbb cccccc* IS IN BOTH SYSMOD *aaaaaaa* AND SYSMOD *dddddd*, BUT *dddddd* HAS NOT BEEN ACCEPTED.

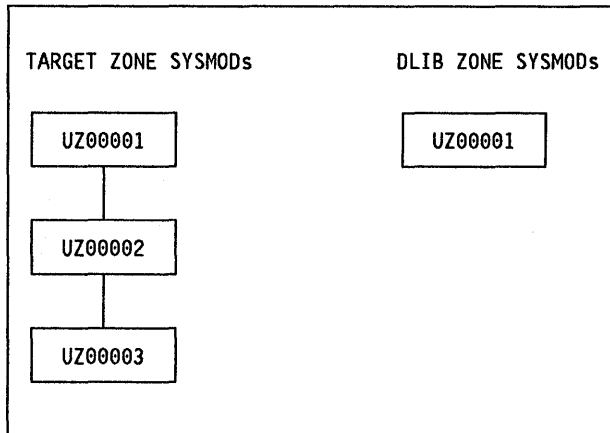
**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *bbbbbbb*—element type
- *ccccc*—element name
- *dddddd*—SYSMOD ID.

SYSMOD *aaaaaaa*, which is being restored, has an element in common with SYSMOD *dddddd*. However, SMP/E cannot restore the element to the level of SYSMOD *dddddd* because SYSMOD *dddddd* has not been accepted.



For example, assume you have installed the PTFs indicated below:



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 *dddddd*, then rerun the RESTORE command for SYSMOD *aaaaaa*.
- Restore both of the SYSMODs.

**GIM43001v** *aaaaaaa* ZONE *bbbb* *cccccc* IS NOT APPLIED.

**Explanation:**

- *aaaaaaa*—target zone name
- *bbbb*—FMID, RMID, or UMID
- *cccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43002v** *aaaaaaa* ZONE *bbbb* *cccccc* WAS NOT RESTORED BECAUSE RELATED SYSMODS FAILED.

**Explanation:**

- *aaaaaaa*—target zone name
- *bbbb*—FMID, RMID, or UMID
- *cccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43003v** *aaaaaaaa ZONE bbbb cccccc IS APPLIED.*

**Explanation:**

- *aaaaaaaa*—target zone name
- *bbbb*—FMID, RMID, or UMID
- *ccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43004v** *aaaaaaaa ZONE bbbb cccccc IS SUPERSEDED.*

**Explanation:**

- *aaaaaaaa*—target zone name
- *bbbb*—FMID, RMID, or UMID
- *ccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43005v** *aaaaaaaa ZONE bbbb cccccc IS DELETED.*

**Explanation:**

- *aaaaaaaa*—zone name
- *bbbb*—FMID, RMID, or UMID
- *ccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43006v** *aaaaaaaa* **ZONE** *bbbb* *cccccc* **IS ACCEPTED BUT NOT APPLIED.**

**Explanation:**

- *aaaaaaaa*—DLIB zone name
- *bbbb*—FMID, RMID, or UMID
- *cccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43007v** *aaaaaaaa* **ZONE** *bbbb* *cccccc* **IS ACCEPTED IN ERROR.**

**Explanation:**

- *aaaaaaaa*—DLIB zone name
- *bbbb*—FMID, RMID, or UMID
- *cccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43008v** *aaaaaaaa* **ZONE** *bbbb* *cccccc* **IS ACCEPTED BUT SUPERSEDED ONLY.**

**Explanation:**

- *aaaaaaaa*—DLIB zone name
- *bbbb*—FMID or UMID
- *cccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx for RESTORE processing.

The indicated SYSMOD is a MODID in a distribution zone element entry, but it is superseded only. For example, it may never have been accepted, but a SYSMOD that superseded it was accepted. Therefore, it cannot be used for RESTORE processing.

**System Action:** Processing of the SYSMOD specified in GIM382xx 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.

**GIM43009v** *aaaaaa* IS THE *bbbb* IN THE DLIB ZONE BUT NOT IN THE TARGET ZONE.

**Explanation:**

- *aaaaaa*—SYSMOD ID
- *bbbb*—FMID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43010v** *aaaaaa* ZONE *bbbb cccccc* DOES NOT SPECIFY ONE OF THE DLIB ZONE MODIDS AS A ++VER PRE OR ++VER SUP VALUE.

**Explanation:**

- *aaaaaa*—target zone name
- *bbbb*—RMID
- *ccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx 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.

**GIM43011v** *aaaaaaa* ZONE *bbbb cccccc* IS APPLIED BUT NOT ACCEPTED.

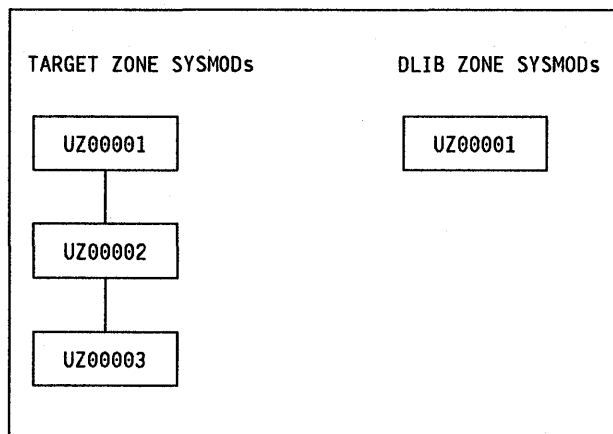
**Explanation:**

- *aaaaaaa*—target zone name
- *bbbb*—FMID, RMID, or UMID
- *ccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 PTFs indicated below:



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 GIM382xx 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.

**GIM43012v** *aaaaaa* ZONE *bbbb* *cccccc* IS NOT SPECIFIED BY ONE OF THE TARGET ZONE MODIDS AS A +-+VER PRE OR +-+VER SUP VALUE.

**Explanation:**

- *aaaaaa*—DLIB zone name
- *bbbb*—UMID
- *cccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx 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 GIM382xx.

**System Action:** Processing of the SYSMOD specified in GIM382xx 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.

**GIM43013v** *aaaaaa* ZONE *bbbb* *cccccc* IS BEING RESTORED.

**Explanation:**

- *aaaaaa*—DLIB zone name
- *bbbb*—UMID
- *cccccc*—SYSMOD ID.

This message further explains the MODID error described by message GIM382xx for **RESTORE** processing. The indicated SYSMOD is being restored. However, no corresponding FMID, RMID, or UMID exists in the distribution zone MODID list.

**System Action:** Processing of the SYSMOD specified in GIM382xx 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.

**GIM43101v** THERE IS NO *aaaaaaaa* VALUE FOR *bbbbbbbb* *cccccccc* IN SYSMOD *ddddddd*.

**Explanation:**

- *aaaaaaaa*—DISTLIB or SYSLIB
- *bbbbbbbb*—element type
- *cccccccc*—element name
- *ddddddd*—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 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.

**GIM43201v** PROGRAM *aaaaaaaa* FAILED WITH A *bbbbbb* ABEND CODE OF *cccccccc*.

**Explanation:**

- *aaaaaaaa*—program name
- *bbbbbb*—SYSTEM or USER
- *cccccccc*—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.

**GIM43301v** SYSMOD *aaaaaaaa* WAS NOT RESTORED BECAUSE *aaaaaaaa* IS SUPERSEDED BY SYSMOD *bbbbbbb*, WHICH IS NOT BEING RESTORED.

**Explanation:**

- *aaaaaaaa*—ID of the superseded SYSMOD
- *bbbbbbb*—ID of superseding SYSMOD.

SMP/E could not restore SYSMOD *aaaaaaaa* because it could not restore SYSMOD *bbbbbbb*. The SYSMOD entry for SYSMOD *aaaaaaaa* indicated that SYSMOD *bbbbbbb* superseded SYSMOD *aaaaaaaa*. (SYSMOD *bbbbbbb* was specified as a SUPBY value, which may have been set by UCLIN.) However, SMP/E could not restore SYSMOD *bbbbbbb* for one of these reasons:

- SYSMOD *bbbbbbb* has already been accepted.
- SYSMOD *bbbbbbb* has not been applied.

As a result, SMP/E also could not restore SYSMOD *aaaaaaaa*.

**System Action:** Processing stops for SYSMOD *aaaaaaaa*.

**Programmer Response:** Eliminate SYSMOD *aaaaaaaa* from RESTORE processing.

**GIM43401v** *aaaaaaaa bbbbbbbb* IN SYSMOD *cccccc* WAS NOT INSTALLED IN ANY TARGET LIBRARY.

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbbb*—element name
- *cccccc*—SYSMOD ID.

The indicated element was not applied to any target libraries.

**Note:** The RMID subentry of the element will be updated, even though the element has not been installed in any target libraries.

**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 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:

- 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.

**GIM43601v** 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.

**GIM43701v** ESTAE ERROR RECOVERY PROCESSING FAILED WITH A RETURN CODE OF *aa*.

**Explanation:**

- *aa*—return code.

An error occurred during ESTAE processing.

**System Action:** System processing stops.

**Programmer Response:** For more information about the ESTAE return code, see the supervisor services and macro instruction manual (ESTAE processing) for your operating system.

**GIM43801v** SMP/E COULD NOT INITIALIZE ITS UTILITY INTERFACE. THE REASON CODE WAS *aa*.

**Explanation:**

- *aa*—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 MVS system. Then rerun the job.

**GIM43802v SMP/E COULD NOT INITIALIZE ITS UTILITY INTERFACE. THE REASON CODE WAS *aa*. THE RETURN CODE WAS *bb*.**

**Explanation:**

- *aa*—two-digit reason code that shows why SMP/E could not initialize the subtask
- *bb*—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 the supervisor services and macro instruction manual for your operating system.

- 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 the supervisor services and macro instruction manual for your operating system.

- 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 the supervisor services and macro instruction manual for your operating system.

**System Action:** SMP/E processing stops.

**Programmer Response:** Fix the error and rerun the job.

**GIM43901v RETRY PROCESSING FAILED FOR THE *aaaaaaa* LIBRARY.**

**Explanation:**

- *aaaaaaa*—ddname of the library.

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. SMP/E processing may stop, depending on where and how the retry processing failed.

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

**GIM43902v RETRY PROCESSING WAS SUCCESSFUL FOR THE *aaaaaaa* LIBRARY.**

**Explanation:**

- *aaaaaaa*—ddname of the library.

SMP/E attempted RETRY processing after a B37–04, D37–04, or E37–04 abend occurred for a library being processed by a utility. RETRY processing was successful.

**System Action:** SMP/E processing continues.

**Programmer Response:** None.

**GIM44001v SYSTEM ABEND *aaaa* OCCURRED WITH A REASON CODE OF *bb* AFTER SMP/E CALLED THE *ccccccc* UTILITY. THE *ddddddd* LIBRARY RAN OUT OF SPACE. SMP/E WILL RETRY THE UTILITY.**

**Explanation:**

- *aaaa*—abend code
- *bb*—abend reason code
- *ccccccc*—name of the load module for the utility
- *ddddddd*—ddname of the library.



The indicated library ran out of space when it was being processed by the utility. The RETRY operand was specified on the command being run, so SMP/E will retry the utility.

**System Action:** RETRY processing is done.

**Programmer Response:** Your action depends on the success or failure of RETRY processing.

**GIM44002v** **USER ABEND *aaaa* OCCURRED WITH A REASON CODE OF *bb* AFTER SMP/E CALLED THE *cccccccc* UTILITY. THE *ddddddd* LIBRARY RAN OUT OF SPACE. SMP/E WILL RETRY THE UTILITY.**

**Explanation:**

- *aaaa*—abend code
- *bb*—abend reason code
- *cccccccc*—name of the load module for the utility
- *ddddddd*—ddname of the library.

The indicated library ran out of space when it was being processed by the utility. The RETRY operand was specified on the command being run, so SMP/E will retry the utility.

**System Action:** RETRY processing is done.

**Programmer Response:** Your action depends on the success or failure of RETRY processing.

**GIM44003v** **SYSTEM ABEND *aaaa* OCCURRED WITH A REASON CODE OF *bb* AFTER SMP/E CALLED THE *cccccccc* UTILITY. THE *ddddddd* LIBRARY RAN OUT OF SPACE. SMP/E WILL NOT RETRY THE UTILITY. THE REASON CODE IS *ee*.**

**Explanation:**

- *aaaa*—abend code
- *bb*—abend reason code
- *cccccccc*—name of the load module for the utility
- *ddddddd*—ddname of the library
- *ee*—reason code that indicates why RETRY processing was not attempted.

The indicated library ran out of space when it was being processed by the utility. However, SMP/E did not retry the utility. The reason is indicated by one of the following codes:

- 01—No system diagnostic work area (SDWA) was provided to the ESTAE routine.
- 02—A user abend occurred.
- 03—The SMP/E command did not specify RETRY.
- 04—RETRY was already in progress when the abend occurred.
- 05—The abend code was not B37, D37, or E37, or the abend reason code was not 04.
- 06—The indicated library was not specified in the RETRYDD list in the OPTIONS entry that was in effect.
- 07—The abend occurred for a library other than the indicated library.  
See the system messages to determine the cause of the failure.
- 08—The RETRY exit routine (exit routine 2) canceled retry processing.
- 09—SMP/E is not running under MVS.

**System Action:** Command processing stops in the situations described by reason codes 01–07 and 09.

For reason code 08, command processing stops with a return code equal to the return code from the RETRY exit routine. If the exit routine issued a return code of 16, all SMP/E processing stops.

**Programmer Response:** Your action depends on the abend code and the reason why SMP/E did not attempt the RETRY.

**GIM44004v** USER ABEND *aaaa* OCCURRED WITH A REASON CODE OF *bb* AFTER SMP/E CALLED THE *ccccccc* UTILITY. THE *ddddddd* LIBRARY RAN OUT OF SPACE. SMP/E WILL NOT RETRY THE UTILITY. THE REASON CODE IS *ee*.

**Explanation:**

- *aaaa*—abend code
- *bb*—abend reason code
- *ccccccc*—name of the load module for the utility
- *ddddddd*—ddname of the library
- *ee*—reason code that indicates why RETRY processing was not attempted.

The indicated library ran out of space when it was being processed by the utility. However, SMP/E did not retry the utility. The reason is indicated by one of the following codes:

- 01—No system diagnostic work area (SDWA) was provided to the ESTAE routine.
- 02—A user abend occurred.
- 03—The SMP/E command did not specify RETRY.
- 04—RETRY was already in progress when the abend occurred.
- 05—The abend code was not B37, D37, or E37, or the abend reason code was not 04.
- 06—The indicated library was not specified in the RETRYDD list in the OPTIONS entry that was in effect.
- 07—The abend occurred for a library other than the indicated library.  
See the system messages to determine the cause of the failure.
- 08—The RETRY exit routine (exit routine 2) canceled retry processing.
- 09—SMP/E is not running under MVS.

**System Action:** Command processing stops in the situations described by reason codes 01–07 and 09.

For reason code 08, command processing stops with a return code equal to the return code from the RETRY exit routine. If the exit routine issued a return code of 16, all SMP/E processing stops.

**Programmer Response:** Your action depends on the abend code and the reason why SMP/E did not attempt the RETRY.

**GIM44005v** SYSTEM ABEND *aaaa* OCCURRED WITH A REASON CODE OF *bb* AFTER SMP/E CALLED THE *ccccccc* UTILITY. SMP/E WILL NOT RETRY THE UTILITY. THE REASON CODE IS *dd*.

**Explanation:**

- *aaaa*—abend code
- *bb*—abend reason code
- *ccccccc*—name of the load module for the utility
- *dd*—reason code that indicates why RETRY processing was not attempted.

An abend occurred during processing by the indicated utility. However, SMP/E did not retry the utility. The reason is indicated by one of the following codes:

- 01—No system diagnostic work area (SDWA) was provided to the ESTAE routine.
- 02—A user abend occurred.
- 03—The SMP/E command did not specify RETRY.
- 04—RETRY was already in progress when the abend occurred.
- 05—The abend code was not B37, D37, or E37, or the abend reason code was not 04.
- 08—The RETRY exit routine (exit routine 2) canceled retry processing.
- 09—SMP/E is not running under MVS.

**System Action:** Command processing stops in the situations described by reason codes 01–05 and 09.

For reason code 08, command processing stops with a return code equal to the return code from the RETRY exit routine. If the exit routine issued a return code of 16, all SMP/E processing stops.

**Programmer Response:** Your action depends on the abend code and the reason why SMP/E did not attempt the RETRY.

**GIM44006v** USER ABEND *aaaa* OCCURRED WITH A REASON CODE OF *bb* AFTER SMP/E CALLED THE *cccccccc* UTILITY. SMP/E WILL NOT RETRY THE UTILITY. THE REASON CODE IS *dd*.

**Explanation:**

- *aaaa*—abend code
- *bb*—abend reason code
- *cccccccc*—name of the load module for the utility
- *dd*—reason code that indicates why RETRY processing was not attempted.

An abend occurred during processing by the indicated utility. However, SMP/E did not retry the utility. The reason is indicated by one of the following codes:

- 01—No system diagnostic work area (SDWA) was provided to the ESTAE routine.
- 02—A user abend occurred.
- 03—The SMP/E command did not specify RETRY.
- 04—RETRY was already in progress when the abend occurred.
- 05—The abend code was not B37, D37, or E37, or the abend reason code was not 04.
- 08—The RETRY exit routine (exit routine 2) canceled retry processing.
- 09—SMP/E is not running under MVS.

**System Action:** Command processing stops in the situations described by reason codes 01–05 and 09.

For reason code 08, command processing stops with a return code equal to the return code from the RETRY exit routine. If the exit routine issued a return code of 16, all SMP/E processing stops.

**Programmer Response:** Your action depends on the abend code and the reason why SMP/E did not attempt the RETRY.

**GIM44101v** SYSMOD *aaaaaaa* DOES NOT SUPERSEDE UMID *bbbbbbb* FOR *cccccccc* *ddddddd*.

**Explanation:**

- *aaaaaaa*—SYSMOD ID
- *cccccccc*—element type
- *ddddddd*—element name
- *bbbbbbb*—UMID specified on the element MCS.

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.

**GIM44201v** 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 your local support group.

**GIM44202v** 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 your local support group.

**GIM44203v** 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 your local support group.

- 
- GIM44204v 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 your local support group.
- GIM44205v 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 your local support group.
- GIM44207v GIMMPVIA - AN UNRECOGNIZABLE VSAM I/O FUNCTION WAS ATTEMPTED.**  
**System Action:** See previous related messages.  
**Programmer Response:** See previous related messages. If necessary, contact your local support group.
- GIM44208v 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 your local support group.
- GIM44209v 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.
- GIM44210v GIMMPVIA - A SEQUENCING ERROR OCCURRED UPDATING A VSAM RECORD.**  
**System Action:** See previous related messages.  
**Programmer Response:** See previous related messages. If necessary, contact your local support group.
- GIM44211v GIMMPVIA - A SEQUENCING ERROR OCCURRED DELETING A VSAM RECORD.**  
**System Action:** See previous related messages.  
**Programmer Response:** See previous related messages. If necessary, contact your local support group.
- GIM44212v GIMMPMRM - IOPTYPE *aaaaaaaa* IS NOT A MAP/REMAPPABLE ENTRY TYPE.**  
**Explanation:**  
• *aaaaaaaa*—the TYPE field of the GIMMPIO parameter list.  
**System Action:** See previous related messages.  
**Programmer Response:** See previous related messages.
- GIM44213v GIMMPMRM - THIS UNKNOWN IOPTYPE IS NOT MAP/REMAPPABLE.**  
**System Action:** See previous related messages.  
**Programmer Response:** See previous related messages.
- GIM44214v GIMMPMRM - THE IOPDSID IS UNKNOWN.**  
**System Action:** See previous related messages.  
**Programmer Response:** See previous related messages.

**GIM44215v GIMMPMRM - IOPFUNCT *aaaaaaaa* IS NOT SUPPORTED.**

**Explanation:**

- *aaaaaaaa*—the FUNCTION field of the GIMMPIO parameter list.

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages.

**GIM44216v GIMMPMRM - THE IOPFUNCT IS UNKNOWN.**

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages.

**GIM44217v GIMMPMRM - *aaaaaaaa* + ' *bbbbbb* ' CALLED GIMMPIO FOR *ccccccc* FOR *dddddd* *eeeeeeee*.**

**Explanation:**

- *aaaaaaaa*—name of the module that called GIMMPIO
- *bbbbbb*—the hexadecimal offset into that module
- *ccccccc*—IOPFUNCT value (such as BLDL, OPEN, READ)
- *dddddd*—IOPTYPE value (such as ASM, DLB, MOD)
- *eeeeeeee*—IOPNAME (element name or PTF number).

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages.

**GIM44218v GIMMPMRM - NO VPLMSGP ERROR TEXT WAS RETURNED FROM GIMMPVIA.**

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages.

**GIM44220v GIMMPMRM - THERE IS AN ERROR IN THE IOPTYPE VALUE PASSED TO GIMADTBL.**

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages.

**GIM44221v GIMMPMRM - A DUPLICATE RECORD WAS DETECTED WHILE TRYING TO ADD TO THE *aaaaaaaa* ZONE.**

**Explanation:**

- *aaaaaaaa*—zone name from the SET command.

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages.

**GIM44222v GIMMPVIA - THE FUNCTION REQUESTED (VPLFUNCT) IS NOT SUPPORTED.**

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages. If necessary, contact your local support group.

**GIM44223v GIMMPVIA - THE VSAM ACCESS ID (VPLACID) IS INVALID.**

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages. If necessary, contact your local support group.

**GIM44224v GIMMPVIA - THE CLUSTER IS NOT OPEN. ACCESS IS DENIED.**

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages. If necessary, contact your local support group.

**GIM44225v GIMMPVIA - VSAM DATA SET aaaaaaaaaa IS OUT OF SPACE.****Explanation:**

- aaaaaaaaaa—CSI data set name.

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages. If necessary, contact your local support group.

**GIM44227v 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.

**GIM44228v GIMMPVIA - THE aaaaaaa ZONE WAS ALREADY OPEN.****Explanation:**

- aaaaaaa—name of zone to be opened.

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages. If necessary, contact your local support group.

**GIM44229v GIMMPVIA - ZONE aaaaaaa CANNOT BE OPENED FOR UPDATE BECAUSE THE PATH CONTAINING THE ZONE IS ALREADY OPENED WITH 'NOUPDATE'.****Explanation:**

- aaaaaaa—name of zone to be opened.

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages. If necessary, contact your local support group.

**GIM44230v 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 your local support group.

**GIM44231v 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 your local support group.

**GIM44232v GIMMPVIA - DYNAMIC ALLOCATION FAILED FOR THE aaaaaaa ZONE, DATA SET bbbbbbb.****Explanation:**

- aaaaaaa—zone name
- bbbbbbb—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.

**GIM44233v GIMMPVIA - A PREALLOCATED VSAM DATA SET WAS NOT FOUND FOR THE *aaaaaaa* ZONE.**

**Explanation:**

- *aaaaaaa*—zone name.

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages. If necessary, contact your local support group.

**GIM44234v GIMMPVIA - SMPCSI DATA SET *aaaaaaaaa* IS BUSY.**

**Explanation:**

- *aaaaaaaaa*—CSI data set name.

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages. If necessary, contact your local support group.

**GIM44236v 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 your local support group.

**GIM44237v 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 your local support group.

**GIM44238v GIMMPRPS - THE FUNCTION (RRLFUNCT) IS NOT RECOGNIZED.**

**System Action:** See previous related messages.

**Programmer Response:** Report the problem to your support group.

**GIM44239v 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 your support group.

**GIM44240v GIMMPRPS - THE ZONE PATH ID IS INVALID.**

**System Action:** See previous related messages.

**Programmer Response:** Report the problem to your support group.

**GIM44241v GIMMPRPS - THE CATEGORY IS INVALID.**

**System Action:** See previous related messages.

**Programmer Response:** Report the problem to your support group.

**GIM44242v GIMMPRPS - A FREEPOOL ERROR OCCURRED. THE POOL IS MISSING.**

**System Action:** See previous related messages.

**Programmer Response:** Report the problem to your support group.

**GIM44243v GIMMPRPS - THE RECORD TO BE DEASSIGNED IS MISSING.**

**System Action:** See previous related messages.

**Programmer Response:** Report the problem to your support group.

**GIM44244v GIMVSMMSG - NO REPRESENTATIONS MET INPUT REQUIREMENTS.**

**System Action:** Processing continues.

**Programmer Response:** See previous related messages.

**GIM44245v GIMVSMMSG - 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.

**GIM44246v GIMVSMMSG - THERE IS A PROBABLE PHYSICAL I/O ERROR.**

**Explanation:** A required device is not available.

**System Action:** SMP/E processing stops.

**Programmer Response:** Contact your local support group.

**GIM44247v GIMVSMMSG - THERE IS A PROBABLE LOGICAL I/O ERROR.**

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44248v GIMVSMMSG - THE RETURN CODE FOR RRLFUNCT IS UNKNOWN.**

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44249v GIMVSMMSG - 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.

**GIM44250v GIMVSMMSG - THE VSAM ERROR ANALYSIS OCCURRED FOR *aaaaaaaa* + X'*bbbb*', RRLFUNCT=' *ccccccc* ', WITH RRLRETN=' *dddd* '.**

**Explanation:**

- *aaaaaaaa*—module name
- *bbbb*—offset into that module
- *ccccccc*—function value for RRLFUNCT
- *dddd*—return code value for RRLRETN.

**System Action:** See previous related messages.

**Programmer Response:** See previous related messages.

**GIM44251v GIMVSMMSG - 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 your local support group.



- GIM44252v GIMVSMMSG - 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.
- GIM44253v GIMVSMMSG - THE REPRESENTATION DID NOT MEET THE ASSERTION CRITERIA.**  
Explanation: The POSU (position for update) macro could not find a record.  
System Action: Depending on the severity of the error, processing may continue or stop.  
Programmer Response: Make sure the record to be processed actually exists. See previous related messages for more information.
- GIM44254v GIMVSMMSG - 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.
- GIM44255v GIMVSMMSG - 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.
- GIM44256v GIMVSMMSG - 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.
- GIM44257v GIMVSMMSG - 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 your local support group.
- GIM44258v GIMVSMMSG - 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 your local support group.
- GIM44259v GIMVSMMSG - 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 your local support group.

- 
- GIM44260v GIMVSMMSG - 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 your local support group.
- GIM44261v GIMVSMMSG - THE ZONE HAS NOT BEEN ASSIGNED.**  
**Explanation:** An attempt to open the zone failed.  
**System Action:** See previous related messages.  
**Programmer Response:** See previous related messages.
- GIM44262v GIMVSMMSG - NO ZONEINDEX EXISTS FOR THE ZONE.**  
**System Action:** Command processing stops.  
**Programmer Response:** See previous related messages. If necessary, define a ZONEINDEX sub-entry for the zone.
- GIM44263v GIMVSMMSG - 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 your support group.
- GIM44264v A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. THE VSAM MACRO SHOWCB THAT REQUESTS THE *bbb* LENGTH FAILED.**  
**Explanation:**
  - *aaaaaaaa*—an identifier for the module that issued this message
  - *bbb*—name of a VSAM control block (ACB or RPL).**System Action:** Command processing stops.  
**Programmer Response:** Report the error to your support group.
- GIM44265v A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. THE VSAM MACRO GENCB THAT CREATES THE *bbb* FAILED.**  
**Explanation:**
  - *aaaaaaaa*—an identifier for the module that issued this message
  - *bbb*—name of a VSAM control block (ACB or RPL).**System Action:** Command processing stops.  
**Programmer Response:** Report the error to your support group.
- GIM44266v A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. OPEN PROCESSING FAILED FOR VSAM DATA SET *bbbbbbbbbb*.**  
**Explanation:**
  - *aaaaaaaa*—an identifier for the module that issued this message
  - *bbbbbbbbbb*—name of a CSI data set.**System Action:** Command processing stops.  
**Programmer Response:** Make sure the OLDCSI operand on the CONVERT command specifies the correct CSI data set. Then rerun the job.

**GIM44267v** A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. THE VSAM MACRO SHOWCB THAT REQUESTS THE KEY AND RECORD LENGTH OF A VSAM DATA SET FAILED.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message.

**System Action:** Command processing stops.

**Programmer Response:** Report the error to your support group.

**GIM44268v** A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. THE KEY LENGTH OF VSAM DATA SET *bbbbbbbbbb* IS INVALID.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbbbb*—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.

**GIM44269v** A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. THE RECORD LENGTH OF VSAM DATA SET *bbbbbbbbbb* IS INVALID.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbbbb*—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.

**GIM44270v** A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. *bbbbbb* FAILED FOR VSAM DATA SET *cccccccccc*.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbb*—VERIFY or ENDREC
- *cccccccccc*—name of a CSI data set.

**System Action:** Command processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44271v** A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. THE VSAM MACRO MODCB THAT MODIFIES THE *bbb* FAILED.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbb*—name of a VSAM control block (ACB or RPL).

**System Action:** Command processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44272v** A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. THE VSAM MACRO POINT FAILED TRYING TO POSITION TO THE BEGINNING OF VSAM DATA SET *bbbbbbbbbb*.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbbbb*—name of a CSI data set.

**System Action:** Command processing stops.

**Programmer Response:** Report the problem to your support group.

- 
- GIM44273v** A VSAM ERROR OCCURRED IN MODULE *aaaaaaaa*. A FAILURE OCCURRED TRYING TO GET THE NEXT RECORD FROM VSAM DATA SET *bbbbbbbbbb*.
- Explanation:**
- *aaaaaaaa*—an identifier for the module that issued this message
  - *bbbbbbbbbb*—name of a CSI data set.
- System Action:** Command processing stops.
- Programmer Response:** Report the problem to your support group.
- GIM44276v** 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.
- GIM44279v** GIMMPVIA - VSAM OPEN FAILED.
- System Action:** See previous related messages.
- Programmer Response:** See previous related messages. If necessary, contact your local support group.
- GIM44280v** 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.
- GIM44281v** GIMMPVIA - A VSAM OPEN SHOWCB MACRO FAILED OBTAINING INITIAL STATISTICS.
- System Action:** See previous related messages.
- Programmer Response:** See previous related messages. If necessary, contact your local support group.
- GIM44282v** GIMMPVIA - A VSAM OPEN SHOWCB MACRO FAILED OBTAINING INDEX CI (CONTROL INTERVAL) SIZE.
- System Action:** See previous related messages.
- Programmer Response:** See previous related messages. If necessary, contact your local support group.
- GIM44283v** GIMMPVIA - A VSAM OPEN SHOWCB MACRO FAILED OBTAINING DATA CI (CONTROL INTERVAL) SIZE.
- System Action:** See previous related messages.
- Programmer Response:** See previous related messages. If necessary, contact your local support group.
- GIM44284v** GIMMPVIA - A VSAM OPEN FAILED WITH THE FOLLOWING INVALID: *aaaaaaaa*
- Explanation:**
- *aaaaaaaa*—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.

---

**GIM44301v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - *bbbbbbbb* *ccccccc* TO BE RETRANSFORMED WAS NOT IN TRANSFORMED FORMAT.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—element type
- *ccccccc*—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 your support group.

**GIM44302v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - *bbbbbbbb* *ccccccc* TO BE RETRANSFORMED IS NOT ON THE SMPWRK6 DATA SET.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—element type
- *ccccccc*—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 your support group.

**GIM44303v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THERE IS AN ERROR IN THE PARAMETER LIST FOR GIMDRS.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message.

**System Action:** SYSMOD processing stops.

**Programmer Response:** Report the error to your support group.

**GIM44304v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - AN INVALID VALUE WAS SPECIFIED ON THE OPT PARAMETER PASSED TO MODULE GIMZDEL.**

**Explanation:**

- *aaaaaaaa*—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 your support group.

**GIM44305v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE RETURN CODE FOR A UCL DEL STATEMENT INDICATED THAT NOTHING WAS DELETED. THE STATEMENT WAS CHECKED IN *aaaaaaaa*.**

**Explanation:**

- *aaaaaaaa*—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 your support group.

**GIM44306v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - 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:**

- *aaaaaaaa*—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 your support group.

**GIM44307v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE RETURN CODE IS NOT ZERO FOR THE VALUE BEING CHECKED IN THE DECODE TABLE. HOWEVER, THAT VALUE WAS PREVIOUSLY ACCEPTED AS VALID.**

**Explanation:**

- *aaaaaaaa*—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:** Command processing stops.

**Programmer Response:** Report the error to your support group.

**GIM44308v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE RETURN CODE FOR A UCL ADD STATEMENT INDICATED THAT NOTHING WAS ADDED. THE STATEMENT WAS CHECKED IN *aaaaaaaa*.**

**Explanation:**

- *aaaaaaaa*—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 your support group.

**GIM44309v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - AN ERROR OCCURRED IN MODULE GIMMPREC. THE MCS ENTRY WAS NOT FOUND IN A CALL TO GIMADTBL.**

**Explanation:**

- *aaaaaaaa*—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 your support group.

**GIM44310v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - GIMDSMGR WAS CALLED FROM *bbbbbbb* + X'*cccccc*'.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbb*—an identifier for the module that called GIMDSMGR
- *cccccc*—offset from the beginning of module *bbbbbbb*.

This message accompanies other GIM443xx messages that were issued during processing by GIMDSMGR.

**System Action:** See the other GIM443xx messages issued with this message.

**Programmer Response:** See the other GIM443xx messages issued with this message.

**GIM44311v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE FIRST PHASE PROCESSED IS NOT PHASE 1 FOR COMMAND *bbbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—an SMP/E command.

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310 is also issued to identify this module. Otherwise, message GIM44318 is issued along with this message.

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44312v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE CURRENT PHASE SHOULD HAVE PRECEDED THE PREVIOUS PHASE FOR COMMAND *bbbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—an SMP/E command.

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310 is also issued to identify this module. Otherwise, message GIM44318 is issued along with this message.

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44313v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE CLEANUP PHASE WAS NOT DONE FOR THE PREVIOUS COMMAND, *bbbbbbbb*. THE CURRENT COMMAND IS *cccccccc*.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—an SMP/E command
- *cccccccc*—an SMP/E command.

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310 is also issued to identify this module. Otherwise, message GIM44318 is issued along with this message.

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44314v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - SMP/E COULD NOT GET EXTRA RESOURCES FOR COMMAND *bbbbbbbb* BECAUSE THEY WERE HELD WITHOUT A REASON.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—an SMP/E command.

An abnormal condition occurred because of an internal processing error. If SMP/E can determine which module called GIMDSMGR, message GIM44310 is also issued to identify this module. Otherwise, message GIM44318 is issued along with this message.

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44315v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE NAME OF THE DATA SET TO BE UPDATED WAS NOT PROVIDED.**

**Explanation:**

- *aaaaaaaa*—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 GIM44310 is also issued to identify this module. Otherwise, message GIM44318 is issued along with this message.

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44316v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE CURRENT PHASE FOR COMMAND *bbbbbbbb* IS UNKNOWN.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—an SMP/E command.

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310 is also issued to identify this module. Otherwise, message GIM44318 is issued along with this message.

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44317v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE *bbbbbbbb* PHASE WAS ATTEMPTED FOR COMMAND *cccccccc* BEFORE THE *dddddddd* PHASE WAS PROCESSED.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—a phase of command processing
- *cccccccc*—an SMP/E command
- *dddddddd*—a phase of command processing.

An abnormal condition occurred because of a phase error. If SMP/E can determine which module called GIMDSMGR, message GIM44310 is also issued to identify this module. Otherwise, message GIM44318 is issued along with this message.

**System Action:** SMP/E processing stops.

**Programmer Response:** Report the problem to your support group.

**GIM44318v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - SMP/E COULD NOT DETERMINE WHICH MODULE CALLED GIMDSMGR.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message.

This message accompanies other GIM443xx messages that were issued during processing by GIMDSMGR.

**System Action:** See the other GIM443xx messages issued with this message.

**Programmer Response:** See the other GIM443xx messages issued with this message.

**GIM44319v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - AN ERROR HAS OCCURRED IN SUBROUTINE *bbbbbbbb*. THE *ccc* ENTRY WAS NOT FOUND IN A CALL TO MODULE GIMADTBL.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—the subroutine name
- *ccc*—the entry type.

**System Action:** Command processing stops.

**Programmer Response:** Report the error to your support group.



**GIM44320v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - AN INVALID PARAMETER WAS PASSED TO MODULE *aaaaaaaa*.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message.

A parameter that was passed to the indicated module was overlaid.

**System Action:** Processing depends on the return code.

**Programmer Response:** Report the error to your support group.

**GIM44321v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - THE CALL FOR ANALYSIS SHOULD NOT HAVE BEEN MADE BECAUSE RRLRETN IS ZERO.**

**Explanation:**

- *aaaaaaaa*—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 your support group.

**GIM44322v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - AN ERROR HAS OCCURRED IN SUBROUTINE XRFPTF1. A RELATION ENTRY WAS NOT FOUND IN A CALL TO MODULE GIMADTBL.**

**Explanation:**

- *aaaaaaaa*—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 your support group.

**GIM44323v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - GIMUTTBL IS TOO SMALL -- LESS THAN 10 BYTES.**

**Explanation:**

- *aaaaaaaa*—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 related appendix in *SMP/E Reference*.

**GIM44324v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - GIMUTTBL DOES NOT HAVE AN END-OF-LIST ENTRY.**

**Explanation:**

- *aaaaaaaa*—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 related appendix in *SMP/E Reference*.

**GIM44325v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - SMP/E COULD NOT ADD THE FMID OF SYSMOD *bbbbbbb* TO THE GLOBAL ZONE, BECAUSE OF A VSAM ERROR.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbb*—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 your support group.

**GIM44326v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - SMP/E COULD NOT FIND REGRESSED SYSMOD *bbbbbbb* IN THE *ccccccc* ZONE.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbb*—SYSMOD ID
- *cccccc*—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.

**System Action:** Command processing stops.

**Programmer Response:** Report the error to your support group.

**GIM44327v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - *bbbbbbb* WAS ATTEMPTED FOR A NONEXISTENT ENTRY - *ccccccc*.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbb*—EXTEND, POSITION, or READ
- *ccccccc*—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 your support group.

**GIM44328v AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - GIMMPIO WAS CALLED FROM *bbbbbbb* + X'*ccccccc*' WITH IOPFUNCT = *ddddddd*, IOPTYPE = *eeeeeee* AND IOPNAME = *ffffff*.**

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbb*—name of the module that called GIMMPIO
- *ccccccc*—offset from the beginning of module *bbbbbbb*
- *ddddddd*—I/O function or UNKNOWN
- *eeeeeee*—entry type or UNKNOWN
- *ffffff*—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 your support group.

**GIM44329v** AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - LMOD ENTRY *bbbbbbbb* HAS NO VALID SYSLIB VALUE.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—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.

**GIM44330v** AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - A RECORD IN THE *bbbbbbbb* ENTRY CONTAINS AN UNKNOWN RELATIONSHIP. THE KEY OF THE RECORD IS X'*cccccc*'.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—entry type
- *cccccc*—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:** Report the error to your support group.

**GIM44331v** AN UNUSUAL CONDITION OCCURRED. *aaaaaaaa* - SYSMOD *bbbbbbbb* COULD NOT BE FOUND IN THE *cccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—an identifier for the module that issued this message
- *bbbbbbbb*—SYSMOD ID
- *cccccc*—type of zone (target or distribution).

SYSMOD *bbbbbbbb* has elements in common with a SYSMOD that failed (and was named in a GIM302xx message). However, SYSMOD *bbbbbbbb* could not be found in the indicated type of zone.

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

**Programmer Response:** None.

**GIM44401v** *aaaaaaaa* WAS NOT ASSEMBLED FOR *bbbbbbbb ccccccc* IN SYSMOD *ddddddd* BECAUSE THERE IS NO ASSEM ENTRY FOR *aaaaaaaa*.

**Explanation:**

- *aaaaaaaa*—name of an ASSEM entry or MOD entry
- *bbbbbbbb*—MAC or SRC
- *ccccccc*—name of the MAC or SRC entry used for the assembly
- *ddddddd*—ID of the SYSMOD that contains the MAC or SRC.

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.

**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 rerun the job.

**GIM44402v** *aaaaaaaa* WAS NOT ASSEMBLED FOR *bbbbbbbb* *ccccccc* IN SYSMOD *ddddddd* BECAUSE THERE IS NO MOD ENTRY FOR *aaaaaaaa*. SMP/E CANNOT DETERMINE THE TARGET LIBRARY FOR THE ASSEMBLER OUTPUT.

**Explanation:**

- *aaaaaaaa*—name of an ASSEM entry or MOD entry
- *bbbbbbbb*—MAC or SRC
- *ccccccc*—name of the MAC or SRC entry used for the assembly
- *ddddddd*—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.

**GIM44501v** CHANGES FOR THE FOLLOWING USERMODS OR APAR FIXES WILL BE LOST BECAUSE SYSMOD *aaaaaaaa* IS BEING INSTALLED.

**Explanation:**

- *aaaaaaaa*—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 GIM44601 follows this message and lists the USERMODs and APAR fixes whose changes were replaced.

**Programmer Response:** None.

**GIM44502v** 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 GIM44601 follows this message and lists the USERMODs whose changes were replaced.

**Programmer Response:** None.

**GIM44601v** *aaaaaaaa* *bbbbbbb* IN *ccccccc* *ddddddd*.

**Explanation:**

- *aaaaaaaa*—USERMOD or APAR
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—element type
- *ddddddd*—element name.

This message lists the USERMODs and APAR fixes whose changes have been replaced. The reason is given in message GIM44501 or GIM44502.

**System Action:** None.

**Programmer Response:** None.

**GIM44701v** MODULE *aaaaaaaa* IN SYSMOD *bbbbbbb* WAS NOT REASSEMBLED BECAUSE THE REUSE OPERAND WAS SPECIFIED. THE OBJECT MODULE SAVED IN SMPWRK3 WAS USED INSTEAD.

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbb*—SYSMOD ID.

The indicated module was first assembled when SMP/E unsuccessfully tried to install SYSMOD *bbbbbbb*. 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.

**GIM44801v** *aaaaaaaa* WAS ASSEMBLED BECAUSE THE ASSEMBLE SUBENTRY WAS SET IN THE MOD ENTRY.

**Explanation:**

- *aaaaaaaa*—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.

**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.

**GIM44802v** *aaaaaaaa* WAS ASSEMBLED BECAUSE ITS NAME MATCHES THE PREFIX VALUE ON A ++MAC OR ++MACUPD STATEMENT.

**Explanation:**

- *aaaaaaaa*—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.

**GIM44803v** *aaaaaaaa* WAS ASSEMBLED BECAUSE IT WAS INCLUDED IN A NEW LOAD MODULE DURING LOAD MODULE BUILD PROCESSING.

**Explanation:**

- *aaaaaaaa*—module name.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM44901v SMP/E ADDED THE *aaaaaaaa* SUBENTRY TO CREATE A VALID ENTRY.****Explanation:**

- *aaaaaaaa*—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.

**GIM45001v AN I/O ERROR OCCURRED FOR *aaaaaaaa bbbbbbbb* ON THE *cccccc* LIBRARY.****Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *cccccc*—ddname of the library.

This message follows message GIM274xx 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.

**GIM45101v LOG RECORDING WAS SWITCHED TO *aaaaaa* BECAUSE *bbbbbb* IS FULL.****Explanation:**

- *aaaaaa*—ddname of the LOG data set that is full
- *bbbbbb*—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.

**GIM45301v DATA IS MISSING FROM THE ++HOLD FOR SYSMOD *aaaaaa*. *bbbbbb* IS REQUIRED.****Explanation:**

- *aaaaaa*—SYSMOD ID
- *bbbbbb*—TYPE, FMID, or REASON.

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.

---

**GIM45401v** *aaaaaaaa bbbbbbbb* IN SYSMOD *ccccccc* WAS NOT PROCESSED BECAUSE THE FMID ON THE ++VER STATEMENT DOES NOT MATCH THE FMID IN THE ELEMENT ENTRY.

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbb*—element name
- *ccccccc*—SYSMOD ID.

**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.

**GIM45601v** THE SMPTLIB DDDEF WAS NOT UPDATED BECAUSE VOLUME SUBENTRY *aaaaaaaa* ALREADY EXISTS.

**Explanation:**

- *aaaaaaaa*—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.

**GIM50001v** THERE IS AN ERROR IN *aaaaaaaa* SYSMOD *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—SYSMOD type
- *bbbbbbb*—SYSMOD ID or blank, if unknown.

**Note:** This message immediately follows message GIM203xx, 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.

**GIM50002v** THERE IS AN ERROR IN AN UNKNOWN TYPE OF SYSMOD *aaaaaaaa*.

**Explanation:**

- *aaaaaaaa*—SYSMOD ID or blank, if unknown.

**Note:** This message immediately follows message GIM203xx, 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.

**GIM50003v** THERE IS AN ERROR IN A ++*aaaaaaaa* MCS FOR ELEMENT *bbbbbbbb* IN *ccccccc* SYSMOD *ddddddd*.

**Explanation:**

- *aaaaaaaa*—MCS type or blank, if unknown
- *bbbbbbbb*—element name or blank, if unknown
- *ccccccc*—SYSMOD type
- *ddddddd*—SYSMOD ID or blank, if unknown.

**Note:** This message immediately follows message GIM203xx, 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.

**GIM50004v** THERE IS AN ERROR IN AN UNKNOWN MCS FOR *aaaaaaaa* SYSMOD *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—SYSMOD type
- *bbbbbbb*—SYSMOD ID or blank, if unknown.

**Note:** This message immediately follows message GIM203xx, 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.

**GIM50005v** THERE IS AN ERROR IN A ++*aaaaaaaa* MCS FOR AN UNKNOWN ELEMENT IN *bbbbbbbb* SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—MCS type
- *bbbbbbbb*—SYSMOD type
- *ccccccc*—SYSMOD ID or blank, if unknown.

**Note:** This message immediately follows message GIM203xx, 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.



**GIM50006v** THERE IS AN ERROR IN A ++IF MCS ASSOCIATED WITH ++VER MCS *aaaaaaaa* IN *bbbbbbbb* SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—number of the ++VER MCS
- *bbbbbbbb*—SYSMOD type
- *ccccccc*—SYSMOD ID or blank, if unknown.

**Note:** This message immediately follows message GIM203xx, 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.

**GIM50007v** THERE IS AN ERROR IN ++VER MCS *aaaaaaaa* IN *bbbbbbbb* SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—number of the ++VER MCS
- *bbbbbbbb*—SYSMOD type
- *ccccccc*—SYSMOD ID or blank, if unknown.

**Note:** This message immediately follows message GIM203xx, 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.

**GIM50008v** THERE IS AN ERROR IN A ++*aaaaaaaa* MCS FOR *bbbbbbbb* SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—MCS type
- *bbbbbbbb*—SYSMOD type
- *ccccccc*—SYSMOD ID or blank, if unknown.

**Note:** This message immediately follows message GIM203xx, 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.

**GIM50101v** *aaaaaaaa* IS NOT ALLOWED BECAUSE *bbbbbbbb* WAS NOT SPECIFIED.

**Explanation:**

- *aaaaaaaa*—the first operand
- *bbbbbbbb*—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 Reference*.

Remove or add the desired operands and rerun the job.

**GIM50201v** *aaaaaaaa* PROCESSING FAILED FOR SYSMOD *bbbbbbbb* BECAUSE *cccccccc* WAS NOT SPECIFIED.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbbbb*—SYSMOD ID
- *cccccccc*—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.

**GIM50301v** *aaaaaaaa* WAS NOT PROCESSED BECAUSE IT IS NOT AN FMIDSET NAME.

**Explanation:**

- *aaaaaaaa*—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 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.

**GIM50401v** 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.

**GIM50501v CONNECT PROCESSING FAILED FOR THE *aaaaaaaa* ZONE.**

**Explanation:**

- *aaaaaaaa*—zone name.

**System Action:** Depending on the return code, either command processing stops or SMP/E processing stops.

**Programmer Response:** Fix the error and rerun the job.

**GIM50502v FREE PROCESSING FAILED FOR THE *aaaaaaaa* ZONE.**

**Explanation:**

- *aaaaaaaa*—zone name.

A physical I/O error caused FREE processing to fail for the indicated zone.

**System Action:** Depending on the return code, either command processing stops or SMP/E processing stops.

**Programmer Response:** Fix the error and rerun the job.

**GIM50504v OPEN PROCESSING FAILED FOR THE *aaaaaaaa* ZONE.**

**Explanation:**

- *aaaaaaaa*—zone name.

SMP/E could not open the indicated zone.

**System Action:** If this message specifies the global zone, SMP/E processing stops. Otherwise, command processing stops.

**Programmer Response:** Make sure that you specified the correct SMPCSI data set name for the indicated zone.

- For the global zone, this name may be specified either on the SMPCSI DD statement or on the CSI parameter of the EXEC statement for GIMSMP.
- For target and DLIB zones, 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.

**GIM50505v CLOSE PROCESSING FAILED FOR THE *aaaaaa* ZONE.**

**Explanation:**

- *aaaaaa*—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.

**GIM50506v LMOD BUILD PROCESSING WAS UNSUCCESSFUL BECAUSE *aaaaaa* PROCESSING FAILED FOR DISTRIBUTION ZONE *bbbbbb*.**

**Explanation:**

- *aaaaaa*—OPEN
- *bbbbbb*—zone name.

**System Action:** Command processing stops.

**Programmer Response:** Make sure that 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.

**GIM50507v SMP/E CANNOT PROCESS ZONE *aaaaaaaa* BECAUSE THE SMPCSI DATA SET CONTAINS TOO MANY ZONES.**

**Explanation:**

- *aaaaaaaa*—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 Reference*.

**GIM50508v *aaaaaaaa* PROCESSING FAILED.**

**Explanation:**

- *aaaaaaaa*—CLOSE, FREE, or CONNECT.

**System Action:** Either command processing stops or SMP/E processing stops, depending on the severity of the error.

**Programmer Response:** Fix the error and rerun the job.

**GIM50509v AN I/O ERROR OCCURRED WHEN SMP/E WAS TRYING TO LOCATE A RECORD FOR ZONE *aaaaaaaa*.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM50601v LINK-EDIT PROCESSING FAILED FOR LMOD *aaaaaaaa* IN THE *bbbbbbb* LIBRARY. THE RETURN CODE WAS *cc*. THE FOLLOWING CSECTS HAVE BEEN DELETED:**

**Explanation:**

- *aaaaaaaa*—load module name
- *bbbbbbb*—ddname of the library
- *cc*—return code.

Link-edit processing to remove a module from the indicated load module failed. Message GIM50701 follows this message and identifies the modules and CSECTs that were 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.
2. Reapply the failing SYSMOD.

**GIM50602v** LINK-EDIT PROCESSING WAS SUCCESSFUL FOR LMOD *aaaaaaaa* IN THE *bbbbbbb* LIBRARY. THE RETURN CODE WAS *cc*. THE FOLLOWING CSECTS HAVE BEEN DELETED:

**Explanation:**

- *aaaaaaaa*—load module name
- *bbbbbbb*—ddname of the library
- *cc*—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.

**GIM50701v** CSECT *aaaaaaaa* IN DLIB MODULE *bbbbbbb* IN SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—CSECT name
- *bbbbbbb*—module name
- *ccccccc*—SYSMOD ID.

This message lists the module and CSECTS within that module that were deleted from the load module identified in message GIM50601 or GIM50602. SYSMOD *ccccccc* is the SYSMOD that caused the module to be deleted from the load module.

**System Action:** None.

**Programmer Response:** None.

**GIM50701v** CSECT *aaaaaaaa* IN DLIB MODULE *bbbbbbb*

**Explanation:**

- *aaaaaaaa*—CSECT name
- *bbbbbbb*—module name.

This message lists the modules and CSECTS within that module that were deleted from the load module identified in message GIM50601 or GIM50602.

**System Action:** None.

**Programmer Response:** None.

**GIM50801v** ZONE *aaaaaaaa* WAS NOT USED BECAUSE IT IS NOT DEFINED BY A ZONEINDEX SUBENTRY IN THE GLOBAL ZONE.

**Explanation:**

- *aaaaaaaa*—zone name.

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 FORZONE, ZONES, or ZONESET operand on the REPORT command
- A ZONESET used with the REJECT command
- The RELATED subentry (in a TARGETZONE or DLIBZONE entry) used during APPLY or ACCEPT command processing
- The input zone on the ZONECOPY command
- The originating zone on the ZONEMERGE command.

**System Action:** Command processing fails. Processing continues with the next valid SET or RESETRC command.

**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.

**GIM50802v LOAD MODULE BUILD PROCESSING STOPPED BECAUSE DISTRIBUTION ZONE *aaaaaaaa* IS NOT DEFINED BY A ZONEINDEX SUBENTRY IN THE GLOBAL ZONE.**

**Explanation:**

- *aaaaaaaa*—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:

- GIM67401 or GIM67402, which lists the load modules that failed
- GIM67501, which lists the modules required to complete the load modules
- GIM22601, 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.

**GIM50901v A SET COMMAND IS REQUIRED BEFORE THE *aaaaaaaa* COMMAND.**

**Explanation:**

- *aaaaaaaa*—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 SMPCTL before the command that requires it. This will point SMP/E to the zone that it is going to process. Then rerun the job.

**GIM50902v SET COMMAND PROCESSING FAILED. DATA IN SMPCTL 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 SMPCTL 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.

**GIM51001v THE *aaaaaaaa* SUBENTRY IS NOT IN *bbbbbbbb* ENTRY *ccccccc*.**

**Explanation:**

- *aaaaaaaa*—subentry type
- *bbbbbbbb*—entry type
- *ccccccc*—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.

**GIM51101v SMP/E PROCESSING STOPPED BECAUSE IT COULD NOT DETERMINE WHICH ZONE TO UPDATE FOR UCLIN PROCESSING.**

**Explanation:** Module GIMPUCD 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.

**GIM51201v THE SPECIFIED ENTRY TYPE IS ONLY ALLOWED FOR A *aaaaaa* ZONE.****Explanation:**

- *aaaaaa*—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 Reference* 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.

**GIM51202v 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.

**GIM51301v THE *aaaaaaaa* OPERAND EXCEEDS THE 26-CHARACTER MAXIMUM.****Explanation:**

- *aaaaaaaa*—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:** UCLIN processing stops for the statement in error. Processing continues with the next UCLIN statement.

**Programmer Response:** Fix the UCLIN statement, and rerun the job.

**GIM51401v THERE IS NO *aaaaaaaa* ENTRY FOR *bbbbbbbb*.****Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—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.)

**System Action:** One of the following occurs:

- If the OPTIONS entry is missing, the system action depends on where the name of the OPTIONS entry was specified and what command was being processed.
  - If the OPTIONS entry was specified on the SET command, command processing stops. SMP/E continues to check the SMP\_CNTL output for syntax.
  - If the OPTIONS entry was specified in the zone definition entry, the system action depends on the command being processed.
    - For the LIST, UNLOAD, UCLIN, and zone commands, processing continues using the default values.
    - For the JCLIN command, command processing stops. SMP/E continues to check the SMP\_CNTL output for syntax.
- If another type of entry was missing, command processing stops. SMP/E continues to check the SMP\_CNTL output for syntax.

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

- If command processing failed, use UCLIN to create the missing entry. Then rerun the job.
- Otherwise, 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.

**GIM51501v** *aaaaaaaa bbbbbbbb* WAS MOVED TO A NEW DISTLIB AND DELETED FROM DISTLIB *ccccccccc*.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—entry name
- *ccccccccc*—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.

**GIM51601v** THE *aaaaaaaa* COMMAND WAS NOT PROCESSED BECAUSE THE SET COMMAND SPECIFIED A *bbbbbbbb* ZONE.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbbbb*—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 SMP\_CNTL 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.

**GIM51602v** THE *aaaaaaaa* COMMAND DID NOT PROCESS THE GLOBAL ZONE BECAUSE THE SET COMMAND SPECIFIED A *bbbbbbbb* ZONE.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbbbb*—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 SMP\_CNTL 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.

**GIM51603v** THE *aaaaaaaa* COMMAND WAS NOT PROCESSED BECAUSE OF A CONFLICT IN ZONE TYPES. THE SET COMMAND SPECIFIED A *bbbbbb* ZONE, BUT THE INPUT ZONE IS A *cccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbb*—zone type
- *cccccc*—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 SMP\_CNTL input for syntax.

**Programmer Response:** Specify a valid combination of zones and rerun the job.

**GIM51701v** THE *aaaaaaaa* COMMAND WAS NOT PROCESSED BECAUSE NO SREL WAS SPECIFIED IN THE ZONE DEFINITION ENTRY.

**Explanation:**

- *aaaaaaaa*—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 SMP\_CNTL input for syntax.

**Programmer Response:** Use UCLIN to define the proper SREL values in the zone definition entry. Then rerun the job.

**GIM51702v** THE *aaaaaaaa* COMMAND WAS NOT PROCESSED BECAUSE NO RELATED ZONE WAS SPECIFIED IN THE ZONE DEFINITION ENTRY.

**Explanation:**

- *aaaaaaaa*—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 may fail, depending on the return code. SMP/E continues to check SMP\_CNTL input for syntax.

**Programmer Response:** Use UCLIN to define the proper RELATED value in the zone definition entry. Then rerun the job.

**GIM51703v** THE *aaaaaaaa* COMMAND WAS NOT PROCESSED BECAUSE NO *bbbbbb* ZONE WAS SPECIFIED ON THE *ccccccc* OPERAND.

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbb*—zone type
- *ccccccc*—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.

**GIM51801v** ZONE *aaaaaaaa* IS DEFINED AS A *bbbbbbbb* ZONE IN THE ZONEINDEX SUBENTRY. THIS CONFLICTS WITH THE ZONE TYPE OF *cccccccc*.

**Explanation:**

- *aaaaaaaa*—zone name
- *bbbbbbbb*—zone type defined in the ZONEINDEX subentry
- *cccccccc*—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.

**GIM51901v** A *++aaaaaaaa* MCS IS NOT ALLOWED WITHIN A SYSMOD.

**Explanation:**

- *aaaaaaaa*—MCS type (++RELEASE or ++NULL.)

During RECEIVE processing, SMP/E found the indicated MCS within a SYSMOD. This MCS is only allowed in the SMPHOLD data set.

**System Action:** SMP/E does not receive the SYSMOD.

**Programmer Response:** Remove the indicated MCS from the SYSMOD and, if necessary, add it to the SMPHOLD data set. Then run RECEIVE again.

**GIM52001v** THE SYSMOD SPECIFIED ON THE *++HOLD* MCS IS NOT THE SYSMOD THAT CONTAINS THE *++HOLD*.

**Explanation:** 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 only use a *++HOLD* within a SYSMOD to hold the containing SYSMOD.

**System Action:** SMP/E does not receive the SYSMOD.

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

- If you meant to hold the containing SYSMOD, fix the SYSMOD specified on the *++HOLD* MCS.
- If you meant to hold the SYSMOD specified on the *++HOLD* MCS, remove the *++HOLD* MCS from the SYSMOD being received and process it through the SMPHOLD data set.

Then rerun the RECEIVE job.

**GIM52101v** THE *++aaaaaaaa* MCS SPECIFIES MORE THAN ONE HOLD CATEGORY. THIS IS NOT ALLOWED.

**Explanation:**

- *aaaaaaaa*—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 the SMPHOLD data set, MCS processing stops.

**Programmer Response:** Fix the indicated MCS and rerun the command.

**GIM52301v** THE *aaaaaa* COMMAND SPECIFIES OPERANDS THAT ARE NOT ALLOWED FOR THE ZONE SPECIFIED ON THE SET COMMAND.

**Explanation:**

- *aaaaaa*—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 chart 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 Reference*.

Operands Not Allowed for Global Zone:	Operands Not Allowed for Target Zone:	Operands Not Allowed for Distribution Zone:
ASSEM	FMSET,FMIDSET	FMSET,FMIDSET
LMOD	OPTIONS	OPTIONS
MAC	UTILITY	UTILITY
MOD	DZONE,DLIBZONE	TZONE,TARGETZONE
SRC	GZONE,GLOBALZONE	GZONE,GLOBALZONE
DLIB	ZONESET	RES,RESTORE
RES,RESTORE		ZONESET
SUP		
NOSUP		
DEL,DELETE		
BYP,BYPASS		
TZONE,TARGETZONE		
DZONE,DLIBZONE		
BACKUP		
Data element entries		

**GIM52401v** UCLIN PROCESSING IS COMPLETE FOR *aaaaaaaa* ZONE *bbbbbbbb*.

**Explanation:**

- *aaaaaaaa*—zone type
- *bbbbbbbb*—zone name.

UCLIN finished processing for the indicated zone.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM52402v** UCLIN PROCESSING IS COMPLETE FOR THE GLOBAL ZONE.

**Explanation:** UCLIN finished processing for the global zone.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM52501v A PERIOD IS MISSING AFTER THE LAST aaaaaaa COMMAND. ONE IS ASSUMED.****Explanation:**

- aaaaaaa—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.

**GIM52601v 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.

**GIM52602v 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.

**GIM52701v THE DSNAME EXCEEDS THE 44-CHARACTER MAXIMUM.**

**Explanation:** An operand on a command or UCL statement specifies a data set name that has more than 44 characters.

**System Action:** Processing stops for the command or UCL statement in error. Processing continues with the next command or UCL statement.

**Programmer Response:** Fix the operand, then rerun the job.

**GIM52702v 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.

**GIM52801v 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, UCL statement, or EXEC statement did not meet this requirement.

**System Action:** One of the following will occur:

- If SMP/E issues this message during command processing, processing stops for the command in error. Processing continues with the next command.

- If SMP/E issues this message during UCLIN processing, processing stops for the statement in error. Processing continues with the next UCL statement.
- If SMP/E issues this message during SMP/E initialization, SMP/E processing stops.

**Programmer Response:** Fix the command, UCL statement, or EXEC parameter. Then rerun the job.

**GIM52901v 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.

**GIM53001v THE CONVERT COMMAND MUST SPECIFY A DATA SET OPERAND.**

**Explanation:** None of the data set operands was specified on the CONVERT command. Therefore, SMP/E does not know which data sets you want to convert. These are the data set operands that may be specified:

- SMP/E data sets:
  - OLDCSI and NEWCSI or
  - SCDS
- SMP4 data sets:
  - CDS, CRQ, SCDS or
  - ACDS, ACRQ or
  - PTS.

**System Action:** CONVERT processing stops.

**Programmer Response:** Add the appropriate data set operands to the CONVERT command and rerun the job.

**GIM53101v aaaa bbbbbb WAS NOT CONVERTED BECAUSE THE SET COMMAND SPECIFIES THE GLOBAL ZONE.**

**Explanation:**

- *aaaa*—type of library
- *bbbbbb*—ddname of the library.

The indicated data set was not processed for one of the following reasons:

- ACDS, ACRQ, CDS, or CRQ—The data set cannot be converted into a global zone.
- SCDS—The data set is not associated with the global zone.

**System Action:** CONVERT processing stops.

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

- Specify an SMP4 data set that can be converted into the global zone.
- Change the SET command to specify the correct zone type for the data set being converted.

The following table shows the types of data sets and the types of zones that must be specified on the SET command in order to convert them.

Zone Type	Data Set Type					
	ACDS	ACRQ	CDS	CRQ	PTS	SCDS
DLIB	Yes	Yes				
Global					Yes	
Target			Yes	Yes		Yes

**GIM53102v** *aaaa bbbbbb* WAS NOT CONVERTED BECAUSE THE SET COMMAND SPECIFIES A *cccccc* ZONE (*ddddddd*).

**Explanation:**

- *aaaa*—type of library
- *bbbbbb*—ddname of the library
- *cccccc*—TARGET or DLIB
- *ddddddd*—zone name.

The indicated data set was not processed for one of the following reasons:

- ACDS, ACRQ, CDS, CRQ, or PTS—The data set cannot be converted into the specified type of zone.
- SCDS—The data set is not associated with the distribution zone.

**System Action:** CONVERT processing stops.

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

- Specify an SMP4 data set that can be converted into the zone type specified on the SET command.
- Change the SET command to specify the correct zone type for the data set being converted.

The following table shows the types of data sets and the types of zones that must be specified on the SET command in order to convert them.

Zone Type	Data Set Type					
	ACDS	ACRQ	CDS	CRQ	PTS	SCDS
DLIB	Yes	Yes				
Global					Yes	
Target			Yes	Yes		Yes

**GIM53103v** THE CSI WAS NOT CONVERTED BECAUSE THE SET COMMAND SPECIFIES A *aaaaaa* ZONE (*bbbbbbb*).

**Explanation:**

- *aaaaaa*—TARGET or DLIB
- *bbbbbbb*—zone name.

The CSI data set was not processed because the SET command did not specify the global zone.

**System Action:** CONVERT processing stops.

**Programmer Response:** Change the SET command to specify the global zone for the CSI data set being converted.

**GIM53201v** A SPECIFIED OPERAND IS NOT ALLOWED FOR THE GLOBAL ZONE.

**Explanation:** One of the operands specified is not allowed for the global zone.

**System Action:** Processing for the UCL statement stops. Processing continues with the next statement.

**Programmer Response:** Check *SMP/E Reference* to determine which operands 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.

**GIM53401v** CONVERT PROCESSING HAS STARTED FOR *aaaaaaaa* ENTRIES IN THE *bbbbbbb cccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type or blank
- *bbbbbbb*—ddname of the library
- *cccc*—type of SMP4 library.

CONVERT processing started for the indicated type of entries in the specified SMP4 data set.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM53402v** CONVERT PROCESSING IS COMPLETE FOR *aaaaaaaa* ENTRIES IN THE *bbbbbbb cccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type or blank
- *bbbbbbb*—ddname of the library
- *cccc*—type of SMP4 library.

CONVERT processing is complete for the indicated type of entries in the specified SMP4 data set.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM53501v** *aaaaaa bbbbbbbb* ENTRIES WERE CONVERTED FROM THE *ccccccc dddd* LIBRARY.

**Explanation:**

- *aaaaaa*—the number of entries that were converted
- *bbbbbbb*—entry type or blank
- *ccccccc*—ddname of the library
- *dddd*—type of SMP4 library.

This message shows the number of entries that were converted.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM53601v** THERE ARE NO *aaaaaaaa* ENTRIES IN THE *bbbbbbb cccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbb*—ddname of the library
- *cccc*—type of SMP4 library.

There were no entries of the specified type in the SMP4 library being converted.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM53701v** *aaaa bbbbbbb* WAS SUCCESSFULLY CONVERTED TO *cccccc* ZONE *ddddddd*.

**Explanation:**

- *aaaa*—type of SMP4 library
- *bbbbbbb*—ddname of the library
- *cccccc*—TARGET or DLIB
- *ddddddd*—zone name.

The specified SMP4 library was successfully converted.

**System Action:** Processing continues with the next command.

**Programmer Response:** None.

**GIM53702v** *aaaa bbbbbb* WAS NOT CONVERTED TO *cccccc* ZONE *ddddddd*.

**Explanation:**

- *aaaa*—type of SMP4 library
- *bbbbbbb*—ddname of the library
- *cccccc*—TARGET or DLIB
- *ddddddd*—zone name.

CONVERT processing failed for the specified SMP4 library.

**System Action:** Command processing stops. The return code determines whether SMP/E will process the next command.

**Programmer Response:** Look at the previous error messages to find the cause of the failure. Fix the error and rerun the job from the point of failure.

**GIM53703v** CONVERSION OF *aaaa bbbbbb* TO *cccccc* ZONE *ddddddd* WAS NOT ATTEMPTED BECAUSE OF A PREVIOUS ERROR.

**Explanation:**

- *aaaa*—type of SMP4 library
- *bbbbbbb*—ddname of the library
- *cccccc*—zone type
- *ddddddd*—zone name.

SMP/E did not attempt to convert the specified SMP4 library because of a previous error for the same CONVERT command. For example, assume you are using a single CONVERT command to convert the ACDS and ACRQ. If an error occurs while SMP/E is converting the ACDS, then SMP/E will not even attempt to convert the ACRQ.

**System Action:** Command processing stops. The return code determines whether SMP/E will process the next command.

**Programmer Response:** Look at the previous error messages to find the cause of the failure. Fix the error and rerun the job from the point of failure.

**GIM53704v** PTS *aaaaaaa* WAS SUCCESSFULLY CONVERTED TO THE GLOBAL ZONE.

**Explanation:**

- *aaaaaaa*—ddname of the library.

The SMP4 PTS data set was successfully converted.

**System Action:** Processing continues with the next command.

**Programmer Response:** None.

**GIM53705v** PTS *aaaaaaa* WAS NOT CONVERTED TO THE GLOBAL ZONE.

**Explanation:**

- *aaaaaaa*—ddname of the library.

CONVERT processing failed for the SMP4 PTS data set.

**System Action:** Command processing stops. The return code determines whether SMP/E will run the next command.

**Programmer Response:** Look at the previous error messages to find the cause of the failure. Fix the error and rerun the job from the point of failure.

**GIM53801v** CONVERT PROCESSING FOR *aaaaaaa bbbbbb* IS INCOMPLETE BECAUSE OF A PREVIOUS ERROR.

**Explanation:**

- *aaaaaaa*—entry type or blank
- *bbbbbbb*—entry name.

A previous error prevented SMP/E from converting the indicated entry. A previous message identifies the error.



**System Action:** Conversion continues with the next entry from the specified data set.

**Programmer Response:** Do the following:

1. Look at the previous messages to find the cause of the error.
2. Do one of the following:
  - Rerun CONVERT to correct the error.
  - Use SMP4 to list the SMP4 data set entry, then use SMP/E to list the SMP/E entry. Use UCLIN to fix the entry that is in error.

**GIM53901v 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.

**GIM54001v 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 the SMPHOLD data set and rerun the RECEIVE command.

**GIM54002v 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 the SMPHOLD data set and rerun the RECEIVE command.

**GIM54101v DYNAMIC ALLOCATION FOR *aaaaaaa* WAS SUCCESSFUL – *bbb...bbb***

**Explanation:**

- *aaaaaaa*—ddname of the library
- *bbb...bbb*—allocation parameters.

SMP/E successfully allocated the indicated data set.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM54201v DYNAMIC ALLOCATION FOR *aaaaaaa* WAS SUCCESSFUL. THE FOLLOWING LIBRARIES WERE CONCATENATED: *bbbbbbb,...,bbbbbbb*.**

**Explanation:**

- *aaaaaaa*—ddname of the library that was allocated
- *bbbbbbb*—ddname 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.

**GIM54301v DYNAMIC DEALLOCATION WAS SUCCESSFUL FOR *aaaaaaa*.****Explanation:**

- *aaaaaaa*—ddname of the library.

SMP/E successfully freed the indicated library.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM54401v DYNAMIC DEALLOCATION FAILED FOR *aaaaaaa*.****Explanation:**

- *aaaaaaa*—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.

**GIM54501v ALLOCATION FAILED FOR *aaaaaaa* BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN THE GLOBAL ZONE.****Explanation:**

- *aaaaaaa*—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:** System action depends on the command being processed and the data set that needed to be allocated. 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.

**GIM54502v ALLOCATION FAILED FOR *aaaaaaa* BECAUSE THERE IS NO DD STATEMENT IN THE JCL AND NO DDDEF ENTRY IN *bbbbbb* ZONE *ccccccc*.****Explanation:**

- *aaaaaaa*—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.

- *bbbbbb*—TARGET or DLIB.
- *ccccccc*—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:** System action depends on the command being processed and the data set that needed to be allocated. 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.

**GIM54601v DYNAMIC ALLOCATION FAILED FOR *aaaaaaa* BECAUSE *bbbbbbb* IS PART OF THE *aaaaaaa* CONCATENATION BUT WAS NOT ALLOCATED.**

**Explanation:**

- *aaaaaaa*—ddname of the library being allocated
- *bbbbbbb*—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. Message GIM545xx states the reason why the concatenated library failed.

**System Action:** System action depends on the command being processed and the library that SMP/E could not allocate. A subsequent message identifies the system action.

**Programmer Response:** Fix the error for the *bbbbbbb* library and rerun the job.

**GIM54701v DYNAMIC ALLOCATION FAILED FOR *aaaaaaa* - *bbbbbbbbbbb*.**

**Explanation:**

- *aaaaaaa*—ddname of the library
- *bbbbbbbbbbb*—the text of the error message from the DAIRFAIL system service that shows the reason for the allocation failure. If the message text is greater than 180 characters, SMP/E prints message GIM54701v again with the second part of the error text.

When SMP/E was trying to allocate the indicated library, an error occurred in SVC99 (the dynamic allocation SVC).

**System Action:** System action depends on the command being processed and the library that SMP/E could not allocate. A subsequent message identifies the system action.

**Programmer Response:** Fix the error and rerun the job.

**GIM54801v ALLOCATION WAS NOT ATTEMPTED FOR *aaaaaaa* BECAUSE A PREVIOUS ALLOCATION ATTEMPT IN THIS ZONE FAILED.**

**Explanation:**

- *aaaaaaa*—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:** System action depends on the command being processed and the library that SMP/E could not allocate. 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.

**GIM54901v *aaaaaaa* WAS ALLOCATED BY THE USER AS SYSIN/SYSOUT.**

**Explanation:**

- *aaaaaaa*—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.

**GIM54902v** *aaaaaaa* WAS ALLOCATED BY THE USER AS A DUMMY DATA SET.**Explanation:**

- *aaaaaaa*—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:**

- If enqueue processing is required, command processing stops.
- Otherwise, processing continues.

**Programmer Response:**

- If this message was issued because an enqueue was attempted for a dummy data set, specify a valid data set name for the data set and rerun the job.
- Otherwise, 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.

**Note:** If enqueue processing was attempted for a dummy data set, this message is also written to the SMPOUT data set.

**GIM54903v** *aaaaaaa* WAS ALLOCATED BY THE USER — *bbb...bbb***Explanation:**

- *aaaaaaa*—ddname of the library
- *bbb...bbb*—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.

**GIM55101v** ZONE *aaaaaaa* WAS SUCCESSFULLY DELETED.**Explanation:**

- *aaaaaaa*—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.

**GIM55201v** THE ZONEINDEX SUBENTRY FOR *aaaaaaa* WAS DELETED.**Explanation:**

- *aaaaaaa*—zone name.

SMP/E deleted the indicated ZONEINDEX subentry.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM55202v THE ZONEINDEX SUBENTRY FOR *aaaaaaaa* WAS NOT DELETED. SMP/E COULD NOT OBTAIN ACCESS TO THE GLOBAL ZONE.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM55301v THERE IS NO DDDEF OR DD STATEMENT FOR THE PARMLIB DATA SET. SMP/E USED THE DEFAULT METHOD OF DETERMINING OPCODES.**

**Explanation:** During JCLIN processing, SMP/E could not find a DDDEF entry or DD statement for the PARMLIB 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 PARMLIB, SMP/E continues processing, using the following method to determine whether a character string is an OPCODE or a macro:

- If the character string is 6 characters or greater, SMP/E treats the string as a macro and creates a MAC entry.
- If the character string is 5 characters or less, SMP/E treats the string as an OPCODE and does not create a MAC entry.

For each character string, SMP/E issues message GIM555xx, showing the following:

- The string that SMP/E found
- The name of the assembly where SMP/E found the string
- An indication of how SMP/E treated the character string.

**System Action:** Processing continues.

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

- Rerun the job with a PARMLIB DD statement.
- Use message GIM555xx to determine whether the default method caused any errors. If so, use UCLIN to create the required entry.

The following is an example of how to use UCLIN to add a MAC entry and to connect it to the proper ASSEM entry:

```
UCLIN.
ADD MAC(aaaaaaaa)
...
GENASM(bbbbbbbb).
ENDUCL.
```

where:

- *aaaaaaaa* is the name of the macro from GIM555xx.
- *bbbbbbbb* is the name of the ASSEM entry from GIM555xx.

**GIM55401v MEMBER GIMOPCDE IS NOT IN THE PARMLIB DATA SET. SMP/E USED THE DEFAULT METHOD OF DETERMINING OPCODES.**

**Explanation:** During JCLIN processing of an assembler step, SMP/E needed to determine whether a character string was a macro or an OPCODE. To do this, SMP/E uses the PARMLIB member specified on the JCLIN command or the ++JCLIN MCS. No OPCODE member was specified, so SMP/E assumed that GIMOPCDE, the member supplied by SMP/E, should be used. However, GIMOPCDE was not in the PARMLIB data set. As a result, SMP/E used the default method of determining whether a character string is a macro or an OPCODE:

- If the character string is 6 characters or greater, SMP/E treats the string as a macro and creates a MAC entry.
- If the character string is 5 characters or less, SMP/E treats the string as an OPCODE and does not create a MAC entry.

For each character string, SMP/E issues message GIM555xx, showing the following:

- The string that SMP/E found
- The name of the assembly where SMP/E found the string
- An indication of how SMP/E treated the character string.

**System Action:** Processing continues.

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

- Rerun the job with a PARMLIB DD statement.
- Use message GIM555xx to determine whether the default method caused any errors. If so, use UCLIN to create the required entry.

The following is an example of how to use UCLIN to add a MAC entry and to connect it to the proper ASSEM entry:

```
UCLIN.
ADD MAC(aaaaaaaa)
...
      GENASMbbbbbbb).
ENDUCL.
```

where:

- *aaaaaaaa* is the name of the macro from GIM555xx.
- *bbbbbbb* is the name of the ASSEM entry from GIM555xx.

**GIM55402v MEMBER aaaaaaaaa IS NOT IN THE PARMLIB DATA SET. SMP/E USED ONLY THE OPCODES IN PARMLIB MEMBER bbbbbbbb.**

**Explanation:**

- *aaaaaaaa*—name of a PARMLIB member
- *bbbbbbb*—name of another PARMLIB member.

During JCLIN processing of an assembler step, SMP/E needed to determine whether a character string was a macro or an OPCODE. To do this, SMP/E uses the PARMLIB member specified on the JCLIN command or the ++JCLIN MCS. An OPCODE member was specified, so SMP/E assumed that the specified member should be concatenated with GIMOPCDE, the member supplied by SMP/E. However, one of these members was not in the PARMLIB data set. As a result, SMP/E used only the OPCODEs that were specified in the member that was found.

**System Action:** Processing continues.

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

- Rerun the job with a PARMLIB DD statement.
- Use message GIM555xx to determine whether the default method caused any errors. If so, use UCLIN to create the required entry.

The following is an example of how to use UCLIN to add a MAC entry and to connect it to the proper ASSEM entry:

```
UCLIN.
ADD MAC(aaaaaaaa)
...
      GENASMbbbbbbb).
ENDUCL.
```

where:

- *aaaaaaaa* is the name of the macro from GIM555xx.
- *bbbbbbb* is the name of the ASSEM entry from GIM555xx.

**GIM55501v** *aaaaaaaa* FOUND IN ASSEMBLY *bbbbbbbb* INPUT IS CONSIDERED A MACRO BECAUSE IT IS GREATER THAN 5 CHARACTERS.

**Explanation:**

- *aaaaaaaa*—character string
- *bbbbbbbb*—name of the assembly.

During JCLIN processing of assembly statements, SMP/E found the indicated character string and determined whether to treat the string as a macro name or an OP CODE.

**System Action:** Processing continues.

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

- Fix the error shown in GIM553xx or GIM554xx and rerun the job.
- Determine whether the default method caused any errors. If so, use UCLIN to create the required entry.

The following is an example of how to use UCLIN to add a MAC entry and to connect it to the proper ASSEM entry:

```
UCLIN.
ADD MAC(aaaaaaaa)
...
GENASM(bbbbbbbb).
ENDUCL.
```

where:

- *aaaaaaaa* is the name of the macro.
- *bbbbbbbb* is the name of the ASSEM entry.

**GIM55502v** *aaaaaaaa* FOUND IN ASSEMBLY *bbbbbbbb* INPUT IS CONSIDERED AN OP CODE BECAUSE IT IS LESS THAN 6 CHARACTERS.

**Explanation:**

- *aaaaaaaa*—character string
- *bbbbbbbb*—name of the assembly.

**System Action:** Processing continues.

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

- Fix the error shown in GIM553xx or GIM554xx and rerun the job.
- Determine whether the default method caused any errors. If so, use UCLIN to create the required entry.

The following is an example of how to use UCLIN to add a MAC entry and to connect it to the proper ASSEM entry:

```
UCLIN.
ADD MAC(aaaaaaaa)
...
GENASM(bbbbbbbb).
ENDUCL.
```

where:

- *aaaaaaaa* is the name of the macro.
- *bbbbbbbb* is the name of the ASSEM entry.

**GIM55503v** *aaaaaaaa* FOUND IN ASSEMBLY *bbbbbbbb* INPUT IS CONSIDERED A MACRO BECAUSE IT WAS NOT FOUND IN PARMLIB MEMBER *cccccccc*.

**System Action:** Processing continues.

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

- Fix the error shown in GIM553xx or GIM554xx and rerun the job.
- Determine whether the default method caused any errors. If so, use UCLIN to create the required entry.

The following is an example of how to use UCLIN to add a MAC entry and to connect it to the proper ASSEM entry:

```
UCLIN.
ADD MAC(aaaaaaaa)
...
GENASM(bbbbbbbb).
ENDUCL.
```

where:

- *aaaaaaaa* is the name of the macro.
- *bbbbbbbb* is the name of the ASSEM entry.

**GIM55601v** *aaaaaaaa* WAS NOT DEFINED. PROCESSING CONTINUES USING *bbbbbbb* INSTEAD.

**Explanation:**

- *aaaaaaaa*—ddname of the missing library
- *bbbbbbb*—SYSPRINT.

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 DD statement to the JCL used to run SMP/E.
- Add a DDDEF entry to the applicable zones.

**GIM55701v** AN ERROR OCCURRED DURING AN ATTEMPT TO READ *aaaaaaaa* *bbbbbbbb*. CONVERT PROCESSING CONTINUES.

**Explanation:**

- *aaaaaaaa*—entry type or blank
- *bbbbbbbb*—entry name.

During CONVERT processing, SMP/E encountered an error while reading an entry from an SMP4 data set.

**System Action:** Processing fails for the indicated entry but continues with the next entry.

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

- If the number of these messages is excessive, fix the problems shown in previous messages and rerun the CONVERT job.
- If only a few of these messages appear, use the SMP4 UNLOAD command to create UCLIN. Use that UCLIN to build the entry in the SMP/E data set.

**GIM55702v** AN ERROR OCCURRED DURING AN ATTEMPT TO STORE *aaaaaaaa* *bbbbbbbb*. CONVERT PROCESSING FAILED.

**Explanation:**

- *aaaaaaaa*—entry type or blank
- *bbbbbbbb*—entry name.

During CONVERT processing, SMP/E encountered an error during an attempt to store the converted entry on the SMP/E data set.

**System Action:** Command processing stops.

**Programmer Response:** Look at the previous messages to find the error and fix the problem. Then rerun the CONVERT job.



**GIM55801v** THE *aaaaaaaa* COMMAND SPECIFIED THE ZONE NAME *bbbbbbb*. THIS DOES NOT MATCH THE ZONE SPECIFIED ON THE PREVIOUS SET COMMAND.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbbb*—zone name.

**System Action:** Command processing stops.

**Programmer Response:** Check which zone name is correct. Fix the error and rerun the job.

**GIM55802v** THE *aaaaaaaa* COMMAND SPECIFIED A *bbbbbb* ZONE. THIS CONFLICTS WITH THE ZONE TYPE OF THE ZONE SPECIFIED ON THE PREVIOUS SET COMMAND.

**Explanation:**

- *aaaaaaaa*—an SMP/E command
- *bbbbbb*—TARGET or DLIB.

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.

**GIM55901v** 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.

**GIM56001v** 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.

**Note:** Each operating system sets its own limits for how many partitioned data sets can actually be concatenated. Therefore, 123 may not be the maximum for your operating system.

**System Action:** Processing stops for the statement in error. Processing continues with the next statement.

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

- If you specified more than 123 DDDEF names, specify 123 or less and rerun the statement.
- If you specified less than 123 names and you got this message, the operating system you are using probably has a lower limit for the number of data sets that may be concatenated. See the data management manual for your operating system to determine the maximum number of data sets you can concatenate. Then fix the statement and rerun the job.

**GIM56101v** A ZONE SPECIFIED ON THE PURGE OPERAND MUST BE A DLIB ZONE. *aaaaaaaa* IS A TARGET ZONE.

A ZONE SPECIFIED ON THE TARGETZONE OPERAND MUST BE A TARGET ZONE. *aaaaaaaa* IS A DLIB ZONE.

**Explanation:**

- *aaaaaaaa*—zone name.

The indicated zone is not the correct type for the operand on which it was specified.

**System Action:** Command processing stops.

**Programmer Response:** Remove the zone from the indicated operand. If it is required for processing, specify it on the appropriate operand. Then rerun the job.

**GIM56301v RELATED ZONE *aaaaaaaa* MUST BE A *bbbbbb* ZONE.**

**Explanation:**

- *aaaaaaaa*—zone name
- *bbbbbb*—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.

**GIM56401v SYSMOD *aaaaaaaa* HAS ALREADY BEEN SPECIFIED ON EITHER THE SELECT OR EXCLUDE OPERAND.**

**Explanation:**

- *aaaaaaaa*—SYSMOD ID.

The indicated SYSMOD 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 only once. Then rerun the job.

**GIM56402v SYSMOD *aaaaaaaa* WAS SPECIFIED MORE THAN ONCE ON A ++ASSIGN MCS. THIS IS NOT ALLOWED.**

**Explanation:**

- *aaaaaaaa*—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).

**GIM56501v THE *aaaaaaaa* SUBENTRY WAS ADDED INSTEAD OF REPLACED BECAUSE IT DID NOT EXIST.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM56601v ENTRY *aaaaaaaa* WAS NOT ADDED BECAUSE IT ALREADY EXISTS.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM56701v** THERE IS AN ERROR IN A ++*aaaaaaaa* MCS FOR SYSMOD *bbbbbbb* IN SMPHOLD.

**Explanation:**

- *aaaaaaaa*—MCS type (HOLD or RELEASE)
- *bbbbbbb*—SYSMOD ID.

Message GIM203xx 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.

**GIM56702v** THERE IS AN ERROR IN A ++*aaaaaaaa* MCS IN SMPHOLD. THE SYSMOD IS UNKNOWN.

**Explanation:**

- *aaaaaaaa*—MCS type (HOLD or RELEASE.)

Message GIM203xx 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.

**GIM56703v** THERE IS AN ERROR IN A ++NULL MCS IN SMPHOLD.

**Explanation:** Message GIM203xx 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.

**GIM56704v** THERE IS AN ERROR IN AN UNKNOWN MCS IN SMPHOLD.

**Explanation:** Message GIM203xx 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.

**GIM56801v** A SUBENTRY VALUE WAS SPECIFIED MORE THAN ONCE FOR THE *aaaaaaaa* SUBENTRY. THIS IS NOT ALLOWED.

**Explanation:**

- *aaaaaaaa*—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.

**GIM56901v** 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.

**GIM57001v 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.

**GIM57101v 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.

**GIM57201v 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.

**GIM57301v THE UCL STATEMENT FOR THE *aaaaaaaa* ENTRY MUST CONTAIN AT LEAST ONE SUBENTRY OPERAND.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM57401v *aaaaaaaa* IS NOT ALLOWED AS THE *bbbbbbb* OPERAND VALUE ON THE CONVERT COMMAND.**

**Explanation:**

- *aaaaaaaa*—ddname
- *bbbb*—operand.

This message may be issued when you are trying to convert either an SMPPTS or an SMPSCDS data set.

- SMPPTS must be the ddname of the SMP/E SMPPTS data set where SMP/E will write the converted data. Therefore, it cannot be specified on the PTS operand as the ddname of the SMP4 PTS data set to be converted.
- SMPSCDS must be the ddname of the SMP/E SMPSCDS data set where SMP/E will write the converted data. Therefore, it cannot be specified on the SCDS operand as the ddname of the SCDS data set to be converted.

**System Action:** CONVERT processing stops.

**Programmer Response:** Do the following:

1. Change the ddname for the indicated data set in the CONVERT command.
2. Specify that ddname in either the DDDEF entry or the DD statement in the JCL used to define the indicated data set.
3. Rerun the CONVERT command.

**GIM57501v 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 Reference*.

**GIM57601v DYNAMIC ALLOCATION FAILED FOR aaaaaaa BECAUSE IT CONCATENATES THE bbbbbbb DDDEF ENTRY, WHICH CONCATENATES OTHER DATA SETS. THIS IS NOT ALLOWED.**

**Explanation:**

- aaaaaaa—ddname of the library being allocated
- bbbbbbb—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. The rerun the job.

**GIM57701v AN I/O ERROR OCCURRED DURING aaaaaaa PROCESSING FOR THE bbbbbbb DATA SET.**

**Explanation:**

- aaaaaaa—type of processing being done (READ, WRITE, STOW, BLDL, OPEN, CLOSE)
- bbbbbbb—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. Fix the error and rerun the SMP/E job.

**GIM57702v AN I/O ERROR OCCURRED DURING aaaaaaa PROCESSING FOR bbbbbbb MEMBER ccccccc IN THE ddddddd DATA SET.**

**Explanation:**

- aaaaaaa—type of processing being done (READ, WRITE, STOW, BLDL, OPEN, CLOSE)
- bbbbbbb—member type
- ccccccc—member name
- ddddddd—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.

**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.

**GIM57703v AN I/O ERROR OCCURRED DURING *aaaaaaaa* PROCESSING FOR *bbbbbbbb* MEMBER *cccccccc* IN THE *ddddddd* DATA SET. APPROXIMATE TTR=*eee*.**

**Explanation:**

- *aaaaaaaa*—type of processing being done (READ, WRITE, STOW, BLDL, OPEN, CLOSE)
- *bbbbbbbb*—member type
- *cccccccc*—member name
- *ddddddd*—ddname of the data set
- *eee*—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 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.

**GIM57801v THE *aaaaaaaa* OPERAND IS NOT ALLOWED ON THE *bbbbbbbb* COMMAND.**

**Explanation:**

- *aaaaaaaa*—operand
- *bbbbbbbb*—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.

**GIM57901v SMPCSI DATA SET *aaaaaaaaaa* IS OUT OF SPACE.**

**Explanation:**

- *aaaaaaaaaa*—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 to:
  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.

**GIM58101v** *aaaaaaaa* IS NOT ALLOWED FOR THE *bbbbbbbb* UTILITY BECAUSE IT IS NOT DEFINED IN GIMUTTBL.

**Explanation:**

- *aaaaaaaa*—program name
- *bbbbbbbb*—utility type.

**System Action:** Command 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 about GIMUTTBL, see *SMP/E Reference*.

**GIM58201v** ZONE *aaaaaaaa* WAS NOT RENAMED BECAUSE THERE IS NO ZONEINDEX SUBENTRY FOR IT.

**Explanation:**

- *aaaaaaaa*—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.

**GIM58301v** *aaaaaaaa* WAS NOT RENAMED BECAUSE THERE IS ALREADY A ZONEINDEX SUBENTRY FOR THE NEW NAME *bbbbbbbb*.

**Explanation:**

- *aaaaaaaa*—current zone name
- *bbbbbbbb*—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.

**GIM58401v** *aaaaaaaa* WAS NOT RENAMED BECAUSE THE NEW ZONE NAME *bbbbbbbb* ALREADY EXISTS IN SMPCSI DATA SET *cccccccc*.

**Explanation:**

- *aaaaaaaa*—name of the zone being renamed
- *bbbbbbbb*—new zone name
- *cccccccc*—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.

**GIM58501v ZONE aaaaaaaa CANNOT BE RENAMED TO bbbbbbbb.**

**Explanation:**

- aaaaaaaa—current zone name
- bbbbbbbb—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.

**GIM58601v ZONE aaaaaaaa WAS RENAMED TO bbbbbbbb.**

**Explanation:**

- aaaaaaaa—old zone name
- bbbbbbbb—new zone name.

The indicated zone was successfully renamed.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM58701v ZONE aaaaaaaa WAS RENAMED TO bbbbbbbb IN SMPCSI DATA SET ccccccccc.**

**Explanation:**

- aaaaaaaa—old zone name
- bbbbbbbb—new zone name
- ccccccccc—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.

**GIM58801v THERE IS A ZONEINDEX SUBENTRY FOR aaaaaaaa BUT THE ZONE DOES NOT EXIST IN THE SMPCSI DATA SET.**

**Explanation:**

- aaaaaaaa—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.



**GIM59001v** ZONE *aaaaaaaa* WAS NOT RENAMED BECAUSE THE NEW ZONE NAME AND THE OLD ZONE NAME ARE THE SAME.

**Explanation:**

- *aaaaaaaa*—zone name.

**System Action:** ZONERENAME processing stops.

**Programmer Response:** Specify a different value for the new name of the zone and rerun the job.

**GIM59201v** 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.

**GIM59202v** "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.

**GIM59203v** "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.

**GIM59301v** THE TOTYPE(TARGET) OPERAND WAS NOT PROCESSED BECAUSE *aaaaaaaa* IS ALREADY DEFINED AS A TARGET ZONE IN THE ZONEINDEX SUBENTRY.

**Explanation:**

- *aaaaaaaa*—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.

**GIM59302v** THE TOTYPE(TARGET) OPERAND WAS NOT PROCESSED BECAUSE *aaaaaaaa* IS ALREADY DEFINED BY A TARGET ZONE ENTRY.

**Explanation:**

- *aaaaaaaa*—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.

**GIM59401v 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.

**GIM59501v OPTIONS SUBENTRY *aaaaaaaa* WAS ADDED.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM59502v THE OPTIONS SUBENTRY WAS CHANGED TO *aaaaaaaa*.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM59503v RELATED SUBENTRY *aaaaaaaa* WAS ADDED.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM59504v THE RELATED SUBENTRY WAS CHANGED TO aaaaaaaa.**

**Explanation:**

- aaaaaaaa—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.

**GIM59505v A ZONE DEFINITION ENTRY WAS ADDED FOR aaaaaaaa.**

**Explanation:**

- aaaaaaaa—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.

**GIM59601v ENQ WAS INITIATED FOR EXCLUSIVE USE OF aaaaaaaaaa FOR bbbbbbbb PROCESSING.**

**Explanation:**

- aaaaaaaaaa—name of the SMPCSI data set or SMPPTS data set
- bbbbbbbb—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.

**GIM59602v ENQ WAS SUCCESSFUL FOR EXCLUSIVE USE OF aaaaaaaaaa FOR bbbbbbbb PROCESSING.**

**Explanation:**

- aaaaaaaaaa—name of the SMPCSI data set or SMPPTS data set
- bbbbbbbb—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.

**GIM59603v ENQ WAS SUCCESSFUL FOR SHARED USE OF aaaaaaaaaa FOR bbbbbbbb PROCESSING.**

**Explanation:**

- aaaaaaaaaa—name of the SMPCSI data set or SMPPTS data set
- bbbbbbbb—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.

**GIM59604v ENQ FAILED FOR EXCLUSIVE USE OF aaaaaaaaaa FOR bbbbbbbb PROCESSING.**

**Explanation:**

- aaaaaaaaaa—name of the SMPCSI data set or SMPPTS data set
- bbbbbbbb—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.

**GIM59605v ENQ FAILED FOR SHARED USE OF *aaaaaaaa* FOR *bbbbbbbb* PROCESSING.**

**Explanation:**

- *aaaaaaaa*—name of the SMPCSI data set or SMPPTS data set
- *bbbbbbbb*—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.

**GIM59606v DEQ WAS SUCCESSFUL FOR EXCLUSIVE USE OF *aaaaaaaa* FOR *bbbbbbbb* PROCESSING.**

**Explanation:**

- *aaaaaaaa*—name of the SMPCSI data set or SMPPTS data set
- *bbbbbbbb*—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.

**GIM59607v DEQ WAS SUCCESSFUL FOR SHARED USE OF *aaaaaaaa* FOR *bbbbbbbb* PROCESSING.**

**Explanation:**

- *aaaaaaaa*—name of the SMPCSI data set or SMPPTS data set
- *bbbbbbbb*—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.

**GIM59608v ENQ WAS INITIATED FOR SHARED USE OF *aaaaaaaa* FOR *bbbbbbbb* PROCESSING.**

**Explanation:**

- *aaaaaaaa*—name of the SMPCSI data set or SMPPTS data set
- *bbbbbbbb*—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.

**GIM59701v THE GLOBAL ZONE WAS UPDATED WITH CHANGES FROM THE *aaaaaaa* ZONE.**

**Explanation:**

- *aaaaaaa*—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.

**GIM59801v THE GLOBAL ZONE WAS NOT UPDATED WITH CHANGES FROM THE *aaaaaaaa* ZONE.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM599v JOB *aaaaaaaa* HAS BEEN WAITING *bb* MINUTES FOR DATA SET *ccccccccc*.**

**Explanation:**

- *aaaaaaaa*—job name
- *bb*—time the task has been waiting (a multiple of 30 minutes)
- *ccccccccc*—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.

**GIM60001v *aaaaaaaa* PROCESSING FAILED BECAUSE AN SMPCSI OR THE SMPPTS DATA SET WAS NOT AVAILABLE.**

**Explanation:** 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. However, this data set has been in use for the past 10 minutes.

As a result, the job failed.

**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.

Then rerun the job.

**GIM60101v 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.

**GIM60201v** ZONE *aaaaaaaa* WAS SUCCESSFULLY WRITTEN TO OUTFILE *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—zone name
- *bbbbbbb*—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.

**GIM60301v** THE ZONE TYPES ARE INCOMPATIBLE. THE RECEIVING ZONE *aaaaaaaa* IS A *bbbbbbb* ZONE AND THE INPUT ZONE *ccccccc* IS A *ddddddd* ZONE.

**Explanation:**

- *aaaaaaaa*—first zone name
- *bbbbbbb*—first zone type
- *ccccccc*—second zone name
- *ddddddd*—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
<b>Notes:</b>	
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.

**GIM60401v** THE SET BOUNDARY ZONE *aaaaaaaa* AND THE INTO ZONE *bbbbbbb* HAVE DIFFERENT NAMES.

**Explanation:**

- *aaaaaaaa*—first zone name
- *bbbbbbb*—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.

**GIM60402v** THE SET BOUNDARY ZONE *aaaaaaaa* AND THE INPUT ZONE *bbbbbbb* HAVE DIFFERENT NAMES.

**Explanation:**

- *aaaaaaaa*—first zone name
- *bbbbbbb*—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.

**GIM60501v** THE INTO ZONE *aaaaaaaa* ALREADY EXISTS IN SMPCSI DATA SET *bbbbbbbbbb*.

**Explanation:**

- *aaaaaaaa*—zone name
- *bbbbbbbbbb*—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.

**GIM60601v** THE GLOBAL ZONE WAS NOT IMPORTED INTO *aaaaaaaaa* BECAUSE THAT DATA SET ALREADY CONTAINS A ZONE.

**Explanation:**

- *aaaaaaaaa*—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.

**GIM60701v** ZONEIMPORT PROCESSING FOR ZONE *aaaaaaaa* FROM INFILE *bbbbbbb* IS INCOMPLETE.

**Explanation:**

- *aaaaaaaa*—zone name
- *bbbbbbb*—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.

**GIM60801v** THE ZONE TYPE OF *aaaaaaaa* WAS CHANGED FROM DLIB TO TARGET.

**Explanation:**

- *aaaaaaaa*—zone name.

The zone being processed was changed from a DLIB zone to a target zone.

**System Action:** None.

**Programmer Response:** None.

**GIM60901v** THERE IS NO ZONE DEFINITION ENTRY FOR ZONE *aaaaaaaa*.

**Explanation:**

- *aaaaaaaa*—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.

**GIM61001v** *aaaaaaaa* IS NOT ALLOWED ON THE ZONEEDIT COMMAND. THE ENTRY TYPE MUST BE DDDEF OR UTILITY.

**Explanation:**

- *aaaaaaaa*—entry type.

The only entry types that may be specified on a ZONEEDIT command are DDDEF and UTILITY. Neither of these was specified on the command being processed.

**System Action:** Command processing stops.

**Programmer Response:**

- If you want to change a DDDEF or UTILITY entry, reenter the ZONEEDIT command with the correct entry type.
- If you want to change any other type of entry, use the UCLIN command.

**GIM61101v** THE *aaaaaaaa* SUBENTRY IS NOT ALLOWED FOR THE *bbbbbbbb* ENTRY.

**Explanation:**

- *aaaaaaaa*—subentry type
- *bbbbbbbb*—entry type.

The command being processed did not specify a subentry that exists in the entry type chosen.

**System Action:** Command processing stops. No changes are made.

**Programmer Response:** Specify a valid subentry on the command and rerun the job. For information about the subentries that may be specified, see *SMP/E Reference*.

**GIM61201v** *aaaaaaaa* WAS NOT MOVED BECAUSE SYSMOD *bbbbbbb* CONTAINS MORE THAN ONE ++MOVE MCS TO MOVE *aaaaaaaa* INTO THE SAME LIBRARY *cccccc*. THIS IS NOT ALLOWED.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—SYSMOD ID
- *cccccc*—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.



**GIM61301v** THE NAME OF THE ZONE ON THE ZONEIMPORT COMMAND (*aaaaaaaa*) IS DIFFERENT FROM THE NAME OF THE ZONE IN THE INFILE DATA SET (*bbbbbbbb*). THE NAMES MUST BE THE SAME.

**Explanation:**

- *aaaaaaaa*—first zone name
- *bbbbbbbb*—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.

**GIM61401v** 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.

**GIM61501v** THE NAME OF THE IMPORTED ZONE WAS CHANGED FROM *aaaaaaaa* TO *bbbbbbbb*.

**Explanation:**

- *aaaaaaaa*—old zone name
- *bbbbbbbb*—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.

**GIM61601v** *aaaaaaaa* PROCESSING FAILED BECAUSE THE INPUT ZONE AND RECEIVING ZONE HAVE THE SAME NAME (*bbbbbbbb*).

**Explanation:**

- *aaaaaaaa*—SMP/E command
- *bbbbbbbb*—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.

**GIM61701v** *aaaaaaaa bbbbbbbb* WAS NOT MOVED BY SYSMOD *cccccc* BECAUSE *bbbbbbb* DOES NOT EXIST IN THE SPECIFIED *dddddd* LIBRARY.

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbb*—element name
- *cccccc*—SYSMOD ID
- *dddddd*—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.

**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.

**GIM61702v** LMOD *aaaaaaaa* WAS NOT RENAMED BY SYSMOD *bbbbbbb* BECAUSE *aaaaaaaa* DOES NOT EXIST IN THE SPECIFIED *cccccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—SYSMOD ID
- *cccccc*—ddname of the library.

There was an entry for the indicated LMOD in the target or distribution 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:** 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.

**GIM61703v** *aaaaaaaa bbbbbbbb* WAS NOT DELETED BY SYSMOD *cccccc* BECAUSE *bbbbbbb* DOES NOT EXIST IN THE SPECIFIED *dddddd* LIBRARY.

**Explanation:**

- *aaaaaaaa*—LMOD or ALIAS
- *bbbbbbb*—element name
- *cccccc*—SYSMOD ID
- *dddddd*—ddname of the library.

There was an entry for the indicated LMOD or alias in the target or distribution zone, but the LMOD or alias 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.

**GIM61704v** *aaaaaaaa* WAS NOT MOVED BY SYSMOD *bbbbbbb* BECAUSE *aaaaaaaa* DOES NOT EXIST IN THE SPECIFIED *ccccccc* LIBRARY.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—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.

**GIM61801v** THE ++MOVE MCS FOR *aaaaaaaa* IN SYSMOD *bbbbbbb* SPECIFIES THE SAME LIBRARY (*ccccccc*) FOR BOTH DISTLIB AND TODISTLIB. THIS IS NOT ALLOWED.

THE ++MOVE MCS FOR *aaaaaaaa* IN SYSMOD *bbbbbbb* SPECIFIES THE SAME LIBRARY (*ccccccc*) FOR BOTH SYSLIB AND TOSYSLIB. THIS IS NOT ALLOWED.

**Explanation:**

- *aaaaaaaa*—member name
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—ddname of the library.

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.

**GIM61901v** *aaaaaaaa* *bbbbbbb* WAS NOT MOVED BY SYSMOD *ccccccc* BECAUSE *bbbbbbb* IS NOT IN THE *ddddddd* ZONE.

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbb*—element name
- *ccccccc*—SYSMOD ID
- *ddddddd*—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 accepting an element with DISTLIB changes, processing stops.
- Otherwise, processing continues.

**Programmer Response:** If processing stops, add the missing entry and rerun the job. Otherwise, no action is needed.

**GIM61902v** *aaaaaaaa bbbbbbbb* WAS NOT RENAMED BY SYSMOD *cccccc* BECAUSE *bbbbbbb* IS NOT IN THE *ddddddd* ZONE.

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbb*—element name
- *cccccc*—SYSMOD ID
- *ddddddd*—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.

**GIM61903v** *aaaaaaaa bbbbbbbb* WAS NOT DELETED BY SYSMOD *cccccc* BECAUSE *bbbbbbb* IS NOT IN THE *ddddddd* ZONE.

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbb*—element name
- *cccccc*—SYSMOD ID
- *ddddddd*—zone name.

SMP/E did not find an entry for the element in the indicated zone, so there was nothing to delete.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM61904v** *aaaaaaaa* WAS NOT MOVED BY SYSMOD *bbbbbbb* BECAUSE *aaaaaaaa* IS NOT IN THE *ccccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—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 accepting an element with DISTLIB changes, processing stops.
- Otherwise, processing continues.

**Programmer Response:** If processing stops, add the missing entry and rerun the job. Otherwise, no action is needed.

**GIM61905v** *aaaaaaaa* WAS NOT RENAMED BY SYSMOD *bbbbbbb* BECAUSE *aaaaaaaa* IS NOT IN THE *ccccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—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.

**GIM61906v** *aaaaaaaa* WAS NOT DELETED BY SYSMOD *bbbbbbb* BECAUSE *aaaaaaaa* IS NOT IN THE *ccccccc* ZONE.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—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.

**GIM62001v** *aaaaaaaa* *bbbbbbb* WAS MOVED BY SYSMOD *ccccccc* FROM *ddddddd* *eeeeeee* TO *ffffff*.

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbb*—element name
- *ccccccc*—SYSMOD ID
- *ddddddd*—DISTLIB or SYSLIB
- *eeeeeee*—ddname of the old library
- *ffffff*—ddname of the new library.

The element was successfully moved from the old library to the new one.

**System Action:** None.

**Programmer Response:** None.

**GIM62002v** *aaaaaaaa* WAS MOVED BY SYSMOD *bbbbbbb* FROM *ccccccc* *ddddddd* TO *eeeeeee*.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—DISTLIB or SYSLIB
- *ddddddd*—ddname of the old library
- *eeeeeee*—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.

**GIM62101v** THE *aaaaaaaa* OPERAND IS REQUIRED BECAUSE THE *bbbbbbb* OPERAND WAS SPECIFIED ON THE ++MOVE MCS IN SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—missing operand
- *bbbbbbb*—specified operand
- *ccccccc*—SYSMOD ID.

The indicated operand, *aaaaaaaa*, is required because the second operand, *bbbbbbb*, was specified. However, *aaaaaaaa* 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.

**GIM62102v** THE *aaaaaaaa* OPERAND IS NOT ALLOWED BECAUSE THE *bbbbbbbb* OPERAND WAS SPECIFIED ON THE ++MOVE MCS IN SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—first operand
- *bbbbbbbb*—second operand
- *ccccccc*—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.

**GIM62201v** THE ++*aaaaaaaa* MCS IS OUT OF SEQUENCE.

**Explanation:**

- *aaaaaaaa*—MCS type.

The indicated MCS is out of order in the SMPPTFIN data set.

**System Action:** SYSMOD processing stops.

**Programmer Response:** Use *SMP/E Reference* or *SMP/E Program Packaging Guide* to find out the correct order for the MCS statements. Make any necessary changes, then rerun the job.

**GIM62401v** A MEMBER TYPE OPERAND IS REQUIRED ON THE ++*aaaaaaaa* MCS FOR *bbbbbbbb* IN SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—MCS type
- *bbbbbbbb*—element name
- *ccccccc*—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.

**GIM62402v** A LIBRARY TYPE OPERAND IS REQUIRED ON THE ++*aaaaaaaa* MCS FOR *bbbbbbbb* IN SYSMOD *ccccccc*.

**Explanation:**

- *aaaaaaaa*—MCS type
- *bbbbbbbb*—element name
- *ccccccc*—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.

**GIM62403v** A *aaaaaaaa* OPERAND IS REQUIRED ON THE ++*bbbbbbbb* MCS FOR *ccccccc* IN SYSMOD *ddddddd*.

**Explanation:**

- *aaaaaaaa*—operand
- *bbbbbbbb*—MCS type
- *ccccccc*—element name
- *ddddddd*—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.

**GIM62501v** *aaaaaaaa* IS NOT A DEFINED DUMP POINT.**Explanation:**

- *aaaaaaaa*—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.

**GIM62502v** MESSAGE ID *aaaaaaaa* IS NOT AN SMP/E MESSAGE.**Explanation:**

- *aaaaaaaa*—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.

**GIM62601v** CURRENT DATA SET NAME *aaaaaaaaaa* WAS NOT CHANGED BECAUSE THE NEW NAME *bbbbbbbbbb* EXCEEDS THE 44-CHARACTER MAXIMUM.**Explanation:**

- *aaaaaaaaaa*—current data set name
- *bbbbbbbbbb*—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.

**GIM62701v** THE ++*aaaaaaaa* MCS FOR *bbbbbbbb* ENTRY *ccccccc* IN SYSMOD *dddddd* WAS NOT PROCESSED BECAUSE THE *eeeeeee* VALUE DOES NOT MATCH THE EXISTING *eeeeeee* VALUE.**Explanation:**

- *aaaaaaaa*—MCS type
- *bbbbbbbb*—entry type
- *ccccccc*—entry name
- *dddddd*—SYSMOD ID
- *eeeeeee*—operand.

SMP/E detected a value specified on the indicated MCS that did not match the value currently in the indicated entry.

**System Action:** If the message severity (*v*) is warning (W), SYSMOD processing continues. However, when processing a ++DELETE MCS, the load module will not be deleted from any of its system libraries. If the message severity (*v*) is error (E), SYSMOD processing stops.

**Programmer Response:** If the message severity is warning, no programmer response is required. If the message severity is error, specify the correct value on the MCS and rerun the job.

**GIM62702v** THE ++*aaaaaaaa* MCS FOR ENTRY *bbbbbbbb* IN SYSMOD *ccccccc* WAS NOT PROCESSED BECAUSE THE *ddddddd* VALUE DOES NOT MATCH THE EXISTING *ddddddd* VALUE.**Explanation:**

- *aaaaaaaa*—MCS type
- *bbbbbbbb*—entry name
- *ccccccc*—SYSMOD ID
- *ddddddd*—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:** If the message severity (*v*) is warning (W), SYSMOD processing continues. However, when a ++DELETE MCS is being processed, the load module will not be deleted from any of its system libraries. If the message severity (*v*) is error (E), SYSMOD processing stops.

**Programmer Response:** If the message severity is warning, no programmer response is required. If the message severity is error, specify the correct value on the MCS and rerun the job.

**GIM62801v 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.

**GIM62901v LMOD *aaaaaaaa* WAS RENAMED TO *bbbbbbbb* IN THE *ccccccc* LIBRARY BY SYSMOD *ddddddd*.**

**Explanation:**

- *aaaaaaaa*—old LMOD name
- *bbbbbbbb*—new LMOD name
- *ccccccc*—ddname of the library
- *ddddddd*—SYSMOD ID.

The LMOD was successfully renamed.

**System Action:** None.

**Programmer Response:** None.

**GIM63201v *aaaaaaaa bbbbbbbb* WAS DELETED FROM THE *ccccccc* LIBRARY BY SYSMOD *ddddddd*.**

**Explanation:**

- *aaaaaaaa*—LMOD or ALIAS
- *bbbbbbbb*—entry name
- *ccccccc*—ddname of the library
- *ddddddd*—SYSMOD ID.

The LMOD or ALIAS was successfully deleted.

**System Action:** None.

**Programmer Response:** None.

**GIM63301v SYSMOD *aaaaaaa* WAS NOT RESTORED BECAUSE IT CONTAINS A ++DELETE MCS.**

**Explanation:**

- *aaaaaaa*—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.

**GIM63401v ZONESET *aaaaaaaa* IS NOT DEFINED IN THE GLOBAL ZONE.**

**Explanation:**

- *aaaaaaaa*—ZONESET name.

The indicated ZONESET is not defined in the global zone.



**System Action:** REPORT 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 to define the ZONESET entry in the global zone.

**GIM63501v 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.

**GIM63901v THE aaaaaaaa OPERAND IS REQUIRED ON THE bbbbbbbb COMMAND.**

**Explanation:**

- aaaaaaaa—operand
- bbbbbbbb—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.

**GIM64001v JOBCARD MEMBER aaaaaaaa IS NOT IN THE bbbbbbbb LIBRARY. A JCL COMMENT WILL BE GENERATED EACH TIME A JOB CARD IS REQUIRED.**

**Explanation:**

- aaaaaaaa—member name
- bbbbbbbb—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 Reference*.) Replace the JCL comment with a valid job card.

**GIM64101v NO ASSEMBLER STEP WAS GENERATED FOR MOD aaaaaaaa BECAUSE THERE IS NO ASSOCIATED ASSEM OR SRC ENTRY.**

**Explanation:**

- aaaaaaaa—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.

**GIM64201v MOD aaaaaaaa WAS NOT INCLUDED IN THE LINK-EDIT STEP FOR LMOD bbbbbbbb BECAUSE aaaaaaaa CANNOT BE ASSEMBLED.**

**Explanation:**

- aaaaaaaa—module name
- bbbbbbbb—load module name.

SMP/E could not assemble the indicated module, as shown in message GIM64101. 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 GIM64101 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.

**GIM64301v A JOB STEP WAS GENERATED TO COPY aaaaaaaa bbbbbbbb FROM cccccc TO ddddddd. aaaaaaaa ENTRY bbbbbbbb HAS NO FMID AND THEREFORE MIGHT NOT HAVE BEEN INSTALLED. AN ERROR WILL OCCUR IF bbbbbbbb IS NOT IN cccccc.**

**Explanation:**

- aaaaaaaa—element type
- bbbbbbbb—element name
- cccccc—ddname of the distribution library
- ddddddd—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.

**GIM64401v A JOB STEP WAS GENERATED TO COPY MOD aaaaaaaa TO LMOD bbbbbbbb IN LIBRARY cccccc. MOD ENTRY aaaaaaaa HAS NO FMID AND THEREFORE MIGHT NOT HAVE BEEN INSTALLED. AN ERROR WILL OCCUR IF MOD aaaaaaaa IS NOT IN THE DISTRIBUTION LIBRARY.**

**Explanation:**

- aaaaaaaa—element name
- bbbbbbbb—load module name
- cccccc—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.

**GIM64402v** A JOB STEP WAS GENERATED TO LINK-EDIT MOD *aaaaaaaa* TO LMOD *bbbbbbbb* IN LIBRARY *ccccccc*. MOD ENTRY *aaaaaaaa* HAS NO FMID AND THEREFORE MIGHT NOT HAVE BEEN INSTALLED. AN ERROR WILL OCCUR IF MOD *aaaaaaaa* IS NOT IN THE DISTRIBUTION LIBRARY.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbbb*—load module name
- *ccccccc*—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.

**GIM64501v** NO LINK-EDIT STEPS WERE GENERATED FOR LMOD *aaaaaaaa* BECAUSE NO MOD ENTRIES SPECIFY *aaaaaaaa* AS AN LMOD SUBENTRY.

**Explanation:**

- *aaaaaaaa*—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.

**GIM64601v** A COMMENT WAS GENERATED FOR THE *aaaaaaaa* JCL STATEMENT IN STEP *bbbbbbbb* OF JOB *ccccccc* BECAUSE THERE IS NO DDDEF ENTRY FOR *aaaaaaaa*.

**Explanation:**

- *aaaaaaaa*—name of the JCL statement
- *bbbbbbbb*—step name
- *ccccccc*—job 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 Reference*.) 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.

**GIM64801v** NO JOB STEP WAS GENERATED FOR *aaaaaaaa* *bbbbbbbb* BECAUSE *bbbbbbbb* HAS NO DISTLIB SUBENTRY.

**Explanation:**

- *aaaaaaaa*—entry type
- *bbbbbbbb*—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.

**GIM64901v** 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.

**GIM64902v** 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.

**GIM65001v** STEP *aaaaaaaa* IN JOB *bbbbbbbb* WAS NOT PROCESSED BECAUSE *ccccccc* WAS NOT SPECIFIED AS A UTILITY ON THE JCLIN COMMAND.

**Explanation:**

- *aaaaaaaa*—step name
- *bbbbbbbb*—job name
- *ccccccc*—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.

**GIM65002v** STEP *aaaaaaaa* IN JOB *bbbbbbbb* WAS NOT PROCESSED BECAUSE *ccccccc* WAS NOT SPECIFIED AS A UTILITY ON THE ++JCLIN MCS IN SYSMOD *ddddddd*.

**Explanation:**

- *aaaaaaaa*—step name
- *bbbbbbbb*—job name
- *ccccccc*—program or procedure name
- *ddddddd*—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.

**GIM65101v ZONE *aaaaaaaa* IS AN EMPTY ZONE.****Explanation:**

- *aaaaaaaa*—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.

**GIM65201v CSI *aaaaaaaa* DOES NOT CONTAIN A GIMZPOOL RECORD.****Explanation:**

- *aaaaaaaa*—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. For commands other than CONVERT, 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.

**GIM65301v THE *aaaaaaaa* SUBENTRY WAS SPECIFIED MORE THAN ONCE ON A ZONEEDIT CHANGE STATEMENT. THIS IS NOT ALLOWED.****Explanation:**

- *aaaaaaaa*—subentry type.

A single ZONEEDIT CHANGE statement specified the same subentry type more than once. This is not allowed.

**System Action:** Command processing stops.

**Programmer Response:** Fix the CHANGE statement to specify the field only once. If there is more than one change for the same field, code a CHANGE statement for each change. Then rerun the job.

**GIM65401v 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.

**GIM65501v ZONE *aaaaaaaa* IN THE FORZONE OPERAND IS NOT DEFINED IN THE *bbbbbbbb* ZONESET.**

**Explanation:**

- *aaaaaaaa*—zone name
- *bbbbbbbb*—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.

**GIM65601v SMP/E ASSUMED THAT ALL THE ZONES IN THE ZONESET WERE DLIB ZONES. *aaaaaaaa* IS A TARGET ZONE.**

**SMP/E ASSUMED THAT ALL THE ZONES IN THE ZONESET WERE TARGET ZONES. *aaaaaaaa* IS A DLIB ZONE.**

**Explanation:**

- *aaaaaaaa*—zone name.

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.

**GIM65701v LOAD MODULE BUILD PROCESSING STOPPED BECAUSE DISTRIBUTION ZONE *aaaaaaaa* WAS NOT AVAILABLE.**

**Explanation:**

- *aaaaaaaa*—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:

- GIM674xx, which lists the load modules that failed
- GIM67501, which lists the modules SMP/E requires to complete the load modules
- GIM22601, 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.

**GIM65801v MODULE *aaaaaaaa* IS NOT IN DISTRIBUTION ZONE *bbbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbbb*—zone name.

SMP/E could not find the indicated module in the distribution zone during load module build processing. Message GIM673xx, which follows this message, identifies the affected load modules and target libraries.

**System Action:** The load module is link-edited without the missing module and processing continues.

**Programmer Response:** Link-edit the module into the load module outside SMP/E.

**GIM65901v** MODULE *aaaaaaaa* WAS NOT LINK-EDITED BECAUSE ITS SERVICE LEVEL IN DISTRIBUTION ZONE *bbbbbbbb* IS DIFFERENT FROM THAT IN THE TARGET ZONE.

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbbb*—name of the distribution zone.

SMP/E required the specified module in order to build a load module. 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 (PTF with a ++MOD, for example) is applied to one zone and not accepted in the other. This service could be source input (++SRC), a complete object deck replacement (++MOD), or a zap (++ZAP) for the module. Message GIM673xx, which follows this message, lists the affected load modules.

**System Action:** The load module is link-edited without the module, and processing continues.

**Programmer Response:** Link-edit the module into the load module outside SMP/E.

**GIM66001v** 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.

**GIM66101v** THE JCLIN CROSS-REFERENCE REPORT WAS NOT PRODUCED BECAUSE SMP/E COULD NOT OBTAIN ENOUGH STORAGE FOR PROCESSING.

**Explanation:** SMP/E could not obtain enough storage to produce the JCLIN Cross-Reference Report.

**System Action:** JCLIN processing continues, but the Cross-Reference report will not be produced.

**Programmer Response:** None.

**GIM66201v** THERE ARE NO *aaaaaaaa* ENTRIES IN THE *bbbbbbbb* ZONE.

**Explanation:**

- *aaaaaaaa*—DDDEF or UTILITY
- *bbbbbbbb*—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.

**GIM66301v** 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.



**GIM66501v** *aaaaaaaa* WAS SPECIFIED MORE THAN ONCE ON THE *bbbbbbbb* OPERAND OF THE ++*ccccccc* MCS FOR *ddddddd* IN SYSMOD *eeeeeee*. THIS IS NOT ALLOWED.

**Explanation:**

- *aaaaaaaa*—operand value
- *bbbbbbbb*—operand type
- *ccccccc*—MCS type
- *ddddddd*—element name
- *eeeeeee*—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.

**GIM66601v** *aaaaaaaa* VALUE *bbbbbbbb* IS THE SAME AS THE ELEMENT NAME ON THE ++*ccccccc* MCS FOR *ddddddd* IN SYSMOD *eeeeeee*. THESE VALUES MUST BE DIFFERENT.

**Explanation:**

- *aaaaaaaa*—operand type
- *bbbbbbbb*—operand value
- *ccccccc*—MCS type
- *ddddddd*—element name
- *eeeeeee*—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.

**GIM66701v** 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.

**GIM66801v** MTSMAC ENTRY *aaaaaaaa* WAS NOT DELETED FROM THE SMPMTS DATA SET BECAUSE THERE IS NO CORRESPONDING MAC ENTRY FOR *aaaaaaaa* IN THE *bbbbbbbb* TARGET ZONE.

STSSRC ENTRY *aaaaaaaa* WAS NOT DELETED FROM THE SMPSTS DATA SET BECAUSE THERE IS NO CORRESPONDING SRC ENTRY FOR *aaaaaaaa* IN THE *bbbbbbbb* TARGET ZONE.

**Explanation:**

- *aaaaaaaa*—entry name
- *bbbbbbbb*—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.

**GIM66802v** BACKUP ENTRIES FOR *aaaaaaaa* WERE NOT DELETED FROM THE SMPSCDS DATA SET BECAUSE THERE IS NO CORRESPONDING SYSMOD ENTRY FOR *aaaaaaaa* IN THE *bbbbbbbb* TARGET ZONE.

**Explanation:**

- *aaaaaaaa*—SYSMOD ID
- *bbbbbbbb*—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.

**GIM66901v** THE ALIAS OPERAND EXCEEDS THE 16-VALUE MAXIMUM ON THE +-DELETE MCS FOR *aaaaaaaa* IN SYSMOD *bbbbbbb*.

THE FMID OPERAND EXCEEDS THE 10-VALUE MAXIMUM ON THE ++MOVE MCS FOR *aaaaaaaa* IN SYSMOD *bbbbbbb*.

THE SYSLIB OPERAND EXCEEDS THE 2-VALUE MAXIMUM ON THE +-DELETE MCS FOR *aaaaaaaa* IN SYSMOD *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—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.

**GIM67001v** THE SAME VALUE (*aaaaaaaa*) WAS SPECIFIED FOR THE RELATED ZONE AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.

**Explanation:**

- *aaaaaaaa*—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.

**GIM67002v** THE SAME VALUE (*aaaaaaaa*) WAS SPECIFIED FOR THE OUTFILE DDNAME AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.

**Explanation:**

- *aaaaaaaa*—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.

**GIM67003v** THE SAME VALUE (*aaaaaaaa*) WAS SPECIFIED FOR THE INFILE DDNAME AND FOR THE ZONE ON THE SET COMMAND. THESE VALUES MUST BE DIFFERENT.

**Explanation:**

- *aaaaaaaa*—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.

**GIM67101v 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 Chapter 1, "SMP/E Return Codes" for a description of ZONEEDIT return codes. Fix the indicated problem and rerun the job.

**GIM67201v LOAD MODULE BUILD PROCESSING STOPPED BECAUSE NO RELATED DISTRIBUTION ZONE IS DEFINED FOR TARGET ZONE *aaaaaaaa*.**

**Explanation:**

- *aaaaaaaa*—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:

- GIM674xx, which lists the load modules that failed
- GIM67501, which lists the modules SMP/E requires to complete the load modules
- GIM22601, 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.

**GIM67301v LOAD MODULE *aaaaaaaa* IN SYSLIB *bbbbbbbb* DOES NOT INCLUDE MODULE *cccccccc* BECAUSE SMP/E COULD NOT FIND *cccccccc*.**

**Explanation:**

- *aaaaaaaa*—load module name
- *bbbbbbbb*—SYSLIB value
- *cccccccc*—module name.

SMP/E could not find a module that it needs to build the indicated load module. 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.

**GIM67302v LOAD MODULE *aaaaaaaa* IN SYSLIBS *bbbbbbbb* AND *cccccc* DOES NOT INCLUDE MODULE *ddddddd* BECAUSE SMP/E COULD NOT FIND *ddddddd*.**

**Explanation:**

- *aaaaaaaa*—load module name
- *bbbbbbbb*—SYSLIB value
- *cccccc*—SYSLIB value
- *ddddddd*—module name.

SMP/E could not find a module that it needs to build the indicated load module. 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.

---

**GIM67401v** LOAD MODULE *aaaaaaaa* WAS NOT BUILT IN SYSLIB *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—load module name
- *bbbbbbb*—SYSLIB value.

SMP/E stopped building the indicated load module because of a previous error. Check for messages (such as GIM67201) that explain this error.

**System Action:** Load module processing ends. SMP/E issues the following related messages:

- GIM67501, which lists the modules SMP/E requires to complete the load module
- GIM22601, which lists the SYSMODs that failed because load modules failed.

**Programmer Response:** Correct the error indicated in the other messages and rerun the job.

**GIM67402v** LOAD MODULE *aaaaaaaa* WAS NOT BUILT IN SYSLIBS *bbbbbbb* AND *ccccccc*.

**Explanation:**

- *aaaaaaaa*—load module name
- *bbbbbbb*—SYSLIB value
- *ccccccc*—SYSLIB value.

SMP/E stopped building the indicated load module because of a previous error. Check for messages (such as GIM67201) that explain this error.

**System Action:** Load module processing ends. SMP/E issues the following related messages:

- GIM67501, which lists the modules SMP/E requires to complete the load module
- GIM22601, which lists the SYSMODs that failed because load modules failed.

**Programmer Response:** Correct the error indicated in the other messages and rerun the job.

**GIM67501v** MODULE *aaaaaaaa* IS REQUIRED FOR LOAD MODULE *bbbbbbb*.

**Explanation:**

- *aaaaaaaa*—module name
- *bbbbbbb*—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 GIM65701 or GIM67201) 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.

**GIM67601v** SMPTLIB *aaaaaaaaa* WAS CATALOGED ON VOLUME *bbbbbb*.

**Explanation:**

- *aaaaaaaaa*—data set name
- *bbbbbb*—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.

**GIM67602v** SMPTLIB *aaaaaaaaa* WAS DELETED FROM VOLUME *bbbbbb*.

**Explanation:**

- *aaaaaaaaa*—data set name
- *bbbbbb*—volume serial number.

During command processing, SMP/E deleted an existing SMPTLIB from the indicated volume.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM67701v LOG RECORDING WAS SWITCHED TO SMPDUMP 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 the SMPDUMP data set.

**System Action:** Processing continues.

**Programmer Response:** Make sure the SMPLOG and SMPLOGA data sets have the same block size.

**GIM67801v DUMPOFF IS NOT ALLOWED FOR *aaaaaaaa* BECAUSE *aaaaaaaa* IS ALREADY OFF.**

**Explanation:**

- *aaaaaaaa*—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.

**GIM67802v 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.

**GIM67901v MOD ENTRY *aaaaaaaa* CONTAINS THE SAME VALUE (*bbbbbbbb*) FOR THE TALIAS AND LMOD SUB-ENTRIES. THE TALIAS SUBENTRY WAS IGNORED AND SHOULD BE DELETED.**

**Explanation:**

- *aaaaaaaa*—entry name
- *bbbbbbbb*—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.

**GIM68001v THE DUMP ID FOR *aaaaaaaa* IS *bbbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—SMP/E message number
- *bbbbbbbb*—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.

**GIM69000v THE DATA WAS NOT TRANSFORMED BECAUSE THE OPEN MACRO FAILED FOR THE *aaaaaaa* DATA SET.**

**Explanation:**

- *aaaaaaa*—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.

**GIM69001v THE DATA WAS NOT TRANSFORMED BECAUSE THE *aaaaaaa* DATA SET IS MISSING.**

**Explanation:**

- *aaaaaaa*—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.

**GIM69002v THE DATA WAS NOT TRANSFORMED BECAUSE THE ATTRIBUTES OF THE *aaaaaaa* DATA SET ARE NOT SUPPORTED.**

**Explanation:**

- *aaaaaaa*—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.

**GIM69003v 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.

**GIM69004v 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.

**GIM69005v 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.

**GIM69006v aaaaaaaaa WAS TRANSFORMED INTO bbbbbbbbb.**

**Explanation:**

- aaaaaaaaa—data set name of the SYSUT1 data set
- bbbbbbbbb—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.

**GIM69007v 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.

**GIM69008v GIMDTS PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE IS aa.**

**Explanation:**

- aa—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.

**GIM69009v aaaaaaaaa bbbbbbbb COULD NOT BE RETRANSFORMED BECAUSE THE ccccccccc LIBRARY'S ATTRIBUTES CONFLICT WITH THE ORIGINAL ATTRIBUTES OF THE DATA.**

**Explanation:**

- aaaaaaaaa—element type
- bbbbbbbb—element name
- ccccccccc—data set name of the library where the element should be installed.

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 to find out the RECFM and LRECL values required for the library where the element should be installed. Specify these values for the library and rerun the job.

**GIM69010v AN ERROR OCCURRED IN RETRANSFORMATION OF *aaaaaaaa bbbbbbbb* ON THE *ccccccccc* LIBRARY. DATA HAS BEEN LOST.**

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbbb*—element name
- *ccccccccc*—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:** Report the error to your support group.

**GIM69011v RETRANSFORMATION OF *aaaaaaaa bbbbbbbb* ON LIBRARY *ccccccccc* WAS UNSUCCESSFUL.**

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbbb*—element name
- *ccccccccc*—data set name of the library where the element should be installed.

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.

**GIM69012v RETRANSFORMATION OF *aaaaaaaa bbbbbbbb* ON LIBRARY *ccccccccc* WAS SUCCESSFUL.**

**Explanation:**

- *aaaaaaaa*—element type
- *bbbbbbbb*—element name
- *ccccccccc*—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.

**GIM69013v 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.



**GIM69014v THE aaaaaaaa ZONE WAS SPECIFIED ON THE bbbbbbbb OPERAND BUT IT DOES NOT EXIST.****Explanation:**

- aaaaaaaa—zone name
- bbbbbbbb—INZONE or COMPAREDTO.

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.

**GIM69015v THE GLOBAL ZONE IS NOT ALLOWED AS THE VALUE OF THE aaaaaaaa OPERAND.****Explanation:**

- aaaaaaaa—INZONE or COMPAREDTO.

GLOBAL is not allowed as the zone name specified on the indicated operand. You must specify the name of a target or distribution zone.

**System Action:** Command processing stops.

**Programmer Response:** Specify the name of a target or distribution zone on the indicated operand and rerun the command.

**GIM69016v 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.

**GIM69017v THE aaaaaaaa ZONE WAS SPECIFIED ON THE bbbbbbbb OPERAND BUT IT CONTAINED NO SYSMOD ENTRIES.****Explanation:**

- aaaaaaaa—zone name
- bbbbbbbb—INZONE or COMPAREDTO.

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.

**GIM69018v THE NEWCSI DATA SET IS NOT EMPTY.**

**Explanation:** The NEWCSI data set contains records other than the GIMZPOOL record. For SMP/E to convert another CSI into the NEWCSI data set, it must be empty except for the GIMZPOOL record.

**System Action:** CONVERT processing stops.

**Programmer Response:** Verify that the data set you specify on the NEWCSI operand is empty except for a GIMZPOOL record. Then rerun the job.

**GIM69019v IF OLDCSI CONTAINS A GLOBAL ZONE, SMPCSI AND NEWCSI MUST SPECIFY THE SAME DATA SET.**

**Explanation:** If you are converting a CSI that contains a global zone, the SMPCSI DD statement must point to the same CSI data set as the NEWCSI operand.

**System Action:** CONVERT processing stops.

**Programmer Response:** Make sure that the SMPCSI DD statement and the NEWCSI operand specify the same data set. Then rerun the job.

**GIM69020v CSI *aaaaaaaa* WAS NOT CONVERTED TO *bbbbbbbbbb*.**

**Explanation:**

- *aaaaaaaa*—name of the old CSI data set
- *bbbbbbbbbb*—name of the new CSI data set.

The CSI data set specified on the OLDCSI operand was not converted to the indicated data set. See other messages for more information about the errors that occurred.

**System Action:** See descriptions of the other messages that were issued to find out the system action.

**Programmer Response:** See descriptions of the other messages that were issued to find out how to correct the errors that occurred.

**GIM69021v SMP/E TRIED TO USE DDNAME *aaaaaaa* FOR INTERNAL ALLOCATION PROCESSING. HOWEVER, THAT DDNAME IS ALREADY DEFINED FOR ANOTHER DATA SET.**

**Explanation:**

- *aaaaaaa*—ddname.

SMP/E allocates certain data sets for its own internal use. It tried to use the indicated ddname for one of those data sets, but that ddname was already defined for another data set.

**System Action:** CONVERT processing stops.

**Programmer Response:** Change the ddname of the existing data set. Then rerun the CONVERT command.

**GIM69023v THE SMPSCDS DATA SET MUST BE EMPTY.**

**Explanation:** The new SMPSCDS must be empty before data from the old SMPSCDS is converted and copied into it.

**System Action:** CONVERT processing stops.

**Programmer Response:** Make sure the new SMPSCDS is empty, then rerun the CONVERT command.

**GIM69024v SCDS *aaaaaaa* WAS SUCCESSFULLY CONVERTED.**

**Explanation:**

- *aaaaaaa*—ddname.

The indicated SMPSCDS data set was successfully converted.

**System Action:** Processing continues.

**Programmer Response:** None.

**GIM69025v SCDS *aaaaaaa* WAS NOT CONVERTED.**

**Explanation:**

- *aaaaaaa*—ddname.

The indicated SMPSCDS data set was not successfully converted.

**System Action:** CONVERT processing stops.

**Programmer Response:** Check the related messages that were issued for this CONVERT command. Make the necessary corrections and rerun the job.

**GIM69026v THE SMPSCDS DATA SET CONTAINS UNCONVERTED MEMBERS.**

**Explanation:** SMP/E checked a member of the SMPSCDS data set and found that it had not been converted to run with this level of SMP/E.

**System Action:** SMP/E processing stops.

**Programmer Response:** Make sure that the SMPSCDS DDDEF entry or DD statement points to the correct data set. If you are using the correct data set, use the CONVERT command to bring the data set to the correct level.

Then rerun the command that failed.

**GIM69027v SMP/E FOUND DUPLICATE INTERNAL RECORDS FOR ZONE *aaaaaaa* AND ZONE *bbbbbbb*. HOWEVER, THE GLOBAL ZONE CONTAINS A ZONEINDEX SUBENTRY FOR EACH OF THESE ZONES. ONE OF THE ZONES IS INCORRECT.**

**Explanation:**

- *aaaaaaa*—first zone name
- *bbbbbbb*—second zone name.

During CONVERT processing, SMP/E found two identical internal records for the indicated zones. SMP/E also found ZONEINDEX subentries for each of these zones in the global zone. The global zone referred to is one of the following:

- If the OLDCSI data set contains a global zone, then the global zone referred to is the one in the OLDCSI data set.
- If the OLDCSI data set does not contain a global zone, then the global zone referred to is the converted one in the CSI data set specified by the SMPCSI DD statement.

This situation may happen if you run a command outside of SMP/E (such as REPRO) to copy information into an existing zone.

**System Action:** CONVERT processing stops.

**Programmer Response:** Determine which of the zones is the correct one for your CSI. Delete the ZONEINDEX subentry for the incorrect zone. Then rerun the CONVERT command.

**GIM69028v IF OLDCSI DOES NOT CONTAIN A GLOBAL ZONE, SMPCSI AND NEWCSI MUST SPECIFY DIFFERENT DATA SETS.**

**Explanation:** If you are converting a CSI that contains target or distribution zones but no global zone, the SMPCSI DD statement must point to the CSI data set that contains the converted global zone. This data set must be different from the one you are converting into, which is specified on the NEWCSI operand.

**System Action:** CONVERT processing stops.

**Programmer Response:** Make sure that the SMPCSI DD statement and the NEWCSI operand specify different data sets. Then rerun the job.

**GIM69029v THE GLOBAL ZONE DOES NOT CONTAIN A ZONEINDEX SUBENTRY FOR ZONE *aaaaaaa*. RECORDS BELONGING TO THIS ZONE WERE NOT CONVERTED.**

**Explanation:**

- *aaaaaaa*—name of the target or distribution zone to be converted.

The global zone does not contain a ZONEINDEX subentry for the indicated zone. The global zone referred to is one of the following:

- If the OLDCSI data set contains a global zone, then the global zone referred to is the one in the OLDCSI data set.
- If the OLDCSI data set does not contain a global zone, then the global zone referred to is the converted one in the CSI data set specified by the SMPCSI DD statement.

**Note:** In general, SMP/E expects the global zone to contain a ZONEINDEX subentry for each zone in the OLDCSI data set. The data set name in each ZONEINDEX subentry must match the OLDCSI data set name.

---

These are the circumstances when this could occur:

---

The OLDCSI data set has already been converted. In this case, this message will be issued for every zone in the data set. No records will be written to the NEWCSI data set.

**System Action:** Command processing continues.

**Programmer Response:** None. This data set has already been converted.

---

The OLDCSI data set is not part of this CSI configuration. This data set was never associated with the converted global zone. In this case, this message will be issued for every zone in the data set. No records will be written to the NEWCSI data set.

**System Action:** Command processing continues.

**Programmer Response:** None. You cannot use the CONVERT command to merge data sets.

---

The CSI containing the global zone was not converted first. In this case, this message will be issued for every zone in the data set. No new records will be written to the NEWCSI data set.

**System Action:** Command processing continues.

**Programmer Response:** Convert the global zone associated with the OLDCSI data set. Then convert the OLDCSI data set.

---

Extraneous internal records are present in the OLDCSI data set. This may happen if a ZONEINDEX subentry was deleted but the zone associated with it was not deleted. None of the records associated with this zone were deleted.

**System Action:** Command processing continues.

**Programmer Response:** If you no longer want this zone, no action is required because no records belonging to this zone were written to the NEWCSI data set.

If you do want this zone, you must make UCLIN changes and rerun the CONVERT command for the OLDCSI data set.

- If the OLDCSI data set contains the global zone, use UCLIN (or the SMP/E Administration dialog) from SMP/E Release 4, or earlier, to add a ZONEINDEX subentry for this zone to the global zone. The data set name in the ZONEINDEX subentry must match the data set containing the zone. Delete the CSI that was used for the conversion and allocate a new CSI. Rerun the CONVERT job.
  - If the OLDCSI does not contain a global zone, use UCLIN (or the Administration dialog) from SMP/E Release 5 or SMP/E Release 6 to delete global zone ZONEINDEX subentries for all the zones in the OLDCSI data set. Then use UCLIN (or the Administration dialog) to add ZONEINDEX subentries to the global zone for all the zones in the OLDCSI to be converted (including the zone indicated in message GIM69029). The data set name in the ZONEINDEX subentry must match the OLDCSI data set name. Delete the NEWCSI data set that was used for the conversion and allocate a new CSI. Rerun the CONVERT job.
- 

The OLDCSI data set contains a duplicate internal record for the indicated zone. This is an irregularity in the zone which may have been caused by running a command outside of SMP/E (such as REPRO) to copy information into an existing zone. If the matching internal record has a ZONEINDEX subentry, the matching zone will be converted and this one will be ignored.

**System Action:** Command processing continues.

**Programmer Response:** You may want to check the records in the zone you are trying to convert because it contains irregularities. You can use the Access Method Services PRINT command to look at these records. If you want to convert the other zone, use UCLIN (or the Administration dialog) to change the ZONEINDEX subentry in the global zone (the new global zone in the SMPCSI data set if you have already converted your global zone, the old global zone if you have not yet converted your global zone). The data set name in the ZONEINDEX subentry must match the data set containing the zone. Delete the CSI that was used for conversion and allocate a new CSI. Rerun the CONVERT job.

---

**GIM69030v SMP/E FOUND DUPLICATE GIMZPOOL RECORDS IN THE OLDCSI DATA SET.**

**Explanation:** During CONVERT processing, SMP/E found more than one GIMZPOOL record in the OLDCSI data set. This situation may occur if you had run a command outside of SMP/E (such as REPRO) to copy information into an existing zone. This is an irregularity in the CSI.

**System Action:** SMP/E ignores the duplicate GIMZPOOL record, and CONVERT processing continues.

**Programmer Response:** Because this situation is an irregularity in the CSI, you may want to check the records in the OLDCSI data set using the Access Method Services PRINT command. If necessary, you can clean up the OLDCSI data set, delete the CSI that was used for conversion, allocate a new CSI, and rerun the CONVERT job.

**GIM69031v SMP/E COULD NOT ADD SYSLIB aaaaaaaa TO bbbbbbbb ccccccc BECAUSE THE MAXIMUM NUMBER OF SYSLIB SUBENTRIES WOULD HAVE BEEN EXCEEDED.**

**Explanation:**

- aaaaaaaa—SYSLIB name
- bbbbbbbb—entry type (LMOD, DLIB, or element)
- ccccccc—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.

- LMOD entries may contain one or two SYSLIB subentries.
- DLIB 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.
```

**GIM69032v SOURCEID aaaaaaaa HAS ALREADY BEEN SPECIFIED ON EITHER THE SOURCEID OR EXSRCID OPERAND.**

**Explanation:**

- aaaaaaaa—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.

**GIM69033v LMOD *aaaaaaaa* WAS NOT RENAMED BY SYSMOD *bbbbbbb* BECAUSE *aaaaaaaa* ALREADY EXISTS IN THE SPECIFIED *ccccccc* LIBRARY.**

**Explanation:**

- *aaaaaaaa*—element name
- *bbbbbbb*—SYSMOD ID
- *ccccccc*—ddname of the library.

There was no entry for the indicated LMOD in the target or distribution zone, but the LMOD 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 element is in the library.

- If it was added, moved, or renamed outside of SMP/E, reconstruct the entry in the appropriate zones.
- If the entry was deleted by UCLIN, decide whether to reconstruct the entry or delete the element from the library.

Then rerun the job.

**GIM69142v PROCESSING FOR ZONE *aaaaaaaa* IN DATA SET *bbbbbbb* FAILED BECAUSE LSR WAS SPECIFIED AND THE LARGEST BUFFER SIZE IN THE CURRENT POOL WAS EXCEEDED.**

**Explanation:**

- *aaaaaaaa*—name of the zone to be opened
- *bbbbbbb*—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.

**GIM69143v SMP/E SUCCESSFULLY OPENED DATA SET *aaaaaaaa* WITHOUT LSR TO COMPLETE COMMAND PROCESSING.**

**Explanation:**

- *aaaaaaaa*—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.



---

## Chapter 3. CBIPO Dialogs—Batch Messages

This chapter explains the format of the messages for the CBIPO dialog batch programs and lists these messages in alphanumeric order.

---

### Message Format

The batch messages for the CBIPO dialogs in SMP/E Release 6 are in the format `CIDMnnnv yy text`.

- *nnn* is the message number.
  - *v* is the severity code, as follows:

I	Informational messages	(return code = 00)
W	Warning messages	(return code = 04)
E	Error messages	(return code = 08)
S	Severe error messages	(return code = 12)
T	Terminating error messages	(return code = 16 or 20).
  - *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

- CIDM002I**    *numerror* ERRORS WERE DETECTED DURING PROCESSING.  
**Explanation:** *numerror* indicates the number of errors that were detected during processing.  
**Dialog Action:** None.  
**System Programmer Response:** Review any preceding error messages.
- CIDM011I**    IDCAMS SUCCESSFULLY RENAMED AND CATALOGED THE DATA SET.  
**Explanation:** The data set was successfully renamed and cataloged by IDCAMS.  
**Dialog Action:** None.  
**System Programmer Response:** None.
- CIDM014S**    A REQUIRED DATA SET NAME WAS NOT SPECIFIED.  
**Explanation:** A required data set name was not specified on the ALLOCATE keyword.  
**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.  
**System Programmer Response:** Using the tailor function of the dialogs, supply the required data set name, then regenerate the JCL and resubmit the job stream.  
**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.
- CIDM015S**    THE DATA SET WAS NOT CATALOGED BECAUSE THE VOLSER WAS NOT SPECIFIED.  
**Explanation:** For the ALLOCATE keyword, no volume serial was specified and the command did not ask for the data set to be cataloged. For the DELETE keyword, no volume serial was specified and the data set was not cataloged.  
**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.  
**System Programmer Response:** Using the tailor function of the dialogs, supply the required volume serial or specify CATALOG, then regenerate the JCL and resubmit the job stream.  
**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.
- CIDM016I**    THE REQUESTED DATA SET WAS ALREADY ALLOCATED. NOW COMPARING ATTRIBUTES:  
**Explanation:** In processing the ALLOCATE keyword, the dialogs found that the requested data set already existed on the specified volume. The dialogs are now comparing the attributes of the existing data set with the attributes specified on the allocation parameters. Additional messages may follow.  
**Dialog Action:** None.  
**System Programmer Response:** Review any subsequent messages.
- CIDM017W**    THE RECORD FORMAT FOR THIS DATA SET WAS NOT SPECIFIED.  
**Explanation:** The record format parameter (RECFM) for this data set was not specified.  
**Dialog Action:** The data set is allocated.  
**System Programmer Response:** Verify that the data set should not have a record format. If the record format should be specified, use the tailor function of the dialogs to supply the required information, then regenerate the JCL and resubmit the job stream.

**CIDM018W THE BLOCK SIZE FOR THIS DATA SET WAS NOT SPECIFIED.**

**Explanation:** The block size was not specified for this data set.

**Dialog Action:** The data set is allocated.

**System Programmer Response:** Verify that the data set should not have a block size. If the block size should be specified, use the tailor function of the dialogs to supply the required information, then regenerate the JCL and resubmit the job stream.

**CIDM019S AN ALLOCATION PARAMETER (TRKS OR CYLS) MUST BE SPECIFIED.**

**Explanation:** No allocation parameter was specified. Allocation in tracks (TRKS) or cylinders (CYLS) is required.

**Dialog Action:** The data set is not allocated. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Using the tailor function of the dialogs, specify either TRKS or CYLS for the data set, then regenerate the JCL and resubmit the job stream.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

**CIDM020S THE PRIMARY ALLOCATION FOR THIS DATA SET MUST BE SPECIFIED.**

**Explanation:** No value was specified for the primary space allocation parameter. This information is required.

**Dialog Action:** The data set is not allocated. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Using the tailor function of the dialogs, specify a value for the primary space allocation parameter for the data set, then regenerate the JCL and resubmit the job stream.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

**CIDM021S THE SECONDARY ALLOCATION FOR THIS DATA SET MUST BE SPECIFIED.**

**Explanation:** No value for the secondary space allocation parameter was specified. This information is required.

**Dialog Action:** The data set is not allocated. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Using the tailor function of the dialogs, specify a value for the secondary space allocation parameter for the data set, then regenerate the JCL and resubmit the job stream.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

**CIDM022S DIRECTORY BLOCKS MUST BE SPECIFIED FOR PARTITIONED DATA SETS.**

**Explanation:** The data set organization for this data set is partitioned, but no value was specified for the directory blocks allocation parameter. This information is required.

**Dialog Action:** The data set is not allocated. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Using the tailor function of the dialogs, supply a value for the directory blocks allocation parameter for the data set, then regenerate the JCL and resubmit the job stream.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

**CIDM023S DIRECTORY BLOCKS ARE NOT ALLOWED FOR SEQUENTIAL DATA SETS.**

**Explanation:** The data set organization for this data set is sequential, but a value was specified for the directory blocks allocation parameter.

**Dialog Action:** The data set is not allocated. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Using the tailor function of the dialogs, remove the value for the directory blocks allocation parameter for the data set, then regenerate the JCL and resubmit the job stream.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

**CIDM024W THE LRECL WAS NOT SPECIFIED FOR THIS DATA SET.**

**Explanation:** The logical record length (LRECL) value was not specified for this data set.

**Dialog Action:** The data set is allocated.

**System Programmer Response:** Verify that the data set should not have an LRECL specified. If the LRECL should be specified, use the tailor function of the dialogs to supply the required information, then regenerate the JCL and resubmit the job stream.

**CIDM025I DYNAMIC ALLOCATION WAS SUCCESSFUL FOR THE DATA SET.**

**Explanation:** Dynamic allocation of the data set was successful.

**Dialog Action:** The data set was allocated.

**System Programmer Response:** None.

**CIDM027S THE EXISTING DATA SET IS PARTITIONED.**

**Explanation:** This message may follow message CIDM016I. The organization of the existing data set is partitioned. However, the allocation parameters indicated that the data set should be sequential.

**Dialog Action:** The data set is not allocated. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Verify that the existing data set is allocated correctly. If it is, use the tailor function of the dialogs to change the value for the data set organization parameter for the data set, then regenerate the JCL and resubmit the job stream.

**CIDM028S THE EXISTING DATA SET IS SEQUENTIAL.**

**Explanation:** This message may follow message CIDM016I. The organization of the existing data set is sequential. However, the allocation parameters indicated that the data set should be partitioned.

**Dialog Action:** The data set is not allocated. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Verify that the existing data set is allocated correctly. If it is, use the tailor function of the dialogs to change the value for the data set organization parameter for the data set, then regenerate the JCL and resubmit the job stream.

**CIDM029S THE EXISTING DATA SET HAS FIXED RECORD FORMAT.**

**Explanation:** This message may follow message CIDM016I. The record format of the existing data set is fixed (RECFM = F\*). This does not match the record format specified on the allocation parameters.

**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM030S THE EXISTING DATA SET HAS A VARIABLE RECORD FORMAT.**

**Explanation:** This message may follow message CIDM016I. The record format of the existing data set is variable (RECFM=V\*). This does not match the record format specified on the allocation parameters.

**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM031S THE EXISTING DATA SET HAS AN UNDEFINED RECORD FORMAT.**

**Explanation:** This message may follow message CIDM016I. The record format of the existing data set is undefined (RECFM=U). This does not match the record format specified on the allocation parameters.

**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM032S THE EXISTING DATA SET HAS AN UNKNOWN RECORD FORMAT.**

**Explanation:** This message may follow message CIDM016I. The existing data set either has an unknown record format or no record format at all, but a record format was specified on the allocation parameters.

**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM033S THE EXISTING DATA SET HAS A BLOCKED RECORD FORMAT.**

**Explanation:** This message may follow message CIDM016I. The record format of the existing data set is blocked (RECFM=\*B). This does not match the record format specified on the allocation parameters.

**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM034S THE EXISTING DATA SET DOES NOT HAVE A BLOCKED RECORD FORMAT.**

**Explanation:** This message may follow message CIDM016I. The record format of the existing data set is not blocked. However, a blocked record format (RECFM=\*B) was specified on the allocation parameters.

**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM035S THE EXISTING DATA SET HAS A SPANNED RECORD FORMAT.**

**Explanation:** This message may follow message CIDM016I. The record format of the existing data set is spanned. This does not match the record format specified on the allocation parameters.

**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM036S THE EXISTING DATA SET DOES NOT HAVE A SPANNED RECORD FORMAT.**

**Explanation:** This message may follow message CIDM016I. The record format of the existing data set is not spanned. However, a spanned record format was specified on the allocation parameters.

**Dialog Action:** Module CIXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM037S THE EXISTING DATA SET HAS ANSI CONTROL CHARACTERS.**

**Explanation:** This message may follow message CIDM016I. The existing data set contains ANSI carriage control characters. However, these were not specified on the allocation parameters.

**Dialog Action:** Module CIXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM038S THE EXISTING DATA SET IS MISSING ANSI CONTROL CHARACTERS.**

**Explanation:** This message may follow message CIDM016I. The existing data set does not contain ANSI carriage control characters. However, these were specified on the allocation parameters.

**Dialog Action:** Module CIXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM039W THE BLOCK SIZE OF THE EXISTING DATA SET IS DIFFERENT FROM THE SPECIFIED BLOCK SIZE.**

**Explanation:** This message may follow message CIDM016I. The block size of the existing data set is different from the block size specified on the allocation parameters.

**Dialog Action:** None.

**System Programmer Response:** Check whether the block size of the existing data set is correct. If it is, use the tailor function of the dialogs to change the value for the block size for the data set, then regenerate the JCL and resubmit the job stream.

**CIDM040S AN ERROR OCCURRED WHILE THE DATA SET WAS BEING RENAMED OR CATALOGED. IDCAMS OUTPUT FOLLOWS:**

**Explanation:** An unrecoverable error occurred while a data set was being renamed or cataloged. IDCAMS diagnostic output follows this message.

**Dialog Action:** Module CIXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Fix the problem and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM041S THE RECORD LENGTH OF THE EXISTING DATA SET IS DIFFERENT FROM THE SPECIFIED RECORD LENGTH.**

**Explanation:** This message may follow message CIDM016I. The record length of the existing data set is different from the record length specified on the allocation parameters.

**Dialog Action:** Module CIXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record length of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

- 
- CIDM042I NO INCONSISTENCIES WERE FOUND WITH THE EXISTING DATA SET.**  
**Explanation:** This message may follow message CIDM016I. The existing data set matches the allocation parameters specified.  
**Dialog Action:** None.  
**System Programmer Response:** None.
- CIDM043I NO HIGH-LEVEL QUALIFIER WAS SPECIFIED. THE DEFAULT OF "CIDTEMP" WAS USED.**  
**Explanation:** The dialogs expected a data set high-level qualifier as an input parameter but none was specified.  
**Dialog Action:** Processing continues. The default parameter CIDTEMP is used.  
**System Programmer Response:** The high-level qualifier should always be specified in all jobs generated by the dialogs, so the default should never be used.  
**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.  
If the JCL generated by the dialogs was not changed, save the JCL that contains this step and contact the IBM Support Center.
- CIDM044W AN EXTRA "ALLOCATE" KEYWORD WAS IGNORED.**  
**Explanation:** The ALLOCATE keyword was specified more than once. The second occurrence was ignored.  
**Dialog Action:** Processing continues.  
**System Programmer Response:** Only one keyword should ever be specified in jobs generated by the dialogs.  
**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.  
If the JCL generated by the dialogs was not changed, save the JCL that contains the step and contact the IBM Support Center.
- CIDM045W AN EXTRA "DELETE" KEYWORD WAS IGNORED.**  
**Explanation:** The DELETE keyword was specified more than once. The second occurrence was ignored.  
**Dialog Action:** Processing continues.  
**System Programmer Response:** Only one keyword should ever be specified in jobs generated by the dialogs.  
**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.  
If the JCL generated by the dialogs was not changed, save the JCL that contains the step and contact the IBM Support Center.
- CIDM046W THE DATA SET WAS NOT DELETED BECAUSE IT WAS NOT FOUND ON THE SPECIFIED VOLUME.**  
**Explanation:** The data set that the dialogs were attempting to clean up could not be found on the volume specified. As a result, the data set was not deleted.  
**Dialog Action:** None.  
**System Programmer Response:** None.
- CIDM047I THE DATA SET WAS SUCCESSFULLY DELETED.**  
**Explanation:** The data set was deleted from the volume.  
**Dialog Action:** None.  
**System Programmer Response:** None.
-

**CIDM048S THE DATA SET WAS NOT DELETED BECAUSE EITHER THE VOLUME WAS NOT MOUNTED OR AN I/O ERROR OCCURRED.**

**Explanation:** The data set was not deleted because either the volume containing the data set was not mounted or a permanent I/O error occurred during processing.

**Dialog Action:** The data set is not deleted. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Check whether the volume is mounted and if the specified VOLSER is correct for the device type. Fix the problem and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM049S THE DATA SET WAS NOT DELETED BECAUSE AN ERROR OCCURRED DURING DEALLOCATION.**

**Explanation:** The allocation parameters specified a data set disposition of DELETE for the data set. However, an error during the deallocation process prevented the data set from being deleted.

**Dialog Action:** The data set is not deleted.

**System Programmer Response:** See the messages accompanying this message to help diagnose the problem. Once the problem is fixed, consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM050S THE DATA SET COULD NOT BE ALLOCATED WITH DISP=(OLD,DELETE).**

**Explanation:** A data set was requested with a disposition of (old,delete). However, the dialogs could not allocate the data set with that disposition.

**Dialog Action:** The data set is not allocated and therefore cannot be deleted.

**System Programmer Response:** See the messages accompanying this message to help diagnose the problem. Once the problem is fixed, consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM051S THE DATA SET HAS MACHINE-READABLE CHARACTERS, BUT RECFM=\*M WAS NOT SPECIFIED.**

**Explanation:** This message may follow message CIDM016I. The data set contains printer-ready data with machine-readable carriage control characters in the first byte of each line. However, the record format parameter (RECFM) on the allocation statement does not specify \*M as part of the record format parameters.

**Dialog Action:** None.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM052S RECFM=\*M WAS SPECIFIED, BUT THE DATA SET DOES NOT CONTAIN MACHINE-READABLE CHARACTERS.**

**Explanation:** This message may follow message CIDM016I. The record format parameter (RECFM) on the allocation statement specifies that the data set contains machine-readable carriage control characters in byte 1 of the print line. However, no machine-readable carriage control characters were found in the data set.

**Dialog Action:** None.

**System Programmer Response:** Either reallocate the existing data set or use the tailor function of the dialogs to change the record format of the data set to match the existing data set. Then regenerate the JCL and resubmit the job stream.

**CIDM054S THE CONTROL STATEMENT WAS IGNORED BECAUSE THE INPUT STREAM CONTAINED AN UNKNOWN KEYWORD.**

**Explanation:** The dialogs did not recognize a keyword in the input stream. As a result, the control statement was ignored. The line containing the unknown keyword precedes this message, with a pointer to the beginning of the unknown keyword.

**Dialog Action:** The control statement is ignored. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** The dialogs should not generate jobs that use unknown keywords.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

If the JCL generated by the dialogs was not changed, save the JCL containing the step and contact the IBM Support Center.

**CIDM055S THE CONTROL STATEMENT WAS IGNORED BECAUSE THE INDICATED CHARACTER WAS NOT A NUMBER FROM 0-9.**

**Explanation:** The character specified is not a number from 0 to 9. As a result, the control statement was ignored. The line containing the invalid character precedes this message, with a pointer to the keyword value that has the invalid character.

**Dialog Action:** The control statement is ignored. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** The dialogs should not generate jobs that specify invalid parameters.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

If the JCL generated by the dialogs was not changed, save the JCL containing the step and contact the IBM Support Center.

**CIDM056S THE CONTROL STATEMENT WAS IGNORED BECAUSE THE INDICATED PARAMETER VALUE IS TOO LONG.**

**Explanation:** The parameter listed above contains a value that is too long. As a result, the control statement was ignored. The line with the parameter precedes this message, with a character pointing to the keyword value that is too long.

**Dialog Action:** The control statement is ignored. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** The dialogs should not generate jobs that specify invalid parameters.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

If the JCL generated by the dialogs was not changed, save the JCL containing the step and contact the IBM Support Center.

**CIDM057S THE VTOC OF THE VOLUME DOES NOT CONTAIN A FORMAT-1 DSCB FOR THE DATA SET.**

**Explanation:** The volume table of contents (VTOC) does not contain a format-1 DSCB for the data set. The Data Facility Product (DFP) RENAME macro returned a status code of 1.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Module CIDXGCND will only try to rename data sets that it previously allocated. It should not be possible for the format-1 DSCB to be missing. Save the JCL containing the step and the output from the job and contact the IBM Support Center.

**Note:** This message is generally not issued, even if the JCL generated by the dialogs was changed.

**CIDM058S THE SPECIFIED PASSWORD WAS NOT CORRECT.**

**Explanation:** During an attempt to rename a data set, a password was required but the correct password was not specified. The Data Facility Product (DFP) RENAME macro returned a status code of 2.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Module CIDXGCND will only try to rename data sets that it previously allocated. It does not specify a password so this message should not occur. Save the JCL containing the step and the output from the job and contact the IBM Support Center.



**Note:** This message is generally not issued, even if the JCL generated by the dialogs was changed.

**CIDM059S A DATA SET WITH THE NEW NAME ALREADY EXISTS.**

**Explanation:** After allocating a data set with a temporary high-level qualifier, the dialogs could not rename the data set because a data set with the new name already exists on the same volume as the data set to be renamed. The Data Facility Product (DFP) RENAME macro returned a status code of 3.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Fix the problem and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM060S EITHER A PERMANENT I/O ERROR WAS ENCOUNTERED OR AN INVALID FORMAT-1 DSCB WAS FOUND.**

**Explanation:** A severe error was encountered on a volume while attempting to rename a data set. The Data Facility Product (DFP) RENAME macro returned a status code of 4.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Fix the problem and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM061S THE DIALOGS COULD NOT VERIFY THAT THE VOLUME WAS MOUNTED AND NO DEVICE WAS AVAILABLE.**

**Explanation:** During an attempt to rename a data set, the dialogs could not verify that the volume was mounted, and no device was available for mounting the volume. The Data Facility Product (DFP) RENAME macro returned a status code of 5.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Fix the problem and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM062S THE OPERATOR WAS UNABLE TO MOUNT THE VOLUME.**

**Explanation:** The operator was unable to mount this volume. For Mass Storage Systems (MSS), a volume mount failure occurred. For a JES3-managed virtual volume, JES3 would not allow the volume to be mounted. The Data Facility Product (DFP) RENAME macro returned a status code of 6.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Fix the problem and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM063S THE DATA SET COULD NOT BE RENAMED BECAUSE IT WAS ALREADY IN USE.**

**Explanation:** After allocating a data set with a temporary high-level qualifier, module CIDXGCND attempted to rename the data set to its proper name. However, the program determined that the data set was already in use. The Data Facility Product (DFP) RENAME macro returned a status code of 7.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Fix the problem and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM064S THE USER IS NOT AUTHORIZED TO RENAME THE DATA SET.**

**Explanation:** After allocating a data set with a temporary high-level qualifier, module CIDXGCND attempted to rename the data set to its proper name. The user does not have sufficient authority to rename the data set. The Data Facility Product (DFP) RENAME macro returned a status code of 8.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Obtain the authority and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM065S RENAME PROCESSING FAILED. RETURN CODE=*rc*, STATUS CODE=*status*.**

**Explanation:** After allocating a data set with a temporary high-level qualifier, module CIDXGCND attempted to rename the data set to its proper name. The Data Facility Product (DFP) RENAME macro returned the following information:

- *rc* – return code
- *status* – status code.

**Dialog Action:** The data set is not renamed. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** See the message accompanying this message to help diagnose the problem. Once the problem is fixed, consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM066S SVC 99 DYNAMIC ALLOCATION FAILED. RETURN CODE=*rc*, REASON CODE=*reason*.**

**Explanation:** Module CIDXGCND issued an SVC 99 to dynamically allocate the specified data set. The dynamic allocation function returned the following:

- *rc* – return code (decimal)
- *reason* – reason code (decimal).

Additional messages from SVC 99 (dynamic allocation) will follow this message.

**Dialog Action:** The data set is not allocated. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Consult the SVC 99 messages to determine how to fix the problem and the appropriate CBIPO dialogs installation guide to determine which job to resubmit.

**CIDM067E LOCATE RETURN CODE=*rc*. PROBABLE CATALOG ERROR.**

**Explanation:** No volume serial number was specified for the ALLOCATE or DELETE keywords. Module CIDXGCND received a return code of *rc* from the LOCATE macro when attempting to find out whether the data set is already cataloged. This message should be accompanied by message CIDM015S.

**Dialog Action:** Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** See the system programmer response for message CIDM015S.

**CIDM068I THE REQUESTED DATA SET IS NOT CATALOGED.**

**Explanation:** The DELETE function was specified, but no volume serial number was specified and the data set is not cataloged.

**Dialog Action:** None.

**System Programmer Response:** Because the dialogs only generate jobs that specify volume serial numbers for this type of deletion, this message should not occur.

**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

If the JCL generated by the dialogs was not changed, save the JCL containing the step and contact the IBM Support Center.

**CIDM069I THE VTOC IS BEING UPDATED TO PREPARE FOR DELETING THE TEMPORARY HIGH-LEVEL QUALIFIER.**

**Explanation:** The DELETE function was requested for a data set but there is an enqueue for the data set (probably for a data set with the same name on a different volume). The program will attempt to update the volume table of contents (VTOC) prior to adding a temporary high-level qualifier to a data set in preparation for deletion.

**Dialog Action:** None.

**System Programmer Response:** None.

**CIDM070E RESERVE PROCESSING FAILED FOR *vo/ser*. THE VOLUME WAS NOT AVAILABLE. RETURN CODE=4.**

**Explanation:** During DELETE processing, a Data Facility Product (DFP) RESERVE macro was issued for volume *vo/ser* in preparation to update the volume table of contents (VTOC). RESERVE failed with a return code of 4, indicating that the volume is not available.

**Dialog Action:** The data set is not deleted. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Check whether another user or job has reserved the volume. When you have fixed the problem, consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM071E RESERVE PROCESSING FAILED FOR *vo/ser*. RETURN CODE=*rc*.**

**Explanation:** During DELETE processing, a Data Facility Product (DFP) RESERVE macro was issued for volume *vo/ser* in preparation to update the volume table of contents (VTOC). RESERVE failed with an unexpected return code of *rc*.

**Dialog Action:** The data set is not deleted. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Check whether another user or job has reserved the volume. Verify what volume the data set is actually on. When you have fixed the problem, consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM072S AN EOF OCCURRED IN THE VTOC BEFORE A FORMAT-1 DSCB WAS FOUND.**

**Explanation:** During DELETE processing for a data set, module CIDXGCND is attempting to find the format-1 data set control block (DSCB) for the data set in the volume table of contents (VTOC). The end-of-file (EOF) was encountered reading the VTOC before the desired DSCB was found.

**Dialog Action:** The data set is not deleted. Module CIDXGCND flushes the rest of the steps in the job.

**System Programmer Response:** Because the dialogs had already determined that a format-1 DSCB was available, this message should never occur. If it does, save the JCL containing the step and the output of the job and contact the IBM Support Center.

**Note:** This message is generally not issued, even if the JCL generated by the dialogs was changed.

**CIDM073I THE VSAM DATA SET WAS NOT RENAMED. A REQUIRED PARAMETER WAS NOT SPECIFIED.**

**Explanation:** When allocating a new data set, module CIDXGCND will first allocate the data set with a temporary high-level qualifier. IDCAMS is usually called later to rename the data set to the proper name. However, if any of the required parameters are missing, the Data Facility Product (DFP) RENAME macro is used instead of IDCAMS.

**Dialog Action:** None.

**System Programmer Response:** Because the dialogs control the parameters of jobs being generated, this message should never be issued. If it is, save the JCL containing the step and contact the IBM Support Center.

- CIDM074S**    **LOCATE FAILED FOR DATA SET *dsname* ON VOLUME *volser*. RETURN CODE=*rc*.**
- Explanation:** This message generally follows message CIDM040S. The message is issued after a data set is allocated and renamed, but the LOCATE macro does not return the expected information about the data set. The dialogs issued a LOCATE command for data set *dsname* on volume *volser*. Processing failed with return code of *rc*.
- Dialog Action:** Module CIDXGCND flushes the rest of the steps of the job.
- System Programmer Response:** Refer the ICKDSF output to help diagnose the problem. Fix the problem and consult the appropriate CBIPO dialogs installation guide to find out which job to resubmit.
- CIDM076I**    **DATA SET *dsname* WAS ALLOCATED.  
VOLUME=*volser*, NUMBER OF TRACKS=*number*.**
- Explanation:** Module CIDXGDYN allocated data set *dsname* on volume *volser* for *number* tracks to account for disk space not used in VSAM allocations.
- Dialog Action:** None.
- System Programmer Response:** None.
- CIDM078I**    **DATA SET *dsname* WAS NOT ALLOCATED.  
THE INDICATED VSAM DATA SET USED UP THE ENTIRE ALLOCATION.**
- Explanation:** Module CIDXGDYN did not allocate *dsname*. No dummy data set was necessary because the VSAM data set used up the entire allocation.
- Dialog Action:** None.
- System Programmer Response:** None.
- CIDM111S**    **OPEN FAILED FOR ISPF TABLE *table\_name*.**
- Explanation:** An ISPF TBOPEN service request was issued to open table *table\_name*. The OPEN request failed.
- Dialog Action:** Module CIDXGDGH abends, and the redistribution order is not usable.
- System Programmer Response:** See the system programmer response for message CIDM135I.
- CIDM112S**    **DATA SET *dsn* IS NOT CATALOGED.**
- Explanation:** No VOLSER was specified for a data set during the data gathering phase of redistribution, and the data set is not cataloged. Since the tailor function requires this for every data set, this should not be possible.
- Dialog Action:** Module CIDXGDGH abends, and the redistribution order is not usable.
- System Programmer Response:** This message is generally not issued unless changes were made to a configuration without using the CBIPO dialogs. Use the tailor function to verify that the data set has a VOLSER and see the system programmer response for message CIDM135I.
- CIDM113S**    **VOLUME *volser* IS NOT ONLINE.**
- Explanation:** The volume *volser* is not online.
- Dialog Action:** Module CIDXGDGH abends, and the redistribution order is not usable.
- System Programmer Response:** See the system programmer response for message CIDM135I.
- CIDM114S**    **DATA SET *dsname* IS NOT ON VOLUME *volser*.**
- Explanation:** The data set *dsname* could not be found on volume *volser*.
- Dialog Action:** Module CIDXGDGH abends, and the redistribution order is not usable.
- System Programmer Response:** See the system programmer response for message CIDM135I. Change the information in the configuration you wish to redistribute.

- CIDM115S** *dsname* CANNOT BE REDISTRIBUTED BECAUSE ITS DSORG IS *type*.  
**Explanation:** Data set *dsname* has a data set organization of *type*, which cannot be read. As a result, the data set cannot be redistributed.  
**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.  
**System Programmer Response:** See the system programmer response for messages CIDM135I. Using the tailor function of the dialogs, change the data set so that it will not be redistributed.
- CIDM116S** THE DSORG FOR *dsname* IS PO, BUT THE CONFIGURATION TABLE SPECIFIES PS.  
**Explanation:** Data set *dsname* is a partitioned data set (DSORG=PO). However, the dialog's configuration table indicates that this should be a sequential data set (DSORG=PS).  
**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.  
**System Programmer Response:** See the system programmer response for message CIDM135I. Using the tailor function of the dialogs, change the data set so that the data set organization is correct.
- CIDM117S** THE DSORG FOR *dsname* IS PS, BUT THE CONFIGURATION TABLE SPECIFIES PO.  
**Explanation:** Data set *dsname* is a sequential data set (DSORG=PS). However, the dialog's configuration table indicates that this should be a partitioned data set (DSORG=PO).  
**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.  
**System Programmer Response:** See the system programmer response for message CIDM135I. Using the tailor function of the dialogs, change the data set so that the data set organization is correct.
- CIDM118E** OPEN FAILED FOR SEQUENTIAL DATA SET *dsname*.  
**Explanation:** The sequential data set *dsname* could not be opened. Therefore, the minimum number of blocks required and the amount of tape needed cannot be determined.  
**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.  
**System Programmer Response:** See the system programmer response for message CIDM135I.
- CIDM119E** OPEN FAILED FOR THE DIRECTORY OF *dsname*.  
**Explanation:** The directory for data set *dsname* could not be opened in order to determine the minimum number of directory blocks required. Access to the directory is also necessary to read each member of the data set.  
**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.  
**System Programmer Response:** See the system programmer response for message CIDM135I.
- CIDM120E** OPEN FAILED FOR PARTITIONED DATA SET *dsname*.  
**Explanation:** The data portion of partitioned data set *dsname* could not be opened. Therefore, the minimum number of blocks required and the amount of tape needed cannot be determined.  
**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.  
**System Programmer Response:** See the system programmer response for message CIDM135I.
- CIDM121E** AN EOF WAS ENCOUNTERED ON THE DIRECTORY OF *dsname*.  
**Explanation:** The end-of-file (EOF) of a partitioned data set's (*dsname*) directory was reached before the required "end of directory" indicator. This indicates there is some sort of internal error within the directory.  
**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.  
**System Programmer Response:** Because of the internal structure of directories, this should never happen. If it does, save the JCL containing the step and the output from the job and contact the IBM Support Center. When the problem is resolved, see the system programmer response for message CIDM135I.  
**Note:** This message is generally not issued, even if the JCL generated by the dialogs was changed.

**CIDM122W** ALIAS *alias\_name* DOES NOT POINT TO A NON-ALIAS. TTR=*ttr*.

**Explanation:** This message indicates that an alias member exists in a data set's directory, but there is no non-alias member with the same TTR. This is normally not possible or desirable but it could happen if a non-alias member existed at one time and was deleted but at least one of its aliases was not also deleted.

**Dialog Action:** Module CIXGDGH will not read the member.

**System Programmer Response:** No response is required. If you wish to delete the member, you may do so with a function such as ISPF/PDF's UTILITY option.

**CIDM123W** NO MEMBERS WERE FOUND IN *dsname*.

**Explanation:** This message is issued by module CIXGDGH. The dialogs tried to read a partitioned data set (*dsname*), but no members were found. The data set will not be copied either to tape or DASD during redistribution.

**Dialog Action:** Processing continues.

**System Programmer Response:** None.

**CIDM128I** DATA SET *current* OF *final*: PROCESSING IS COMPLETE FOR *dsname* ON VOLUME *volser*.

**Explanation:** This message is issued by module CIXGDGH to indicate that processing has completed with data set *dsname* and that this data set is number *current* out of *final* data sets in the configuration.

**Dialog Action:** Processing continues.

**System Programmer Response:** None.

**CIDM129I** THE FIRST PASS OF PROCESSING HAS STARTED FOR *usrdsn* ON *usrvol*.

**Explanation:** This message is issued by module CIXGDGH. The program is performing the first pass on data set *usrdsn* on volume *usrvol*. The first pass checks to make sure the data set is on the volume as specified in the configuration.

**Dialog Action:** Processing continues.

**System Programmer Response:** None.

**CIDM130I** THE FIRST PASS OF PROCESSING IS COMPLETE. PROCESSING CONTINUES WITH BLOCK READS.

**Explanation:** This message is issued by module CIXGDGH after it has completed its first pass without any errors.

**Dialog Action:** Processing continues with the second pass, which reads the blocks of the data sets.

**System Programmer Response:** None.

**CIDM131I** PROCESSING HAS REACHED THE END OF THE CONFIGURATION DEFINITION.

**Explanation:** This message is issued by module CIXGDGH. The program has reached the end of the configuration.

**Dialog Action:** Processing ends.

**System Programmer Response:** None.

**CIDM132E** SVC 99 DYNAMIC ALLOCATION FAILED FOR DDNAME *ddname*. RETURN CODE=*rc*, REASON CODE=*reason*.

**Explanation:** Module CIXGDGH called SVC 99 to allocate a data set to ddname *ddname* and received the following information back:

- *rc*—return code from SVC 99 (decimal)
- *reason*—reason code from SVC 99 (hexadecimal).

Additional messages from SVC 99 follow this message to help you diagnose the problem.

**Dialog Action:** Processing of the data set ends. Module CIXGDGH abends, and the redistribution order is not usable.

**System Programmer Response:** See the system programmer response for message CIDM135I.

**CIDM133E OPEN FAILED FOR VSAM DATA SET *dsname*.**

**Explanation:** The VSAM data set *dsname* could not be opened; therefore, the amount of tape the data set might need cannot be determined.

**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.

**System Programmer Response:** See the system programmer response for message CIDM135I.

**CIDM134E AN ERROR OCCURRED WHILE THE DIALOGS WERE READING FILE *dsname*.**

**Explanation:** An error was encountered during an attempt to read a block of a VSAM data set *dsname*. A non-zero return code was returned by the Data Facility Product (DFP) GET macro.

**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.

**System Programmer Response:** System messages in the job output can help you diagnose the problem. See the system programmer response for message CIDM135I to find out what actions to take once the problem is resolved.

**CIDM135I ERRORS CAUSED THE DATA GATHERING PROCESS TO FAIL.  
USE THE DIALOGS TO DO THE FOLLOWING:**

1. DELETE THE REDISTRIBUTION ORDER FOR WHICH THE DATA IS BEING GATHERED.
2. FIX THE PROBLEMS.
3. REIDENTIFY AN ORDER TO BE REDISTRIBUTED.

**Explanation:** These messages are issued in a group by module CIXGDGH. During the data gathering operation of the redistribution process, errors occurred that prevented the process from continuing.

**Dialog Action:** Module CIXGDGH abends, and the redistribution order is not usable.

**System Programmer Response:** Using the CBIPO dialogs, do the following:

1. Delete the redistribution order for which the data is being gathered.
2. Correct the problems identified in previous messages.
3. Reidentify the original configuration to be redistributed as a redistribution order.

**CIDM140I THE FOLLOWING LINES ARE BEING WRITTEN TO MEMBER *member* IN *dsname*:**

**Explanation:** Module CIXGUPD is echoing the lines that are being placed in the *member* member of the data set *dsname*.

**Dialog Action:** Processing continues.

**System Programmer Response:** None.

**CIDM141W MEMBER *member* WAS NOT UPDATED BECAUSE IT WAS NOT FOUND IN *dsname*.**

**Explanation:** Module CIXGUPD could not find the member it was asked to update and could not create the member.

**Dialog Action:** The member was not updated.

**System Programmer Response:** This message is normally an indication that the data set must be reallocated with more directory blocks in order to add a new member. Once the problem is resolved, see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM142I MEMBER *member* DOES NOT EXIST. THE DIALOGS WILL ATTEMPT TO CREATE THE MEMBER.**

**Explanation:** Module CIXGUPD could not find the member (*member*) it was asked to update. Instead, it will try to create the member.

**Dialog Action:** The program attempts to create the member.

**System Programmer Response:** None.

**CIDM143S MEMBER *member* WAS NOT CREATED. STOW RETURN CODE=*rc*.**

**Explanation:** Module CIDXGUPD could not find the member (*member*) it was asked to update and could not create the member. The Data Facility Product (DFP) STOW macro to add a new directory entry returned with a return code of *rc*.

**Dialog Action:** The member was not created.

**System Programmer Response:** This message is normally an indication that the data set must be reallocated with more directory blocks in order to add a new member. Once the problem is resolved, see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM144T SVC 99 DYNAMIC ALLOCATION FAILED FOR DDNAME *ddname*. RC=*rc*, EC=*long\_error*, IC=*long\_info*.**

**Explanation:** A permanent or temporary data set could not be allocated by module CIDXGUPD using SVC 99:

- *ddname* – *ddname* for which allocation was attempted
- *rc* – return code from SVC 99 (decimal)
- *long\_error* – SVC 99 error code (hexadecimal)
- *long\_info* – SVC 99 information code (hexadecimal).

**Dialog Action:** The data set is not allocated, and the member cannot be updated.

**System Programmer Response:** Fix the problem and see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM145S ZAP PROCESSING IS COMPLETE. RETURN CODE=*rc*. AMASPZAP OUTPUT FOLLOWS:**

**Explanation:** Module CIDXGUNZ invoked AMASPZAP (SUPERZAP) to update a load module but received a return code of *rc* from SUPERZAP. The output from SUPERZAP follows this message to aid in diagnosing the problem.

**Dialog Action:** The load module is not updated.

**System Programmer Response:** The output from SUPERZAP should help diagnose the problem. Fix the problem and see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**CIDM146S THE CIDXGUNZ WORK AREA HAS RUN OUT OF SPACE. MODULE *module* IS TOO BIG.**

**Explanation:** Module CIDXGUNZ attempted to update a load module (*module*) by first dumping the module using AMASPZAP and reading the output into storage. The storage buffer filled before all of the module could be read in. This should not be possible and is probably an indication that there is something wrong with the module the program is trying to update. The type of module that CIDXGUNZ is designed to update should never exceed the size of its buffer.

**Dialog Action:** The load module is not updated.

**System Programmer Response:** Because the program was designed to avoid problems such as this, the message should never be issued. If it is, save the JCL containing the step and the output of the job and contact the IBM Support Center. Once the problem is resolved, see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**Note:** This message is generally not issued, even if the JCL generated by the dialogs was changed.

**CIDM147S NO VALID INPUT WAS FOUND FROM SUPERZAP OUTPUT. THE FORMAT MAY HAVE CHANGED.**

**Explanation:** Module CIDXGUNZ attempted to update a load module by first dumping the module using AMASPZAP and reading the output into storage. The program found no output lines from AMASPZAP containing the contents of the module to be updated. This may indicate that the output from AMASPZAP has changed from the format that CIDXGUNZ expects.

**Dialog Action:** The load module is not updated.

**System Programmer Response:** Save the JCL containing the step and the output of the job and contact the IBM Support Center. Once the problem is resolved, see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.

**Note:** This message is generally not issued, even if the JCL generated by the dialogs was changed.



- CIDM148I** ENTRY *entryname* WAS NOT FOUND IN *module* **MODULE**.  
**Explanation:** Module CIDXGUNZ did not find entry *entryname* anywhere in module *module*. This probably indicates that IMS was not installed on the system that was redistributed.  
**Dialog Action:** The module is not updated.  
**System Programmer Response:** Verify that IMS is not installed on the system being redistributed. If IMS is installed, save the JCL containing the step and the output of the job and contact the IBM Support Center.
- CIDM149I** ZAP PROCESSING SUCCESSFULLY DELETED ENTRY *entryname* FROM MODULE *module*.  
**Explanation:** Module CIDXGUNZ was able to successfully remove all occurrences of entry *entryname* from module *module*.  
**Dialog Action:** The module is updated.  
**System Programmer Response:** None.
- CIDM150W** ZAP PROCESSING TO DELETE ENTRY *entryname* FAILED. RETURN CODE=*rc*.  
**Explanation:** Module CIDXGUNZ attempted to invoke the AMASPZAP (SUPERZAP) utility to remove entry *entryname* from a load module but SUPERZAP issued non-zero return code *rc*. Because SUPERZAP was previously called to read the same module without problems, this may be an indication that the user does not have the authorization to update the data set containing the load module.  
**Dialog Action:** The load module is not updated.  
**System Programmer Response:** Check authorization. Fix the problem and see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.
- CIDM151W** *entryname* WAS FOUND IN *module* BUT NO ZAP CHANGES NEED TO BE MADE.  
**Explanation:** Module CIDXGUNZ found entry *entryname* in module *module* at least once, but after comparing the old module with what the program would replace, no differences were found. This is probably an indication that there is an error internal to the program.  
**Dialog Action:** The load module is not updated.  
**System Programmer Response:** Save the JCL containing the step and the output of the job and contact the IBM Support Center.  
**Note:** This message is generally not issued, even if the JCL generated by the dialogs was changed.
- CIDM152W** "*verb*" IS NOT A KNOWN VARIABLE.  
**Explanation:** Module CIDXGUNZ encountered an unknown variable keyword *verb* while parsing command input.  
**Dialog Action:** Processing continues.  
**System Programmer Response:** Because the dialogs control which keywords are generated in jobs, this situation should never occur.  
**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.  
 If the JCL generated by the dialogs was not changed, save the JCL containing the step and the output of the job and contact the IBM Support Center.
- CIDM153T** A REQUIRED VARIABLE WAS NOT SPECIFIED.  
**Explanation:** A required variable keyword was not specified in the command input.  
**Dialog Action:** The module is not updated.  
**System Programmer Response:** Fix the problem and see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.  
**Note:** This message is generally not issued unless the JCL job stream generated by the dialogs has been changed.

- 
- CIDM169I THE DIALOGS ARE PREPARING TO DEACTIVATE THE VTOC INDEX FOR VOLUME *vo/ser*.**  
**Explanation:** Module CIDXGCND is attempting to update the VTOC of volume *vo/ser* to delete a data set with an enqueue on it. It was determined that the volume has a VTOC index, and to avoid disastrous results, CIDXGCND will deactivate the index before updating the VTOC. The index will be reactivated and its information refreshed from the VTOC after the data set has been deleted.  
**Dialog Action:** Module CIDXGCND attempts to deactivate the VTOC index.  
**System Programmer Response:** None.
- CIDM170S AN ERROR OCCURRED DURING AN ATTEMPT TO DEACTIVATE THE INDEXED VTOC. RC=*rc*. ICKDSF OUTPUT FOLLOWS:**  
**Explanation:** Module CIDXGCND attempted to deactivate a VTOC index prior to updating a volume and received a return code of *rc* from the ICKDSF utility. The output from the utility follows this message.  
**Dialog Action:** The VTOC index was not deactivated. Module CIDXGCND flushes the rest of the steps of the job.  
**System Programmer Response:** See the ICKDSF output to help diagnose the problem. Fix the problem and see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.
- CIDM171I THE DIALOGS ARE PREPARING TO REACTIVATE THE VTOC INDEX FOR VOLUME *vo/ser*.**  
**Explanation:** Module CIDXGCND temporarily deactivated the VTOC index for volume *vo/ser* in order to update the VTOC. Before processing a different volume or before the program ends, the VTOC index will be reactivated and its information refreshed from the VTOC.  
**Dialog Action:** Module CIDXGCND attempts to reactivate the VTOC index.  
**System Programmer Response:** None.
- CIDM172W AN ERROR OCCURRED DURING AN ATTEMPT TO REACTIVATE THE VTOC INDEX. RC=*rc*. ICKDSF OUTPUT FOLLOWS:**  
**Explanation:** Module CIDXGCND temporarily deactivated the VTOC index for volume *vo/ser* in order to update the VTOC. An attempt to reactivate the index for the volume has failed. The ICKDSF utility has issued a return code of *rc*, and its output follows this message.  
**Dialog Action:** The VTOC index was not reactivated. The volume is still functional as a non-indexed volume. The index must be reactivated for it to function as an indexed volume. Module CIDXGCND flushes the rest of the steps of the job.  
**System Programmer Response:** See the ICKDSF output to help diagnose the problem. Fix the problem and see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.
- CIDM173S THE DATA SET WAS NOT DELETED BECAUSE MODULE CIDXGCND IS NOT AUTHORIZED.**  
**Explanation:** The CBIPO dialog module CIDXGCND must reside in an authorized system library in order to perform some of its functions. When the module began execution, the system determined it was not executing from an authorized system library. In this state, the program cannot delete a data set for which an enqueue has been issued. Therefore, the data set was not deleted.  
**Dialog Action:** Module CIDXGCND flushes the rest of the steps of the job.  
**System Programmer Response:** Authorize the library that module CIDXGCND resides in and see the appropriate CBIPO dialogs installation guide to find out which job to resubmit.
- CIDM174W MESSAGE *msgid* COULD NOT BE LOCATED.**  
**Explanation:** A CBIPO dialog module issued a request to display a message with the identifier *msgid*. However, the message could not be located.  
**Dialog Action:** Processing continues.  
**System Programmer Response:** Save the output of the job and contact the IBM Support Center.
-

**CIDM175W NO BLOCKS WERE READ FOR *dsname*.**

**Explanation:** No blocks were found in a VSAM data set *dsname* that was to be redistributed. The data set will not be copied either to tape or DASD during redistribution.

**Dialog Action:** Processing continues.

**System Programmer Response:** None.

**CIDM176S RECFM *recfm* FOR DATA SET *dsname* IS NOT SUPPORTED FOR REDISTRIBUTION.**

**Explanation:** The RECFM value (*recfm*) for data set *dsname* is not supported by the CBIPO dialogs for redistribution. As a result, the minimum number of blocks required and the amount of tape needed cannot be determined.

**Dialog Action:** Module CIDXGDGH abends, and the redistribution order is not usable.

**System Programmer Response:** Remove the data set from the original configuration. It cannot be redistributed with the CBIPO dialogs. See the system programmer response for message CIDM135I for further information.

**CIDM200S AN OPEN PARENTHESIS MUST FOLLOW THE "MODULES" KEYWORD.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAxx member from a SYS1.PARMLIB data set. An open parenthesis must follow the MODULES keyword according to the MVS/ESA\* SP Version 4 syntax rules for this member.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either add the missing parenthesis after the MODULES keyword, or delete the entire INCLUDE statement from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAxx to update or specify that no IEALPAxx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM201S THE "MODULES" KEYWORD MUST FOLLOW THE LIBRARY KEYWORD.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAxx member from a SYS1.PARMLIB data set. The MODULES keyword must follow the LIBRARY keyword according to the MVS/ESA SP Version 4 syntax rules for this member.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either add the MODULES keyword after the LIBRARY keyword, or delete the entire INCLUDE statement from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAxx to update or specify that no IEALPAxx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM202S A CLOSING PARENTHESIS MUST FOLLOW THE LIBRARY NAME.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAXx member from a SYS1.PARMLIB data set. A closing parenthesis must follow the library name of the LIBRARY keyword according to the MVS/ESA SP Version 4 syntax rules for this member.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either add the closing parenthesis after the library name, or delete the entire INCLUDE statement from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAXx to update or specify that no IEALPAXx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM203S A DATA SET NAME MUST BE SPECIFIED FOR THE "LIBRARY" KEYWORD.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAXx member from a SYS1.PARMLIB data set. A library name must be specified for the LIBRARY keyword according to the MVS/ESA SP Version 4 syntax rules for this member. The end-of-file was encountered before a library name could be read.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either add the missing LIBRARY keyword, or delete the entire INCLUDE statement from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAXx to update or specify that no IEALPAXx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM204S AN OPEN PARENTHESIS MUST FOLLOW THE "LIBRARY" KEYWORD.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAXx member from a SYS1.PARMLIB data set. An open parenthesis must follow the LIBRARY keyword according to the MVS/ESA SP Version 4 syntax rules for this member.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either add the missing parenthesis after the LIBRARY keyword, or delete the entire INCLUDE statement from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAXx to update or specify that no IEALPAXx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM205S THE "LIBRARY" KEYWORD MUST FOLLOW THE "INCLUDE" KEYWORD.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAXx member from a SYS1.PARMLIB data set. The LIBRARY keyword must follow the INCLUDE keyword according to the MVS/ESA SP Version 4 syntax rules for this member.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either add the missing LIBRARY keyword after the INCLUDE keyword, or delete the entire INCLUDE statement from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAXx to update or specify that no IEALPAXx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM206S THE "INCLUDE" KEYWORD IS MISSING.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAXx member from a SYS1.PARMLIB data set. Each statement of this member must begin with the INCLUDE keyword according to the MVS/ESA SP Version 4 syntax rules for this member, but something other than INCLUDE was found starting a statement.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either add the missing INCLUDE keyword, or delete or comment out the offending characters from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAXx to update or specify that no IEALPAXx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM207S THE END-OF-FILE WAS ENCOUNTERED BEFORE A CLOSING PARENTHESIS FOUND.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAXx member from a SYS1.PARMLIB data set. While reading module names from the MODULES keyword of a statement, the end-of-file was encountered before a closing parenthesis could be found. The MODULES keyword must be terminated with a closing parenthesis according to the MVS/ESA SP Version 4 syntax rules for this member.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either add the missing closing parenthesis to end the MODULES keyword, or delete the entire INCLUDE statement from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAXx to update or specify that no IEALPAXx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM208S NO MODULES WERE SPECIFIED FOR LIBRARY *library*.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAXx member from a SYS1.PARMLIB data set. No modules were specified in the MODULES keyword for the *library* library. According to the MVS/ESA SP Version 4 syntax rules for this member, at least one module must be specified for each library specified.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** The line number of the error will be found in accompanying message CIDM209S. Do one of the following:

- Either specify at least one module as a MODULES keyword, or delete the entire INCLUDE statement from the PARMLIB member. Then see the appropriate CBIPO dialogs installation guide to determine which job to resubmit.
- Use the tailor function to specify a different IEALPAXx to update or specify that no IEALPAXx member should be updated. Then regenerate and resubmit the installation job streams.

**CIDM209S A SYNTAX ERROR WAS DETECTED ON LINE *linenum* of member.**

**Explanation:** Module CIDXGUPD found a syntax error while reading an IEALPAXx member from a SYS1.PARMLIB data set. The error was detected on line *linenum* of member *member* of the SYS1.PARMLIB data set being updated.

**Dialog Action:** The step generates a return code of 12. The PARMLIB member is not updated.

**System Programmer Response:** Refer to the message issued after this message for instructions on how to proceed.

---

**CIDM210W MVS VERSION 4 MLPA SYNTAX IS BEING GENERATED BECAUSE THE DESIRED SYNTAX CANNOT BE DETERMINED.**

**Explanation:** Because the IEALPAxx PARMLIB member to be updated does not exist, module CIDXGUPD will create it. However, the module does not know which syntax rules to use:

- For MVS/XA\* and MVS/ESA systems:  
SYS1.LINKLIB A,B,C
- For MVS/ESA SP Version 4 systems:  
INCLUDE LIBRARY(SYS1.LINKLIB) MODULES(A,B,C)

**Dialog Action:** The PARMLIB member is created using the IEALPAxx member syntax for MVS/ESA SP Version 4 systems.

**System Programmer Response:** Verify that the MVS/ESA SP Version 4 syntax of this member is desired. If the systems using this member are pre-MVS/ESA SP Version 4, you must change the member to use the appropriate syntax.

**CIDM211S PROGRAM HAS INSUFFICIENT AUTHORIZATION TO DELETE THIS DATA SET**

**Explanation:** The RACROUTE macro returned a non-zero return code indicating that the program does not have sufficient authorization to delete the data set. Authorization is established using a security system such as RACF.

**Dialog Action:** The data set is not deleted. Module CIDXGCND flushes the rest of the steps of the job.

**System Programmer Response:** Establish ALTER authorization to the data set using your security system. When you have authorization to delete the data set, consult the appropriate CBIPO dialogs installation guide to find out what job to resubmit.



---

## Glossary

This glossary defines terms used in SMP/E publications. Additional terms can be found in the index of this book, in prerequisite publications, and in *IBM Dictionary of Computing*, SC20-1699.

### 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.

**ACDS.** Alternate control data set.

**alternate control data set (ACDS).** An SMP4 data set used to record information about a distribution library.

**ACRQ.** Alternate conditional requisite queue.

**alternate conditional requisite queue (ACRQ).** An SMP4 data set used to record information about a distribution library.

**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 at a later date 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.

**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.

**ASSEM entry.** An SMP/E entry that contains 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 were 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.

**base function.** A SYSMOD that defines 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. OS/VS2 (MVS) Release 3.8 is an example of what would be considered a base level system.

**bypass.** In SMP/E, to circumvent errors that would normally 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.<sup>1</sup> See *root cause analysis*.

**CBIPO.** MVS Custom-Built Installation Process Offering.

**CBIPO driver.** A load-and-go MVS system designed to help users install a CBIPO if there is no MVS system to use as a driver.

**CBIPO Process Aids.** RIMs for a model MVS system, without any product code.

**CBIPO service tape.** A tape containing SYSMODs associated with the products in a CBIPO but not integrated into the CBIPO distribution libraries.

**CBPDO.** MVS Custom-Built Product Delivery Offering.

**CDS.** Control data set.

**control data set (CDS).** An SMP4 data set used to record information about a target library.

**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*.

**conditional requisite queue (CRQ).** An SMP4 data set used to record information about a target library.

**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*.

**CONVERT.** The SMP/E command used to move records from data sets for a previous release of SMP/E or SMP4 into data sets for the latest release of SMP/E.

**convert.** In SMP/E, to change records from SMP/E or SMP4 data sets into the format required for the latest release of SMP/E. This is done with the CONVERT command.

**corequisite SYSMODs.** SYSMODs that require each other to be installed properly. 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 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 may be used to check information saved from ++IF statements and determine where any cross-zone requisites should be installed.

**CRQ.** Conditional requisite queue.

**CSI.** Consolidated software inventory data set. See *SMPCSI*.

**CSSF.** Customer Software Support Facility.

**CUM tape.** Cumulative service tape.

**cumulative service tape (CUM tape).** The tape sent with an order for a new function that contains all the current PTFs for that function.

**Customer Software Support Facility (CSSF).** A data base in Information/Access that can be used to research APARs and to obtain information about preventive service planning.

### 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 that contains the information SMP/E needs to dynamically allocate a particular data set.

---

<sup>1</sup> Information about causer SYSMODs appears only when the root cause SPE (PTF UR90259 for FMID HMP1600 and, if the Japanese feature is installed, PTF UR90260 for FMID JMP1611) is installed.

**DEBUG.** The SMP/E command used to obtain additional information for problem determination. For example, DEBUG can be used 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.

**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.

**DFP.** Data Facility Product.

**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 that describes a distribution library that was totally copied into a target library.

**DLIBZONE entry.** An SMP/E entry that contains information used by SMP/E to process a specific distribution zone and the associated distribution libraries.

## 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 when that element is 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 using UCL statements.

**environment.** The functions (FMIDs) that are 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.

**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.

**explicitly deleted function.** A function that was deleted because it was specified on the DELETE operand of a ++VER statement in another SYSMOD.

**exported zone.** A zone that was copied into a sequential data set using the SMP/E ZONEEXPORT command.

**external HOLDDATA.** ++HOLD statements contained in SMPHOLD. Contrast with *internal HOLDDATA*.

## F

**FE.** Field engineering.

**feature.** See *dependent function*.

**FMID.** Function modification identifier.

**FMIDSET.** A group of FMIDs to be used when processing an SMP/E command. For example, it may be used to indicate that SYSMODs applicable to certain functions should be installed.

**FMIDSET entry.** An SMP/E entry that defines an FMIDSET.

**function.** In SMP/E, a product (such as a system component or licensed program) that can be optionally installed in a user's system. 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.

**global zone.** A group of records in a CSI data set used to record information about SYSMODs that have been received for a particular system. The global zone also contains information that (1) allows SMP/E to access target and distribution zones in that system, and (2) allows you to tailor aspects of SMP/E processing.

**GLOBALZONE entry.** An SMP/E entry that contains information used by SMP/E to process the global zone, the associated target and distribution zones, and the SMPPTS data set.

## H

**header MCS.** An ++APAR, ++FUNCTION, ++PTF, or ++USERMOD statement. The header MCS indicates the type of SYSMOD.

**hierarchy.** In SMP/E, the top-down structure of function and service SYSMODs, where each SYSMOD is dependent on the one above it.

**higher functional level.** An element version that contains all of 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 that contains ++HOLD statements that were either received from SMPHOLD (external HOLDDATA) or that were within a SYSMOD that was received (internal HOLDDATA).

## I

**ICF.** Integrated Catalog Facility.

**IFREQ.** A conditional requisite. Conditional requisites are specified on the REQ operand of the ++IF statement.

**IMASZAP.** The system utility program used to install superzaps, which are changes for modules, load modules, or CSECTs within modules.

**implicitly deleted function.** A function that was deleted because of its dependency on an explicitly deleted function (a function specified on the DELETE operand of the ++VER statement).

**imported zone.** A zone that was copied from a sequential data set into another zone using the SMP/E ZONEIMPORT command.

**IMS.** Information Management System.

**IMSGEN.** IMS generation.

**indicator.** A subentry in an SMP/E data set entry 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.

**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.

**Information/Access.** A feature of Information/System 1, an interactive retrieval program. Information/Access provides direct customer access to a database that contains APAR and PTF information (CSSF), APAR fixes and PTFs, and preventive service planning information.

**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.** In SMP/E, one of the following:

- The SMP/E command used to process data from the SMPJCLIN data set.
- The **++JCLIN** statement, which is associated with JCLIN data that is included in a SYSMOD.
- The SMPJCLIN data set. 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. JCLIN data may be optionally used by SMP/E to update the distribution zone when the SYSMOD is accepted.

**JCLIN input.** The JCL statements contained in the SMPJCLIN data set 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 that contains 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 that contains all the information needed to replace or update a given load module.

**load module.** The output of the linkage editor; a program in a format suitable for loading into main storage for execution.

**LOG.** In SMP/E, one of the following:

- The SMP/E command used to write user-supplied information to the SMPLOG data set.
- 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 that contains a copy of a SYSMOD exactly as it was received from the SMPPTFIN data set. MCS entries are in the SMPPTS data set, which is used as a warehouse for SYSMODs.

**MOD.** The SMP/E entry or MCS that describes an object module.

**MODID.** Modification identifier.

**modification.** In SMP/E, an alteration or correction to an 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 were 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.** A program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading; for example, the input to or output from an assembler, compiler, or linkage editor.

**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 Installation Process Offering (CBIPO).** A software delivery offering used to create or replace an entire MVS, NCP, CICS, or IMS system.

**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.

**MVS/ESA.** Multiple Virtual Storage/Enterprise Systems Architecture (MVS/SP\* Version 3).

**MVS/SP.** Multiple Virtual Storage/System Product.

**MVS/XA.** Multiple Virtual Storage/Extended Architecture (MVS/SP Version 2).

**MVS/370.** Multiple Virtual Storage for System/370\* (MVS/SP Version 1).

## 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 module.** A module that is the output of an assembler or compiler and is input to a linkage editor.

**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 that defines processing options that are to be used by SMP/E.

**OS/VS.** Operating System/Virtual Storage (VS1 or VS2)

**OS/VS1.** Operating System/Virtual Storage 1

**OS/VS2.** Operating System/Virtual Storage 2

---

System/370 and MVS/SP are trademarks of the IBM Corporation.

## P

**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.

**PARMLIB.** The data set that contains members to define parameters, such as macros and assembler operation codes.

**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 either be already installed or be installed along with another SYSMOD for that other SYSMOD to be successfully installed. It is defined by the PRE operand on the ++VER statement.

**preventive service.** In SMP/E, one of the following:

- The mass installation of PTFs to avoid rediscoveries of the APARs fixed by those PTFs.
- The SYSMODs delivered on the program update tape.

**preventive service planning (PSP).** Installation recommendations and HOLDDATA for a product or PUT. PSP information may be obtained through CSSF or the IBM Support Center.

**product.** Generally, a software package, such as a licensed program or user application. A product may contain one or more functions and may 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 was 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 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 tape containing preventive service PTFs. The tape is shipped periodically from the program libraries, such as IBM Software Distribution (ISD).

**PSP.** Preventive service planning.

**PTF.** Program temporary fix.

**PTS.** PTF temporary store data set. See *SMPPTS*.

**PTFIN.** PTF input data set. See *SMPPTFIN*.

**PUT.** Program update tape.

## R

**RACF.** Resource Access Control Facility.

**RECEIVE.** The SMP/E command used to read in SYSMODs and other data from the SMPPTFIN and SMPHOLD data sets.

**receive.** In SMP/E, to read SYSMODs and other data from the SMPPTFIN and SMPHOLD data sets, and store them on the global zone for subsequent SMP/E processing. This is done with the RECEIVE command.

**regressed SYSMOD.** A SYSMOD that has one or more of its elements 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 data set.

**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 CBIPO, task-oriented documentation, jobs, sample exit routines, procedures, parameters, and examples developed by IBM. RIMs are designed to help users install, generate, and use systems created from a CBIPO.

**related SYSMOD.** A SYSMOD associated with other SYSMODs by the FMID, PRE, REQ, or SUP operands.

**related zone.** The zone named in the RELATED sub-entry of a TARGETZONE or DLIBZONE entry. For a target zone, the related zone is generally the distribution zone for the libraries that were used to create the target libraries. For a distribution zone, the related zone is generally the target zone for the libraries that were built from the distribution libraries.

**relative file tape (RELFILE tape).** A standard label tape made up of two or more files. It contains a file of the MCSs for one or more function SYSMODs and one or more relative files containing unloaded source data sets and unloaded, link-edited object data sets at the distribution library level. A relative file tape is one way of packaging SYSMODs, and is typically used for function SYSMODs.

**relative files (RELFILES).** Files that contain modification text and JCL input data associated with a SYSMOD.

**release.** A distribution of a new product or new function and APAR fixes for an existing product. Contrast with *modification level* and *version*.

**RELFILES.** Relative files.

**RELFILE tape.** Relative file tape.

**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 CROSSZONE:** Lists conditional requisites that must be installed in certain zones because of SYSMODs that are installed in other zones.
- **REPORT ERRSYSMODS:** Determines whether any SYSMODs that have already been 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. It may be defined by either the PRE or REQ operands on the ++VER statement, or by the REQ operand on the SYSMOD's associated ++IF statement.

**RESETRC.** The SMP/E command used to set the return codes for the previous commands to zero, thus allowing SMP/E to 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. This is done with the RESTORE command.

**restore group.** All the SYSMODs that have a direct or indirect relationship with a SYSMOD being restored using the GROUP operand.

**RETAIN.** A data base accessible through Information/Access that contains information about APARs and PTFs. The customer version of this data base is called the Customer Service Support Facility (CSSF).

**RIM.** Related installation material.

**RMID.** Replacement modification identifier.

**root cause analysis.** Processing done by SMP/E for the ACCEPT, APPLY, and RESTORE commands to identify causer SYSMODs (SYSMODs whose failure led to the failure of other SYSMODs).<sup>2</sup> 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 (x37 abends)
- Missing DD statements and other allocation errors
- ID errors (SYSMOD does not supersede an RMID or UMID or specify it as a prerequisite)
- JCLIN failures (syntax and other errors)

## S

**SCDS.** Save control data set. See *SMPSCDS*.

**SCP.** System control program.

**select-mode processing.** In SMP/E, processing that includes SYSMODs that were individually selected.

**SERV.** CBIPO service tape.

**service.** PTFs and APAR fixes.

---

<sup>2</sup> Root cause analysis information appears only when the root cause SPE (PTF UR90259 for FMID HMP1600 and, if the Japanese feature is installed, PTF UR90260 for FMID JMP1611) is installed.

**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. This is not a new release of the function, so 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.

**SMP.** System Modification Program.

**SMPCNTL.** The SMP/E data set 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 that contains a dump that was 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.** System Modification Program Extended.

**SMP/E commands.** Commands that define 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.** The SMP/E file or data set that contains HOLDDATA (++HOLD and ++RELEASE statements) to be processed by the RECEIVE command.

**SMPJCLIN.** The SMP/E data set 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 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 may 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.

**SMPMPTS.** The SMP/E data set used as a temporary target library for macros that only exist in a distribution library (such as macros in SYS1.AMODGEN). The SMPMPTS allows the current version of these macros to be used for assemblies during APPLY processing.

**SMPOBJ.** The SMP/E data set used for source-maintained products. SMPOBJ contains preassembled modules that can be used instead of 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 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 the SMPLIST and SMPRPT data sets are not defined.

**SMPPTFIN.** The SMP/E file or data set that contains SYSMODs and ++ASSIGN statements to be processed by the RECEIVE command.

**SMPPTS.** The SMP/E data set that contains SYSMODs that have been received from the SMPPTFIN data set. The SMPPTS is a sort of warehouse, and is the source of SYSMODs that are installed in the target and distribution libraries.

**SMPSPUNCH.** The SMP/E data set 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 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 may request a snap dump of SMP/E storage when specified messages are issued, or may 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 temporary target library for source that only exists in a distribution library. The SMPSTS allows 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 that are 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 linkage-editor 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.

**SMP4.** System Modification Program Release 4.

**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 (such as from a particular PUT). 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.

**SRC.** The SMP/E entry or MCS statement that describes a source.

**SRCUPD.** The MCS used to update a source.

**SREL.** System release identifier.

**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.

**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 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 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 PTF's SYSMOD entry.

**subentry indicator.** A field in an entry that does not have a data value associated with it. An example of a subentry indicator is the APP indicator in a SYSMOD entry.

**SUP.** Supersede.

**superseded-only SYSMOD.** A SYSMOD that has not been installed, but that has been superseded by another SYSMOD that was 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. This is done by specifying the superseded SYSMOD on the SUP operand of the ++VER statement.

**superzap.** A generic term for the process performed by IMASPZAP. It can also refer to the module updates processed by IMASPZAP.

**SYSGEN.** System generation.

**SYSLIB.** In SMP/E, one of the following:

- A subentry used to identify the target library where an element is installed.
- A concatenation of macro libraries to be used by the assembler.
- A set of routines used by the linkage editor to resolve unresolved external references.

**SYSMOD.** System modification.

**SYSMOD entry.** An SMP/E entry that contains information about a SYSMOD that has been received into the 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 that contains object modules that were 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 or 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 or 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 Modification Program (SMP).** A program used to install software and software changes on OS/VS1 and OS/VS2 systems. In this book, SMP refers to SMP Release 4.

**System Modification Program Extended (SMP/E).** A licensed program used to install software and software changes on OS/VS1 and OS/VS2 systems. In addition to providing the services of SMP, 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. In this book, SMP/E refers to the current release of SMP/E and any earlier releases of SMP/E, unless specified otherwise.

**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 that contains 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 that contains information used by SMP/E to process a specific target zone and the associated target libraries.

**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 are deleted when processing is done.

**text library (TXLIB).** A data set that contains 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.

**TLIB.** Temporary library. See *SMPTLIB*.

**transformed data.** Data that has been processed by the GIMDTS service routine so that it can be packaged inline in fixed-block 80 records.

**TXLIB.** Text library.

## U

**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 sub-entries 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. This is done with 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 either 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 that contains 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 that contain elements that are functionally lower than elements contained 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.

**VS1.** Operating System/Virtual Storage 1.

**VS2.** Operating System/Virtual Storage 2.

**VTOC.** Volume table of contents.

## Z

**ZAP.** In SMP/E, one of the following:

- The MCS used to update an object module.
- A shortened name for the IMASPZAP 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 that defines a ZONESET.



---

# Communicating Your Comments to IBM

System Modification Program Extended  
Messages and Codes  
Release 6

Publication No. SC28-1108-09

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

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.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:
  - United States & Canada: 914 + 385-0662
  - Other Countries: (country code + 1) + 914 + 385-0662
- If you prefer to send comments electronically, use this network ID:
  - IBMLINK: KGNRCF at KGNVMC
  - IBM Mail Exchange: USIB27BQ at IBMMAIL
  - Internet: KGNRCF@KGNVMC.vnet.ibm.com

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.



---

## Readers' Comments

**System Modification Program Extended  
Messages and Codes  
Release 6**

**Publication No. SC28-1108-09**

You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you. Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

**Note:** Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.

Today's date: \_\_\_\_\_

What is your occupation?

Newsletter number of latest Technical Newsletter (if any) concerning this publication:

How did you use this publication?

- |                          |                               |                          |                        |
|--------------------------|-------------------------------|--------------------------|------------------------|
| <input type="checkbox"/> | As an introduction            | <input type="checkbox"/> | As a text (student)    |
| <input type="checkbox"/> | As a reference manual         | <input type="checkbox"/> | As a text (instructor) |
| <input type="checkbox"/> | For another purpose (explain) |                          |                        |

---

Is there anything you especially like or dislike about the organization, presentation, or writing in this manual? Helpful comments include general usefulness of the book; possible additions, deletions, and clarifications; specific errors and omissions.

Page Number:

Comment:

\_\_\_\_\_  
Name

\_\_\_\_\_  
Address

\_\_\_\_\_  
Company or Organization

\_\_\_\_\_  
Phone No.



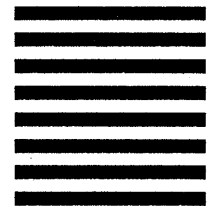
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES



---

# BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

---

POSTAGE WILL BE PAID BY ADDRESSEE

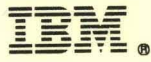
International Business Machines Corporation  
Information Development  
Department 52QA MS 911  
NEIGHBORHOOD ROAD  
KINGSTON NY 12401-0000



Fold and Tape

Please do not staple

Fold and Tape



File Number: S370-37  
Program Number: 5668-949

Printed in USA



SC28-1108-09

