

System Automation for z/OS
Version 3 Release 4

Messages and Codes



Note!

Before using this information and the product it supports, read the information in "Notices" on page v.

This edition applies to IBM Tivoli System Automation for z/OS (5698-SA3) Version 3 Release 4, an IBM licensed program, and to all subsequent releases and modifications until otherwise indicated in new editions.

| This edition replaces SC34-2574-02.

| IBM welcomes your comments. You may forward your comments electronically, or address your comments to:

| IBM Deutschland Research & Development GmbH

| Department 3248

| Schoenaicher Strasse 220

| 71032 Boeblingen

| Germany

| FAX: (Germany): 07031 16-3456

| FAX: (Other countries): +49 7031 16-3456

| Internet e-mail: s390id@de.ibm.com

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 IBM Corporation 1996, 2012.

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

Contents

Notices	v
Trademarks	vi

Accessibility	vii
Using assistive technologies	vii
Keyboard navigation of the user interface	vii
z/OS information	vii

How to send your comments to IBM	ix
If you have a technical problem	ix

About This Book.	xi
Who Should Use This Book	xi
How to Use This Book	xi
Where to Find More Information	xi
The System Automation for z/OS Library	xi
Related Product Information	xii
Using LookAt to look up message explanations	xii

Chapter 1. About System Automation for z/OS Messages.	1
Format of System Automation for z/OS Messages	1
Message Text	2
Who Receives Messages	2
Displaying Notification Operator Settings	2
Message Classes	3
MVS Descriptor Codes	4

Chapter 2. Messages AOF000I to AOF820I	5
---	----------

Chapter 3. Messages EVE120I to EVE842E (CICS Automation)	113
Error Codes	117

Chapter 4. Messages EVI022I to EVI844A (IMS automation)	121
--	------------

Chapter 5. Messages EVJ000I to EVJ440I (TWS Automation)	123
Codes Posted to TWS by TWS Automation	133

Chapter 6. Messages HSAH001I to HSA6288I	135
---	------------

Chapter 7. Messages HSAM1000I to HSAM5411I	147
---	------------

Chapter 8. Messages IHVA100I to IHVZ760.	169
---	------------

Chapter 9. Messages ING001I to INGY1203I.	225
--	------------

Chapter 10. Messages ISQ001I to ISQ901I	307
--	------------

Chapter 11. Messages ISQX998I and ISQX999I	365
---	------------

Chapter 12. Messages ISQtt001I to ISQtt0723I.	367
--	------------

Appendix A. Return Codes, Status Codes, and Reason Codes	385
I/O Operations Return Codes	385
I/O Operations Status Codes for a Dynamic Switch	385
I/O Operations Reason Codes	385
Reason Codes Associated With Several Return Codes	385
Reason Codes Associated With RC=0	387
Reason Codes Associated With RC=4 or RC=8	387
Reason Codes Associated With RC=12	390
Reason Codes Associated With RC=16	390
Reason Codes Associated With RC=20	393

Appendix B. Abend Codes and Reason Codes Issued By the Automation Manager	395
Communication DST and Automation Manager Abend Codes	395

Appendix C. Response Messages, Error Strings, Condition Codes.	397
Response Messages (AOFA0000–AOFA0099)	397
Asynchronous Response Messages (AOFA0100–AOFA0999)	411
Response Messages (AOFB0000–AOFB0099)	415
Asynchronous Response Messages (AOFB0100–AOFB0999)	433
Condition Codes	434
Hardware Communication Task Condition Codes "00B00xxx"	434
Data Exchange Services "0B100xxx"	438
Command Services "0B200xxx"	440
Internal Transport Services "0Bx00xxx"	442
Ensemble HMC API Exchange Services Condition Codes	446
Ensemble Communication Task Condition Codes "00B600xxx"	449

Appendix D. Sense Codes, Hardware Object Status Summary	455
--	------------

Sense Codes 455
Hardware Object Status Summary 456

**Appendix E. Automation Manager
Reason Codes 459**

**Appendix F. Sysplex Communication
Services Return and Reason Codes. . 463**

**Appendix G. End-to-End Automation
Adapter Messages 467**

Glossary 477

Notices

This information was developed for products and services offered in the U.S.A.

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Deutschland Research and Development GmbH
Department 3248
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms, used in this book, are trademarks of the IBM Corporation in the United States or other countries:

CICS/ESA	BladeCenter	CICS®
DataPower	CICSplex®	Current
DFSMS/MVS	DB2®	DFS
	ESCON®	First Failure Support Technology/2™
FFST/2™	FICON®	IBM
IMS™	IMS/ESA®	MQSeries®
MVS™	NetView	OMEGAMON
OS/390®	POWER	POWER7
PowerVM	PR/SM™	PS/2
RACF®	System x	System z
S/390®	SP	Sysplex Timer®
System/390®	Tivoli®	Tivoli Enterprise Console®
VM/ESA®	VSE/ESA	VTAM®
z/OS®	z/VM	z10
z9	zEnterprise	zSeries®

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Accessibility

Publications for this product are offered in Adobe Portable Document Format (PDF) and should be compliant with accessibility standards. If you experience difficulties when using PDF files, you may view the information through the z/OS Internet Library website or the z/OS Information Center. If you continue to experience problems, send an email to mhvrcfs@us.ibm.com or write to:

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
U.S.A.

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS enable users to:

- Use assistive technologies such as screen readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size

Using assistive technologies

Assistive technology products, such as screen readers, function with the user interfaces found in z/OS. Consult the assistive technology documentation for specific information when using such products to access z/OS interfaces.

Keyboard navigation of the user interface

Users can access z/OS user interfaces using TSO/E or ISPF. Refer to *z/OS TSO/E Primer*, *z/OS TSO/E User's Guide*, and *z/OS ISPF User's Guide Vol I* for information about accessing TSO/E and ISPF interfaces. These guides describe how to use TSO/E and ISPF, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

z/OS information

z/OS information is accessible using screen readers with the BookServer or Library Server versions of z/OS books in the Internet library at:

<http://www.ibm.com/systems/z/os/zos/bkserv/>

How to send your comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or give us any other feedback that you might have.

Use one of the following methods to send us your comments:

1. Send an email to s390id@de.ibm.com
2. Visit the SA z/OS home page at http://www.ibm.com/systems/z/os/zos/features/system_automation/
3. Visit the Contact z/OS web page at <http://www.ibm.com/systems/z/os/zos/webqs.html>
4. Mail the comments to the following address:
IBM Deutschland Research & Development GmbH
Department 3248
Schoenaicher Str. 220
D-71032 Boeblingen
Federal Republic of Germany
5. Fax the comments to us as follows:
From Germany: 07031-16-3456
From all other countries: +(49)-7031-16-3456

Include the following information:

- Your name and address
- Your email address
- Your telephone or fax number
- The publication title and order number:
IBM Tivoli System Automation for z/OS V3R4.0 Messages and Codes
SC34-2648-00
- The topic and page number related to your comment
- The text of your comment.

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.

IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you submit.

If you have a technical problem

Do not use the feedback methods listed above. Instead, do one of the following:

- Contact your IBM service representative
- Call IBM technical support
- Visit the IBM zSeries support web page at www.ibm.com/systems/z/support/.

About This Book

This book contains problem determination information for IBM Tivoli System Automation for z/OS (SA z/OS). This information includes messages, return codes, reason codes and status codes.

Who Should Use This Book

This book is intended for operators and system programmers of System Automation for z/OS (SA z/OS).

How to Use This Book

This book lists all SA z/OS messages, return codes, and reason codes. The messages are sorted alphabetically by the first four characters.

Where to Find More Information

The System Automation for z/OS Library

Table 1 shows the information units in the System Automation for z/OS library:

Table 1. System Automation for z/OS Library

Title	Order Number
<i>IBM Tivoli System Automation for z/OS Planning and Installation</i>	SC34-2645
<i>IBM Tivoli System Automation for z/OS Customizing and Programming</i>	SC34-2644
<i>IBM Tivoli System Automation for z/OS Defining Automation Policy</i>	SC34-2646
<i>IBM Tivoli System Automation for z/OS User's Guide</i>	SC34-2647
<i>IBM Tivoli System Automation for z/OS Messages and Codes</i>	SC34-2648
<i>IBM Tivoli System Automation for z/OS Operator's Commands</i>	SC34-2649
<i>IBM Tivoli System Automation for z/OS Programmer's Reference</i>	SC34-2650
<i>IBM Tivoli System Automation for z/OS Product Automation Programmer's Reference and Operator's Guide</i>	SC34-2643
<i>IBM Tivoli System Automation for z/OS TWS Automation Programmer's Reference and Operator's Guide</i>	SC34-2651
<i>IBM Tivoli System Automation for z/OS End-to-End Automation Adapter</i>	SC34-2652
<i>IBM Tivoli System Automation for z/OS Monitoring Agent Configuration and User's Guide</i>	SC34-2653

The System Automation for z/OS books are also available on CD-ROM as part of the following collection kit:

IBM Online Library z/OS Software Products Collection (SK3T-4270)

SA z/OS Home Page

For the latest news on SA z/OS, visit the SA z/OS home page at http://www.ibm.com/systems/z/os/zos/features/system_automation

Related Product Information

You can find books in related product libraries that may be useful for support of the SA z/OS base program by visiting the z/OS Internet Library at <http://www.ibm.com/systems/z/os/zos/bkserv>

Using LookAt to look up message explanations

LookAt is an online facility that lets you look up explanations for most of the IBM messages you encounter, as well as for some system abends and codes. Using LookAt to find information is faster than a conventional search because in most cases LookAt goes directly to the message explanation.

You can use LookAt from these locations to find IBM message explanations for z/OS elements and features, z/VM[®], z/VSE[®], and Clusters for AIX[®] and Linux:

- The Internet. You can access IBM message explanations directly from the LookAt Website at www.ibm.com/systems/z/os/zos/bkserv/lookat/index.html
- Your z/OS TSO/E host system. You can install code on your z/OS or z/OS.e systems to access IBM message explanations using LookAt from a TSO/E command line (for example: TSO/E prompt, ISPF, or z/OS UNIX System Services).
- Your Microsoft Windows workstation. You can install LookAt directly from the *z/OS Collection* (SK3T-4269) or the *z/OS and Software Products DVD Collection* (SK3T-4271) and use it from the resulting Windows graphical user interface (GUI). The command prompt (also known as the DOS > command line) version can still be used from the directory in which you install the Windows version of LookAt.
- Your wireless handheld device. You can use the LookAt Mobile Edition from www.ibm.com/systems/z/os/zos/bkserv/lookat/lookatm.html with a handheld device that has wireless access and an Internet browser (for example: Internet Explorer for Pocket PCs, Blazer or Eudora for Palm OS, or Opera for Linux handheld devices).

You can obtain code to install LookAt on your host system or Microsoft Windows workstation from:

- A CD-ROM in the *z/OS Collection* (SK3T-4269).
- The *z/OS and Software Products DVD Collection* (SK3T-4271).
- The LookAt Website (click **Download** and then select the platform, release, collection, and location that suit your needs). More information is available in the LOOKAT.ME files available during the download process.

Chapter 1. About System Automation for z/OS Messages

This book describes System Automation for z/OS messages and codes. It enables system operators and system programmers to respond to messages appropriately and efficiently.

Format of System Automation for z/OS Messages

Each System Automation for z/OS system message begins with a unique message identifier, followed by message text.

In general, the message format is as follows:

XXXynnt MESSAGE TEXT

where:

XXX or XXXX

Is the **program identifier**. It identifies where in System Automation for z/OS the message originated from. It can have three or four characters

- **AOF**: System operations
- **EVE**: CICS-related messages
- **EVI** : IMS-related messages
- **EVJ**: TWS-related messages
- **HSA**: Automation manager
- **IHV**: I/O operations
- **ING**: System Automation for z/OS, general parts
- **ISQ**: Processor operations

Y Is the SA z/OS component specific identifier.

nnn Is the **message identification number**. It is a three-digit or four-digit number that uniquely identifies each message.

t Is the **message type**. It is a letter that helps indicate the purpose of the message and the type of response required for the message. The message types are:

- A** Immediate Action. Requires the system operator to perform an action. The associated task does not continue until the requested action is taken.
- D** Immediate Decision. Requires the system operator to make a decision from the available options specified on the message. For example, if a request for a particular tape drive cannot be satisfied, a system operator will have to communicate a decision (REPLY "RETRY" or "CANCEL") back to the system. The associated task is suspended until the operator communicates the decision to the system.
- E** Eventual Action. Requires eventual action by the system operator. The associated task continues independent of the operator action.
- I** Information. Requires no action. This message is used only for advisory purposes.
- W** System Wait. Requires system operator action immediately. The system or a major subsystem must be restarted.

Message Text

The message text provides concise information about an event or condition of interest to an operator or system programmer. Variable text is used to identify system components and to provide other detailed information. In this book, variable text in messages is italicized, for example, *job_name*.

Some messages include two or more alternate text strings, only one of which is generated with any single occurrence of the message. The text string generated depends on the condition the message is reporting. In this book, alternate text strings are surrounded by braces ({ and }) and are separated by the OR bar (|). For example: {*domain* | *operator*}.

In some cases, messages include optional text. Optional text might or might not appear with the message, depending on the condition reported by the message. In this book, optional text is surrounded by brackets ([and]).

In some cases " - FROM=*sysname*" is appended to the message text, where *sysname* indicates the system where the message originated. This normally applies where the message was received by a slave process running on a remote system.

Who Receives Messages

System Automation for z/OS messages are received by automation operators and notification operators. *Automation operators* are automated operator tasks or *autotasks* that can respond to operating system, subsystem, and network messages without requiring a human operator.

Notification operators are NetView[®] operators who receive System Automation for z/OS notification messages, or messages about actions System Automation for z/OS either detects or takes. Notification messages are routed to one or more assigned, logged-on operators. Notification operators can be defined the following ways:

- Using the Operator Notification panel in the customization dialogs. Refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy* for further information on how to use the customization dialogs.
- Using the INGNTFY command. Refer to *IBM Tivoli System Automation for z/OS User's Guide* for more information on how to use the INGNTFY command.

Displaying Notification Operator Settings

If you want to find out who is currently receiving SA z/OS messages, use the INGNTFY command dialog. INGNTFY displays a list of notification operators, whether they are logged on and are receiving messages, and a description of the operator from either the automation control file or the INGNTFY command dialog.

```

AOFKAANT          SA z/OS - Command Dialogs      Line 1   of 3
Domain ID = IPSNO ----- INGNTFY -----      Date = 09/15/11
Operator ID = OPID                                     Time = 10:22:29

Cmd:  A Add      C Show/Change settings    D Delete    O Turn OFF msg receiving

Cmd Operator  System  Log  Rcv  Description  Classes
-----
-  AUTNOTI1   AOC4   Y   N   MSG AOP
-  NOTI1      AOC4   N   Y
-  NOYI2     AOC4   Y   Y
                                     40 41 42 43 44
                                     80 81 82

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Ro11
               PF9=Refresh  PF12=Retrieve
    
```

Figure 1. Example DISPNTFY Command Dialog

Notification operators can be logged on without receiving messages if they have used the INGNTFY command to turn off messages. Refer to *IBM Tivoli System Automation for z/OS User's Guide* for further information.

Message Classes

Some messages generated by System Automation for z/OS have a message class. During automated operations, the class of the message is compared to the class of each notification operator. Any operator with a class matching the message's class receives the message.

To find out which classes of messages are assigned to each operator, use the customization dialogs. For further information on the customization dialogs, refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

The message classes are shown in the following table:

Table 2. Message Classes

Message Class	Description
General Message Classes	
0	Errors
1	Debug
Automation Agent-Related Message Classes	
40	All Automation agent messages
41	RESERVED
42	RESERVED
43	Information
44	Action
45	Status
46	User intervention
50	Critical threshold setting exceeded

About System Automation for z/OS Messages

Table 2. Message Classes (continued)

Message Class	Description
51	Frequent threshold setting exceeded
52	Infrequent threshold exceeded
60	IMS Automation messages
Automation Manager-Related Message Classes	
80	All automation manager messages
81	Intervention required
82	Automation denied
83	Automation impacted
84	Awaiting automation
85	In automation
86	Degraded: Target resources
87	Degraded: All resources
88	Satisfactory: Target resources
89	Satisfactory: All resources
90	SA z/OS Infrastructure/Environment

MVS Descriptor Codes

The System Automation for z/OS AOCMSG common automation routine handles descriptor codes to assist in the management of the SA z/OS messages at master consoles (with respect to highlighting, color, and hold/nohold). The following convention exists:

W	1
A	2
D	2
I	6

For more details, refer to *z/OS MVS System Messages*.

Chapter 2. Messages AOF000I to AOF820I

AOF000I MESSAGE *message_number* ISSUED BUT DOES NOT EXIST IN MESSAGE TABLE *table* - CALL IGNORED

Explanation: System automation attempted to issue a message, but no text could be found for the message in the message table.

The variable *message_number* shows the number of the message that text was missing for.

The variable *table* shows the name of the message library member that the message should have been found in.

System action: None.

Operator response: None.

System programmer response: This message indicates that there might have been a problem installing System Automation for z/OS. Make sure the message library SINGNMSG is allocated to DSIMSG DD.

Classes: None.

AOF001I REQUEST *request* WAS SUCCESSFUL FOR *entry_type*

Explanation: The request for the record has been completed successfully.

The variable *request* shows the function performed on the record.

The variable *entry_type* shows the function or key of the record that was processed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF002I END OF MULTILINE MESSAGE GROUP

Explanation: This message identifies the end of a multiline message group.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF003I *function* IS NOT ACTIVE

Explanation: An operator dialog issues this message. An inactive PF key was pressed.

The variable *function* shows the PF key that was pressed.

System action: None.

Operator response: Press a key that is active. For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF004I PROCESSING FAILED FOR *command* COMMAND

Explanation: The entered command (*command*) failed to process. This could be because of an error, or because the command requested something that was already in effect. Other messages preceding this one indicate the reason for the failure.

System action: None.

Operator response: Make sure the proper command was entered and look for messages preceding this message to determine the cause of the failure. If you cannot determine the cause of the failure, contact your system programmer.

System programmer response: Review the netlog (which was active at the time the command was rejected) to see whether a message was routed correctly.

Classes: None.

AOF005I MEMBER *member* CURRENTLY BEING *action* FOR THE CONTROL FILE

Explanation: This is the first line of a multiline message group.

The variable *member* shows the name of the member currently being used.

The variable *action* shows where the automation control file resides. It can be one of the following:

USED The status display is for the current in-storage member

SAVED

The status display is for the currently saved member

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF006I *text*

Explanation: This message is part of a multiline message group that begins with message AOF005I.

The variable *text* shows detailed information about an automation control file. It can be one of the following:

```
BUILT BY name ON date AT TIME
LOADED BY name ON date at time
CONFIGURATION TOKEN = token
CONFIGURATION DATASET = dataset_name
```

The variable *name* shows the name of an operator.

The variable *date* shows the date the action was done.

The variable *time* shows the time the action was done.

The variable *token* shows the current configuration token.

The variable *dataset_name* shows the name of current configuration data set.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF007I **A RESOURCE TYPE IS REQUIRED FOR THIS SELECTION**

Explanation: A resource type was not entered as a selection on the prompt screen.

System action: None.

Operator response: Enter the resource type.

System programmer response: None.

Classes: None.

AOF008I **A SELECTION MUST BE MADE**

Explanation: An operator dialog issues this message. This message indicates that you must make a selection on the prompt screen.

System action: None.

Operator response: Make a selection on the prompt screen. For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF009I **MAXIMUM NUMBER OF *keyword* OPERANDS EXCEEDED**

Explanation: The maximum number of operands allowed for the keyword identified in this message was exceeded.

The variable *keyword* shows the keyword that too many operands were specified for.

System action: None.

Operator response: Enter the command again, making sure that the correct number of operands is specified for the keyword you specify.

System programmer response: None.

Classes: None.

AOF010I **WRONG NUMBER OF PARAMETERS ENTERED**

Explanation: Too few or too many parameters were entered with a command or request.

System action: None.

Operator response: Enter the command or request again, this time using the right number of parameters.

System programmer response: None.

Classes: None.

AOF011I **EXPECTING AN ENDING *delimiter* FOR *parameter***

Explanation: The operand was entered without a closing delimiter.

The variable *delimiter* shows the character expected as the delimiter.

The variable *parameter* shows the parameter that was in error.

System action: None.

Operator response: Enter the command again using the correct delimiter.

System programmer response: None.

Classes: None.

AOF012I *message_ID message_text*

Explanation: General purpose message entry. Specifically used for message forwarding of messages from users or other components. The AOF012I message identifier is not displayed to an operator who receives this message.

The variable *message_ID* shows the message identifier of the forwarded message.

The variable *message_text* shows the message text of the forwarded message.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF013I **SPECIFIED OPERAND** *operand*
INVALID FOR *parameter*

Explanation: An operand that is not valid was specified for this parameter.

The variable *operand* shows the operand that is not valid.

The variable *parameter* shows the parameter that was entered.

System action: None.

Operator response: Enter the command again using the correct operands.

System programmer response: None.

Classes: None.

AOF014I **SPECIFIED PARAMETER** *parameter*
INVALID

Explanation: The specified parameter (*parameter*) is not valid for this command.

System action: None.

Operator response: Enter the command again using the correct parameters.

System programmer response: None.

Classes: None.

AOF015I **INVALID DELIMITER OF** *delimiter*
ENCOUNTERED

Explanation: A delimiter that is not valid (*delimiter*) was entered for a command.

System action: None.

Operator response: Enter the command again using the correct syntax.

System programmer response: None.

Classes: None.

AOF017I **EXPECTED PARAMETER(S) MISSING**
FOR REQUEST *request*

Explanation: Required parameters for this request are missing or null. If a parameter was entered, no operand was specified.

The variable *request* shows the request used.

System action: None.

Operator response: Enter the command again using the correct syntax.

System programmer response: None.

Classes: None.

AOF018I **INVALID SEARCH RANGE SPECIFIED**

Explanation: The specified starting search range is greater than the ending search range.

System action: None.

Operator response: Enter the command again using the correct syntax.

System programmer response: None.

Classes: None.

AOF019I *time*: **INVALID SYNTAX FOR**
COMMAND *cmd_name parmstr*

Explanation: Command *cmd_name* was invoked with invalid syntax on the command line.

The variable *time* shows the time the error was generated.

The variable *cmd_name* is the name of the failing command.

The variable *parmstr* is the parameter string passed to the failing command, where available.

System action: Processing is terminated.

Operator response: If operator initiated, correct and reissue the command, otherwise contact your system programmer.

System programmer response: Correct and reissue the command.

Classes: None.

AOF020A *macro/type* **REQUEST WAS**
UNSUCCESSFUL RC=rc A=xx I=yy FOR
function

Explanation: The program made a macro type request. The macro was not able to satisfy the request and returned values for use in problem determination. These values are displayed in this message.

The variable *macro/type* shows the request type.

The variable *rc* shows the return code from the macro.

The variable *xx* shows the major return code from a VSAM request.

The variable *yy* shows the minor return code from a VSAM request.

The variable *function* shows the name of the macro.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Check the VSAM return code values to determine the cause of the problem.

Classes: None.

AOF021I CONFLICTING PARAMETERS
parameters SPECIFIED for command

Explanation: An invalid combination of parameters was specified for *command*. The variable *parameters* lists the conflicting parameters. The variable *command* shows the name of the issued command.

System action: None.

Operator response: Enter the command again with a valid combination of parameters

System programmer response: None

Classes: None.

AOF022E MODULE *module* DETECTED ERROR IN SERVICE *service*. RC=*rc*, RS=*rs*

Explanation: A service was invoked and returned an unexpected return code or reason code. This message is logged in the netlog.

The variable *module* shows the name of the module that invoked the service.

The variable *service* shows the name of the service that was invoked.

The variable *rc* shows the return code that was received.

The variable *rs* shows the reason code that was received.

System action: Processing stops. The message is logged in the netlog.

Operator response: Refer to the documentation of the service that was invoked for more information.

System programmer response: None.

Classes: None.

AOF024I A NULL FIELD WAS ENTERED

Explanation: A null field is not acceptable.

System action: None.

Operator response: Correct the command syntax and reissue the command.

System programmer response: None.

Classes: None.

AOF025I SYNTAX ERROR FOR *command*

Explanation: A syntax error was encountered in the command that was entered.

System action: None.

Operator response: Correct the command syntax and reissue the command.

System programmer response: None.

Classes: None.

AOF026I *module* DETECTED A VERSION MISMATCH. REASON *nm*

Explanation: Module *module* detected a version mismatch.

The variable *module* shows the name of the module that invoked the service.

The variable *rs* shows the reason code. Valid reason codes are:

- 01** The version of AOFRASFX does not match.
- 02** The version of a status file record does not match.

System action: The request is not processed.

Operator response: Check the system setup and reissue the command.

System programmer response: None.

Classes: None.

AOF027I THE FOLLOWING SA AUTOMATION OPERATORS COULD NOT BE STARTED:

Explanation: This is the first message of a multiline message group that lists all automation operators that could not be started.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF028I *automation operator name* - NETVIEW TASK *taskname* status

Explanation: This message is part of a multiline message group that begins with message AOF027I.

The variable *automation operator name* shows the name of the SA z/OS automation operator that could not be started.

The variable *taskname* shows the name of the NetView task that caused the problem.

The variable *status* shows:

- UNDEFINED if the NetView task is currently not defined.

- INACTIVE if it could not be started.

System action: None.

Operator response: None.

System programmer response: Define the missing NetView task names or review the netlog for reasons why it could not be started.

Classes: None.

AOF029I DYNAMIC ADD OF SA AUTOMATION OPERATORS FAILED

Explanation: This message follows the multiline message group that begins with message AOF027I.

System action: None.

Operator response: None.

System programmer response: Define the missing NetView task names (see message AOF028I) or review the netlog for reasons why it could not be started.

Classes: None.

| AOF030I *task_name* IS INACTIVE

| **Explanation:** The called task (*task_name*) is not active at this time.

| **System action:** None.

| **Operator response:** Start the task by issuing START TASK=xxxxxxx, where xxxxxx is:

- | • AOFSTSTS for the status file

| **System programmer response:** None.

| **Classes:** None.

AOF032I USAGE *util* %, LONG LINEAR *pred1*, SHORT LINEAR *pred2*, 1ST ORDER *pred3*

Explanation: This message is posted to SDF when JES2 spool recovery has been initiated and SA z/OS makes predictions about future spool usage. There are three different prediction methods:

Long Linear

This method looks at the first data point and the latest data point, and does a linear extrapolation to predict how long it will take for the spool to reach 100% utilization.

Data points occur if either the pass processor runs, or a spool recovery initialization message (HASP050 or HASP355) is received.

Short Linear

This method is like long linear, but is based on the last two data points.

Short 1st Order

This method takes the last 3 data points,

works out the rate of change between each pair of consecutive points, then works out the variation between these two rates of change. It then extrapolates using this quadratic variation to predict when the spool will reach 100% utilization.

The variable *util* shows the current percent utilization of all active spool volumes.

The variables *pred1*, *pred2*, *pred3* are spool usage predictions. They can have the following values:

hh:mm:ss

The predicted time for the spool to reach 100% utilization.

NONE Insufficient data points were collected to make the prediction.

STABLE

There is no change in spool usage between the data points. Spool usage is neither increasing nor decreasing.

SAFE The projected spool usage falls below the lowest data point rather than reaching 100% utilization.

DROPPING

The latest data point is lower than the others.

FULL The current data point shows 100% spool utilization.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF035I MEMBER *member* I/O ERROR OCCURRED WHILE PROCESSING

Explanation: An I/O error occurred during processing of the member identified by the variable *member*.

System action: Processing ends.

Operator response: Make sure the member identified in the message is valid. If the member identified in the message is valid, contact your system programmer for further analysis.

System programmer response: Verify that the member name exists in the file pointed to by the procedure in use at the time of the error. Review the netlog being used at the time of the error to see whether there was a system error that could have prevented the member from being found.

Classes: None.

AOF040I CONTROL FILE INACTIVE

Explanation: A request to the automation policy was not successful because the automation control file was not active. This message can also be a response to the ACF STATUS command.

System action: None.

Operator response: To activate the automation control file, specify the ACF REFRESH or the INGAMS command.

System programmer response: None.

Classes: None.

AOF041I UNABLE TO FIND *entry type*

Explanation: The task did not find any data for the specified entry-type pair.

The variable *entry* shows the entry field of the entry-type pair the task tried to locate.

The variable *type* shows the type field of the entry-type pair the task tried to locate.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF042I MEMBER *name* NOT FOUND

Explanation: SA z/OS attempted to prepare the member named in the message but could not find it.

The variable *name* is the name of member that could not be found.

Operator response: None.

System programmer response: Make sure that the member identified in the message is valid. Add the member in the DSIPARM concatenation chain or correct the SDF panel definitions that refer to the erroneous member.

System action: Processing continues.

AOF043I *task* IS ACTIVE

Explanation: The task *task* has been activated.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF044I *task* IS TERMINATING

Explanation: The task *task* is ending.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF045I *task* TERMINATED

Explanation: The task *task* has ended.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF057I REQUEST FAILED. TIMERID =*timer_ID* ALREADY EXISTS

Explanation: An attempt was made to add a timer ID that already exists.

The variable *timer_ID* shows the timer ID that already exists.

System action: None.

Operator response: Specify a timer ID that does not already exist.

System programmer response: None.

Classes: None.

AOF058I AOFCTMSG ERROR - *message_number*

Explanation: The status display facility attempted to issue the message indicated, but no text could be found for the message in its internal message table (AOFCTMSG).

The variable *message_number* shows the number of the message that text was missing for.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Determine what operation was being performed when the error occurred, and note the message number. This information should be forwarded to your IBM product support contact.

Classes: None.

AOF059I REQUEST FOR *name* LOCKS TO BE CLEARED WAS CANCELED

Explanation: The request to clear locks was canceled. No locks were cleared.

The variable *name* shows the subsystem name whose locks were to be cleared.

System action: No locks were cleared.

Operator response: None.

System programmer response: None.

Classes: None.

AOF060I OPERATOR ID *operator_ID* NOT FOUND

Explanation: The operator ID identified in this message (*operator_ID*) was not found.

System action: None.

Operator response: Specify the operator ID again.

System programmer response: None.

Classes: None.

AOF064I *time: ident* WAITING FOR *event*

Explanation: Automation is waiting for an event to complete.

The variable *time* shows the time the message was issued.

The variable *ident* shows the CLIST that issued the message.

The variable *event* shows the event the CLIST is waiting for.

When *event* is "ENVIRON SETUP EXITS TO FINISH" it means that the automation initialization-related processing is suspended. Automation initialization will not complete until the exits have run.

System action: System automation waits until the event has completed.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF069I *data*

Explanation: This message is given in conjunction with a subsequent error message that explains what is incorrect. It shows the SDF panel statement that is incorrect.

The variable *data* identifies the buffer data in error.

System action: None.

Operator response: None.

System programmer response: Correct the panel definitions.

Classes: None.

AOF076I REPEAT FIND WILL NOT WORK UNTIL YOU HAVE ISSUED A FIND

Explanation: To use the repeat function a find command must have been entered previously.

System action: The requested command is not performed.

Operator response: Specify a find command first, for example, FIND abc.

System programmer response: None.

Classes: None.

AOF077I USE P OR N TO SPECIFY FIND DIRECTION

Explanation: An invalid value for the find direction has been specified. P and N are valid values.

System action: The requested command is not performed.

Operator response: Specify P (previous) or N (next) for the direction of the FIND command, for example, FIND abc N.

System programmer response: None.

Classes: None.

AOF078I *subsys* SPOOL RECOVERED TO BELOW *nn%*, *text*

Explanation: System automation SPOOL recovery was successful.

The variable *subsys* shows the name of the subsystem.

The variable *nn%* shows the spool percentage.

The variable *text* shows either RECOVERY STOPPED or RECOVERY DONWGRADED TO SHORT.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF081I SECURITY FACILITY IS NOT ACTIVE

Explanation: A request or function was not performed, because the security facility was not active.

System action: None.

Operator response: Take steps to activate the security facility. Contact your system programmer.

System programmer response: Determine why the security facility was not active and take steps to ensure that it is available when necessary.

Classes: None.

AOF082I SAVE REQUEST REJECTED – ORIGINAL DATA HAS BEEN MODIFIED

Explanation: A request to save changed data was issued. However, the original data that you have changed has in the meantime been modified by another process and is no longer up-to-date. The request is therefore rejected.

System action: The requested function is not performed.

Operator response: Refresh the displayed data and redo the changes based on the new data.

System programmer response: None.

Classes: None.

AOF083I FIELD *field_name* EXCEEDS THE ALLOWABLE FIELD LENGTH

Explanation: The field identified in this message (*field_name*) is too long.

System action: None.

Operator response: Determine the correct length for the field and try the operation again.

System programmer response: None.

Classes: None.

AOF084I PARAMETERS EXCEED THE MAXIMUM LENGTH

Explanation: The command could not be processed because the parameters exceeded the maximum length.

System action: None.

Operator response: Try to reduce the length of the parameters.

System programmer response: None.

Classes: None.

AOF085I PRESS PF5 TO SAVE OR PF3 AGAIN TO IGNORE CHANGES

Explanation: Function key PF3 has been issued to leave a panel without having saved temporary modifications of the displayed data. The requested function is rejected.

System action: The requested function is not performed.

Operator response: Follow the instruction of the message.

System programmer response: None.

AOF086I *time* : *phase* DEFINITIONS TO *subsystem* RESTORED BY *task*

Explanation: This message is written to the NetView log when previously stored start or stop definitions to an application have been restored after they have been modified.

The variable *time* shows the time this message was generated.

The variable *phase* shows whether START or STOP definitions have been modified.

The variable *subsystem* shows the name of the resource.

The variable *task* shows the name of the task that requested the function.

System action: Processing continues.

Operator response: None.

System programmer response: None.

AOF087I *action* REQUEST REJECTED - *phase* IN PROGRESS

Explanation: A request was issued to save modified start or stop definitions for an application or to discard changes by resetting the previously loaded definitions.

The variable *action* shows whether a save or reset function has been requested.

The variable *phase* shows the reason why the requested function was rejected:

START

A startup of the application is in progress.

STOP

A shutdown of the application is in progress.

INITIALIZATION

SA z/OS is not initialized.

System action: The requested function is not performed.

Operator response: Reissue the request either after the start or stop of the application has completed or after SA z/OS has initialized.

System programmer response: None.

AOF088I *command*: COMMAND ONLY ALLOWED IN FULLSCREEN MODE

Explanation: A command was issued that is only allowed in fullscreen mode. The command is rejected because the executing task does not support fullscreen mode.

The variable *command* shows the command that was issued.

System action: None.

Operator response: None.

System programmer response: None.

AOF089I *time* : *phase* **DEFINITIONS TO** *subsystem*
MODIFIED BY *task*

Explanation: This message is written to the NetView log when start or stop definitions to an application have been modified during runtime.

The variable *time* shows the time this message was generated.

The variable *phase* shows whether START or STOP definitions have been modified.

The variable *subsystem* shows the name of the resource.

The variable *task* shows the name of the task that requested the function.

System action: Processing continues.

Operator response: None.

System programmer response: None.

AOF090I **IPLDATE=** *date*, **IPLTIME=** *time*

Explanation: This message shows the date and time of the last initial program load (IPL).

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF094I *operator_ID* **IS NOT AUTHORIZED TO**
USE INGNIFY COMMAND

Explanation: An operator dialog issues this message. An operator who is not authorized to use the INGNIFY command attempted to do so.

The variable *operator_ID* shows the ID of the operator who attempted to use the command.

System action: None.

Operator response: If you need more information about this message, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF095I *function_name* **FUNCTION**
SUCCESSFULLY COMPLETED

Explanation: An operator dialog issues this message. Processing of a function called by a command has been successfully completed.

The variable *function_name* identifies the function that processing has been completed for.

System action: None.

Operator response: None. For more information about

this message, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF096I **A COMMAND MUST BE SPECIFIED**

Explanation: You are expected to enter a command.

System action: None.

Operator response: Enter a command.

System programmer response: None.

Classes: None.

AOF097I **A RESOURCE NAME IS REQUIRED**
FOR THIS SELECTION

Explanation: You made an operator dialog panel selection that requires a resource name, but you did not enter a resource name.

System action: None.

Operator response: Enter a resource name. For more information, request help for this operator dialog panel.

System programmer response: None.

Classes: None.

AOF098I **COMMAND FAILED - REFER TO**
NETVIEW LOG

Explanation: Processing of a request or function failed to be completed successfully. The netlog file active at the time of the failure might contain information indicating why processing failed.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Review the netlog for messages related to the request that failed. Determine the cause of the failure by analyzing these messages. If there are no related messages, determine whether the command is still active on the maintenance level that was active when this message was generated.

Classes: None.

AOF099I **FUNCTION COMPLETED**

Explanation: An operator dialog issues this message. It indicates that processing of the command completed. If no error message is shown previously, it also indicates that the function completed successfully.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF100I *time: command* **COMMAND ISSUED** *data*

Explanation: The request you made has been processed or has been issued and is processing.

The variable *time* shows the time this message was generated.

The variable *command* shows the name of the command that failed.

The variable *data* shows detailed information about where the command was issued.

System action: None.

Operator response: None.

System programmer response: None.

Classes. 40 43.

AOF101I **CONFIRMATION RESPONSE IS REQUIRED**

Explanation: An operator dialog issues this message.

System action: None.

Operator response: For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF102I **CORRECT FIELDS IN ERROR**

Explanation: An operator dialog issues this message.

System action: None.

Operator response: For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF103I **CURRENT TIME PAST REQUESTED TIME** *time*

Explanation: A time that has already passed was specified. The specified time cannot be processed.

The variable *time* shows the time specified in the request.

System action: None.

Operator response: Specify a valid time in the request.

System programmer response: None.

Classes: None.

AOF104I **DEFAULT (IF ANY) IN WHITE, NO NEED TO ENTER SEL NUMBER**

Explanation: On the NNT selection panel, the first defined session is the default selection for initiation/termination. It is selected automatically if you do not enter a selection number.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF105I **DELETE SUCCESSFUL**

Explanation: A request to delete is complete.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF106I **ENTER A SELECTION**

Explanation: An operator dialog issues this message.

System action: None.

Operator response: For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF107I **ENTER TO DISPLAY**

Explanation: An operator dialog issues this message. This message is usually displayed after an invalid command is entered.

System action: None.

Operator response: Make a selection from the list displayed on the panel and press the key designated as the Enter key.

System programmer response: None.

Classes: None.

AOF108E **ERROR OCCURRED DURING** *process* **PROCESS**

Explanation: An error occurred during the process identified in this message.

The variable *process* shows the name of the process during which the error occurred.

System action: None.

Operator response: Notify your system programmer.

System programmer response: Examine other messages related to the process to identify the error and its cause.

Classes: None.

AOF109I EXITS CANNOT BE ENABLED - EXITS DO NOT EXIST

Explanation: An attempt was made to enable exits. Because no exits exist, no action was performed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF110I *name* BEING USED FOR THE AUTOMATION CONTROL FILE - *origin*

Explanation: The automation control file member was either loaded, restored, or refreshed successfully.

The variable *name* shows the name of the member that was loaded.

The variable *origin* shows where the automation control file came from. It can be:

LOADED from data sets in the DSIPARM concatenation.

RESTORED using NetView Save/Restore facility.

REFRESHED as for LOADED but from changed data only.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF111I AUTOMATION CONFIGURATION DISPLAY - ENTRY= *entry*

Explanation: This is the first message of a multiline message group that displays the current automation configuration.

The variable *entry* shows the name of the entry that will be displayed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF112I ACTIVE TYPE= *active_type* , DESIRED TYPE= *desired_type*

Explanation: This message is part of a multiline message group that begins with message AOF111I.

The variable *active_type* shows the name of the type that satisfied the request.

The variable *desired_type* shows the name of the type that was requested.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF113I DATA IS *data*

Explanation: This message is part of a multiline message group that begins with message AOF111I.

The variable *data* shows the data area of the entry and type being displayed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF114I REQUESTED STATE *state* FOR SUBSYSTEM *subsystem* IS ALREADY SET

Explanation: An operator dialog issues this message.

The variable *state* shows the requested state.

The variable *subsystem* shows the affected subsystem.

System action: None.

Operator response: For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF115I REQUESTED TIMER *timer_identification* WAS CHANGED

Explanation: The requested timer change was made.

The variable *timer_identification* shows the name of the timer that was changed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF116I REQUESTED TIMERS WERE DELETED

Explanation: The request to delete timers was completed successfully. The timers specified in the request were deleted.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF117I RETURN CODE *return_code* RECEIVED FROM LIST TIMER COMMAND

Explanation: A non-zero return code was generated during processing of the LIST TIMER command. Processing of the request is not complete.

The variable *return_code* shows the return code generated.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Refer to the description of the NetView LIST *TIMER* command in *Tivoli NetView for z/OS Command Reference Vol. 1* for an explanation of the return code.

Classes: None.

AOF118I SECOND PARAMETER SHOULD BE BLANK OR T

Explanation: A non-blank value other than "T" was specified for the second parameter of the entered command. This parameter must be blank or have a value of T.

System action: None.

Operator response: Enter the command again, specifying valid values for all parameters.

System programmer response: None.

Classes: None.

AOF119I SELECT ONE PARAMETER THEN PRESS THE ENTER KEY

Explanation: This message is a prompt for you to select a parameter.

System action: None.

Operator response: Select a parameter and press the Enter key.

System programmer response: None.

Classes: None.

AOF121I HIT ENTER KEY

Explanation: This message is a prompt for you to press the Enter key.

System action: None.

Operator response: Press the key on your keyboard that performs the Enter function.

System programmer response: None.

Classes: None.

AOF122I INVALID CONFIRMATION REQUEST OF *function* ENTERED

Explanation: An operator dialog issues this message. A confirmation request that is not valid was entered.

The variable *function* shows the request entered.

System action: None.

Operator response: Enter a valid confirmation request. For more information about this message, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF123D INVALID INPUT -- ENTER COMMAND, SET COMMAND, T TO TERMINATE OR I TO INITIATE

Explanation: Data that is not valid was entered.

System action: None.

Operator response: Enter one of the following:

- A valid command
- SET
- T
- I

System programmer response: None.

Classes: None.

AOF124D INVALID INPUT -- ENTER IMS COMMAND, SET COMMAND OR T TO TERMINATE

Explanation: Data that is not valid was entered.

System action: None.

Operator response: Enter the data again, making sure that it is valid.

System programmer response: None.

Classes: None.

AOF125A INVALID OPERATOR ID - PLEASE RE-ENTER

Explanation: An operator ID that is not valid was entered.

System action: None.

Operator response: Enter a valid operator ID.

System programmer response: None.

Classes: None.

AOF126A INVALID PASSWORD - PLEASE RE-ENTER

Explanation: A password that is not valid was specified.

System action: None.

Operator response: Enter a valid password.

System programmer response: None.

Classes: None.

AOF129I INVALID SELECTION

Explanation: An operator dialog issues this message.

System action: None.

| **Operator response:** For more information, call help
 | from the panel that this message appears on. Choose an
 | option that is within the valid range.

System programmer response: None.

Classes: None.

AOF130I INVALID SUBSYSTEM NAME
invalid_name

Explanation: An operator dialog issues this message. This message may also be issued during the automation control file load or refresh.

The variable *invalid_name* shows the invalid name.

System action: For automation control file load or refresh the Entry/Type is ignored.

Operator response: For more information, call help from the panel that this message appears on. If the message is issued during the configuration file load or refresh then the Entry/Type in the file must be corrected.

System programmer response: None.

Classes: None.

AOF131I MISSING OR INVALID *parameter*

Explanation: A parameter or keyword in the entered command is missing or is not valid.

The variable *parameter* shows the parameter or keyword that is missing or is not valid.

System action: None.

Operator response: Specify valid parameters and reissue the command.

System programmer response: None.

Classes: None.

AOF132I NO AUTOMATION FLAGS SET

Explanation: An operator dialog issues this message.

System action: None.

Operator response: For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF133I NO ENTRIES HAVE BEEN DEFINED

Explanation: The information that was requested does not exist.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF134I NO FUNCTION WAS SELECTED

Explanation: You did not specify a function.

System action: None.

Operator response: Specify a function.

System programmer response: None.

Classes: None.

AOF135I NO HELP PANEL IS AVAILABLE

Explanation: Help information was requested when none is available.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF136I NO MORE HELP IS AVAILABLE

Explanation: All available help text has been displayed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF137I NO NOTIFY OPERATORS DEFINED

Explanation: The request was not performed, because there are no notify operators.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF138I NO SCHEDULED AUTOMATION OVERRIDES

Explanation: An operator dialog issues this message.

System action: None.

Operator response: For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF139I NO THRESHOLDS DEFINED - AT LEAST ONE THRESHOLD VALUE IS REQUIRED

Explanation: No threshold values have been defined. You must define at least one threshold value.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Define at least one threshold value. For information about defining threshold values, refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

Classes: None.

AOF140I NO TIMERS ARE SCHEDULED

Explanation: No timers are scheduled, so there is nothing to display.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF141I NUMBER 0 IS NOT VALID

Explanation: The value "0" was specified for the field. This is not a valid value for the field.

System action: None.

Operator response: Specify a valid value for the field.

System programmer response: None.

Classes: None.

AOF142I ONLY ONE DATE/DAY MAY BE SPECIFIED

Explanation: More than one day or date was specified. Only one day or date can be accepted.

System action: None.

Operator response: Specify only one day or date.

System programmer response: None.

Classes: None.

AOF143I ONLY ONE SELECTION IS ALLOWED, PLEASE RESPECIFY

Explanation: More than one selection was made. Only one selection can be accepted.

System action: None.

Operator response: Make only one selection.

System programmer response: None.

Classes: None.

AOF144I PARAMETER *parameter_name* INVALID

Explanation: An operator dialog issues this message. A parameter that is not valid (*parameter_name*) was entered.

System action: None.

Operator response: For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF145I PARAMETER MISSING

Explanation: An operator dialog issues this message.

System action: None.

Operator response: For more information, call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF146I PARAMETER MUST BE NUMERIC

Explanation: A non-numeric value was entered in a field that requires a numeric value.

System action: None.

Operator response: Enter a numeric value for the field.

System programmer response: None.

Classes: None.

**AOF147I REPLY FAILED - REFER TO
NETWORK LOG**

Explanation: The reply failed. The netlog contains additional information about the failure.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Make sure the reply specification is correct. Identify and correct any other causes of the failure.

Classes: None.

**AOF148I REPLYID IS NOT VALID, CORRECT
AND RE-ENTER**

Explanation: The reply ID is not valid.

System action: None.

Operator response: Enter a valid reply ID.

System programmer response: None.

Classes: None.

**AOF149I REQUEST FAILED. DATE/TIME (*date*
time) IS INVALID**

Explanation: The date or time specified is not valid.

The variable *date* shows the date specified.

The variable *time* shows the time specified.

System action: None.

Operator response: Specify a valid date and a valid time. Make sure that the day you specify is valid for the month and year you specify, and that the time is specified in 24-hour-clock format.

System programmer response: None.

Classes: None.

**AOF150I STATISTICS DISPLAY REQUESTED
FOR *from_resource* THRU *to_resource***

Explanation: This is the first message in a group of messages that display statistics from a resource structure.

The variable *from_resource* shows the highest resource name in the resource structure requested to be displayed.

The variable *to_resource* shows the lowest resource name in the resource structure requested to be displayed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF151I ID= *resource* , TYPE= *type* , STATUS=
*status***

Explanation: This message is part of a multiline message group that begins with message AOF150I.

The variable *resource* shows the name of the resource being displayed.

The variable *type* shows the type of resource being displayed.

The variable *status* shows the status of the resource being displayed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF152I LAST UPDATED BY *operator_ID* ,
RECORD TYPE= *record_type***

Explanation: This message is part of a multiline message group that begins with message AOF150I.

The variable *operator_ID* shows the ID of the operator who last updated the record for the resource.

The variable *record_type* shows the type of the record that was updated.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF153I LAST *event_type* EVENT: TS= *date time* ,
TYPE=*type***

Explanation: This message is part of a multiline message group that begins with message AOF150I.

The variable *event_type* shows the event type, which is one of START, AVAIL, STOP, or DOWN. The event types describe the lifecycle milestones of the resource.

AOF154I • AOF162I

The variable *date time* shows the time stamp in the form mm/dd/yyyy hh:mm:ss.

The variable *type* shows the start or stop type that is associated with the event.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF154I LAST TERMCODE= *code*

Explanation: This message is part of a multiline message group that begins with message AOF150I.

The variable *code* shows the termination (abend) code.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF155I OPERATOR NOTIFIED: *notify*

Explanation: This message is part of a multiline message group that begins with message AOF150I. This message indicates whether an operator has been notified of the current error.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF156I LAST STATUS CHANGE DATE= *mm/dd/yy* , TIME= *hh.mm.ss* , OPER= *operator_ID*

Explanation: This message shows the date and time of the last status change, and the operator ID at the point of the status change.

The variables *mm*, *dd*, and *yy* show the month, day, and year, respectively, of the last status change.

The variables *hh*, *mm* and *ss* show the hour, minute, and second of the last status change.

The variable *operator_ID* shows the operator ID at the point of the last status change.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF158I USER = *information*

Explanation: Informational message returning up to 20 characters of data.

The variable *information* shows information by user.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF159I NO DATA AVAILABLE

Explanation: This message is part of a multiline message group that begins with message AOF150I. There is no error data available for the resource.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF160I ERROR COUNT DATE TIME

Explanation: This message is part of a multiline message group that begins with message AOF150I. This message provides the headings for the data shown in message AOF161I.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF161I *nn mm/dd/yy hh:mm:ss*

Explanation: This message is part of a multiline message group that begins with message AOF150I. This message shows the current error count (*nn*) and the date and time of the last error.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF162I INVALID SCROLL AMOUNT - SPECIFY "HALF", "PAGE", "MAX", OR A NUMBER

Explanation: An invalid value was entered in the scroll field.

System action: None.

Operator response: Correct the scroll amount and retry.

System programmer response: None.

Classes: None.

**AOF163I LAST THRESHOLD EXCEEDED -
 threshold**

Explanation: This message is part of a multiline message group that begins with message AOF150I. The last threshold setting for the resource has been exceeded.

The variable *threshold* shows the last threshold setting.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF165I REQUEST FAILED. TASK *task_name*
 NOT ACTIVE**

Explanation: The task identified in this message must be active for processing of the request to be completed successfully. Because this task was not active, the request failed.

The variable *task_name* shows the name of the task that was not active.

System action: None.

Operator response: Contact your system programmer. Start the task identified by *task_name* and issue the original request again.

System programmer response: Determine why the task identified by *task_name* was not active and correct the problem.

Classes: None.

**AOF166I REQUEST FOR *number* LOCKS TO BE
 CLEARED WAS SUCCESSFULLY
 COMPLETED**

Explanation: A request to clear locks was successful.

The variable *number* shows the number of locks cleared.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF169I THE LAST LINE IS ALREADY
 DISPLAYED ON THE SCREEN**

Explanation: You tried to scroll forward to display more data on the screen, but there is no more data to display; the last line of data is already being displayed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF170I THE TOP LINE IS ALREADY
 DISPLAYED ON THE SCREEN**

Explanation: You tried to scroll backward to display more data on the screen, but there is no more data to display; the first line of data is already being displayed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF171I TIMEOUT OF *number* SECONDS
 EXPIRED**

Explanation: Processing of the request was not completed within the number of seconds shown by this message.

The variable *number* shows the number of seconds taken by processing of the request before processing was suspended.

System action: Normal system processing proceeds; processing of the request was suspended.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF172I TIMER ID MUST BE CHANGED FOR
 ADD PROCESSING**

Explanation: The timer ID was not changed for an add operation.

System action: None.

Operator response: Specify the timer ID again.

System programmer response: None.

Classes: None.

**AOF173I TIMER SCHEDULED BUT TIMER
 DISPLAY FAILED**

Explanation: The requested time was set. However, in an attempt to verify the request by displaying the timer ID, the NetView LIST *TIMER* command failed.

System action: None.

Operator response: Verify that your requested timer was set.

System programmer response: None.

Classes: None.

**AOF174I UNKNOWN ERROR OCCURRED,
REFER TO NETWORK LOG FOR
DETAILS**

Explanation: An unexpected return code was received, indicating an error condition. Additional information about the error condition can be found in the NetView log.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Check the netlog for messages related to the request that resulted in the error condition.

Classes: None.

**AOF175I YOU MUST ENTER YOUR
PASSWORD**

Explanation: A password was not entered.

System action: None.

Operator response: Enter your password.

System programmer response: None.

Classes: None.

AOF176I YOU MUST ENTER YOUR USER ID

Explanation: A user ID was not specified.

System action: None.

Operator response: Enter your user ID.

System programmer response: None.

Classes: None.

**AOF177I YOUR REQUESTED NUMBER IS NOT
ON THE LIST**

Explanation: Your selection number is greater than the number in the list of valid selections on this panel.

System action: None.

Operator response: Select a number that is included in the selection list.

System programmer response: None.

Classes: None.

**AOF179I REQUEST NOT COMPLETED - WAIT
TIME EXPIRED**

Explanation: Processing of the request was not complete after the length of time specified as the "wait time." Processing of the request is suspended. Other processing continues.

System action: Processing of the request might be completed after this message is issued or the request is incomplete.

Operator response: Retry your request or determine whether it subsequently completes.

System programmer response: None.

Classes: None.

**AOF180I REQUEST NOT COMPLETED BY
command_name COMMAND FAILURE**

Explanation: Processing of a request was not completed because the command identified in this message failed.

The variable *command_name* shows the name of the command that failed.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Check the netlog for related error messages to determine the reason why the command failed.

Classes: None.

**AOF181I MULTIPLE SELECTIONS NOT
ALLOWED**

Explanation: A command entered on the selection line was interpreted as an attempt to select more than one option. Only one option can be selected.

System action: None.

Operator response: Select one of the options displayed on the panel and press the key designated as the Enter key.

System programmer response: None.

Classes: None.

**AOF186I *parameter* PARAMETER SPECIFIED
MORE THAN ONCE**

Explanation: A parameter was used in a command more than once, or was specified more than once for a subsystem definition in the automation control file.

The variable *parameter* shows the duplicated parameter.

System action: The entered command is not processed. For automation control file processing the parameter is ignored.

Operator response: Enter the command again, using the correct syntax.

System programmer response: None.

Classes: None.

AOF187I SPECIFIED OPERAND *operand*
INVALID FOR *parameter* FOR A CROSS
DOMAIN REQUEST TO *cross_domain*

Explanation: The specified operand (*operand*) cannot be used with the specified parameter (*parameter*) for a cross domain request to the specified cross domain (*cross_domain*).

System action: The entered command is not processed.

Operator response: Enter the command again, using the correct syntax.

System programmer response: None.

Classes: None.

AOF188I FROM= AND TO= PARAMETERS ARE
INVALID FOR REQ=REPL|DEL
REQUESTS

Explanation: The FROM parameter and TO parameter specified in the command are not valid for replace or delete requests.

System action: The entered command is not processed.

Operator response: Enter the command again, using the correct syntax.

System programmer response: None.

Classes: None.

AOF189I FOR REQ=DEL THE ONLY VALID
PARAMETER IS: ID=

Explanation: A parameter other than ID= was entered. This is the only valid parameter.

System action: The entered command is not processed.

Operator response: Enter the command again, using the correct syntax.

System programmer response: None.

Classes: None.

AOF199I FOR REQ=DISP THE ONLY VALID
PARAMETERS ARE: ID=, FROM=, and
TO=

Explanation: A parameter other than ID=, FROM=, or TO= was entered. These are the only valid parameters.

System action: The entered command is not processed.

Operator response: Enter the command again, using the correct syntax.

System programmer response: None.

Classes: None.

AOF200I *time*: COMMAND *init_cmd* FAILED.
RECEIVED RETURN CODE *rr* FROM
fail_cmd

Explanation: A command issued by the initiating command ended with a non-zero return code.

The variable *time* shows the time that this message was generated.

The variable *init_cmd* shows the name of the initiating command.

The variable *rr* shows the condition code that was returned by the failing command.

The variable *fail_cmd* shows the name of the failing command.

System action: Processing is terminated.

Operator response: Notify your system programmer.

System programmer response: Determine why the command failed and correct the problem.

Classes: 0.

AOF201I *time* : CLIST *clist_name*: *msg_input*

Explanation: An error occurred during processing of a command list.

The variable *time* shows the time this message was generated.

The variable *clist_name* shows the name of the command list that was being processed when the error occurred.

The variable *msg_input* describes the error.

System action: Processing of the command list ends with an appropriate return code.

Operator response: Contact your system programmer.

System programmer response: Perform problem determination on the command.

Classes: 0.

AOF203I *clist_name* INVOKED BY
UNAUTHORIZED USER (*user_ID*).
REQUEST DENIED: *request*

Explanation: A command list was called by a user who is not authorized to do so. The request to process the command list was denied.

The variable *clist_name* shows the command list that was called.

The variable *user_ID* shows the ID of the user who called the command list.

The variable *request* shows the request that accompanied the call the command list.

System action: The command list identified in the message is not processed.

Operator response: If you feel that you should be authorized for this command list, request authorization from your system programmer.

System programmer response: Evaluate the need the operator has for this function. If authorization should be granted, notify the security administrator.

Classes: 0.

AOF204I *time:* EXPECTED PARAMETERS
MISSING OR INVALID FOR REQUEST
clist_name - parameter_name

Explanation: One or more parameters that must be passed to the command list identified in this message are missing or are not valid.

The variable *time* shows the time this message was generated.

The variable *clist_name* shows the name of the command list that the variables must be passed to.

The variable *parameter_name* shows the name of the parameter that was missing or not valid.

System action: None.

Operator response: Make sure that the syntax is correct and enter the request again. The netlog will show both the message and the request as processed. Notify your system programmer if the request fails again.

System programmer response: Review the user input to determine the validity of the request and instruct the operator on the proper procedure. If the user input is correct, set debugging on for the command list to determine the source of the error.

Classes: 0.

AOF205A *time : command* COMMAND FAILED
FOR *clist_name: interval - WAIT TIME*
EXPIRED

Explanation: Processing of the command identified by this message failed because it took too much time. The timer interval for command processing was exceeded.

The variable *time* shows the time this message was generated.

The variable *command* shows the name of the command that failed.

The variable *clist_name* shows the name of the command list that was requested.

The variable *interval* shows the timer interval that was exceeded.

System action: None.

Operator response: Check the netlog that was active when the message was received to see whether other

command timers have expired. If no other timers have expired, review the syntax for the command and issue the command again. If other command timers have been exceeded or if processing of the command fails again, notify your system programmer.

System programmer response: Review the user input to determine the validity of the request and instruct the operator on the proper procedure. If the user input is correct, review the netlog that was active at the time of the message to see whether there is a reason for delays in command list processing. Note the value of the timer interval to see whether the timer default should be changed.

Classes: 0, 40, 44.

AOF206I *time :* "command" COMMAND FAILED
FOR *clist_name: RECEIVED received_ID*
"received_value"

Explanation: Processing of the command identified in this message failed. The command list identified in this message received a return code or a message indicating the problem.

The variable *time* shows the time this message was generated.

The variable *command* shows the command that was processed.

The variable *clist_name* shows the command list that was requested.

The variable *received_ID* shows either a message ID or the text string "RETURNCODE=".

The variable *received_value* shows the message text or the return code.

System action: None.

Operator response: Notify your system programmer.

System programmer response: Check the netlog for further information. Perform problem determination on the command list identified in this message.

Classes: 0.

AOF207E *time:* NO DEFAULTS SET FOR
resource_name, CLIST HALTED

Explanation: Command list processing has been halted because automation defaults have not been set for the resource identified in this message.

The variable *time* shows the time this message was generated.

The variable *resource_name* shows the name of the resource that automation defaults are not defined for.

System action: The process is ended.

Operator response: Contact your system programmer.

System programmer response: Correct the automation

control file entries for the resource identified in this message.

Classes: 0.

AOF211E *time:* NO AUTOOPS ENTRIES HAVE BEEN DEFINED - AUTOMATION CANNOT CONTINUE

Explanation: No automated operator (AUTOOPS) entries have been defined in the automation control file for the automation environment.

The variable *time* shows the time this message was generated.

System action: Automation initialization is ended.

Operator response: Notify your system programmer.

System programmer response: Add the appropriate automated operator entries in the automation control file using the customization dialogs. See *IBM Tivoli System Automation for z/OS Defining Automation Policy* for more information.

| In addition, since processing of the AUTOOPS
| ENTRIES is performed on task AUTO2, verify AUTO2
| is active and available (and not scheduled to process
| other work that might prevent SA z/OS from using it).

Classes: 0.

AOF216I *time: command* YOUR COMMAND CANNOT BE PROCESSED - THE AUTOMATION ENVIRONMENT HAS NOT BEEN INITIALIZED

Explanation: The command that was entered cannot be processed because the automation environment has not been completely initialized.

The variable *time* shows the time when this message was generated.

The variable *command* shows the command that was not processed, or the name of the REXX script processing the command.

System action: The command is not processed.

Operator response: If the automation environment has been completely initialized, message AOF540I is issued. Then reenter the command.

System programmer response: None.

Classes: None.

AOF217I *time: command* IS NOT SUPPORTED

Explanation: The command that was entered cannot be processed because this command is restricted for licensed SA z/OS usage.

The variable *time* shows the time when this message was generated.

The variable *command* shows the command that was not processed, or the name of the clist processing the command.

System action: The command is not processed.

Operator response: None.

System programmer response: None.

Classes: None.

AOF219I "OPID=" ONLY VALID WHEN "STATUS=" IS SPECIFIED

Explanation: You can change the operator ID (OPID) only when the status is specified to be changed.

System action: The command is not processed.

Operator response: Enter the command again and specify both the STATUS= and OPID= parameters or do not specify the OPID= parameter.

System programmer response: None.

Classes: None.

AOF221I *time:* NO PRIMARY FOCAL POINT DOMAIN HAS BEEN DEFINED - DEFAULT OF *domain_ID* USED

Explanation: A primary focal point domain was not defined in the automation control file member. The domain ID shown by the variable *domain_ID* will be assigned to the primary focal point.

The variable *time* shows the time this message was generated.

System action: The current domain is assigned as the primary focal point.

Operator response: None.

System programmer response: If the default value is not desired, use the customization dialogs to correct the automation control file member. For more information, refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

Classes: None.

AOF222I *time:* NO BACK UP FOCAL POINT DOMAIN HAS BEEN DEFINED - DEFAULT OF *domain_ID* USED

Explanation: A backup focal point domain was not defined in the automation control file member. The domain ID shown by the variable *domain_ID* will be assigned to the backup focal point.

The variable *time* shows the time this message was generated.

System action: The current domain is assigned as the backup focal point.

Operator response: None.

System programmer response: If you do not want the default backup focal point domain to be used, use the customization dialogs to specify a backup focal point domain in the automation control file member. For more information, refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

Classes: None.

**AOF223I SETSTATE REJECTED - SUBSYSTEM
name INVOLVED IN A SHUTDOWN**

Explanation: A SETSTATE command has been requested for a subsystem but it or one of its dependants is in the process of shutting down.

The variable *name* shows the subsystem involved in the shutdown.

System action: The request is denied.

Operator response: Wait for the shutdown to complete then reissue the SETSTATE request.

System programmer response: None.

Classes: None.

**AOF224I time: JOB jobname HAS BEEN DEFINED
TO RUN ONLY ONCE - STATE NOT
SET**

Explanation: A transient job has been defined to run only once and a request was made for its status to be changed.

The variable *time* shows the time this message was generated.

The variable *jobname* shows the job involved.

System action: Request rejected for this job.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

**AOF225I DUPLICATE KEYWORD keyword
FOUND FOR SUBSYSTEM subsystem.
FIRST OCCURRENCE USED**

Explanation: While loading an automation control file, a duplicate keyword was found for a subsystem, its value was ignored.

The variable *keyword* shows the duplicated keyword.

The variable *subsystem* is the name of the subsystem containing the keyword.

System action: The first occurrence of the keyword value is used.

Operator response: None.

System programmer response: Remove the duplicate entry from the automation control file.

Classes: 40, 43.

AOF227I UNABLE TO PROCESS command - reason

Explanation: System automation is not able to process the requested action.

The variable *command* shows the affected command.

The variable *reason* gives an explanation.

System action: Processing stops.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF229I time: NO ENTRY FOR resname (entry
type) HAS BEEN DEFINED - DEFAULT
OF value USED**

Explanation: The system type was not defined in the automation control file.

The variable *time* shows the time this message was generated.

The variable *resname* shows the name of the keyword being defined.

The variable *entry* shows the name of the entry in the automation control file member.

The variable *type* shows the name of the type in the automation control file member.

The variable *value* shows the value assigned to the entry within the specified resource.

System action: Default value is assigned.

Operator response: None.

System programmer response: If the default value is not desired, use the customization dialogs to correct the automation control file member. For more information, refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

Classes: None.

**AOF233I SHUTDOWN REJECTED -
SUBSYSTEM name INACTIVE**

Explanation: The subsystem you are trying to shutdown is inactive.

The variable *name* shows the name of the subsystem.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF240I REQUESTED TIMER *timerid* WAS ADDED

Explanation: This message is a response to the TIMER ADD command. Processing of the command was completed successfully.

The variable *timerid* shows the ID of the timer.

System action: The timer is scheduled, with the requested name and function.

Operator response: None.

System programmer response: None.

Classes: None.

AOF241I *time*: EXPECTED PARAMETERS MISSING OR INVALID FOR REQUEST *clist_name* - PARAMETER *parm_list*

Explanation: One or more of the parameters that needed to be passed to a command list during shutdown processing were missing or were not valid.

The variable *time* shows the time this message was generated.

The variable *clist_name* identifies the requested command list.

The variable *parm_list* lists all of the parameters that were missing or were not valid.

System action: The shutdown process ends.

Operator response: Make sure valid parameters are entered.

System programmer response: Make sure valid parameters are entered. Check the parameters in the automation control file.

Classes: 0.

AOF244I *time*: SHUTDOWN PROCESSING ON *restype resname* (JOB *job_name*) SUSPENDED FOR *interval* MINUTES

Explanation: Shutdown processing has been suspended for the job identified by the variable *job_name* for the number of minutes shown by the variable *interval*. This allows users of the job to end their activities before the job is shut down.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource.

The variable *resname* shows the name of the resource.

System action: The Shutdown process begins again after the number of minutes specified in this message.

Operator response: None.

System programmer response: None.

Classes: 40, 43, 44, 46.

AOF245I *time*: *restype resname* (JOB *job_name*) STATUS IS *status* - SHUTDOWN WILL BYPASS ISSUING *msgtype* COMMANDS

Explanation: The shutdown processing has determined that the job identified by the variable *job_name* is already shutting down. The shutdown processing will bypass issuing further commands.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource.

The variable *resname* shows the name of the resource.

The variable *status* shows the status of the resource.

The variable *msgtype* shows the command type to be issued (SHUTINIT,SHUTDOWN).

System action: The status of the job is either stopping or abending. The job is checked to see whether it is down, so that shutdown processing may complete. If the job does not end, manual intervention will be required.

Operator response: None. This message is sent to the netlog only.

System programmer response: Check why the job was in stopping or abending status.

Classes: 40, 43.

AOF246A *time*: SHUTDOWN PROCESSING FAILED FOR SUBSYSTEM *subsystem_name* - SHUTDOWN MAY NOT COMPLETE SUCCESSFULLY

Explanation: An error occurred during shutdown processing. A message describing the error was issued previously.

The variable *time* shows the time when this message was generated.

The variable *subsystem_name* shows the subsystem that the message was for.

System action: None.

Operator response: See the message describing the error that was issued previously to determine why shutdown processing failed.

System programmer response: None.

Classes: 40, 43.

AOF247E *time*: SHUTDOWN PROCESSING FAILURE FOR *restype resname* (JOB *job_name*) - ALL COMMANDS/REPLIES MAY NOT HAVE BEEN ISSUED - *command_type error_type* FOR PASS

AOF249E • AOF255I

pass_number HAD ERROR

Explanation: Shutdown processing has failed for the subsystem identified by the resource name (*resname*) and the job name (*job_name*).

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource. *command_type* shows the command type to be issued (SHUTDOWN,SHUTDOWN).

The variable *error_type* shows whether a command or reply had errors.

The variable *pass_number* shows the pass number in error.

System action: The shutdown processing ends.

Operator response: Check for the type of errors identified by *error_type* and contact your system programmer.

System programmer response: Review the automation control file, looking for the job identified by this message. Correct the errors observed by the operator. Look for message AOF206I in the netlog that was active at the time this message was issued. Message AOF206I contains information you can use to determine why processing of the command or reply failed. The syslog contains commands and replies that can be checked against what should have been issued.

Classes: 40, 43.

AOF249E *time: resource* CGLOBAL *cglobal* WAS NOT PROPERLY INITIALIZED

Explanation: The *cglobal* variable identified in this message was not properly initialized for the resource identified in this message.

The variable *time* shows the time this message was generated.

System action: Current[®] processing might not be completed successfully.

Operator response: Review the netlog that was active at the time of this message for related messages that help define the problem. Contact your system programmer.

System programmer response: Review the initialization procedures and the control file to determine why the *cglobal* variable is not being initialized properly. You might need to set debugging on to identify the problem. Correct the problem.

Classes: None.

AOF251I SHUTDOWN *type* REJECTED - *subsystem_type* ALREADY IN PROGRESS

Explanation: A shutdown has been requested but a shutdown is currently executing involving this

subsystem or one of its children.

The variable *type* shows the shutdown type.

The variable *subsystem_type* shows the name of the subsystem.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF252I *time: TIMER ID = timer_ID* NOT SCHEDULED - *reason*

Explanation: The timer identified by the variable *timer_ID* could not be scheduled.

The variable *time* shows the time this message was generated.

The variable *reason* gives an explanation of failure, such as TASK NOT ACTIVE.

System action: The timer identified in the message is not scheduled.

Operator response: Find the reason the timer was not scheduled, correct the problem, and issue the time request again.

System programmer response: None.

Classes: 40, 43.

AOF253I *time: REQUESTED TIMER = timer_ID* NOT FOUND

Explanation: This message is a response to an attempt to delete (purge) a timer (identified by the variable *timer_ID*). The timer to be deleted could not be found.

The variable *time* shows the time this message was generated.

System action: Processing continues.

Operator response: Verify that the timer does not exist by refreshing the timer display.

System programmer response: None.

Classes: 0, 40, 43.

AOF255I *time: EXIT exit_name* IN *entry type* IS INVALID, RC=*return_code*

Explanation: An exit name that is not valid (shown by the variable *exit_name*) was found when calling user exits.

The variable *time* shows the time this message was generated.

The variable *entry* identifies the entry field in the automation control file where the exit was defined.

The variable *type* identifies the type field in the automation control file where the exit was defined.

The variable *return_code* shows a return code that identifies why the exit was not valid: If the return code is 04, the exit name was not 1 to 8 characters long; if the return code is 08, the exit name was not a valid module or command list.

System action: The exit code for the routine that is not valid is bypassed.

Operator response: Contact your system programmer.

System programmer response: Correctly define the exit routine or remove it from the exit list.

Classes: 0, 40, 44.

AOF256I *time*: OPTIONAL EXITS WERE NOT FOUND FOR *resource_name* - EXIT CODE WAS NOT INVOKED (*resource_type*)

Explanation: No optional exits were found for the resource identified in this message. The exit code was not called.

The variable *time* shows the time this message was generated.

The variable *resource_name* shows the name of the resource.

The variable *resource_type* shows the type of the resource.

System action: Processing continues, using default actions.

Operator response: Contact your system programmer.

System programmer response: Review the control file entry for the resource identified in this message to determine why exits were not found.

Classes: 0, 40, 44.

AOF262E MESSAGE ID *message_ID* INVALID, MUST BE "NNN", "ABCNNN" OR "ABCDNNN"

Explanation: The specified message ID is invalid. Three message ID forms are valid:

- Three decimal digits (*NNN*)
- A three-character prefix followed by three decimal digits (the prefix may contain numeric values)
- A four-character prefix followed by three decimal digits (the prefix may contain numeric values).

The variable *message_ID* shows the message ID specified as the second parameter of the AOCMSG command. This message ID is not valid.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Correct the message ID specified as the second parameter of the AOCMSG command.

Classes: None.

AOF263I MESSAGE ID NUMERIC *message_ID* IS NOT NUMERIC

Explanation: The specified message ID is not valid because it does not end with three numeric digits.

The variable *message_ID* shows the message ID specified as the second parameter of the AOCMSG command. This message ID is not valid.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Correct the message ID specified as the second parameter of the AOCMSG command.

Classes: None.

AOF264I TOO FEW PARAMETERS ON AOCMSG COMMAND, 2 IS MINIMUM

Explanation: The AOCMSG command was issued without one or more required parameters. At least two parameters must be specified for the command to be processed successfully.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Make sure that the coding of the AOCMSG command specifies a message ID value as the second parameter. If you do not want to specify a value for the first parameter, code a comma in its position as a place holder (AOCMSG ,001).

Classes: None.

AOF271I *clist_name*: DATATYPE "*data_type*" FOR entry NOT KNOWN, IGNORED

Explanation: The *data_type* keyword specified in entry *entry* is not recognized. It is being ignored.

The variable *clist_name* shows the name of the command list that generated this message.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40.

AOF282I *clist_name*: SDF DATA LOADED

Explanation: The status display facility (SDF) portion of system automation initialization is complete.

The variable *clist_name* shows the name of the command list that generated this message.

System action: None.

AOF283I • AOF291I

Operator response: None.

System programmer response: None.

Classes: None.

AOF283I *clist_name: number* **SUBSYSTEMS
CURRENT STATUS PRIMED**

Explanation: This message reports the number of subsystems processed.

The variable *clist_name* shows the name of the command list that generated this message.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF284I *clist_name: number* **NOTIFY OPERATORS
READIED**

Explanation: This message reports the number of notification operators that have been processed and initialized.

The variable *clist_name* shows the name of the command list that generated this message.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF286I *clist_name:* **SUBSYSTEM STATUS
VERIFIED**

Explanation: Subsystem status information has been verified with the active cglobal variables.

The variable *clist_name* shows the name of the command list that generated this message.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF287I *clist_name:* **COMPLETE**

Explanation: Processing of the command list has been completed normally.

The variable *clist_name* shows the name of the command list that generated this message.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF288I *clist_name: subsystem_name* **JOBNAME IS
DUPLICATE, SUBSYSTEM IGNORED**

Explanation: Subsystem *subsystem_name* has a job name that is a duplicate. This subsystem is ignored.

The variable *clist_name* shows the name of the command list that generated this message.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Make sure that the subsystem *subsystem_name* does not have a duplicate job name.

Classes: 40.

AOF290I *clist_name:* **SUBSYSTEM -
subsystem_name SCHEDULING
SUBSYSTEM scheduling_subsystem_name
INVALID**

Explanation: The *scheduling_subsystem_name* is not defined as a valid Job Entry Subsystem, and therefore cannot be used to schedule subsystem *subsystem_name*.

The variable *clist_name* shows the name of the command that generated this message.

System action: The primary JES subsystem will be used.

Operator response: Contact your system programmer.

System programmer response: Correct the subsystem definition to identify a subsystem of type JES2 or JES3.

Classes: 40.

AOF291I *clist_name:* **NO JES2 OR JES3
SUBSYSTEMS DEFINED; SUBSYS =
subsystem_name SET**

Explanation: No subsystems were defined as being type JES2 or JES3, and therefore the validity of the subsystem identified as the primary Job Entry Subsystem (JES) cannot be validated.

The variable *clist_name* shows the name of the command list that generated this message.

System action: The identified subsystem will be used without validation.

Operator response: Contact your system programmer.

System programmer response: Correct the subsystem definition to properly identify the one or more Job Entry Subsystems, and verify that the correct subsystem is identified as the primary JES.

Classes: 40.

AOF293I *time*: THERE IS NO OUTSTANDING
REPLY STORED BY SA z/OS FOR
SUBSYSTEM *sname*.

Explanation: System automation cannot determine from its CGLOBALS the reply ID for this subsystem.

The variable *time* shows the time the error message was generated.

The variable *sname* shows the subsystem to be replied to.

System action: Processing continues.

Operator response: Notify your system programmer.

System programmer response: Check why the reply ID is not stored by SA z/OS.

Classes: 40, 43, 44.

AOF294I SPOOL *type* RECOVERY COMMANDS
EXHAUSTED, SPOOL TGS IS NOT
RELIEVED, OPERATION
INTERVENTION IS REQUIRED

Explanation: While attempting to recover from a spool shortage, SA z/OS ran out of defined recovery commands. User has chosen not to reissue the spool commands.

The variable *type* shows the type of spool problem being recovered.

System action: The spool recovery has stopped.

Operator response: You need to have a look at your spool usage .SA z/OS has issued all the defined spool recovery passes and the spool TGS is not relieved. Manual intervention by operators is required.

System programmer response: Review your spool recovery commands.

Classes: 40, 43, 44.

AOF296I TARGET AND RESOURCE_SYSTEM
SPECIFICATION ARE DIFFERENT
(TARGET IGNORED)

Explanation: The target and resource system specification in the INGREQ command are different. Therefore the target specification has been removed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF300I A REPLY MUST BE SPECIFIED

Explanation: You must specify a reply.

System action: None.

Operator response: Specify a reply.

System programmer response: None.

Classes: None.

AOF302I *time*: REQUEST *clist_name* action BY
oper_id IS COMPLETED FOR *resname*

Explanation: The request to set a function ON or OFF for the specified resource was successful.

The variable *time* shows the time this message was generated.

The variable *clist_name* shows the name of the requested command list or command.

The variable *action* shows the function of the requested command list or command.

The variable *oper_ID* shows the name of the operator that issued the request.

The variable *resname* shows the name of the resource affected.

System action: The requested function is performed on the specified resource.

Operator response: None.

System programmer response: None.

Classes: None.

AOF306I *time*: NO *auto_type* AUTOMATION
REQUIRED FOR *restype* *resname*

Explanation: Automation was requested for the resource and it was found that no action was required.

The variable *auto_type* shows the type of automation requested.

The variable *restype* shows the type of resource.

The variable *resname* shows the name of the resource.

The variable *time* shows the time this message was generated.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF307I *time*: REQUEST *clist_name* action BY
oper_id WAS UNSUCCESSFUL FOR
resname, STATE IS ALREADY SET

Explanation: The request to set a function ON or OFF for the specified resource was not successful.

The variable *time* shows the time this message was generated.

The variable *clist_name* shows the name of the requested command list or command.

The variable *action* shows the function of the requested command list or command.

The variable *resname* shows the name of the resource.

The variable *oper_id* shows the name of the operator that issued the request.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF309I *time*: OPERATOR *operator_ID*
EXECUTED TIMER *timer_ID* FOR
CATCHUP - COMMAND: *cmd*

Explanation: The command *cmd* was executed for timer *timer_ID* because the current time was past the execution time and CATCHUP=YES was specified.

The variable *time* shows the time this message was generated.

The variable *operator_ID* shows the name of the operator that executed the command.

The variable *timer_ID* shows the name of the timer that was executed.

The variable *cmd* shows the command that was executed.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF310I *time* : *resname* *autotype* IS SET ON -
autotype *action* NOT FOUND FOR
msg_type WITH SELECTION *selection*

Explanation: The specified automation flag is on, but the command or reply to be issued could not be found in the automation control file.

The variable *time* shows the time when this message was generated.

The variable *resname* shows the name of the resource.

The variable *autotype* shows the automation flag that has been checked.

The variable *action* shows the type of action that should have been taken. Possible values are COMMAND and REPLY.

The variable *msg_type* shows the message ID that the specified action has been searched under in the automation control file.

The variable *selection* shows the pass or selection string that the entry that was searched for could not be found on.

System action: Processing continues.

Operator response: None.

System programmer response: Use the customization dialog to check the entries specified by the resource name, the message ID and the selection string in the automation control file. For further information, refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

Classes: 40, 43.

AOF311I *time* : *restype* *resname* (JOB *job_name*) DID
NOT HAVE AN OUTSTANDING
REPLY ON *msg_type* PASS *pass_number*

Explanation: The automation policy for shutting down a resource indicates that a reply should be issued for the command type identified in this message. However, there was no outstanding message that a reply could be issued for on the pass shown by *pass_number*.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource.

The variable *resname* shows the name of the resource.

The variable *job_name* shows the name of the job.

The variable *msg_type* shows the command type to be issued (SHUTINIT,SHUTDOWN).

System action: Processing continues.

Operator response: Contact your system programmer.

System programmer response: Make sure that the automation control file policy for shutting down the resource identified in this message is accurate.

Classes: 40, 43.

AOF313I *time* : START FOR *restype* *resname* (JOB
job_name) WAS NOT ATTEMPTED -
reason

Explanation: SA z/OS attempted to start a resource but could not for the reason given.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of the resource.

The variable *resname* shows the name of the resource.

The variable *job_name* shows the name of the job.

The variable *reason* shows the reason that the resource could not be started.

System action: If the resource was not started because a flag was turned off it will be left in its current state and the flag rechecked each time the monitor cycle runs. No attempt will be made to start its parents until the flag is turned on. If the resource was not started because it (or one of its parents) was involved in a

shutdown its status may be changed to AUTODOWN. It will not start until the shutdown has finished.

Operator response: If the resource failed to start because a flag was set off and you want it to start you should use INGAUTO to change the flag. If the resource did not start because of a shutdown and you want it to start you must either wait for the shutdown to finish or clear the shutdown.

System programmer response: None.

Classes: 40, 43.

AOF314I *time* : CATCHUP TIMER ID *timer_ID*
NOT EXECUTED. TASK *operator_ID*
NOT LOGGED ON.

Explanation: A timer was unable to be restored because the scheduled time had already occurred. Catch-up processing unsuccessfully attempted to issue the command. The command failed because the operator (*operator_ID*) was not logged on to receive and run the command.

The variable *time* shows the time this message was generated.

The variable *timer_ID* shows the name of the timer that was not able to be restored.

The variable *operator_ID* shows the name of the operator where the timer command should have run.

System action: Processing continues.

Operator response: Issue the command manually under the appropriate operator ID.

System programmer response: None.

Classes: 0, 40, 43.

AOF315I *time* : TIMER *timer_ID* PURGED by
operator *operator_ID*

Explanation: A timer was deleted from the NetView timer list by the operator.

The variable *time* shows the time this message was generated.

The variable *timer_ID* shows the name of the timer that was deleted (purged).

The variable *operator_ID* shows the name of the operator who deleted the timer.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF316I *time* : SHUTDOWN FOR *restype resname*
(JOB *job_name*) DELAYED *interval*
MINUTES - RECEIVED EXIT CODE
code FROM *exit_name*

Explanation: Shutdown processing has been suspended for a job because a return code received from an exit matched a shutdown return code that temporarily stops processing.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource.

The variable *resname* shows the name of the resource.

The variable *job_name* shows the name of the job that processing has been suspended for.

The variable *interval* shows the number of minutes processing will be suspended.

The variable *code* shows the return code received from the exit.

The variable *exit_name* shows the exit that the return code was received from.

System action: Shutdown will be retried after the interval. Retries will continue until subsystem is shutdown.

Operator response: If the delay is excessive, investigate the reason and contact your system programmer.

System programmer response: If delay is excessive, investigate why the exit is temporarily suspending shutdown.

Classes: 40, 43.

AOF319I *time* : OPERATOR *operator1*
SCHEDULED TIMER *timer_ID* UNDER
OPERATOR *operator2* - COMMAND: *text*

Explanation: An operator created and scheduled a new timer.

The variable *time* shows the time this message was generated.

The variable *operator1* shows the name of the operator who created the timer.

The variable *timer_ID* shows the name of the timer that was scheduled.

The variable *operator2* shows the operator identification where the timer was scheduled.

The variable *text* shows the text of the timer command that was scheduled.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF320I *time* : ASSIST LOG FOR:
 resource_type/resource_name - **KEY**:
 entry/type/keyword - **COMMAND**: *text*

Explanation: This message is generated when assist mode is active for a resource and a command is issued for that resource. This message identifies the resource associated with the command and shows the text of the command. This message is logged in the netlog.

The variable *time* shows the time this message was generated.

The variable *resource_type* shows the type of resource associated with the command.

The variable *resource_name* shows the name of the resource associated with the command.

The variable *entry* shows the entry field from the automation control file entry that defines the command. If it is internally generated, this value will be FUNC.

The variable *type* shows the type field from the automation control file entry that defines the command. If it is internally generated, this value will be ISSUED.

The variable *keyword* shows the command selection field from the automation control file entry.

The variable *text* shows the text of the command that was issued, after variable substitution.

System action: This message is logged in the netlog.

Operator response: None.

System programmer response: Review the netlog to verify that the action reported by this message was intended and that variable substitution worked properly.

Classes: None.

AOF323I *time* : ASSIST LOG FOR:
 resource_type/resource_name - **KEY**:
 entry/type/keyword - **REPLY**: *text*

Explanation: This message is generated when assist mode is active for a resource and a reply is issued for that resource. This message identifies the resource associated with the reply and shows the text of the reply. This message is logged in the netlog.

The variable *time* shows the time this message was generated.

The variable *resource_type* shows the type of resource associated with the reply.

The variable *resource_name* shows the name of the resource associated with the reply.

The variable *entry* shows the entry field from the automation control file entry that defines the reply. If it is internally generated, this value will be FUNC.

The variable *type* shows the type field from the automation control file entry that defines the reply. If it is internally generated, this value will be ISSUED.

The variable *keyword* shows the reply selection field from the automation control file entry.

The variable *text* shows the text of the reply that was issued, after variable substitution.

System action: This message is logged in the netlog.

Operator response: None.

System programmer response: Review the netlog to verify that the action reported by this message was intended and that variable substitution worked properly.

Classes: None.

AOF325I **CHRON COMMAND IN ERROR:**
 chron_command

Explanation: The CHRON command processor rejected the command.

The variable *chron_command* shows the name of the command that could not be executed.

System action: Processing continues.

Operator response: Reenter the correct command, if applicable.

System programmer response: None.

Classes: None.

Module: AOFRAAT1

AOF326I **STOP TIME MUST BE LATER THAN
 START TIME**

Explanation: The stop time of the EVERY timer is earlier than the start time.

System action: Processing continues.

Operator response: Reenter the correct command, if applicable.

System programmer response: None.

Classes: None.

Module: INGRYTIM

AOF331I **SUBSYSTEM *name* IS NOT DEFINED
 ON THE LOCAL SYSTEM**

Explanation: The specified subsystem is defined more than once within the systems in the sysplex. Because the command is operating in line mode and the subsystem is not defined locally, system automation cannot determine which of the subsystems the command should be routed to.

An operator dialog issues this message.

The variable *name* shows the specified subsystem name.

System action: None.

Operator response: Reenter the correct command. Specify the TARGET parameter to identify the system that the command should be routed to.

System programmer response: None.

Classes: None.

AOF332I **SUBSYSTEM *name* COULD NOT BE LOCATED ON *target***

Explanation: The specified subsystem is neither defined in the local system nor in any other system of the sysplex.

If the target is outside the local sysplex, there could be a problem with the RMTCMD link. If this is the case, additional NetView messages have been issued.

The variable *name* shows the specified subsystem name.

The variable *target* shows the target, or the list of targets where the subsystem is not defined.

System action: None.

Operator response: Reenter the correct command.

System programmer response: None.

Classes: None.

AOF333I ***function* CALL FAILED WITH RC=
return_code IN CLIST *name***

Explanation: The specified clist detected an error while attempting to schedule a remote procedure call (RPC) to run a clist or a command on a remote system in the sysplex. The return code indicates which type of error occurred:

- 12 An invalid handle was specified. The handle is currently in use, or the handle was already deleted, or it is being deleted.
- 16 Processing of the remote procedure call (RPC) failed.
- 24 The target is not active, or it is not contained in the member table.
- 32 REXX variable pool problem.
- 36 Other severe error.
- 40 The Sysplex Communication Manager Environment is not available, for example, because the Sysplex Communication Manager task was not started.
- 44 An ABEND occurred.

An operator dialog issues this message.

The variable *function* shows the service that causes the problem.

The variable *return_code* shows the return code from the failing function.

The variable *name* shows the name of the clist that invoked the failing function.

System action: None.

Operator response: Issue the command again. If the problem persists, contact your IBM Support Center for further assistance.

System programmer response: Examine the return code of the failing service for the cause of the error. For a list of possible error code values and their meaning, refer to the explanation of message AOF350E. If you cannot resolve the problem, contact your IBM Support Center.

Classes: 0.

AOF334I **UNABLE TO RECEIVE THE OUTPUT FROM THE REQUESTED SYSTEM(S)
*system***

Explanation: One or more remote procedure calls (RPCs) were scheduled to get data from the remote system(s). It is possible that the commands were successfully processed on the remote system(s). However, the "OK" result was not returned in time.

The variable *system* shows the list of systems that did not provide the output in time.

System action: None.

Operator response: Check whether processing has completed successfully, then reenter the command. If the problem persists, contact your IBM Support Center.

System programmer response: Examine the netlog for additional information. If necessary, increase the maximum wait time, "aofrpcwait". If communication is via the NetView RMTCMD, increase the default CORR CMD wait time value in the DSICCDEF or use "aofrmtcmdwait".

Classes: None.

AOF335I **UNRECOGNIZED ACTION CODE**

Explanation: An action code that is not valid was entered.

System action: None.

Operator response: Enter a correct action code. For a list of valid action codes (or selection codes), call help from the panel that this message appears on.

System programmer response: None.

Classes: None.

AOF336I **SUBSYSTEM *name* NOT DEFINED ON SYSTEMS *sysname_list***

Explanation: The specified subsystem is not defined on the system(s) identified with the TARGET parameter or the appropriate panel field.

An operator dialog issues this message.

The variable *name* shows the specified subsystem name.

The variable *sysname_list* shows the systems that do not have the subsystem defined.

System action: None.

Operator response: Reenter the correct command.

System programmer response: None.

Classes: None.

AOF337I **ONLY ONE TARGET IS ALLOWED**

Explanation: More than one system name or domain ID was specified for the TARGET parameter. However, only one name is allowed.

System action: None.

Operator response: Reenter the correct command.

System programmer response: None.

Classes: None.

AOF338I **CONFLICTING TARGET SPECIFICATION**

Explanation: ALL and at least one other system name or domain ID have been specified for the TARGET parameter.

An operator dialog issues this message.

System action: None.

Operator response: Reenter the correct command.

System programmer response: None.

Classes: None.

AOF339I ***name* DOES NOT CORRESPOND TO AN ACTIVE TARGET WITHIN THE *environment***

Explanation: The specified name is neither the system name nor the network domain ID of an active target within the specified environment.

The following reasons can cause the dialog to issue this message:

- The specified target name is not valid.
- The specified target SA z/OS is a release that is earlier than Release 3

- If the specified target is outside the local sysplex, the GATEWAY connection to that target could be inactive.

- If the target is outside the local sysplex, there could be a problem with the RMTCMD link. If this is the case, additional NetView messages have been issued.

The variable *name* shows the unrecognized name.

The variable *environment* shows the environment the dialog searched for. It is either *sysplex* or *enterprise*.

System action: None.

Operator response: Reenter the correct command.

System programmer response: None.

Classes: None.

AOF340I **NO RESPONSE RECEIVED FROM SYSTEM *system_name***

Explanation: A remote procedure call (RPC) was sent to the specified system, but the remote system did not reply with an answer within the expected time period.

The variable *system_name* identifies the system that did not respond.

System action: None.

Operator response: Issue the command again. If the problem persists, contact your local support personnel for further assistance.

System programmer response: Examine the netlog for additional information. If necessary, increase the maximum waittime "aofrpcwait."

Classes: None.

AOF341I ***time* : *clist_name* RETURN CODE: *xx* SYSTEM: *name* ERROR: *error***

Explanation: This is an internal message that is used to transmit command completion information from the system that processed the command back to the system that requested the command execution. The message will not appear on any screen.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF343I **NOTHING TO DISPLAY**

Explanation: No data is available for display.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF344I CLIST *clist_name* DOES NOT EXIST

Explanation: The specified clist could not be found.
The variable *clist_name* shows the name of the clist.

System action: None.

Operator response: Issue the command again.

System programmer response: None.

Classes: None.

| **AOF345I** RUNMODE CHANGED FOR *system*.
| PLEASE REVISE RUNMODE
| QUALIFICATIONS

| **Explanation:** The runmode has been changed and
| there are still resources that have been qualified earlier
| by an INGRUN command.

| The variable *system* shows the system for which
| runmode was changed.

| **System action:** None.

| **Operator response:** Revise the displayed list and
| decide whether the runmode qualification for those
| resources can be deleted.

| **System programmer response:** None.

| **Classes:** None.

| **AOF346I** RUNMODE *runmode* ALREADY SET
| FOR *system*

| **Explanation:** INGRUN was called to set a runmode
| which was already set on the given system.

| The variable *runmode* shows the name of the
| runmode.

| The variable *system* shows the name of the affected
| system.

| **System action:** None.

| **Operator response:** None, because the runmode is
| already set. You can specify another runmode.

| **System programmer response:** None.

| **Classes:** None.

| **AOF347I** NO RUNMODE DEFINED FOR *system*

| **Explanation:** INGRUN was called to set a runmode
| but there are none defined on the given system.

| The variable *system* shows the name of the affected
| system.

| **System action:** None.

| **Operator response:** Contact your system programmer.

| **System programmer response:** Define runmodes for
| the given system in the automation control file.

| **Classes:** None.

AOF350E COMMUNICATION ERROR:
SYSTEM=*system name* ERRTYPE=*error*
type ERRCODE=*error code*
SERVICE=*service* RET=*return code*
RSN=*reason code* DIAG=*module remainder*

Explanation: An error was detected during
communication to another system. This message always
contains the name of the system where the error
occurred, the type of error and the error code. The
other keywords are optional.

The variable *system name* shows the name of the
OS/390 or z/OS system where the error occurred.

The variable *error type* shows a decimal number that
specifies the type of the error. It can have the following
values:

- 4 Input checking failed. User specified input, for example, global variables, that were not set up correctly. The variable *error code* provides further information.
- 8 The system automation communication manager Data Service Task (DST) could not be initialized.
- 12 Termination error of the system automation communication manager Data Service Task.
- 16 An internal communication pipe service provided by a system automation system failed.
- 20 A remote procedure call (RPC) Handler failed.
- 24 A system service failed.
- 28 A REXX API failed.
- 32 A storage problem occurred.
- 36 An internal error occurred.
- 40 The system automation communication services could not be initialized.

The variable *error code* shows a unique number that
specifies where in the code the error occurred.

The variable *service* shows the name of the service that
caused the error.

The variable *return code* shows the return code of the
failing service (hex number). This variable will only be
returned in combination with the *service* variable.

The variable *reason code* shows the reason code of the
failing service (hex number). This variable will only be
returned in combination with the *service* variable.

The *module remainder* variable shows diagnostic
information. It specifies the name of the failing module.
The remaining items vary depending on the error code.
The *module remainder* variable data is intended for your
IBM Support Center.

The following table shows the possible combinations of
error types and error codes and their meaning. This
information also applies to message HSAM1050E.

Table 3. Combinations of Error Types and Error Codes

Error Type	Error Code	Meaning
4	72 136 140 284	Explanation Input checking error. Invalid format of a handle. Action Contact your IBM Support Center (ISC).
4	260	Explanation Input checking error. The sysplex group name is not known. The reason is that the retrieval of the common global variable AOFXCFGROUP failed or it contains a null string. Action Set AOFXCFGROUP up properly.
4	268	Explanation Input checking error. Target sysplex member, representing the target NetView or the automation manager, is not valid. Action Specify a valid target sysplex member or start the SA z/OS sysplex communication manager in NetView for the specified system or start the automation manager.
4	541 550	Explanation Input checking error. Target sysplex member, representing the primary automation manager, is not active. Action Check whether the member name in the DIAGNOSTIC field is a valid XCF member name. Specify a valid target sysplex member or start the automation manager.
8	13 16 20	Explanation DST initialization error. There is not enough storage available. Action Increase the region size.
8	31 170	Explanation DST initialization error. An SA z/OS communication service module cannot be loaded. Action Check return code of the load service.
8	208	Explanation DST initialization error. An SA z/OS communication environment cannot be created. Action Check the z/OS return code.
8	244 248 573 574	Explanation DST initialization error. An SA z/OS communication provided NetView command processor cannot be added to the NetView environment. Action Check the NetView return code.
8	300 348	Explanation DST initialization error. An attempt was made to start SA z/OS communication on an OS/390 system lower than Release 3. Action Install an OS/390 or z/OS Release supporting XCF Ordered Message Delivery.
8	571	Explanation DST initialization error. The DST receiver task (DSR) cannot be started. Action Check the return and reason codes. If necessary contact your IBM Support Center.

Table 3. Combinations of Error Types and Error Codes (continued)

Error Type	Error Code	Meaning
8	489	<p>Explanation Automation manager initialization error. The MTIB could not be initialized.</p> <p>Action Increase the region size and if necessary contact your IBM Support Center.</p>
8	517	<p>Explanation Automation manager initialization error. A subtask could not be initialized.</p> <p>Action Contact your IBM Support Center.</p>
8	520	<p>Explanation Automation manager initialization error. The subtask shell logon to the Communication Manager failed.</p> <p>Action Contact your IBM Support Center.</p>

Table 3. Combinations of Error Types and Error Codes (continued)

Error Type	Error Code	Meaning
8	532	<p>Explanation</p> <p>Automation manager initialization error. A control block cannot be created that is anchored to the MTIB. One of the following return codes in hexadecimal will be supplied:</p> <p>X'64' (100) Message control block (MSIB) creation failed</p> <p>X'190' through X'1F3' (400 + nn) PKVT cannot be initialized</p> <p>X'194' (404) PKVT cannot be initialized due to errors in the automation manager parmlib member.</p> <p>X'1F8' (504) Pre-initialization of the control block failed</p> <p>X'1FC'(508) Post-initialization of the control block failed</p> <p>X'256' (598) Message control block (MSIB) creation failed</p> <p>X'257' (599) Message control block (MSIB) creation failed</p> <p>X'2BC' through X'31F' (700 + nn) Initialization of the CTCB failed</p> <p>X'320' through X'383'(860 + nn) Deletion of the CTCB failed. This is a setup problem of the component trace of the system logger.</p> <p>X'384' through X'3E7' (900 + nn) Initialization of the MTIB failed</p> <p>X'3E8' through X'7CF' (1000 + nn) where if: nn=13,16,20,472</p> <p>it is a storage problem. If: nn=31,170</p> <p>it is a load module problem.</p> <p>Communication initialization failed</p> <p>If RET=X'000496' (1174) then there might be too many automation manager instances on the MVS image. The limit is 9.</p> <p>2707 No longer MODE=SUP</p> <p>Action Check the return code and other messages that may be issued with this message. If necessary contact your IBM Support Center.</p>
8	533	<p>Explanation</p> <p>Automation manager initialization error. Not all modules could be loaded.</p> <p>Action Check the STEPLIB concatenation. If necessary contact your IBM Support Center.</p>

Table 3. Combinations of Error Types and Error Codes (continued)

Error Type	Error Code	Meaning
8	536	<p>Explanation Automation manager initialization error. Cannot open the task lib.</p> <p>Action Check the HSAMODLE DD statement in the start procedure of the automation manager and if necessary contact your IBM Support Center.</p>
8	561	<p>Explanation Automation manager initialization error. The MVS console interface could not be initialized.</p> <p>Action Contact your IBM Support Center.</p>
8	564	<p>Explanation DST initialization error. The error occurred when parsing the DST init member INGXINIT.</p> <p>Action Check the DSIPARM member INGXINIT for syntax errors.</p>
8	566	<p>Explanation DST initialization error. A global NetView variable could not be created.</p> <p>Action Increase the region size.</p>
12	11	<p>Explanation DST Termination. The SA z/OS communication manager is waiting to terminate the DST because there is another task that is still using its resources. The waiting time depends on the value of the global variable XDOMTIME. If XDOMTIME contains a null string or an invalid value then DST termination will continue after 60 seconds. Otherwise, the value of XDOMTIME is multiplied by 5 seconds. For example, if XDOMTIME=120 then the DST will wait 600 seconds.</p> <p>Action Stop the REXX program that is using the SA z/OS communication services.</p>
12	12	<p>Explanation DST Termination. The SA z/OS communication manager termination continued but the SA z/OS sysplex communication block was not freed since another task still holds a lock.</p> <p>Action Stop the REXX program that is using the old SA z/OS communication environment.</p>
12	535	<p>Explanation Automation manager termination error. The subtask shell termination failed.</p> <p>Action Contact your IBM Support Center.</p>
12	562	<p>Explanation Automation manager termination. The cleanup of a control block anchored to the MTIB failed. The return code that will be supplied with this message is one of those explained in the row of ERROR TYPE 4 and ERROR CODE 532.</p> <p>Action Check the return code. If necessary contact your IBM Support Center.</p>
12	602 572	<p>Explanation DSR Termination. The DST Receiver Task could not be terminated.</p> <p>Action Contact your IBM Support Center.</p>
16	24	<p>Explanation RPC Pipe error. Pipe cannot be created. There is probably not enough storage.</p> <p>Action Increase the region size.</p>

AOF350E

Table 3. Combinations of Error Types and Error Codes (continued)

Error Type	Error Code	Meaning
16	44 100 503 527	<p>Explanation RPC Pipe error. Write failed, probably due to lack of storage.</p> <p>Action For return codes 12, 16 or 56 increase the region size. Return code 84 indicates that the write was canceled because of a DST termination request, and no action is required. In case of any other return code, contact your IBM Support Center.</p>
16	48 64 132 146 328 504 514 529 555	<p>Explanation RPC Pipe error. Close failed.</p> <p>Action Contact your IBM Support Center.</p>
16	56 128 144 510 525 554 592	<p>Explanation RPC Pipe error. An open failed because the RPC handle to be opened cannot be found, is currently in use or is disabled. Error code 592 is a failing open during retry.</p> <p>Action Contact your IBM Support Center.</p>
16	60 513	<p>Explanation RPC Pipe error. Read request failed, probably due to lack of storage.</p> <p>Action For return codes 12, 16 or 56 increase the region size. Return code 84 indicates that the read was canceled because of a DST termination request, and no action is required. In case of any other return code contact your IBM Support Center.</p>
16	150 154 512 526	<p>Explanation RPC Pipe error. Pipe is broken.</p> <p>Action Look for additional AOF350E messages that describe the reason for the broken pipe, for example messages with error type 20. Contact your IBM Support Center.</p>
16	320 505	<p>Explanation RPC Pipe error. Create request for a remote pipe failed, probably due to lack of storage for the pipe object.</p> <p>Action Increase the NetView region size.</p>
16	372	<p>Explanation RPC Pipe error. Pipe is broken. The SA z/OS communication manager on a pipe's target system was terminated. The pipe object is disabled, communication is no longer possible.</p> <p>Action Restart the SA z/OS communication manager (INGPXDST), or recycle NetView.</p>
16	392	<p>Explanation RPC Pipe error. The DST or the automation manager cannot send the pipe data to the target. This is either due to an XCF problem, or an internal problem. The return code (RET) is the return code of the RPC pipe service. The reason code (RSN) describes the return code of the package service. The return and reason codes are for diagnostic purposes only.</p> <p>Action Check whether additional messages were issued, for example, HSAM1050E with error type 24 and analyze whether an XCF error occurred. If necessary contact your IBM Support Center.</p>

Table 3. Combinations of Error Types and Error Codes (continued)

Error Type	Error Code	Meaning
16	521	<p>Explanation RPC Pipe error. A subtask cannot receive the data buffer.</p> <p>Action Contact your IBM Support Center</p>
16	575 597 598 599 623 638	<p>Explanation RPC Pipe error. The DST or the automation manager cannot send the pipe data to the target system. The send request was retried but the retry limit exceeded.</p> <p>Action The communication path is probably broken. Make sure that communication can work properly. In case of XCF customize the XCF subsystem. For example, increase the number of XCF buffers or install a new message buffer class dedicated to the DST, or do both. Check whether there are additional AOF350 or HSAM1050 messages that might explain the reason why communication does not work. If necessary contact your IBM Support Center.</p>
20	400	<p>Explanation RPC Command Handler error. The RPC command handler INGRXCMD failed. The reason code contains the REXX line number where the error condition was trapped.</p> <p>Action Contact your IBM Support Center.</p>
20	404 408 412	<p>Explanation RPC Command Handler error. The RPC command handler INGRXCMD failed.</p> <p>Action Look for the corresponding additional error message AOF350E with error type 16 for more information. If necessary, contact your IBM Support Center.</p>
20	416	<p>Explanation RPC Command Handler error. The RPC command handler INGRXCMD cannot execute the command because the NetView PIPE failed.</p> <p>Action Check the return code of the NetView PIPE command. Contact your IBM Support Center.</p>
24	40 174 178 190 232 233 234 236 476 480 482 483	<p>Explanation System Service error. The specified XCF system service failed.</p> <p>The return and reason codes of system service IXCxxx, for example, IXCMSSGO, are described in <i>z/OS MVS Programming Sysplex Services Reference</i> (SA22-7618).</p> <p>Action Check the return code and reason code of the system service. If necessary contact your IBM Support Center.</p>
24	272 336	<p>Explanation System Service error. The specified NetView service failed while a NetView command cannot be scheduled.</p> <p>Action Check the return code and reason code of the system service. If necessary contact your IBM Support Center.</p>
24	516	<p>Explanation System Service error. The activate recovery failed (ESTEVA).</p> <p>Action Check the return code and reason code of the system service. If necessary contact your IBM Support Center.</p>

Table 3. Combinations of Error Types and Error Codes (continued)

Error Type	Error Code	Meaning
24	523 556	Action Ignore the error if this error occurs while the automation manager is terminating.
24	537	Explanation System Service error. A task cannot be attached. Action Check the return code and reason code of the system service. If necessary contact your IBM Support Center.
24	568 569 576 577 601 620 643	Explanation System Service error. The specified MVS system service failed. Action Check the return code and reason code of the system service. If necessary contact your IBM Support Center.
24	674 675 676 677 683 684	Explanation NetView PPI service failed. Action Check the return code of the PPI service. It is given in RSNCODE for 683,674 through 677, or RETCODE for 684. If necessary contact your IBM Support Center.
24	679	Explanation NetView service failed while adding or removing a command processor. Action Check the return code and reason code of the PPI service. If necessary contact your IBM Support Center.
28	104	Explanation REXX API error. A REXX API failed while a stem element retrieval was attempted from the REXX variable pool, caused by INGPXSND() or INGPXRPC(). Action Check the return code and reason code of the REXX IRX EXCOM API. If necessary contact your IBM Support Center.
28	112	Explanation REXX API error. A REXX API failed while attempting to set a stem element in the REXX variable pool, caused by INGPXRCV(). Action Check the return code and reason code of the REXX IRX EXCOM API. If necessary contact your IBM Support Center.
28	116	Explanation REXX API error. A REXX API failed while INGPXRCV() attempted to set the number of stem elements variable (stem.0) in the REXX variable pool. Action Check the return code and reason code of the REXX IRX EXCOM API. If necessary contact your IBM Support Center.
28	120 296 312	Explanation REXX API error. A call to the REXX API failed. Action Check the return code and reason code of the REXX IRX EXCOM API. If necessary contact your IBM Support Center.

Table 3. Combinations of Error Types and Error Codes (continued)

Error Type	Error Code	Meaning
32	84 88 92 96 182 186 198 204 216 220 292 332 340 485 487 488 493 496 498 500 543 560 567 585 586 594 595 596 608 613 624 626 631 621 640 678 680 685	<p>Explanation Storage problem. There is insufficient storage for the queue, table or buffer operation requested.</p> <p>Action Increase the region size.</p>
36	523	<p>This means that the data of an XCF request was received but it cannot be scheduled to the Automation Manager receiver-queue because the corresponding thread is no longer registered to receive requests. If this happens during termination of the Automation Manager the message can be ignored.</p> <p>Error code 556 is caused by error code 523 so message HSAM1050E with error code 556 may also appear.</p>
36	<i>nnn</i>	<p>Explanation Internal error.</p> <p>Action Contact your IBM Support Center.</p>
40	440 444 448 452 456 460 464 468	<p>Explanation Environment error.</p> <p>Action Start the DST INGPXDST.</p>

System action: None.

Operator response: Refer to the documentation of message AOF333I.

System programmer response: Contact your IBM Support Center.

Classes: 0.

**AOF351E COMMUNICATION MANAGER
INITIALIZATION FAILED**

Explanation: An error was detected during initialization of the system automation communication manager Data Service Task (DST).

System action: The Data Service Task terminates.

Operator response: Analyze the problem, referring to the explanation of message AOF350E. Correct your input and retry the task.

System programmer response: None.

Classes: 0.

**AOF352I COMMUNICATION MANAGER
INITIALIZATION COMPLETE**

Explanation: The initialization module of the system automation communication manager has successfully initialized the Data Service Task (DST).

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

**AOF355E INCORRECT CALL TO FUNCTION
 name RC=rc**

Explanation: An error was detected during parameter list validation of a system automation REXX API function.

System action: The REXX function terminates due to a REXX syntax error.

Operator response: Analyze the return code. Invoke the REXX function *name* with a correct parameter list. The return code *rc* can have one the following values:

Value	Meaning
4	An incorrect number of arguments was specified.
8	Either the value of the data parameter is no stem, or stem.0 is nonnumeric, or a negative number.
12	The value of the handle parameter is too long.
16	The value of the records parameter is zero or nonnumeric.

20	The value of the task parameter is too long.
24	The value of the target parameter is too long.
40	The handle parameter is required but no value was specified.
44	The data parameter is required but no value was specified.
48	The task parameter is required but no value was specified.
52	The records parameter is required but no value was specified.

System programmer response: None.

Classes: 0.

**AOF356E COMMAND HANDLER INGRXCAM
FAILED: ERROR=xx FN=yy RC=zz**

Explanation: The command handler failed with a return code. It can have the following values:

Value	Meaning
4	Invalid RPC type
8	REXX function failed. If <i>xx</i> is 8 then <i>yy</i> shows the return code from the REXX function.
12	NetView pipe failed. If <i>xx</i> is 12 then <i>yy</i> =null and <i>zz</i> shows the return code from the pipe.
16	Buffer.0 is wrong

System action: The command handler terminates.

Operator response: Retry the operation. If the problem persists, contact your system programmer.

System programmer response: Retry the operation. If the problem persists, contact your system programmer.

**AOF359I DIAGNOSTIC INFO: MOD=module
INCIDENT=incident diagnostic-text**

Explanation: The purpose of this message is to provide diagnostic information to IBM service staff.

The variable *module* shows name of the module that detected the incident.

The variable *incident* shows a unique ID for an internally detected incident.

The variable *diagnostic-text* shows useful text in free format. The text depends on the incident that occurred. For example, for INCIDENT=100, this is the duplicate handle + seq-number.

System action: Processing continues.

Operator response: None.

System programmer response: None.

**AOF364I type entry INVALID TYPE OF
SUBSYSTEM. EXPECTING value1, GOT
value2, FOR MESSAGE msg**

Explanation: A SA z/OS message trap routine is

processing a message. The job that issued the message is associated with a subsystem that has a type that is not valid for the message.

This typically happens when either the type is unspecified or incorrectly specified, for example, specified CICS for an IMS subsystem.

The variable *type* is the SUBSYSTEM.

The variable *entry* is the name of the subsystem in error.

The variable *value1* is the TYPE that was expected for this message.

The variable *value2* is the TYPE specified in the ACF.

The variable *msg* is the message that triggered the problem.

System action: Processing stops. Automation for the message specified is not processed. This may cause further automation errors if the message is required for automation information, for example, UPMSG, etc.

Operator response: Report this error to your system programmer. This is a configuration error.

System programmer response: Correct the configuration of the subsystem specified in the Policy Database. Rebuild the Automation Control File and reload.

Classes: None.

AOF365I NO ACTIVE LINK FOUND

Explanation: The link between a consumer subsystem and a provider subsystem is already in the requested state. The link status can be either ACTIVE or INACTIVE.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF366I NO DYNAMIC LINK CONFIGURED IN THE POLICY BETWEEN *consumer* AND *provider*

Explanation: Activating or deactivating a link between a consumer subsystem and a provider subsystem failed because the link was either not defined at all or it was not defined as DYNAMIC in the automation policy.

The variable *consumer* is the subsystem name of the consumer.

The variable *provider* is the subsystem name of the provider.

System action: None.

Operator response: Verify that you defined the correct consumer and provider names. If these are correct,

notify your system programmer.

System programmer response: Define a dynamic link in the automation policy between the consumer and the provider.

Classes: None.

AOF367I LINK ACTIVATED BETWEEN *consumer* AND *provider*

Explanation: Link activation was successful between the consumer and the provider application.

The variable *consumer* is the subsystem name of the consumer.

The variable *provider* is the subsystem name of the provider.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF368I LINK DEACTIVATED BETWEEN *consumer* AND *provider*

Explanation: Link deactivation was successful between the consumer and the provider application.

The variable *consumer* is the subsystem name of the consumer.

The variable *provider* is the subsystem name of the provider.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF401E UNEXPECTED TIMEOUT FROM *command* in *clist*. PROCESSING TERMINATED.

Explanation: A message was expected but was not received before the 'WAIT' time expired.

System action: The requested function has not completed successfully.

Operator response: Retry the operation. If the problem persists, contact your system programmer.

System programmer response: Contact your IBM Support Center.

Classes: None.

AOF403E UNEXPECTED WAIT EVENT (WC=*wc*) FROM *command* IN *clist*. PROCESSING TERMINATED.

Explanation: A message was expected, but instead an unexpected event code was returned.

System action: The requested function has not completed successfully.

Operator response: Refer to *NetView Customization: Writing Command Lists* for an explanation of the code. If it is an unexplained wait code, for example, an error code, contact your IBM Support Center. If it is a G code, for example, GO, do not enter GO while waiting for that command to process.

System programmer response: None.

Classes: None.

AOF408I MISSING OR INVALID PARAMETER *parm* DETECTED IN *command*. PROCESSING TERMINATED.

Explanation: The command *command* contains an invalid or missing parameter.

System action: Processing stops. The requested function is not completed successfully.

Operator response: None.

System programmer response: Correct the function call.

Classes: None.

AOF415I NO SERVICE PERIODS DEFINED FOR *applid*

Explanation: The Automation engine found no service periods defined in the control file for this application.

System action: Processing continues.

Operator response: None. This is an internal message.

System programmer response: None.

Classes: None.

AOF432I INVALID EVENT NAME

Explanation: The event name was not found. Either it is not defined in system automation, or it contains invalid characters.

System action: Processing continues.

Operator response: Enter a valid event name.

System programmer response: None.

Classes: None.

AOF435I TYPE AN EVENT NAME AND PRESS ENTER

Explanation: System automation is waiting for an event name to be specified.

System action: Processing continues.

Operator response: Type a valid event name.

System programmer response: None.

Classes: None.

AOF436I TYPE A RESOURCE NAME AND PRESS ENTER

Explanation: System automation is waiting for an application name to be specified.

System action: Processing continues.

Operator response: Type a valid application name.

System programmer response: None.

Classes: None.

AOF438I EVENT *event* DEFINED BUT NOT USED BY ANY TRIGGER

Explanation: The event is defined in the automation control file (ACF), but it is not used by any trigger in the sysplex.

System action: Processing continues.

Operator response: None.

System programmer response: None.

AOF440I NO TRIGGER DEFINED FOR *applid*

Explanation: There is no trigger defined for the specified application.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

AOF441E SETEVENT FAILED: FUNCTION PARAMETER MUST BE SET OR UNSET

Explanation: SETEVENT/INGEVENT failed because the function parameter was neither SET nor UNSET.

System action: Processing continues.

Operator response: Specify the correct function parameter SET or UNSET.

System programmer response: None.

Classes: None.

AOF442I SET|UNSET EVENT *event* DONE FOR ALL AFFECTED APPLICATIONS ON *system*

Explanation: The SETEVENT/INGEVENT routine successfully completed the status file update for this event and all affected applications.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

AOF447I DATE OUT OF RANGE

Explanation: The date could generate a schedule conflict because it is in the range of 12/26/yy and 12/31/yy where yy is computed by the common global variable AOCCENTURYSWITCH and the current year.

System action: Processing continues.

Operator response: Change the date.

System programmer response: None.

Classes: None.

AOF448I INVALID DATE

Explanation: The specified date is not valid.

System action: Processing continues.

Operator response: Specify a valid date.

System programmer response: None.

Classes: None.

AOF449I OVERLAP EXISTS BETWEEN ACTIVE *type* TIMESLOT *timeslot* AND SPECIFIED *type* TIMESLOT *timeslot*

Explanation: The specified timeslot is in conflict with the currently active timeslot. A timeslot is active when the start time has passed, but not the end time.

The variable *type* shows the type of the timeslot (UP or DOWN).

The variable *timeslot* shows the start and end time of the time window.

System action: None.

Operator response: Correct the command.

System programmer response: None.

Classes: None.

AOF450I ORIGINAL TIME HAS PASSED AND CANNOT BE CHANGED

Explanation: A time specified on the service periods panel is before the current time.

System action: Processing continues.

Operator response: Enter service period times later than the current time.

System programmer response: None.

Classes: None.

AOF451I PREVIOUS STOP TIME IS NOT '****'

Explanation: The '****' in service periods shows that the service period will remain active the next day. If the last stop time is '****', the next start time must also be '****' indicating the continuity. Here, the operator modified the service hours and specified '****' as the first start time, but the previous stop time is not '****'.

System action: Processing continues.

Operator response: Either change the last stop time to '****', or change the first start time appropriately.

System programmer response: None.

Classes: None.

AOF452I PREVIOUS STOP TIME IS '****'.

Explanation: The operator modified the service hours and did not specify '****' as the first start time, but the previous day's last stop time is '****'.

System action: Processing continues.

Operator response: Either change the last stop time to '****', or change the first start time appropriately.

System programmer response: None.

Classes: None.

AOF453I NEXT START TIME IS '****'

Explanation: The next day's first start time is '****', but when the operator modified the service hours, '****' was not specified as the last stop time.

System action: Processing continues.

Operator response: Either change the start time to '****', or change the last stop time accordingly.

System programmer response: None.

Classes: None.

AOF454I NEXT START TIME IS NOT '**'**

Explanation: The '****' in service periods indicates that the service period will remain active the next day. If the last stop time is '****', the next start time must also be '****' indicating the continuity. In this case, the operator modified the service hours and specified '****' as the last stop time, but the next start time is not '****'.

System action: Processing continues.

Operator response: Either change the start time to '****', or change the last stop time accordingly.

System programmer response: None.

Classes: None.

AOF456I No times allowed after 'DOWN'

Explanation: When 'DOWN' is the first start or stop time, no times can be entered later that day. The operator modified the service hours and specified 'DOWN' as the first start time, but additional times follow it.

System action: Processing continues.

Operator response: Correct the service hours.

System programmer response: None.

Classes: None.

AOF457I BOTH 'FROM' AND 'TO' TIMES MUST BE ENTERED, OR BOTH MUST BE BLANK

Explanation: A 'From' or 'To' time was entered. Either both 'From' and 'To' times must be entered, or both must be blank.

System action: Processing continues.

Operator response: Enter both 'From' and 'To' times, or blank both out.

System programmer response: None.

Classes: None.

AOF458I NO TIMES ALLOWED AFTER '**'**

Explanation: Because '****' shows that a service period will remain active until the next day, no times are allowed after the '****'.

System action: Processing continues.

Operator response: Correct the service hours.

System programmer response: None.

Classes: None.

AOF459I INVALID TIME

Explanation: The specified time is not valid. The time must be between 0000 and 2359.

System action: Processing continues.

Operator response: Enter a valid time.

System programmer response: None.

Classes: None.

AOF461I TIME MUST ALSO BE 'DOWN'.

Explanation: Either in the first Start or in the Stop field of the window, DOWN was specified. DOWN must be specified in both, the first Start and first Stop window.

System action: Processing continues.

Operator response: Either specify DOWN in both, the Start and Stop first service period fields, or specify valid times in both.

System programmer response: None.

Classes: None.

AOF462I ONLY THE FIRST START TIME MAY BE '**'**

Explanation: '****' was specified in a Start field other than the first one. '****' is a valid entry only in the first Start field, because it indicates that a service period continues from the previous day.

System action: Processing continues.

Operator response: Correct the time.

System programmer response: None.

Classes: None.

AOF463I TIME MUST BE NUMERIC

Explanation: The 'From' or 'To' time specified is not numeric, asterisks, or DOWN.

System action: Processing continues.

Operator response: Correct the time.

System programmer response: None.

Classes: None.

AOF464I THE FIRST 'FROM' AND 'TO' TIME CANNOT BE BLANK

Explanation: The first 'From' and 'To' time must be specified. They are required parameters.

System action: Processing continues.

Operator response: Correct the time by specifying valid times in the first fields.

System programmer response: None.

AOF465I TRIGGER *trigger* HAS NO SERVICE PERIODS

Explanation: No service periods are defined for this trigger.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

AOF466I CHANGED TIME MUST BE LATER THAN CURRENT TIME

Explanation: The time change specified was prior or equal to the current time.

System action: Processing continues.

Operator response: Specify a time change later than the current time, or quit the function.

System programmer response: None.

Classes: None.

AOF467I DELETION NOT ALLOWED BECAUSE IT CREATES A SCHEDULE CONFLICT

Explanation: The overrides cannot be deleted because this would create a conflict in the existing schedule(s). At first, the earliest start time of the regular schedule and of the override is determined. Then the latest stop time of both schedules is determined. If the actual time of the day is between those two times specified, this results in a schedule conflict. DOWN-DOWN is treated as '0000-2400'.

System action: Processing continues.

Operator response: Change the schedule to eliminate the conflict, or quit the function.

System programmer response: None.

Classes: None.

AOF468I TYPE 'D' IF DELETE REQUIRED

Explanation: An invalid character was specified in the Delete field.

System action: Processing continues.

Operator response: Type 'D' to delete the override.

System programmer response: None.

Classes: None.

AOF469I 'FROM' TIME MUST BE LATER THAN PREVIOUS 'TO' TIME

Explanation: The 'From' time of the service window is not later than the 'To' time of the previous service window.

System action: Processing continues.

Operator response: Correct either the 'From' or the 'To' time.

System programmer response: None.

AOF470I 'TO' TIME MUST BE LATER THAN 'FROM' TIME

Explanation: The 'To' time of the service window is not later than the 'From' time of the same service window.

System action: Processing continues.

Operator response: Correct either the 'From' or the 'To' time.

System programmer response: None.

AOF475I EVENT *event* IS NOT USED BY RESOURCE *applid*

Explanation: The event is not defined in the trigger that is used by the resource.

The variable *event* shows the event that is not used.

The variable *applid* shows the APPLID of the resource.

System action: The command terminates immediately.

Operator response: Check the scope parameters of the command, correct the value(s). Then issue the command again.

System programmer response: None.

AOF476I EVENT *event* IS NOT USED ON SYSTEM *system*

Explanation: The event is defined but it is not used by any trigger on the specified system.

The variable *event* shows the event that is not used.

The variable *system* shows the name of the system.

System action: The command terminates immediately.

Operator response: Check the RANGE parameter of the command and correct the value. Then issue the command again.

System programmer response: None.

AOF478I WILDCARDS ARE NOT SUPPORTED

Explanation: The INGEVENT command supports down-level systems. Because a down-level system does not know a wildcard, this support is suppressed for the time these down-level systems are supported.

System action: The command is not executed.

Operator response: Correct the parameter and reissue the command.

System programmer response: None.

Module: INGRYEVO

AOF479I SERVICE PERIOD REQUESTS TO UP-LEVEL SYSTEMS ARE NOT SUPPORTED

Explanation: A service period request in line mode to an up-level system is not supported. The reason for this is that the definitions of service periods on two different system levels, for example SA OS/390 1.3 and SA OS/390 2.1, are not compatible.

System action: The request is rejected.

Operator response: None.

System programmer response: None.

Module: INGRVP90

AOF501E *time* : RECOVERY FOR *restype resname* HALTED - *number* ERRORS SINCE *error_time* ON *error_date* - CRITICAL ERROR THRESHOLD EXCEEDED

Explanation: The resource has experienced the indicated number of errors during the time interval defined in the automation control file for the critical error threshold.

The variable *time* shows the time the exception message was generated.

The variable *restype* shows the type of the resource.

The variable *resname* shows the name of the resource.

The variable *number* shows the number of errors that has occurred.

The variable *error_time* shows the time of the first error to occur within the time interval.

The variable *error_date* shows the date of the first error to occur within the time interval.

System action: If *resname* is a network resource, the recovery process is ended. If *resname* is a system resource, the recovery process is ended or the appropriate commands are issued.

Operator response: None.

System programmer response: None.

Classes: 40 41 44 51.

AOF502I *time* : RECOVERY FOR *restype resname*, CONTINUING - *number* ERRORS SINCE *error_time* ON *error_date* - FREQUENT ERROR THRESHOLD EXCEEDED

Explanation: The frequent error threshold has been exceeded for a resource. This message reports the number of errors that occurred during the time interval defined in the automation control file for the frequent error threshold.

The variable *time* shows the time the exception message was generated.

The variable *restype* shows the type of the resource.

The variable *resname* shows the name of the resource.

The variable *number* shows the number of errors that occurred.

The variable *error_time* shows the time of the first error that occurred within the time interval.

The variable *error_date* shows the date of the first error that occurred within the time interval.

System action: If the resource identified in this message is a network resource, the recovery process continues. If it is a system resource, the appropriate commands are issued.

Operator response: Contact your system programmer.

System programmer response: Make sure the policy for shutting down this resource in the automation control file is accurate. Determine whether other action needs to be taken.

Classes: 40 43 51.

AOF503I *time* : RECOVERY FOR *restype resname* CONTINUING - *number* ERRORS SINCE *error_time* ON *error_date* - INFREQUENT ERROR THRESHOLD EXCEEDED

Explanation: The infrequent error threshold has been exceeded for a resource. This message reports the number of errors that occurred during the time interval defined in the automation control file for the infrequent error threshold.

The variable *time* shows the time the exception message was generated.

The variable *restype* shows the type of the resource.

The variable *resname* shows the name of the resource.

The variable *number* shows the number of errors that has occurred.

The variable *error_time* shows the time of the first error to occur within the time interval.

The variable *error_date* shows the date of the first error to occur within the time interval.

System action: If the resource identified in this message is a network resource, the recovery process continues. If it is a system resource, the appropriate commands are issued.

Operator response: Contact your system programmer.

System programmer response: Make sure the policy for shutting down this resource in the automation control file is accurate. Determine whether other action needs to be taken.

Classes: 40 43 52.

AOF510I *time : rename flag* **FLAG HAS BEEN SET TO *status* BY OPERATOR *operator_ID***

Explanation: The flag status for the resource identified in this message has been set as indicated.

The variable *time* shows the time when this message was generated.

The variable *rename* shows the name of the resource.

The variable *flag* shows the name of the automation flag that is set.

The variable *status* shows the automation status.

The variable *operator_ID* shows the name of the operator who changed the automation status.

System action: The automation status is set.

Operator response: You can issue the DISPFLGS command to view actual and effective values for the specific resource flag.

System programmer response: None.

Classes: 40, 43.

AOF511I *time : member* **AUTOMATION CONTROL FILE COMMON VALUES HAVE BEEN INITIALIZED**

Explanation: The automation control file data has been used to create common variables used throughout the automation process.

The variable *time* shows the time this message was generated.

The variable *member* shows the name of the configuration member used.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF512I *time : rename flag* **FLAG HAS BEEN RESET BY OPERATOR *operator_ID*. THE EFFECTIVE FLAG VALUE IS: *efinfo***

Explanation: The flag status for the resource identified in this message has been set as indicated.

The variable *time* shows the time when this message was generated.

The variable *rename* shows the name of the resource.

The variable *flag* shows the name of the automation flag that is set.

The variable *operator_ID* shows the name of the operator who changed the automation status.

The variable *efinfo* shows the effective value of flag after RESET operation.

The additional EFFECTIVE FLAG VALUE information is only displayed for INGAUTO RESET with SCOPE=ONLY.

System action: The automation status is set.

Operator response: You can issue the DISPFLGS command to view actual and effective values for the specific resource flag.

System programmer response: None.

Classes: 40, 43.

AOF521I **MONITORING *target* FAILED : *text***

Explanation: Monitoring failed for the reason specified in the message text.

The variable *target* shows the name of the target system.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Determine the cause of the problem from the message text. Contact the IBM support representative if required.

Classes: 0.

AOF522I **SYSTEM *target* IS LOADING: *text***

Explanation: Target system is loading. The message text indicates the system loading status.

The variable *target* shows the name of the target system.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 0.

AOF523W SYSTEM *target* IS INACTIVE: *text*

Explanation: Definition or hardware problem detected.

The variable *target* shows the name of the target system.

System action: System target status changed to inactive.

Operator response: Contact your system programmer.

System programmer response: Determine the cause of the problem from the message text. Correct the problem as indicated in the message text.

Classes: 0.

AOF524I MONITORING *target* : *text*

Explanation: Operational or configuration problem detected.

The variable *target* shows the name of the target system.

System action: None.

Operator response: Try to correct the problem. Contact your system programmer if required.

System programmer response: Correct the problem as indicated in the message text.

Classes: 0.

AOF530I *time* : CONFIGURATION TIMER PROCESSING HAS BEEN COMPLETED

Explanation: This message issued when all the timer entries in the automation control file have been processed and the required actions have been performed.

The variable *time* shows the time this message was generated.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF531I *time* : OUTSTANDING REPLY PROCESSING HAS BEEN COMPLETED

Explanation: This message is issued when outstanding replies have been captured and forwarded to SDF (Status Display Facility). It is produced at system automation startup, when SDF is recycled, or whenever the gateway connection to the focal point is established, changed, or reset.

The variable *time* shows the time this message was generated.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF532I *time* : AUTOMATION ENVIRONMENT HAS BEEN INITIALIZED

Explanation: The automation environment has been initialized.

The variable *time* shows the time this message was generated.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF534I *time* : AUTOMATION TIMERS HAVE BEEN INITIALIZED OR RESTARTED

Explanation: The automation timers have been initialized or restarted.

The variable *time* shows the time this message was generated.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF535I *time* : AUTOMATION ENVIRONMENT HAS BEEN CLEARED

Explanation: After an ACF COLD command is issued to reload an automation control file, the existing common global variables are cleared in readiness for fresh data.

The variable *time* shows the time this message was generated.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF537I *time* : *rename status - restype resname cmd* COMMAND(S) NOT FOUND FOR *console - text*

Explanation: One or more commands issued from the

console identified in this message could not be processed.

The variable *time* shows the time this message was generated.

The variable *resname* shows the name of the resource.

The variable *status* shows the status of the resource.

The variable *restype* shows the type of the resource.

The variable *cmd* shows the command to be processed.

The variable *console* shows the console-ID or console-CUU having difficulty.

The variable *text* shows the short form of the message.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Determine whether a configuration error or a user error exists, by reviewing the values supplied and the automation control file.

Classes: 40 43 52.

AOF538I *time* : OUTSTANDING REPLY: MVS -
reply_ID msg_text

Explanation: Automation will not reply to an outstanding reply.

The variable *time* shows the time this message was generated.

The variable *reply_ID* identifies the outstanding reply that will not be replied to.

The variable *msg_text* shows the message text for the outstanding reply.

System action: None.

Operator response: Reply to outstanding reply accordingly.

System programmer response: None.

Classes: 40 41.

AOF540I *time* : INITIALIZATION RELATED
PROCESSING HAS BEEN
COMPLETED

Explanation: Processing required to complete initialization after the automation environment has been initialized has been completed.

The variable *time* shows the time that the message was generated.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF541I *time* : SYSTEM MONITORING
FUNCTION COMPLETED FOR
resourcelist

Explanation: Indicates completion of a monitor cycle initiated either by using a timer or by operator request.

The variable *time* shows the time that the message was generated.

The variable *resourcelist* shows one of:

- ALL SUBSYSTEMS WITH NO INDIVIDUAL MONITORING (MONITOR \$SYSTEM)
- ALL SUBSYSTEMS (MONITOR ALL or *)
- A list of resources that monitoring has been completed for as the result of a single subsystem or wildcard request

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF543E *time* : COMMAND *cmd* WAS
CANCELED. RC=*retcode*

Explanation: Command *cmd* was canceled because the prevailing environmental conditions for system automation were not adequate for the command to complete successfully.

The variable *time* shows the time this message was generated.

The variable *cmd* shows the command that was affected.

The variable *retcode* shows the condition code identifying the reason for the cancelation of the command. outstanding reply.

System action: The command is terminated.

Operator response: Try re-issuing the command later.

System programmer response: None.

Classes: 40, 43.

AOF545I *clist*: KEYWORD *keyword* HAS AN
INVALID VALUE *value*, PARAMETER
IGNORED

Explanation: Either the keyword identified in this message or the value specified for it is not valid. The parameter containing this value will be ignored.

The variable *clist* shows the name of the command list.

The variable *keyword* shows the keyword that is not valid or has a value that is not valid.

The variable *value* shows the value specified for the keyword.

System action: None.

Operator response: None.

System programmer response: Correct the problem by changing the coding for the keyword or value.

Classes: None.

AOF546I *clist:* NO FUNCTION REQUESTED,
NOTHING DONE

Explanation: No function keyword was specified to cause an action. No action was taken.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Determine the cause of the problem. See whether message AOF545I was also issued. It is likely that a coding error caused the problem. For example, a keyword might have been specified with no default value possible, due to a coding error.

Classes: None.

AOF550I STATUS OF *monitor* MONITOR :
OBSERVED=*observed* HEALTH=*health* -
text

Explanation: This message is generated when either the status of a monitor resource or the health status of the objects that the monitor is watching changes. It shows the resource's new status.

The variable *monitor* shows the name of the monitor resource.

The variable *observed* shows the observed status of the monitor resource.

The variable *health* shows the health status of the monitored objects.

The variable *text* shows additional information.

System action: None.

Operator response: None.

System programmer response: None.

AOF551I *group* GROUP STATUS FOR *type*
APPLGROUP IS *observed*

Explanation: This message is generated when the status of an application group changes. It shows the group's new status.

The variable *group* shows the name of the application group.

The variable *type* shows whether the application group belongs to a system or a sysplex.

The variable *observed* shows the observed status of the application group.

System action: None.

Operator response: None.

System programmer response: None.

AOF561A *time* : JES2 RESOURCE DRAINING - *opt*
COMMAND NOT FOUND TO FORCE
DRAIN. OPS ACTION REQUIRED

Explanation: The indicated resource is shutting down, but no commands were found to drain the indicated resource with the method of FORCE.

The variable *time* shows the time this message was generated.

The variable *opt* shows the parameter for FORCE DRAIN.

System action: None.

Operator response: Manually drain the indicated resource.

System programmer response: Use the customization dialogs to add commands for the indicated action to the automation control file. For further information, refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

Classes: 40, 43, 44, 46.

AOF562I *time* : JES2 RESOURCE DRAINING -
resource IS *status*. OPS ACTION
REQUIRED TO DRAIN

Explanation: The JES2 resource status is not ACTIVE, INACTIVE, DRAINING, or DRAINED.

The variable *time* shows the time this message was generated.

The variable *resource* shows the name of the resource.

The variable *status* shows the JES2 resource status.

System action: None.

Operator response: Manually drain the resource.

System programmer response: None.

Classes: 40, 43, 44, 46.

AOF563A *time* : JES2 RESOURCE DRAINING -
UNKNOWN RESTYPE: *resource*. OPS
ACTION REQUIRED TO DRAIN

Explanation: The resource identified in this message is unknown.

The variable *time* shows the time this message was generated.

The variable *resource* shows the name of the resource.

System action: None.

Operator response: Manually drain the resource.

System programmer response: Check the netlog for further information and perform problem determination.

Classes: 40, 43, 44, 46.

AOF564A *{time: JES2 SHUTTING DOWN - message
- OPERATOR INTERVENTION
REQUIRED | title | task_names |
job_names | device_names}*

Explanation: This message reports that JES2 is being shut down and there are jobs or tasks (or both) that are active, or devices that have not yet been drained, or all of these. This message is followed by one or more AOF654A messages that lists the jobs or tasks that are active (*job_names*, *task_names*) or the devices that have not been drained (*device_names*). Each of these messages shows a list of job names, task names or device names.

The variable *time*

The variable *message*

The variable *title* is one of:

TASKS
JOBS
DEVICES

The variable *task_names* shows the names of tasks that are still active.

The variable *job_names* shows the names of jobs that are still active.

The variable *device_names* shows the names of devices that are still active.

System action: The system waits for devices and initiators to be drained. This message is repeated until JES2 is shut down.

Operator response: Wait for devices and jobs to complete and automation to shutdown JES2. Commands can be issued to force jobs or drain devices. Consult your system programmer if the situation requires manual intervention.

System programmer response: If this situation requires immediate shutdown instruct operations personnel as to how to accomplish this.

Classes: 40, 43, 44, 46.

AOF565A *time : JES2 RESOURCES DRAINING
BUT DRAIN COMMAND TURNED
OFF ON subsystem_function ENTRY FOR
resource_name - recommendation*

Explanation: JES2 resources are draining. However, the drain command is turned off for the identified subsystem function (*subsystem_function*) and resource (*resource_name*).

The variable *time* shows the time this message was generated.

The variable *subsystem_function* shows the function being performed.

The variable *resource_name* shows the name of the resource.

The variable *recommendation* shows an action recommended for solving the problem.

System action: None.

Operator response: Issue commands to drain the resource identified in this message. Contact your system programmer.

System programmer response: If you want to automate draining of the resource identified in this message, use the customization dialogs to correct the automation control file entry for the subsystem function and resource. For more information, refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

Classes: 40, 43, 44, 46.

AOF566I *time: JES2 PROCESSING REJECTED IN
command - JES2 NOT IN SHUTDOWN
PROCESS*

Explanation: Shutdown related JES2 processing rejected in routine *command*, because shutdown isn't in progress for JES2.

The variable *time* shows the time this message was generated.

The variable *command* shows the name of the routine, rejecting JES2 processing.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Check the netlog for further information on why command has been issued outside the shutdown process of JES2.

Classes:None

AOF568I *time : STATUS OF domain OUTBOUND
GATEWAY TO DOMAIN target_domain
IS status*

Explanation: This message shows the status of the outbound gateway from this domain to the target domain.

The variable *time* shows the time this message was generated.

The variable *domain* shows the domain initiating the connection to the target domain.

The variable *target_domain* shows the domain that the gateway is connected to.

The variable *status* shows the current status of the gateway connection.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40, 43, 44, 46.

AOF569I *time* : STATUS OF *domain* INBOUND
GATEWAY FROM DOMAIN
target_domain IS *status*

Explanation: This message shows the status of the inbound gateway from the requested target domain to this domain.

- The variable *time* shows the time this message was generated.
- The variable *domain* shows the name of this domain.
- The variable *target_domain* shows the requested target domain connected to this domain.
- The variable *status* shows the current status of the gateway connection.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40, 43, 44, 46.

AOF570I *time* : ISSUED "*command*" FOR *restype*
resname - *text*

Explanation: A command has been issued for the resource identified in the message.

- The variable *time* shows the time this message was generated.
- The variable *command* shows the command that has been issued.
- The variable *restype* shows the type of the resource.
- The variable *resname* shows the name of the resource.
- The variable *text* shows additional information.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF571I *time* : *resname* SUBSYSTEM STATUS
FOR JOB *job_name* IS *status* - *text*

Explanation: This message is generated when the status of a resource changes. It shows the resource's new status.

- The variable *time* shows the time this message was generated.
- The variable *resname* shows the name of the resource.
- The variable *job_name* shows the name of the job.
- The variable *status* shows the new status of the resource.
- The variable *text* shows additional information.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40 43 45.

AOF572I CGLOBALS NOT INITIALIZED FOR
AUTOMATED FUNCTION *autofunc* -
UNABLE TO ROUTE COMMAND
command, *operand_1*, *operand_2*, *operand_3*

Explanation: The command identified in this message could not be routed properly. The first three operands of the command are shown in the message, to aid analysis of the problem.

- The variable *autofunc* shows the automated function that the command could not be routed to. If no automated function was specified, this will be blank.
- The variable *command* shows the command that was not routed properly.
- The variable *operand_1* shows the first operand of the command. This variable is blank if the command has no required operands.
- The variable *operand_2* shows the second operand of the command. This variable is blank if the command has less than two required variables.
- The variable *operand_3* shows the third operand of the command. This variable is blank if the command has less than three required variables.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Check the netlog for messages related to an initialization failure. Determine the cause of the failure and correct the problem.

Classes: 40 43 45.

AOF573I *time* : AUTOMATED *task* *restype* HAS
BEEN RESTARTED

Explanation: An automated task has been restarted.

- The variable *time* shows the time this message was generated.
- The variable *task* shows the task identifier of the task that has been restarted.
- The variable *restype* shows the type of the resource.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40 43 45.

AOF574I *time* : RECOVERY INITIATED, JOB
job_name, - RECEIVED MSG "text"

Explanation: Recovery has been started for a job.

The variable *time* shows the time this message was generated.

The variable *job_name* shows the name of the job that recovery has been started for.

The variable *text* shows the message text received.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40 43 45.

AOF575A *time* : JOB *job_name* HAS ENDED -
AUTOMATED RECOVERY NOT IN
PROGRESS - OPERATION
INTERVENTION REQUIRED

Explanation: A job has ended and automation has determined that operator intervention is required.

The variable *time* shows the time this message was generated.

The variable *job_name* shows the name of the job that ended.

System action: Automation is discontinued for the specified job.

Operator response: Determine what is wrong with the job and contact your system programmer.

System programmer response: Review the netlog to determine when and why processing of the job failed. You might need to modify the automation control file to correct the problem if you can find no other cause of the failure.

Classes: 40 41 44 45 46.

AOF576A *time* : JOB *job_name* HAS *status* - *autotype*
AUTOMATION SET OFF -
OPERATION INTERVENTION
REQUIRED

Explanation: A job has ended and automation is set to OFF for the resource. Operator intervention is required to restart the resource.

The variable *time* shows the time this message was generated.

The variable *job_name* shows the name of the job that ended.

The variable *status* shows the status of the resource.

The variable *autotype* shows the type of automation.

System action: None. No automation is attempted.

Operator response: Restart the resource.

System programmer response: None.

Classes: 40 41 44 45 46.

AOF577E *time* : RECOVERY FOR *restype resname*
(JOB *job_name*) HALTED - CRITICAL
THRESHOLD EXCEEDED

Explanation: Automated recovery of a resource will not occur because the critical error threshold set for the resource has been exceeded. Operator intervention is required to restart the resource.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource.

The variable *resname* shows the name of the resource.

The variable *job_name* shows the name of the job.

System action: No automated recovery of the resource occurs.

Operator response: Contact your system programmer.

System programmer response: Check the netlog for further information on why the resource has failed.

Classes: 40 41 44 50.

AOF578I *time* : RECOVERY FOR *restype resname*
(JOB *job_name*) CONTINUING -
FREQUENT THRESHOLD EXCEEDED

Explanation: A resource has failed and automated recovery is occurring. The frequent error threshold set for the resource has been exceeded, but automation continues the recovery.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource that failed.

The variable *resname* shows the name of the resource that failed.

The variable *job_name* shows the name of the job.

System action: Automated recovery of the resource continues.

Operator response: Notify your system programmer.

System programmer response: Check the netlog for further information on why the resource has failed.

Classes: 40 43 51.

AOF579I *time* : RECOVERY FOR *restype resname*
(JOB *job_name*) CONTINUING -
INFREQUENT THRESHOLD
EXCEEDED

Explanation: Automated recovery is occurring for a resource. The infrequent error threshold set for the

AOF580I • AOF583I

resource has been exceeded, but automation continues the recovery.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource that automated recovery is occurring for.

The variable *resname* shows the name of the resource that automated recovery is occurring for.

The variable *job_name* shows the name of the job.

System action: Automated recovery for the resource continues.

Operator response: Notify your system programmer.

System programmer response: Check the netlog for further information as to why the indicated resource failed.

Classes: 40 43 52.

AOF580I *time* : *auto* **AUTOMATION FOR** *restype*
resname **(JOB** *job_name* *job_number*) **IS**
SET OFF - AUTOMATION NOT
ATTEMPTED FOR MSG *text*

Explanation: Automation has been set to OFF for a resource. As a result, no automation has been attempted for the message shown.

The variable *time* shows the time this message was generated.

The variable *auto* shows the type of automation.

The variable *restype* shows the type of resource that automation has been set OFF for.

The variable *resname* shows the name of the resource that automation has been set OFF for.

The variable *job_name* shows the name of the job.

The variable *job_number* shows the number of the job.

The variable *text* shows the message that has not been automated.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40, 43, 44.

AOF581I *time* : *autotype* **SUCCESSFUL FOR**
resource - *text*

Explanation: Automation caused by a message was successful for a resource.

The variable *time* shows the time this message was generated.

The variable *autotype* shows the type of automation.

The variable *resource* shows the name of the resource that automation was successful for.

The variable *text* shows the message that caused automation to be initiated.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40 43 45.

AOF582A *time* : **RECOVERY UNSUCCESSFUL -**
JOB *job_name* **- OPERATOR**
INTERVENTION REQUIRED FOR
MESSAGE *text*

Explanation: Recovery was not successful for a job. Operator intervention is required for the message shown, which could not be automated.

The variable *time* shows the time this message was generated.

The variable *job_name* shows the name of the job that recovery was not successful for.

The variable *text* shows the message that could not be automated.

System action: None.

Operator response: Manual intervention is required to handle the problem.

System programmer response: Determine from the netlog why recovery was unsuccessful. Review the commands coded in the automation control file.

Classes: 40 43 44 45 46.

AOF583I *time* : *auto* **AUTOMATION FOR** *restype*
resname **(JOB** *job_name*) **IS SET OFF -**
AUTOMATION NOT ATTEMPTED

Explanation: Automation has been set to OFF for resource. As a result, no automation has been attempted.

The variable *time* shows the time this message was generated.

The variable *auto* shows the type of automation flag.

The variable *restype* shows the type of resource that automation has been set OFF for.

The variable *job_name* shows the name of the job.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40, 43, 44.

AOF584I *time* : *rename autotype* IS SET ON -
autotype action NOT FOUND FOR *restype*
rename - "text"

Explanation: Automation was set ON for a resource, but no commands or replies were found in the automation control file for the type of automation described by the variable *autotype*.

The variable *time* shows the time this message was generated.

The variable *rename* shows the resource name.

The variable *autotype* shows the type of automation that no commands or replies were found for.

The variable *action* shows the type of action that should have been taken. Possible values are COMMAND and REPLY.

The variable *restype* shows the type of resource.

The variable *text* shows the message that called automation.

System action: Automation is not performed for the message shown.

Operator response: Notify your system programmer.

System programmer response: Check the entries in the automation control file and correct them as needed.

Classes: 40, 43, 44.

AOF585I *time* : *autotype* OF *resname* IS ALREADY
 IN PROGRESS - *text*

Explanation: Automation is in progress for a resource. No additional automation of the same type will occur until the current automation is complete.

The variable *time* shows the time this message was generated.

The variable *autotype* shows the type of automation in progress.

The variable *resname* shows the name of the resource that automation is in progress for.

The variable *text* shows the message that called automation.

System action: The current automation continues.

Operator response: None.

System programmer response: None.

Classes: 40, 43, 44.

AOF586A *time* : TIME OUT OCCURRED : MVS D
 C,B FAILED - AUTOMATION
 RECOVERY ATTEMPT TERMINATED

Explanation: A console buffer shortage recovery attempt has timed out. The recovery attempt is terminated.

The variable *time* shows the time when this message was generated.

System action: None.

Operator response: Notify your system programmer.

System programmer response: Determine why the MVS D C,B system command is not responding. Should the console be overloaded with excessive messages, issue the MVS K command to resolve console buffer shortages. Refer to *MVS/ESA System Commands* for further details.

Classes: 40, 43, 44, 46.

AOF587I *time* : RECOVERY FOR *restype* *resname*
 CONTINUING - CRITICAL
 THRESHOLD *action*

Explanation: Recovery for a resource is in progress and the critical error threshold for the resource has been exceeded. The commands specified to be issued when this happens have been issued.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of the resource that recovery is in progress for.

The variable *resname* shows the name of the resource that recovery is in progress for.

The variable *action* is either 'COMMANDS ISSUED' or 'EXCEEDED'

System action: If *action* is 'COMMANDS ISSUED' then automation issues the commands specified in the automation control file for critical threshold. If *action* is 'EXCEEDED' processing continues.

Operator response: None.

System programmer response: Determine why the critical threshold has been reached for this resource.

Classes: 40 41 44 50.

AOF588I *time* : RECOVERY FOR *restype* *resname*
 CONTINUING - FREQUENT
 THRESHOLD *action*

Explanation: Recovery is in progress for a resource and the frequent error threshold has been exceeded. The commands specified to be issued when this happens have been issued.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of the resource that recovery is in progress for.

The variable *resname* shows the name of the resource that recovery is in progress for.

The variable *action* is either 'COMMANDS ISSUED' or 'EXCEEDED'.

System action: If *action* is 'COMMANDS ISSUED' then automation issues the commands specified in the automation control file for the frequent error threshold.

If *action* is 'EXCEEDED' processing continues.

Operator response: None.

System programmer response: Determine why the frequent error threshold for this resource was exceeded.

Classes: 40 43 51.

AOF589I *time* : RECOVERY FOR *restype* *resname*
CONTINUING - INFREQUENT
THRESHOLD *action*

Explanation: Recovery is in progress for a resource and the infrequent error threshold for the resource has been exceeded. The commands specified to be issued when this happens have been issued.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of the resource that recovery is in progress for.

The variable *resname* shows the name of the resource that recovery is in progress for.

The variable *action* is either 'COMMANDS ISSUED' or 'EXCEEDED'

System action: If *action* is 'COMMANDS ISSUED' then automation issues the commands specified in the automation control file for the infrequent error threshold. If *action* is 'EXCEEDED' processing continues.

Operator response: None.

System programmer response: Determine from the netlog why the errors are occurring.

Classes: 40 43 52.

AOF592E SHUTDOWN REJECTED - *autotype*
FLAG FOR *subsystem* IS OFF

Explanation: Shutdown processing is rejected for a resource that automation is set to OFF for.

The variable *autotype* shows the type of automation (AUTOMATION or TERMINATION).

The variable *subsystem* indicates the resource in question.

System action: None.

Operator response: Shut down the resource manually or change the automation flag to ON, and re-invoke the INGREQ=STOP command.

System programmer response: None.

Classes: None.

AOF593A *time* : ALL SHUTDOWN
COMMANDS/REPLIES FOR *restype*
resname (JOB *job_name*) HAVE BEEN
ISSUED - OPERATOR INTERVENTION
REQUIRED

Explanation: All shutdown commands in the automation control file have been issued and a resource still has not shut down. Operator intervention is required to shut down the resource.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of resource that has not shut down.

The variable *resname* shows the name of the resource that has not shut down.

The variable *job_name* shows the name of the job.

System action: Automation waits for the resource to be shut down manually.

Operator response: Shut down the resource manually and notify your system programmer.

System programmer response: Determine from the netlog why the resource did not shut down.

Classes: 40, 43, 44, 46.

AOF595E *time* : *restype* *resname* (JOB *job_name*)
STATUS NOT AUTODOWN - STATUS
IS *status* - RESTART NOT ATTEMPTED

Explanation: A resource has been shut down but its status is not AUTODOWN. No restart will be attempted.

The variable *time* shows the time this message was generated.

The variable *restype* shows the type of the resource that was shut down.

The variable *resname* shows the name of the resource that was shut down.

The variable *job_name* shows the name of the job.

The variable *status* shows the status of the resource.

System action: None.

Operator response: Notify your system programmer.

System programmer response: Determine why the status of the resource is not AUTODOWN after shutdown. Either you must restart the resource manually or you must change the status to DOWN using the SETSTATE command and allow automation to start the resource.

Classes: 40, 43.

AOF597A *time : percent%* **MAX PASSES REACHED FOR console_ID - suggestion**

Explanation: Commands have been exhausted for a console.

The variable *time* shows the time this message was generated.

The variable *percent* shows the percentage of WTO buffer shortage.

The variable *console_ID* identifies the console that reached the buffer storage percentage.

The variable *suggestion* shows a recommended action for resolving the condition reported by the message.

System action: Recovery ends.

Operator response: Check for any outstanding REPLY, MOUNT, or multiline messages that are being held. These messages use the WTO buffer storage.

System programmer response: None.

Classes: 40 43 52.

AOF598A *time : percent%* **MAX PASSES REACHED FOR ALL CONSOLES USING > count BUFFERS - autotype ATTEMPT ENDED - suggestion**

Explanation: Commands have been exhausted for all consoles.

The variable *time* shows the time this message was generated.

The variable *percent* shows the percentage of WTO buffer shortage.

The variable *count* shows the number of buffers used.

The variable *autotype* shows the type of automation.

The variable *suggestion* shows a recommended action for resolving the condition reported by the message.

System action: Recovery ends.

Operator response: Check for any outstanding REPLY, MOUNT, or multiline messages that are being held. These messages use WTO buffer storage.

System programmer response: None.

Classes: 40 43 52.

AOF599A *time : percent%* **NO CONSOLES FOUND USING MORE THAN count BUFFERS - autotype ATTEMPT ENDED**

Explanation: No console has reached the threshold set for number of buffers. No recovery is attempted.

The variable *time* shows the time this message was generated.

The variable *percent* shows the percentage of WTO buffer shortage.

The variable *count* shows the number of buffers used.

The variable *autotype* shows the type of automation.

System action: None.

Operator response: If you suspect a problem with WTO buffers, additional checking is required to correct the problem. Contact your system programmer.

System programmer response: Either verify that the "number of buffers" threshold is set to a value that will accommodate demands, or allow recovery to detect that the threshold has been exceeded.

Classes: 40 43 52.

AOF601E *product_name* **IS DOWNLEVEL - PREREQ FOR SA z/OS IS NOT FULFILLED**

Explanation: The prerequisites for SA z/OS are not fulfilled. SA z/OS cannot be started.

The variable *product_name* indicates the product in question

System action: Processing terminates.

Operator response: None.

System programmer response: Install the required prerequisites.

Classes: None.

AOF602D **ENTER 'RETRY' OR 'CANCEL' - DOMAIN domid**

Explanation: A severe error, indicated by other messages, occurred during base initialization.

The variable *domid* shows the NetView domain ID.

System action: The system waits for an operator response.

Operator response: An associated message will provide more details about the problem. If necessary, contact your system programmer.

System programmer response: Find and fix the problem with your customization of system automation. If you cannot determine the cause of the problem, contact your IBM Support Center.

Classes: None.

AOF603D **ENTER AUTOMATION OPTIONS OR 'R' (RE-DISPLAY) - DOMAIN domid**

Explanation: This message is issued at the beginning of system automation initialization. Its purpose is to give the operator the opportunity to stop, pause, or override automation. A reply is generated automatically

for this message if there is no response from the operator.

The variable *domid* shows the NetView target domain ID.

System action: The system waits two minutes for a reply. If there is no reply after two minutes, automation continues.

Operator response: No reply is necessary if you want automation to continue with the default actions. Pressing Enter with no input causes automation to continue immediately using the default actions. Other options that may be specified are documented in the preceding multiline message, AOF767I. See AOF767I help for details or reply 'R' to redisplay message AOF767I.

System programmer response: None.

Classes: None.

AOF604I AUTOMATION PAUSED BY OPERATOR REQUEST

Explanation: Automation has been paused due to an operator request.

System action: Automation is paused.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF605I AUTOMATION DISABLED BY OPERATOR REQUEST

Explanation: Automation has been disabled as a result of the operator replying STOP to message AOF603D or message AOF606D.

System action: Automation is disabled.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF606D AUTOMATION PAUSED, ENTER OPTIONS OR 'R' (RE-DISPLAY) – DOMAIN *domid*

Explanation: Automation has been paused as a result of the operator replying PAUSE to message AOF603D.

The variable *domid* shows the NetView target domain ID.

System action: Automation pauses and waits for a reply to this message.

Operator response: Pressing Enter with no input causes automation to continue immediately using default values. Other options that may be specified are documented in the preceding multiline message,

AOF767I. See AOF767I help for details or reply 'R' to redisplay message AOF767I.

System programmer response: None.

Classes: 40, 43.

AOF607I AN *action* OF THE AUTOMATION ENVIRONMENT IS IN PROGRESS

Explanation: The automation is either initializing or reloading its automation environment.

The variable *action* shows which action is being performed in the automation environment. It can be one of the following:

BUILD The automation is either initializing, reloading, or restoring its automation environment

REFRESH

The automation is refreshing its automation environment

System action: If the *action* is **BUILD**, the automation is unavailable until the environment is established.

If the *action* is **REFRESH**, processing continues.

Operator response: None.

System programmer response: None.

Classes: 40.

AOF608E REXX Function Package INGRXFPG not installed

Explanation: The SA z/OS required REXX function INGRXFPG is not installed. However, the function package is mandatory for running SA z/OS.

System action: Processing terminates.

Operator response: None.

System programmer response: Install the required REXX function package. See *IBM Tivoli System Automation for z/OS Planning and Installation* for further guidance.

Classes: None.

AOF609I INVALID REPLY: *reason*

Explanation: May be one of the following:

- An operator command such as INGREQ or INGMOVE has issued a WTOR but the response to the WTOR is wrong

- The reply you gave in response to AOF603D or AOF606D WTOR is incorrect

The variable *reason* gives a brief explanation of what went wrong.

System action: Depending on the explanation, may be one of the following:

- | • The command re-issues the WTOR asking the operator for the correct answer.
- | • Automation is paused (if it was not already paused), the AOF767I message is reissued, and an AOF606D WTOR is issued so you can enter a correct response.

Operator response: Reply to the new WTOR.

Classes: 40 44 46.

AOF610I SHUTDOWN REJECTED -
subsystem_name **HAS NO SHUT** *shut_type*
DEFINED

Explanation: The shutdown type specified for this shutdown request has not been defined in the automation control file for the indicated subsystem.

The variable *subsystem_name* indicates the subsystem in question.

The variable *shut_type* shows the type of shutdown, NORM, IMMED, or FORCE.

System action: None.

Operator response: Either shut down the subsystem manually, or reissue the shutdown request with a higher priority shutdown type.

System programmer response: Include the shutdown type specification in the automation control file definition.

Classes: None.

AOF611I SHUTDOWN REJECTED -
subsystem_name **SHUT** *shut_type* **HAS NO**
COMMAND/REPLY

Explanation: The shutdown type specified for this shutdown request has no command or reply defined in the automation control file for the indicated subsystem.

The variable *subsystem_name* indicates the subsystem in question.

The variable *shut_type* shows the type of shutdown, NORM, IMMED, or FORCE.

System action: None.

Operator response: Either shut down the subsystem manually, or reissue the shutdown request with a higher priority shutdown type.

System programmer response: Include the shutdown type specification in the automation control file definition.

Classes: None.

AOF613D SPECIFY NEW RUNMODE.
RUNMODE *runmode* **NO LONGER**
EXISTS - DOMAIN *domid*

Explanation: The message is issued at the beginning of system automation initialization. A runmode is

currently set which does not exist in the policy.

The variable *runmode* shows the name of the current runmode.

The variable *domid* shows the NetView target domain id.

System action: The system waits for a reply.

Operator response: Specify a valid runmode or *ALL for RUNMODE=keyword. RUNMODE=? lists all possible runmodes. Although it is not recommended, you can keep the current runmode followed by an exclamation mark(!).

System programmer response: None.

Classes: None.

AOF614I TIMER *timerid* **IS DEFINED BUT IS**
MORE THAN 2 YEARS FROM NOW

Explanation: The requested timer is defined successfully but the YY portion of the requested date is more than 2 years past the current year.

The variable *timerid* shows the ID of the timer being defined.

System action: The requested timer is defined.

Operator response: Check that you have specified the year that you intended. If necessary, update the timer definition using the SETTIMER command.

System programmer response: None.

Classes: None.

AOF615I UNABLE TO START TASK *task*

Explanation: The task could not be started.

System action: Automation is paused. The message is followed by AOF604I and AOF767I.

Operator response: Reply to AOF767I. If the problem persists, contact your system programmer.

System programmer response: Check your DSIDMN file, or NetView style sheet DSIPARM member for the task in question.

| Also study the SYSLOG for associated IEC* messages to indicate an associated dataset issue.

Classes: 40, 43.

AOF617I SA Z/OS INITIALIZATION STARTED

Explanation: Initialization of SA z/OS has started.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

AOF618I NO VALID ACF FOUND FOR *sysname* -
reason

Explanation: No automation control file was found that could be loaded by system automation.

The variable *sysname* shows the system name that an automation control file could not be found for.

The variable *reason* shows the reason for the failure. It can be one of the following:

NO DEFAULT ACF FOUND IN AOFACFMP

The ACF mapping member does not contain a configuration for this system.

ACF MEMBER NOT AVAILABLE

The ACF member could not be accessed.

ACF TOKEN MISMATCH

The ACF does not have the same token as the automation manager's configuration file.

ACF CONTENT NOT VALID

The ACF was not built using the correct version of the customization dialogs or was corrupted.

ALLOCATION OF *dataset* FAILED

System automation was not able to dynamically allocate the ACF data set.

AUTOMATION TABLE IN ERROR

The test of a NetView automation table failed. Browse the netlog for further details and see the INGERRLS listing in the DSILIST data set.

MESSAGE REVISION TABLE IN ERROR

The test of a NetView message revision table failed. Browse the netlog for further details and see the INGERRLS listing in the DSILIST data set.

ACF CHECK FAILED WITH RC=*rc*

An error occurred while checking the ACF. Contact your IBM Support Center.

| CURRENT RUNMODE *runmode* NO LONGER
| EXISTS

| The ACF does not contain the runmode which
| is currently set.

System action: ACF load/refresh processing terminates.

Operator response: Contact your system programmer.

System programmer response: Fix the problem by providing a valid configuration for the automation agent:

1. Copy the configuration that the automation manager was loaded from back into your configuration data set and instruct the automation agent to Retry its cold start. This is only possible if you still have a copy of that data.

2. Instruct the automation manager to load the configuration currently on disk with F AM, REFRESH, *.
3. Instruct the automation manager and the automation agent to load their configuration from a different configuration file with F AM, REFRESH, *file_name*. Like option 1, this only works if you have an alternative file available.

Classes: 40, 43.

AOF620I *clist time* LENGTH ERROR, FORMAT SHOULD BE HH.MM, H.MM, MM, M, HH.M, HH., H., H.M or HH.MM.SS

Explanation: A timer value specified is not valid.

The variable *clist* shows which CLIST issues this message.

The variable *time* shows the timer interval entered.

System action: None.

Operator response: Enter the command again, specifying a valid value.

System programmer response: None.

Classes: None.

AOF621I *clist INVALID TIME SPECIFIED*
time_value/maximum_value

Explanation: A timer value specified is not valid.

The variable *clist* shows which CLIST issues this message.

The variable *time_value* shows the timer interval value.

The variable *maximum_value* shows the maximum value for either hours (24), minutes (59) or seconds (59).

System action: None.

Operator response: Enter the command again, specifying a valid value.

System programmer response: None.

Classes: None.

AOF622A NEW STATUS OF *auto_status* NOT VALID WHEN THE *type* STATUS IS *other_status*

Explanation: The automation status specified in the SETSTATE command is not valid for the current application monitor routine status or ARM status of the resource.

The variable *auto_status* shows the automation status specified in the SETSTATE command.

The variable *type* shows the type of resource status that is being compared with the automation status. The *type* value can be either MONITOR or ARM.

The variable *other_status* shows the application monitor or ARM status that the automation status is not valid for.

System action: None.

Operator response: Correct the automation status and reissue the SETSTATE command, or investigate the status of the resource.

System programmer response: None.

Classes: None.

AOF624I **NEW STATUS of *auto_status* NOT VALID FOR SUBSYSTEM OF TYPE *type***

Explanation: The automation status specified in the SETSTATE command is not valid for this type of subsystem.

The variable *auto_status* shows the automation status specified in the SETSTATE command.

The variable *type* shows the type of the subsystem, for example, MVS.

System action: None.

Operator response: Correct the automation status and reissue the SETSTATE command.

AOF625I **CONFIGURATION REFRESH STAGED FOR THE FOLLOWING REASON(S):**

Explanation: This message is the header line of a multiline message. A configuration refresh was delayed for the reasons listed with messages AOF626I.

System action: The configuration refresh is retried after 5 minutes.

Operator response: None.

Classes: None.

AOF626I *restype resname* **IS IN action**

Explanation: This message is part of a multiline message. It provides information about the reasons for a configuration refresh staging.

The variable *restype* is the resource type (that is, SUBSYSTEM).

The variable *resname* is the resource name.

The variable *action* is the action that is currently performed for the resource.

System action: The configuration refresh is delayed for this resource.

Operator response: None.

Classes: None.

AOF632I **REFRESHING OF SDF PANELS WILL NOT OCCUR**

Explanation: This message indicates that the SDFQNM command was not executed successfully during a reload or recycle, due to mismatching sysnames.

System action: Processing continues.

Operator response: Notify your system programmer.

System programmer response: Ensure the sysname matches the root system name defined in AOFTRREE.

AOF641E *time* : CLIST *name* **FAILED. PPI RECEIVER ID *id* DOES NOT EXIST. MESSAGES WILL NOT BE FORWARDED TO TEC**

Explanation: During the initialization of the TEC Notification service, it was detected that the PPI receiver ID of the GEM Event Adapter is not available.

The variable *time* shows the time when this message was generated.

The variable *name* shows the CLIST that issued this message.

The variable *id* shows the ID of the PPI receiver of the GEM Event Adapter.

System action: Events will not be forwarded to TEC.

Operator response: Notify your system programmer.

System programmer response: Check whether the GEM Event Adapter is up and running. If this is the case, check whether the PPI receiver ID used by the GEM Event Adapter is the same as the one that is expected by the TEC Notification service.

Classes: 0, 40, 44.

AOF642I *time* : TIVOLI ENTERPRISE CONSOLE NOTIFICATION SERVICE IS **ENABLED**

Explanation: The T/EC Notification service is fully initialized and the E/AS Message Adapter that the TEC Notification sends messages to for conversion to TEC Events is up and running.

The variable *time* shows the time when this message was generated.

System action: From now on system automation messages are forwarded to TEC via the E/AS Message Adapter.

Operator response: None.

System programmer response: None.

Classes: 40.

AOF643I *time* : TIVOLI ENTERPRISE CONSOLE NOTIFICATION SERVICE IS DISABLED

Explanation: The TEC Notification service has stopped forwarding messages to the E/AS Message Adapter.

The variable *time* shows the time when this message was generated.

System action: Messages are no longer forwarded to TEC.

Operator response: Notify your system programmer.

System programmer response: The PPI receiver ID of the E/AS Message Adapter was no longer available to the TEC Notification Service. Ensure that the E/AS Message Adapter is up and running.

Classes: 40.

AOF644E *time* : CLIST *name* FAILED. FOCAL POINT VARIABLE *varname* NOT SET. EVENTS WILL NOT BE FORWARDED TO TEC

Explanation: The system tried to retrieve the system automation primary focal-point name from the NetView global variable *varname*. If the name cannot be retrieved, message AOF644E is issued. The E/AS Message Adapter is expected to run on the system automation primary focal-point system.

The variable *time* shows the time when this message was generated.

The variable *name* shows the name of the CLIST that issued the message.

The variable *varname* shows the name of the focal point system.

System action: Events will not be forwarded to TEC.

Operator response: Notify your system programmer.

System programmer response: Make sure that you have customized the focal point forwarding path correctly.

Classes: 0 40 44.

AOF646I *time* : TIVOLI ENTERPRISE CONSOLE NOTIFICATION SERVICE IS INITIALIZED

Explanation: The Tivoli Enterprise Console Notification service is initialized successfully. The TEC Notification service is ready to forward messages to the E/AS Message Adapter. As soon as the E/AS Message Adapter is up and running, the AOF642I message is issued indicating that from now on messages will be forwarded to the E/AS Message Adapter.

The variable *time* shows the time when this message was generated.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: 40.

AOF660I **CURRENT FOCAL POINT IS**
domain_ID_1 , **WAS** *domain_ID_2* , **FOR**
domain_ID_3 **SYSTEM** *system_name*

Explanation: The focal point has been established or changed.

The variable *domain_ID_1* shows the domain ID of the current focal point.

The variable *domain_ID_2* shows the domain ID of the prior focal point.

The variable *domain_ID_3* shows the domain ID that is using *domain_ID_1* as its focal point.

The variable *system_name* shows the name of this domain.

System action: Communications to the focal point from *domain_ID_3* will now use *domain_ID_1*.

Operator response: None.

System programmer response: None.

Classes: 40.

AOF661I **DOMAIN** *domain_ID_1* **IS**
COMMUNICATING WITH *domain_ID_2*
, VIA *domain_ID_3* , *system_info* , *product*
version

Explanation: Communication is established to *domain_ID_2* from *domain_ID_1* via *domain_ID_3*.

The variable *domain_ID_1* shows the domain ID that established communications.

The variable *domain_ID_2* shows the domain ID that is now ready for communications.

The variable *domain_ID_3* shows the domain ID that is first in the communication path to *domain_ID_2*. This is equal to *domain_ID_2* if the communication is direct.

The variable *system_info* shows the sysplex name, the system name, and the sdfroot of *domain_ID_2* in the format sysplex.system(sdfroot). The sysplex (including the dot) is only shown if SA z/OS on *domain_ID_2* runs in a parallel sysplex.

The variable *product* shows the automation product running in *domain_ID_2*.

The variable *version* shows the version of that product in the format VnRnMn.

System action: Any other domains that were waiting on the activation of this domain are notified. If this domain is defined as a focal point, focal point reassignment is considered. If *domain_ID_1* has domains that have not yet established communication, and if this domain has been defined as a path to it, the other

domain is queried in *domain_ID_2* for its status.

Operator response: None.

System programmer response: None.

Classes: 40.

AOF662I **DOMAIN** *domain_ID_1* **HAS STOPPED COMMUNICATING WITH** *domain_ID_2* *system_name*

Explanation: Communication has been interrupted for the specified domain.

The variable *domain_ID_1* shows the domain ID that stopped communicating.

The variable *domain_ID_2* shows the domain ID that cannot be reached.

The variable *system_name* shows the name of *domain_ID_2*.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40.

AOF663I **DOMAIN** *domain_ID* **STATUS IS** *status*, **REPORTED BY** *reporting_domain*

Explanation: The reporting domain has reported the status of domain ID and the status was not a known status. Known statuses are ACTIVE, INACTIVE, and INVALID.

The variable *domain_ID* shows the domain ID whose status is being reported.

The variable *status* shows the status value of the domain ID.

The variable *reporting_domain* shows the domain ID of the domain reporting.

System action: None.

Operator response: Notify your system programmer.

System programmer response: The status is not a known status; further action is not possible. The origin of this status is reporting domain; it is either in error or it supports domain statuses that this product does not.

Classes: 1 40.

AOF664I **NO FOCAL POINT AVAILABLE**

Explanation: There is no focal point system available at this time.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40.

AOF667I **REQUEST TO DOMAIN** *domain_ID_1* **WAS REJECTED AT** *domain_ID_2*, **NO ROUTE DEFINED**

Explanation: A request issued for *domain_ID_1* was rejected by *domain_ID_2* because no route to *domain_ID_1* is defined.

The variable *domain_ID_1* shows the domain ID that the request was issued to.

The variable *domain_ID_2* shows the domain ID that rejected the request.

System action: None.

Operator response: None.

System programmer response: The message can be used to analyze gateway activity.

Classes: 1.

AOF668I **DOMAIN** *domain_ID* **REJECTED REQUEST:** *request*

Explanation: The identified domain rejected the request shown.

The variable *domain_ID* shows the domain ID that rejected the request shown.

The variable *request* shows the request that was rejected.

System action: None.

Operator response: None.

System programmer response: Either the domain ID does not support the request or the request is not valid.

Classes: 1.

AOF669I **REQUEST LOOPED BACK TO** *domain_ID_1*, **SENT ON** *domain_ID_2*, **RECEIVED FROM** *domain_ID_3*, **REQUEST:** *request*

Explanation: A request that originated in this domain looped back to this domain. The loop is stopped here and logged through this message.

The variable *domain_ID_1* shows the domain ID that received its own request back.

The variable *domain_ID_2* shows the domain ID that the request was probably sent out on. This domain is the domain that is currently being used to communicate to the target domain; this may have changed since this request was sent.

The variable *domain_ID_3* shows the domain ID that the request was received from. That domain should have been the next domain en route to the target domain.

The variable *request* shows the looping request. The target domain is contained in this request.

System action: None.

Operator response: None.

System programmer response: Scrutinize the gateway definitions in the path towards the target domain, including the alternate routing definitions. Expect to find a path that leads back to this domain that would result in an endless loop that never reaches the target domain.

Classes: 1.

AOF670I **COMMAND** (*command_text*) **WAS EXECUTED IN DOMAIN** *domain_ID* **BY OPERATOR** *operator_ID*

Explanation: A command was issued by this operator in this domain.

The variable *command_text* shows the text of the command executed.

The variable *domain_ID* shows the domain that the command was executed in.

The variable *operator_ID* shows the ID of the operator issuing the command.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF671I **COMMAND** (*command_text*) **FAILED EXECUTION IN DOMAIN** *domain_ID*, **RC=***return_code*

Explanation: An INGSSEND with RESP=ACK was issued by an operator from this domain. This message is an acknowledgment that the command executed on the target domain, but failed.

The variable *command_text* shows the text of the command attempted.

The variable *domain_ID* shows the domain that the command was attempted in.

The variable *return_code* shows the value returned for the failing command.

System action: None.

Operator response: The return code is specific to the command. Preserve the information for follow up action.

System programmer response: Check the command return codes for a match to determine the cause of failure and possible resolution.

Classes: None.

AOF672I **COMMAND** (*command_text*) **WAS EXECUTED IN DOMAIN** *domain_ID* **BY A** *crm* **COMMAND FROM OPERATOR** *operator_ID*

Explanation: An INGSSEND with RESP=ACK was issued by an operator from this domain. This message is an acknowledgment that the command executed successfully on the target domain.

The variable *command_text* shows the text of the command executed.

The variable *domain_ID* shows the target domain that the command was executed in. If the command routing mechanism used was RMTCMD, the domain may be qualified by a network ID, or "*" indicating that the default network ID was used to route the command.

The variable *crm* shows the command routing mechanism used. Possible values are **ROUTE** or **RMTCMD**.

The variable *operator_ID* shows the ID of the operator who issued the INGSSEND command.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF700I **CLIST** *name* **EXECUTING ON** *task* **WITH PARMS** > *parmstr*

Explanation: When debugging is active, this message is written to the netlog each time a command list is processed.

The variable *name* is the name of the clist.

The variable *task* is the name of the task that the clist ran on.

The variable *parmstr* shows the parameters processed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF701A **MESSAGE TABLE** *table* **WAS NOT LOADED - DETECTED ERRORS**

Explanation: Loading of the message table failed. Errors were detected while attempting to load *table*. Preceding messages describe the errors.

The variable *table* shows the name of the message table.

System action: A WTOR (AOF715D) is issued to ask the operator for a new table name to load.

Operator response: Contact your system programmer.

System programmer response: Follow the response instructions given in the description of the preceding messages. Correct the errors and have the operator respond to the WTOR with the new table name.

Classes: 40 44 46.

AOF702I **CLIST** *name* **COMPLETED ON** *task*
WITH RETURN CODE *rc*

Explanation: When debugging is active, this message is written to the netlog each time a command list has finished.

The variable *name* is the name of the clist.

The variable *task* is the name of the task that the clist ran on.

The variable *rc* is the return code given by the clist.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF705I *user* **ISSUED COMMAND** *command*
WITH *parameters*

Explanation: The specified user issued the command with the specified parameters. The message is only issued if command logging is enabled.

The variable *user* is the NetView user ID.

The variable *command* is the command that was issued.

The variable *parameters* lists the various parameters for the command (either specified or defaulted).

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF708I **THE LINE NUMBER ENTERED IS**
INVALID

Explanation: When attempting to change the number of the line that is displayed, an invalid line number was entered.

System action: None.

Operator response: Enter a valid line number.

System programmer response: None.

Classes: None.

AOF709I **THE COMMAND ENTERED IS**
INVALID

Explanation: The command that was specified is not valid.

System action: None.

Operator response: Type a valid command.

System programmer response: None.

Classes: None.

AOF710A *request* **INPUT AND THEN PRESS**
ENTER

Explanation: Input displayed on an operator dialog needs updating or validating before a command can be issued.

The variable *request* specifies the action required.

System action: None.

Operator response: Perform the requested action.

System programmer response: None.

Classes: None.

AOF711I **AUTOMATION OFF FOR** *subsys* , *value*
EFFECTIVE WHEN AUTOMATE SET
ON

Explanation: Automation has been set to OFF for a resource. When automation is reset to ON, the requested update to the flag will become enabled.

The variable *subsys* shows the name of the resource.

The variable *value* shows the flag that is to be updated and the value the flag is to be updated to.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF712I **INVALID CHARACTER** *character* **IN**
field_type *field*

Explanation: A data field of an automation control file contains an incorrect character.

The variable *character* shows the incorrect character.

The variable *field_type* shows the type of the field that contains the incorrect character. The field type can be either ENTRY or TYPE.

The variable *field* shows the field that contains the incorrect character.

System action: None.

Operator response: Specify valid entry/type fields with correct characters. Refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy* for

AOF713A • AOF719I

information about the correct syntax.

System programmer response: None.

Classes: None.

AOF713A CALL TO *macro* FAILED, RC *rc*

Explanation: A system automation program detected an unacceptable return code from a macro that it called. The meaning of the return code depends on the macro that issued it. Common examples are: DSICDS,28 : attempt to set a REXX variable failed as not called from REXX DSIDKS(FIND),4 : member not found or empty.

The variable *macro* shows the name of the macro that returned the error.

The variable *rc* shows the return code.

System action: The routine that called the macro continues processing if possible. If the routine cannot continue it will terminate with a non-zero return code. Other messages may be issued indicating that a problem occurred.

Operator response: Take appropriate corrective action. The action needed will depend upon the problem that caused the failure. For example the routine should be run from REXX if necessary. For empty or missing members either correct the member name or create the necessary member.

System programmer response: If the fixes under operator responses do not work make a note of the issuing routine, the macro and the return code and contact your IBM service representative for assistance.

Classes: None.

AOF715A ENTER THE NAME OF A VALID AUTOMATION TABLE

Explanation: This message is issued during system automation initialization. Errors were encountered during the load of the NetView automation table.

System action: The system waits for an operator response. SA z/OS initialization will only proceed after a valid NetView automation table is loaded.

Operator response: Enter the name of the NetView automation table that you wish to load. If you wish to load the table specified in your automation control file you need not enter the table name.

System programmer response: None.

Classes: None.

AOF717I SPECIFIED MEMBER NAME IS TOO LONG

Explanation: A string passed to a routine as the name of a member is more than 8 characters long.

System action: The routine terminates with a non-zero return code.

Operator response: If issuing the command directly, reissue it with the correct member name. If the command is issued from a CLIST refer the problem to your system programmer.

System programmer response: If the command is issued from a CLIST, fix the CLIST.

Classes: None.

AOF718A SYS-OPS UNABLE TO CONTINUE - OPERATOR ASSISTANCE ESSENTIAL

Explanation: The system automation program has encountered a problem that it is unable to bypass. Such problems are generally related to corrupt message tables or automation control files that are missing essential information.

System action: System automation stops working. Some message automation may occur if there was an active NetView automation table when the problem was found.

Operator response: An associated message will give more details of the problem. If necessary, contact your system programmer.

System programmer response: Find and fix the problem with your customization of system automation. If you cannot identify the cause of the problem, contact your IBM Support Center.

Classes: 40, 46.

AOF719I *clist* : *type* COMMAND *command* FAILED - *reason*

Explanation: A CLIST issued a command and did not get the expected result.

The variable *clist* shows the name of the CLIST that issued the command.

The variable *type* shows the type of command that was issued, for example MVS, or JES.

The variable *command* shows the command that failed.

The variable *reason* gives a description of problem encountered.

System action: CLIST processing usually stops.

Operator response: Notify your system programmer.

System programmer response: If the reason is timeout, change the timeout values in the automation control file. Fix any customization errors. If the problem persists, contact your IBM Support Center.

Classes: 40 46.

AOF720E NO VALID DATA FOUND IN AUTOMATION CONTROL FILE

Explanation: The automation control file is empty or all data was invalid.

System action: None.

Operator response: Notify your system programmer.

System programmer response: Correct the automation control file. Refer to *IBM Tivoli System Automation for z/OS Defining Automation Policy* for information on the syntax.

Classes: None.

AOF721I EASY MESSAGE MANAGEMENT DISABLED BY AOFSMARTMAT SETTING

Explanation: Easy Message Management is not active. The NetView automation table built by the customization dialogs will not be used.

System action: Processing continues with the AT from DSIPARM.

Operator response: None.

System programmer response: To enable Easy Message Management set the AAO AOFSMARTMAT to 2 using the NetView stylesheet.

Classes: 40 43.

AOF722E MEMBER *member* NOT FOUND IN *dataset* - USING DEFAULT *defmem* FROM *ddname*

Explanation: Easy Message Management is enabled, but the NetView automation table built by the customization dialogs could not be found.

The variable *member* is the name of the missing AT fragment.

The variable *dataset* is the name of the configuration data set.

The variable *defmem* is the name of the default AT fragment.

The variable *ddname* is the name of the DD statement within the NetView startproc that is used to search for *defmem*.

System action: The default AT delivered by SA z/OS is used.

Operator response: Notify your system programmer.

System programmer response: Determine why the dynamically built AT fragment is not available.

Classes: 40 46.

AOF723I MEMBER *member* FROM *dataset* LOADED INTO STORAGE AS AT INCLUDE FRAGMENT *defmem*

Explanation: Easy Message Management is enabled and the NetView automation table built by the customization dialogs replaces the default AT fragment delivered with SA z/OS.

The variable *member* is the name of the missing AT fragment.

The variable *dataset* is the name of the configuration data set.

The variable *defmem* is the name of the default AT fragment.

System action: Processing continues with the replaced AT fragment.

Operator response: None.

System programmer response: None.

Classes: 40 43.

AOF724I MEMBER *member* FROM *dataset* LOADED INTO STORAGE AS MRT *defmem*

Explanation: Easy Message Management is enabled and the NetView message revision table (MRT) built by the customization dialogs is activated, replacing any previously active MRT.

The variable *member* is the name of the missing MRT fragment.

The variable *dataset* is the name of the configuration data set.

The variable *defmem* is the name of the activated MRT.

System action: Processing continues with the replaced MRT.

Operator response: None.

System programmer response: None.

Classes: 40 43.

AOF726I FUNCTION SUCCESSFULLY COMPLETED - WARNINGS WRITTEN TO NETWORK LOG

Explanation: The function requested has been successfully executed. During execution warning messages have been written to the network log.

System action: None.

Operator response: Refer to the netlog for further information.

System programmer response: None.

Classes: None.

AOF727I *request* EXITS CANNOT BE ENABLED,
EXITS DO NOT EXIST - FLAG SET TO
ON

Explanation: A request has been made to enable the exits for a resource automation flag. Because no exits have been coded in the automation control file for this resource flag, the flag has been set to ON.

The variable *request* shows the resource name and automation flag.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF728I *subsystem type* RECOVERY
COMMANDS EXHAUSTED, BACK TO
PASS 1

Explanation: While attempting to recover a spool or buffer shortage system automation ran out of defined recovery commands before recovery was complete. System automation will re-execute the recovery command sequence starting from the first pass.

This situation may occur if the earlier recovery commands have disposed of excessive spool or buffer but more has accumulated while the earlier passes were running.

The variable *subsystem* shows the subsystem name.

The variable *type* shows the type of shortage problem being recovered. Possible values are SHORT and FULL.

The variable *resource* shows the resource having a shortage problem.

System action: The shortage recovery continues.

Operator response: You need to have a look at your spool or buffer usage. System automation may need some assistance to keep it in control. Look for tasks generating spool or consuming buffer that the recovery commands do not affect.

System programmer response: Review your recovery commands. You may wish to make them more aggressive or to add some more of them.

Classes: 40 45.

AOF729I *subsystem* SPOOL *type* RECOVERY
SUSPENDED ON PASS *pass* : *reason*

Explanation: While attempting to recover a spool shortage system automation found a reason why it could not continue with spool recovery. The most likely reason is that the Recover automation flag is or has been turned off. While the flag remains off spool usage will be monitored, but no recovery commands will be issued. If the flag is reset, the recovery commands will

be reissued from the pass indicated in the message.

The variable *subsystem* shows the subsystem name.

The variable *type* shows the type of spool problem being recovered. Possible values are SHORT and FULL.

The variable *pass* shows the recovery pass that was to be issued.

The variable *reason* shows the reason that system automation has not issued the recovery pass.

System action: The spool recovery continues. No recovery passes are issued.

Operator response: Reset the Recover automation flag if that is the problem and you wish the suspended recovery commands to be reissued.

System programmer response: Investigate the cause of the blockage. Take appropriate action to stop it from occurring again.

Classes: 40 45.

AOF730I SPECIFIED OPERAND *operand*
INVALID FOR *request*

Explanation: An operand specified on a command conflicts with the request.

The variable *operand* shows the operand that is invalid.

The variable *request* shows the parameter that the operand conflicts with and why.

System action: None.

Operator response: Reenter the request with the correct operand.

System programmer response: None.

Classes: None.

AOF732I SHUTDOWN PROCESSING RESUMED

Explanation: Shutdown processing has been resumed. Shutdown attributes have *not* been updated.

The variable *shutdown_id* shows the ID of the shutdown that has been resumed.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF735I UNABLE TO PROCESS SHUTDOWN -
AUTOMATION IS SET OFF FOR
subsystem

Explanation: Shutdown processing was unable to shut down the subsystem because the Terminate flag was set to OFF. Because the option for overriding the Terminate

flag was set to NO, the shutdown could not be performed.

The variable *subsystem* shows the name of the affected subsystem.

System action: None.

Operator response: If you want to shut down the subsystem regardless of its flag setting, reissue the INGREQ REQ=STOP request specifying OVERRIDE=FLG, or ALL if you also want to override other conditions.

System programmer response: None.

Classes: None.

AOF742D ENTER >GO< TO PROCEED OR
>CANCEL< TO CANCEL REQUEST OR
>R< TO REDISPLAY, USER=*userid*

Explanation: An operator command such as INGREQ or INGMOVE has been entered with VERIFY=WTOR specified. AOF742D is preceded by AOF778I that lists the resources that will be affected by the request

The variable *userid* shows the operator ID where the request originated. This can be used to tie AOF742D to the associated AOF778I.

System action: The operator is prompted by a WTOR.

Operator response: Verify the list of affected resources and reply with either GO or CANCEL to the WTOR or reply R to redisplay message AOF778I.

System programmer response: None.

Classes: None.

AOF743I SHUTDOWN WILL NOT
(RE)PROCESS SUBSYSTEM *subsystem*
AS IT IS *status*

Explanation: The shutdown considered shutting the named subsystem down, but then did not because it was not in a suitable state.

The variable *subsystem* shows the name of the subsystem being processed.

The variable *status* shows the current state of the subsystem.

System action: The thread of the shutdown process that found the problem will cease. The message does not necessarily indicate a problem.

Operator response: If the subsystem is in an error state you should recover it manually.

System programmer response: This message normally indicates that the shutdown process has already started to shut a subsystem down. This message may also be issued if the subsystem is in a STOPPING or DOWN status.

Classes: 40, 43.

AOF745E SHUTDOWN FOR SUBSYSTEM
subsystem : SUBSYSTEM NOT FOUND
AND NO FINAL TERMINATION
MESSAGE RECEIVED. SHUTDOWN
SUSPENDED FOR *time* MINUTES

Explanation: The shutdown monitoring has detected that the named subsystem has cleared address space, but that its final termination message was not received within at least 60 seconds of this happening.

The variable *subsystem* shows the name of the subsystem being processed.

The variable *time* shows the number of minutes the system will wait.

System action: The system waits up to *time* minutes for the subsystem's final termination message to arrive. If it does not, the shutdown will assume that the subsystem has terminated and issue the appropriate final message for the subsystem.

Operator response: Check that the subsystem has properly terminated. If it has not, you must help it manually as no more shutdown commands will be issued for it. If it has terminated normally, refer the problem to your system programmer.

System programmer response: Determine the reason that the final termination message was missed. You may need to correct your NetView automation table, or change it to use a better message. MPF is another place where the message could have been lost.

Classes: 40 46.

AOF746A SHUTDOWN FOR SUBSYSTEM
subsystem : NORMAL SHUTDOWN
PROCESSING HAS BEEN
SUSPENDED AS THE SUBSYSTEM IS
status

Explanation: A subsystem being shutdown by a normal shutdown (SHUTNORM) has entered either an ABENDING or BREAKING status. No further SHUTNORM commands will be issued.

The variable *subsystem* shows the name of the subsystem being processed.

The variable *status* shows the current status of the subsystem.

System action: The shutdown is suspended until the subsystem stops or the shutdown is modified to use a more aggressive set of shutdown commands. Both immediate and force shutdown commands can be issued against such subsystems.

Operator response: Determine the state of the subsystem. If it will eventually terminate you don't have to do anything. If more commands are needed to effect a shutdown, either issue them manually or modify the type of the shutdown for that subsystem. Refer the problem to your system programmer.

System programmer response: Examine the shutdown commands and the NetView automation table entries. If the normal shutdown commands cause the subsystem to abend this situation will recur.

Classes: 40 46.

AOF747A SHUTDOWN FOR SUBSYSTEM
subsystem IS STUCK. THE SUBSYSTEM IS NOT SHUT DOWN BUT THERE ARE NO MORE *type* PASSES TO ISSUE

Explanation: SA z/OS has issued all defined shutdown passes for a subsystem, but the subsystem has not stopped. Operator assistance is required to continue with the shutdown.

The variable *subsystem* shows the name of the subsystem being processed.

The variable *type* shows the type of shutdown commands being issued.

System action: The shutdown of the subsystem is unable to proceed. It will hang until either the subsystem is stopped by an external agency or the shutdown is cleared.

Operator response: Determine the status of the subsystem. If it is in the process of stopping, wait. If the subsystem is not stopping, either upgrade the type of shutdown commands being issued or shut it down manually. Inform your system programmer of the problem.

System programmer response: Examine your shutdown commands. If they are not sufficient to stop the subsystem, upgrade them. If the subsystem is consistently taking too long to stop either increase the subsystem's delay or add some dummy passes to the end of the shutdown commands.

Classes: 40 46.

AOF748E SHUTDOWN FOR SUBSYSTEM
subsystem : SUBSYSTEM NOT FOUND AND NO FINAL TERMINATION MESSAGE RECEIVED. AUTOMATED SHUTDOWN RESUMES

Explanation: The system is resuming a shutdown that was suspended when a subsystem cleared address space without issuing a final termination message. No such message has since been received, and it seems unlikely that one will be.

The variable *subsystem* shows the name of the subsystem being processed.

System action: System automation issues the final termination message on behalf of the subsystem. The shutdown proceeds as if the message had arrived normally.

Operator response: Inform your system programmer about the failure of the system to recognize the

subsystem's final termination message.

System programmer response: Examine your shutdown commands and NetView automation table entries. It is likely that they are not trapping the subsystem's final termination message, and this is causing unnecessary delays in your shutdown processing.

Classes: 40 46.

AOF749I SHUTDOWN COMPLETE FOR *resource*.
 (RESTART=*restart*, TYPE=*shuttype*)

Explanation: The indicated shutdown has completed successfully.

The variable *resource* shows the name of the resource that has been shut down.

The variable *restart* indicates what happens to the resource after the shutdown is complete. The value of *restart* can be YES.

The variable *shuttype* indicates the shutdown type.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF750I SHUTDOWN FOR SUBSYSTEM
subsystem MAY HAVE FAILED TO ISSUE SOME *shuttype itemtype* ON PASS *pass*

Explanation: While issuing the shutdown commands for the named pass return codes from AOF747E or AOF748E indicated that they may have failed to issue some of their commands or replies.

The variable *subsystem* shows the name of the subsystem being processed.

The variable *shuttype* shows the type of shutdown passes being issued.

The variable *itemtype* shows the COMMANDS or REPLIES indicating what failed.

The variable *pass* shows the pass of shutdown processing that it occurred on.

System action: The shutdown continues. If it did fail to issue the commands the shutdown may become stuck and issue an AOF747E message when it runs out of passes.

Operator response: Check the netlog and the system log to see whether the proper shutdown commands were issued. If they were not, you may need to shut the subsystem down manually.

System programmer response: Check that the shutdown commands are all valid when issued from a NetView console. If they are not, fix them. If you still get regular occurrences of this message try altering

your shut delay times or contacting your IBM Support Center, or both.

Classes: 40 46.

AOF751I CLEAN UP FOR SHUTDOWN DID NOT CHANGE THE STATUS OF SUBSYSTEM *subsystem* TO RESTART BECAUSE *reason*

Explanation: While a shutdown with RESTART=YES was being cleaned up, the status of the specified subsystem was not changed to RESTART for the reason given.

The variable *subsystem* shows the name of the subsystem being processed.

The variable *reason* shows the reason that the change was not made.

System action: The clean up continues. The named subsystem and its children will not be restarted when it is complete.

Operator response: To restart the subsystem, use the SETSTATE command.

System programmer response: If the message was issued because the subsystem abended or broke, you should check it out and, if the process was a normal part of its shutdown, change your automation so that it no longer enters an error state.

Classes: 40 43.

AOF752E SHUTDOWN FOR SUBSYSTEM *subsys* SUSPENDED. THE TERMINATE FLAG IS CURRENTLY TURNED OFF

Explanation: When the shutdown attempted to shut the specified subsystem down it checked its Terminate flag and found it to be turned off. No passes will be processed for the subsystem until the flag is turned on.

The variable *subsys* shows the name of the subsystem being processed.

System action: The shutdown is suspended. The shutdown CLISTs will be redriven every shut delay to recheck the Terminate flag. As soon as they find it turned on the next pass of shutdown commands will be issued.

Operator response: If you do nothing the shutdown remains suspended until the flag is turned on (possibly at the end of a NOAUTO or a TIMEOUT period). You could force the flag to change with the INGAUTO command.

System programmer response: Check whether the reason why the flag was turned off was valid. Ensure that the operators have instructions to deal with this situation should it arise again.

Classes: 40 46.

AOF755I SHUTDOWN FOR SUBSYSTEM *subsystem* : type PASS pass : AOFSHUTMOD SET TO *value*

Explanation: The shutdown processing found that the AOFSHUTMOD variable had been set to a non-null value by the subsystem's shutdown commands.

The variable *subsystem* shows the name of the subsystem being processed.

The variable *type* shows the type of shutdown being processed.

The variable *pass* shows the pass of the shutdown being processed.

The variable *value* shows the value of the TGLOBAL.

System action: The shutdown process continues. If the value is recognized by the shutdown process it will be used to alter the shutdown's flow. If it is not recognized it will be ignored. Recognized keywords are NOW, DELAY, ABORT and NEXTPASS. Note that both DELAY and NEXTPASS require their parameters to be correct.

Operator response: Notify your system programmer if the value is invalid. Note that if the parameter on a DELAY statement was invalid the automated shutdown will have stopped and you should shut the affected subsystem down manually.

System programmer response: If the value is invalid, set the value in the AOFSHUTMOD variable correctly.

Classes: 40, 43.

AOF757I UNABLE TO PROCESS *action* FOR *resource* - *reason*

Explanation: You invoked INGREQ for a resource.

The variable *action* shows the action that cannot be performed, for example, STARTUP.

The variable *resource* shows the name of the affected resource.

The variable *reason* shows an explanation why the action could not be performed.

System action: INGREQ stops processing.

Operator response: Change the parameters in the INGREQ call. If necessary, contact your system programmer.

System programmer response: Ensure that the resource can be managed by system automation (start and stop commands defined, automation flags turned on, etc.).

Classes: None.

AOF759A **AUTOMATED STARTUP FOR SUBSYSTEM** *subsystem* **MAY HAVE FAILED : RC** *rc* **FROM "ACFCMD MSGTYPE=STARTUP,ENTRY=** *subsystem"*

Explanation: System automation tried to start a subsystem. The ACFCMD call failed, giving the indicated return code.

The variable *subsystem* shows the name of the subsystem being processed.

The variable *rc* shows the return code from the ACFCMD call.

System action: The startup will probably fail. The subsystem's status will have been updated to INACTIVE with an appropriate message. If the subsystem does start, it will be processed normally.

Operator response: You must start the subsystem manually. Refer the problem to your system programmer.

System programmer response: The definition for the subsystem is either wrong or incomplete. Fix the definitions through the customization dialogos.

Classes: 40 46.

AOF760E *condition* **CONDITION TRAPPED IN** *clist* **LINE** *sourceline* **(CODE** *error_num*) **DESCRIPTION** *description*

Explanation: A REXX CLIST has encountered a fatal error condition.

The variable *condition* shows one of NOVALUE, SYNTAX, HALT, or FAILURE.

The variable *clist* shows the name of the CLIST that the error was trapped in.

The variable *sourceline* shows the source line number where the condition occurred.

The variable *error_num* shows N/A or SYNTAX error number or FAILURE return code.

The variable *description* shows the NOVALUE variable name, SYNTAX error description etc.

System action: The CLIST ends with code -5, causing calling CLISTs to end.

Operator response: Report the problem to your system programmer, unless the condition is operator caused (for example, NetView CLOSE causing a HALT condition).

System programmer response: Report the problem to your IBM Support Center.

Classes: 40 46.

AOF761E *entry* **NOT ACTIVE;** *description*

Explanation: The ACF loader did not find an expected entry to be ACTIVE.

The variable *entry* shows the entry that was expected to be ACTIVE.

The variable *description* shows details about the probable consequences of the entry not being ACTIVE.

System action: Processing terminates.

Operator response: None.

System programmer response: Investigate why this entry is not ACTIVE and correct the error. Then issue a cold start.

Classes: 40 46.

AOF762I *entry* **NOT ACTIVE;** *description*

Explanation: The ACF loader did not find an expected entry to be ACTIVE.

The variable *entry* shows the entry that was expected to be ACTIVE.

The variable *description* shows details about the probable consequences of the entry not being ACTIVE.

System action: None.

Operator response: None.

System programmer response: Examine the description and determine whether or not you wish this entry to be ACTIVE. If you do not require this entry to be ACTIVE, correct the reason why it is not ACTIVE. Then issue a cold start.

Classes: 40 46.

AOF763I **GATEWAY EXCMD TO OPERATOR** *operator_ID* **FAILED, RC=** *return_code* , **COMMAND (** *command_text* **)**

Explanation: A Gateway operation cannot be completed because the EXCMD to transfer command has failed.

The variable *command_text* shows the command that failed.

The variable *operator_ID* shows the ID of the operator that the command was sent to.

The variable *return_code* shows the return code from the failed command.

System action: The CLIST continues. The command will not have been actioned.

Operator response: Report the problem to your system programmer.

System programmer response: Determine why the operator is inactive or the EXCMD failed. If the

command is AOFMSG, check whether another domain is sending messages to be issued by a non-existent operator.

Classes: 40, 43.

AOF764I *field_name* **DOMAIN NAME** *domain_ID*
INVALID - reason

Explanation: While reading the control file or when processing the domain operand of a command or panel a *domain_ID* has been rejected.

The variable *field_name* shows the field where the *domain_ID* is declared

The variable *domain_ID* shows the *domain_ID* that has been rejected. The variable *reason* shows the explanation for the rejection.

System action: The control file entry, or the command is ignored.

Operator response: If a control file error, report the problem to your system programmer. If a command error, correct the domain name if in error.

System programmer response: Correct the domain name in the control file entry, or define the domain in the DSIPARM data set, member DSIDMN, or the NetView style sheet.

Classes: None.

AOF765E **NO entry DEFINED;** *description*

Explanation: The ACF loader did not find an expected entry in the automation control file.

The variable *entry* shows the entry that was expected.

The variable *description* shows details about the probable consequences of not supplying the entry.

System action: Processing terminates.

Operator response: None.

System programmer response: Add the required data through the customization dialogs.

Classes: 40 46.

AOF766I **NO entry DEFINED;** *description*

Explanation: The loader did not find an expected entry in the automation control file.

The variable *entry* shows the entry that was expected

The variable *description* details the likely consequences of not supplying the entry.

| Note that when the description indicates that
| PLEXOPER is not defined and "PROCESSOR AND/OR
| IMAGECNTL DATA IGNORED", the ISQSTART
| command will fail and ProcOps will be unavailable

| unless the Automated Function PLEXOPER is correctly
| defined in the OPERATORS Policy Item.

System action: None.

Operator response: None.

System programmer response: Examine the description and determine whether or not you wish to include the entry.

Classes: 40 46.

AOF767I **AUTOMATION OPTIONS:**

Explanation: This is the first line in a multiline message block that appears at system automation initialization. The lines of the message block define the options that can be specified for system automation initialization.

Note: Your response to this message block and its following WTORs is the chief means of determining how your system is automated. You should understand clearly the details of the message block as explained in the *IBM Tivoli System Automation for z/OS User's Guide*.

System action: AOF767I is followed by either an AOF603D or an AOF606D message. These are WTORs that can be responded to. System automation replies automatically to an AOF603D after two minutes with default values unless this has been changed as a part of your site's automation policy. System automation does not automatically reply to an AOF606D message unless this has been changed as a part of your site's automation policy.

Operator response: Varies. Refer to *IBM Tivoli System Automation for z/OS User's Guide* for details. Generally, reply to the AOF603D or AOF606D WTOR, or, for the AOF603D WTOR, let it time out.

System programmer response: Varies. Refer to *IBM Tivoli System Automation for z/OS User's Guide* for details.

Classes: None.

AOF768I **SUBSYSTEM** *subsys_name* **REQUIRES**
entry : **NOT FOUND IN AUTOMATION**
CONTROL FILE

Explanation: An entry that is required for a specific subsystem was not found in the automation control file. You may have a problem using this subsystem.

The variable *subsys_name* shows the name of the subsystem in question.

The variable *entry* shows the entry missing from the automation control file.

System action: None.

Operator response: None.

AOF769I • AOF775I

System programmer response: Consider adding the appropriate entry to the automation control file.

Classes: 40 46.

AOF769I TIMER IDS STARTING WITH *prefix* ARE RESERVED FOR SYS-OPS SYSTEM USE

Explanation: An attempt was made to delete, change or add a timer with a timer ID that starts with the character sequence *prefix*.

| The value of the variable *prefix* is either 'AOF',
| 'ING', or 'ITMR'.

| SA z/OS uses a timer ID prefix of 'AOF', 'ING', or
| 'ITMR' to distinguish its timers from other NetView
| timers and SA z/OS user timers. The operator interface
| prevents these timers from being altered. The control
| file initialization will not set a timer that starts with
| 'AOF', 'ING' or 'ITMR' because it might override a
| SA z/OS system timer. Note timers with a prefix of
| 'ITMR' cannot be deleted.

System action: The timer request was not processed.

Operator response: Change the timer ID to have another prefix. If you want to delete a SA z/OS system timer, use the NetView PURGE TIMER command. Do not purge any SA z/OS system timer unless you are certain of the consequences.

System programmer response: None.

Classes: 40, 43.

AOF770I THIS FUNCTION IS NOT AVAILABLE IN LINEMODE

Explanation: An attempt was made to execute a CLIST as a linemode command when the CLIST does not support the linemode facility.

System action: None.

Operator response: Issue command on an OST task to see command output.

System programmer response: None.

Classes: None.

AOF771I REQUEST *command* REJECTED, *action* IN PROGRESS

Explanation: The issued request cannot be processed at this time because a startup or shutdown is still in progress.

The variable *command* shows the request that failed.

The variable *action* shows either STARTUP or SHUTDOWN.

System action: Request is canceled.

Operator response: Wait for startup or shutdown to

complete or clear shutdown before reissuing request.

System programmer response: None.

Classes: None.

AOF772I PARSING ERROR PROCESSING *input - error*

Explanation: While processing the indicated item of input the parsing routine found an error.

The variable *input* shows the input that was being parsed when the error was detected.

The variable *error* shows the error that occurred.

System action: The command stops processing.

Operator response: Correct the parameter strings and rerun the command.

System programmer response: If the error occurred from within a CLIST you must edit the CLIST to fix the problem.

Classes: None.

AOF773I YOUR INSTALLATION AUTOMATION POLICY DISALLOWS AN ACF RELOAD

Explanation: A reload of the automation control file has been requested but your automation flags indicate that a reload is not allowed.

System action: The reload process is abandoned.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF774A YOUR INSTALLATION AUTOMATION POLICY REQUIRES CONFIRMATION OF AN ACF RELOAD - ENTER "GO" TO CONTINUE OR "CANCEL" IF YOU DO NOT WISH TO PROCEED

Explanation: A reload of the automation control file has been requested but your installation automation policy requires confirmation from the operator.

System action: Waits for operator response.

Operator response: Enter "GO" or "CANCEL".

System programmer response: None.

Classes: None.

AOF775I OPERATOR ENTERED *response - action*

Explanation: This message confirms a response entered by an operator.

The variable *response* shows the operator response.

The variable *action* shows the action taken by the system.

System action: As described in *action*.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

AOF776I CONNECTION FROM DOMAIN
domain_id_1 *sdroot_name* REJECTED AT
DOMAIN *domain_id_2* - *reason*

Explanation: An attempt to establish a cross-domain gateway has failed.

The variable *domain_id_1* shows the domain attempting to connect.

The variable *sdroot_name* shows the system attempting to connect.

The variable *domain_id_2* shows the domain rejecting the connection.

The variable *reason* shows the reason for rejecting the connection.

System action: The connection will be tried again at gateway monitor time.

Operator response: None.

System programmer response: The gateway definitions in the automation control file on either or both domains need correction to ensure compatibility.

Classes: 40, 43.

AOF778I VERIFY AFFECTED RESOURCES FOR
REQUEST "*request*", **USER=***userid*

Explanation: An operator command such as INGREQ or INGMOVE has been entered with VERIFY=WTOR specified. This multiline WTO lists all the resources that will be affected by the request.

The variable *request* shows the type of request entered.

The variable *userid* shows the operator ID originating the request. This can be used to tie AOF778I to the associated AOF742D.

System action: WTOR message AOF742D will be issued after AOF778I.

Operator response: Verify the list of affected resources before replying to message AOF742D.

System programmer response: None.

Classes: None.

AOF779I VALID RUNMODES ARE:

Explanation: This is the first line in a multiline message block that appears when RUNMODE=? was entered as a response to AOF603D. The lines of the message block show all valid runmodes and their runtokens for the current system.

System action: AOF767I and AOF603D are re-displayed.

Operator response: None.

System programmer response: None.

Classes: None.

AOF780E INTERNAL ERROR AT LOCATION *loc*.
UNABLE TO CONTINUE

Explanation: An internal error occurred at the specified location in the program code.

System action: Processing is terminated.

Operator response: Contact your system programmer.

System programmer response: Contact IBM Support Center.

Classes: None.

AOF781I SERVICE REQUEST *req* **FAILED AT**
LOCATION *loc* **WITH RC=** *rc*

Explanation: A service routine ended with a non-zero return code.

The variable *req* identifies the service request that failed.

The variable *loc* identifies the location in the code where the failure occurred.

The variable *rc* is the return code from the service routine.

System action: Processing terminates.

Operator response: Contact your system programmer.

System programmer response: Perform problem determination and correct the problem.

Classes: None.

AOF782I AUTOMATION CONTROL FILE
PROCESSING COMPLETED

Explanation: The automation control file was processed and completed with no errors.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

AOF783A AUTOMATION CONTROL FILE PROCESSING HAS FAILED

Explanation: The processing of the automation control file resulted in a fatal error. No control information is processed.

System action: Processing is terminated.

Operator response: Contact your system programmer.

System programmer response: Correct the problem identified by other messages.

Classes: None.

AOF784I AUTOMATION CONTROL FILE MEMBER *file* IS EMPTY OR DOES NOT EXIST

Explanation: An INCLUDE member was processed but did not contain any records or the member does not exist.

The variable *file* shows the name of member that caused the error.

System action: Processing continues.

Operator response: Contact your system programmer.

System programmer response: Correct the automation control file.

Classes: 40, 43.

AOF785I INVALID INCLUDE NAME IN MEMBER *member name*

Explanation: An INCLUDE member name in *member* is invalid, blank or contains invalid characters. The member name is ignored.

The variable *member name* shows the name of member that caused the error.

System action: Processing continues.

Operator response: Contact your system programmer.

System programmer response: Correct the member name.

Classes: 40, 43.

AOF786A NO DATA FOUND IN AUTOMATION CONTROL FILE *file*

Explanation: When the automation control file was processed only comments, blank lines and INCLUDE statements were found. There was no system automation control information.

System action: Processing stops.

Operator response: Contact your system programmer.

System programmer response: Correct the automation control file.

Classes: None.

AOF787A TERMINATING ERROR WHEN PROCESSING CONTROL FILE *file*

Explanation: Input from automation control file file contained an error that prevented any system automation control information from being processed.

System action: Processing stops.

Operator response: Correct the command or contact your system programmer.

System programmer response: Correct the automation control file or command error as identified by other diagnostic messages.

Classes: None.

AOF789I COMMAND LINE PARAMETER *parm* INVALID

Explanation: Command line parameter *parm* is not valid for the command issued.

System action: Processing is terminated.

Operator response: Reissue the command or contact your system programmer.

System programmer response: Correct the command.

Classes: None.

AOF790I THRESHOLD INTERVALS MUST BE: CRITICAL < FREQUENT < INFREQUENT

Explanation: For threshold checking to function correctly, the error intervals (time interval divided by error count) must be organized correctly. The error interval for the critical threshold should be a smaller interval than that for the frequent threshold, which should be smaller than the infrequent threshold.

System action: The threshold values are not changed.

Operator response: Issue the INGTRES command again and enter count and time data so that the intervals are correct.

System programmer response: None.

Classes: None.

AOF791I AUTOMATION CONTROL FILE PROCESSING COMPLETED WITH ERRORS

Explanation: The specified automation control file was processed and control data were obtained but some errors were encountered during processing

System action: Processing continues.

Operator response: Contact your system programmer.

System programmer response: Correct the errors identified by other diagnostic messages.

Classes: None.

AOF797A YOUR INSTALLATION POLICY HAS STOPPED SYS-OPS INITIALIZATION BECAUSE OF ERRORS

Explanation: During a load or reload of the automation control file errors were detected. Your installation has set the automation global variable AOFACFINIT to stop initialization.

System action: Initialization processing stops.

Operator response: Notify your system programmer.

System programmer response: Correct the errors to do with the automation control file and reload it.

Classes: 40 46.

**AOF800I FOR IBM SERVICE USE ONLY:
MODULE: *module_name* RC: *return_code***

Explanation: This message is intended for your IBM Support Center to use in debugging internal system automation errors.

The variable *module_name* shows the module that issued the message.

The variable *return_code* shows the return code from the module.

System action: None.

Operator response: None.

System programmer response: Contact IBM Support Center.

Classes: 0.

AOF801E REMOTE PROCESSING ENCOUNTERED AN INTERNAL ERROR UNEXPECTED RESULTS MAY OCCUR AS PROCESSING CONTINUES.

Explanation: Processing will continue but the results will be unpredictable.

System action: None.

Operator response: Notify your system programmer.

System programmer response: Contact IBM Support Center.

Classes: 40.

AOF802I INTERNAL ERROR DETECTED IN *clist*, ERROR DATA:

Explanation: An internal error occurred in the specified CLIST.

The variable *clist* shows the name of the CLIST.

System action: The requested function is not performed.

Operator response: Contact your system programmer.

System programmer response: Contact your IBM Support Center.

Classes: None.

AOF803I *data*

Explanation: This message is part of a multiline message that begins with message AOF802I.

The variable *data* contains error data specific to the failed CLIST.

System action: None.

Operator response: None.

System programmer response: Contact your IBM Support Center.

Classes: None.

AOF816I EXTENDED MCS CONSOLE *console* FOR TASK *task* HAS BEEN OBTAINED

Explanation: The automation obtained an extended MCS console for the named task because the default extended MCS console is already in use.

The variable *console* shows the name of the extended MCS console obtained for the task.

The variable *task* shows the name of the task.

System action: None.

Operator response: If an MVS command was issued before receiving this message, the command should be reissued.

System programmer response: None.

Classes: None.

AOF817I UNABLE TO OBTAIN AN EXTENDED MCS CONSOLE *console* FOR TASK *task* *reason*

Explanation: The automation was unable to obtain an extended MCS console for the named task.

The variable *console* shows the name of the extended MCS console obtained for the task.

The variable *task* shows the name of the task.

The variable *reason* explains why the command AOCGETCN failed.

System action: None.

Operator response: Notify your system programmer.

System programmer response: If the console is already in use, update the common global variable AOFCNMASK to define a unique console name.

Classes: None.

AOF818I **RODM RETURN CODE** *return_code* ,
REASON CODE *reason_code*

Explanation: A call to RODM has resulted in an unexpected return code.

The variable *return_code* shows the return code.

The variable *reason_code* shows the reason code.

System action: None.

Operator response: None.

System programmer response: Examine the return and reason codes in association with other messages displayed.

Classes: 40.

AOF821I **DSIDKS FAILED FOR** *reason*
PROCESSING, INTERNAL ERROR

Explanation: The NetView DSIDKS macro has returned a non-zero return code.

The variable *reason* shows the reason for the failure. This is either CONNECT, READ, or DISCONNECT.

System action: The request terminates.

Operator response: None.

System programmer response: Run the NetView trace. Retry the request.

Classes: None.

AOF824I **NEVER SHOULD OCCUR, INVALID DATA, CONTACT IBM SERVICE WITH DIAGNOSTIC DUMP**

Explanation: Invalid data has been detected.

System action: Produces a diagnostic dump.

Operator response: Notify your system programmer.

System programmer response: Contact your IBM Support Center.

Classes: None.

AOF826I **M** *dev,volser*, **RECEIVED ON** *mm/dd/yy*
AT *hh:mm:ss* **COMPLETED ON** *mm/dd/yy*
AT *hh:mm:ss*

Explanation: This message is created by tape mount monitoring. It is generated by a delete operator message (DOM) for an outstanding tape mount. The

NetView automation table should be configured to trap this message and initiate the deletion of the associated tape mount icon on the workstation.

The variable *dev* shows the device number that a tape mount request for was satisfied or canceled.

The variable *volser* shows the volume serial that a tape mount request for was satisfied or canceled.

The variables *mm/dd/yy* and *hh:mm:ss* show date and time respectively.

System action: The workstation tape mount icon for the nominated device and volume is deleted.

Operator response: None.

System programmer response: None.

Classes: None.

AOF830I **A SPECIFIED LIST ITEM WAS NOT FOUND, AND WAS REMOVED FROM THE LIST**

Explanation: A search item that was specified was not found and has been removed from the search list.

System action: Either re-specify search criteria or leave as is.

Operator response: None.

System programmer response: None.

Classes: None.

AOF831I **THE WILDCARD CHARACTER "*" VOIDS ALL OTHER ITEMS IN THE LIST**

Explanation: The "*" is a global wildcard and overrides all other search criteria (if any).

System action: None.

Operator response: If you do not want to override all other search criteria respecify search criteria.

System programmer response: None.

Classes: None.

AOF832I **A DUPLICATE ITEM WAS SPECIFIED IN THE LIST AND HAS BEEN REMOVED**

Explanation: The search criteria had duplicate items or two items were specified and were reduced to a single item, for example, JES and JES2 are reduced to JES.

System action: Either re-specify search criteria or leave as is.

Operator response: None.

System programmer response: None.

Classes: None.

AOF833I AN INCORRECT STATUS WAS SPECIFIED AND HAS BEEN REMOVED

Explanation: A status was specified that is not known to system automation.

System action: Respecify status search criteria.

Operator response: None.

System programmer response: None.

Classes: None.

AOF834I INCORRECT NESTING LEVEL SPECIFIED, IT MUST BE A NUMBER OR AN "*"

Explanation: An incorrect nesting level was specified. The nesting level should be re-specified as either a number (0-9) or an "*".

System action: Re-specify status search criteria.

Operator response: None.

System programmer response: None.

Classes: None.

AOF835I *clist_name*: ENTRY SPECIFICATION: *subsystem_name* LENGTH EXCEEDS *length*, PLEASE RE-ENTER

Explanation: An entry name has more than the maximum number of characters allowed. Reenter the entry name.

The variable *clist_name* shows the name of the command list that generated this message.

The variable *subsystem_name* shows the offending entry name.

The variable *length* shows the maximum permissible length of the entry name.

System action: None.

Operator response: Reenter entry specification.

System programmer response: None.

Classes: None.

AOF836I *clist_name*: TYPE SPECIFICATION: *subsystem_name* LENGTH EXCEEDS *length*, PLEASE RE-ENTER

Explanation: A type name has more than the maximum number of characters allowed. Reenter the type name.

The variable *clist_name* shows the name of the command list that generated this message.

The variable *subsystem_name* shows the offending type name.

The variable *length* shows the maximum permissible length of the type name.

System action: None.

Operator response: Reenter type specification.

System programmer response: None.

Classes: None.

AOF837I A COMMAND WAS ENTERED WITH A JES2 PREFIX *character* THAT IS NOT DEFINED IN THE AUTOMATION CONTROL FILE

Explanation: The command that was entered began with a character that is defined as a command synonym for the AOC/MVS JES2 command. However, it has not been defined in an automation control file entry for JES2. The variable *characters* shows the prefix.

System action: The command is not processed.

Operator response: Reenter the command with the correct prefix.

System programmer response: If the character is not intended as a JES2 prefix removed it from the command synonym member (usually AOF837I). Else, define it in the customization dialog - Application Definition panels for JES2.

Classes: None.

AOF838A *subsystem type* RECOVERY COMMANDS EXHAUSTED

Explanation: While attempting to perform JES3 recovery, system automation ran out of defined recovery commands before recovery was complete. System automation will re-execute the recovery command sequence starting from the first pass.

The variable *subsystem* indicates the name of the subsystem.

The variable *type* shows the type of JES3 recovery being performed. Possible values are:

SPOOLSHORT for SPOOL

MDSCOUNTA for ALLOCATION Queue

MDSCOUNTB for BREAKDOWN Queue

MDSCOUNTE for ERROR Queue

MDSCOUNTF for FETCH Queue

MDSCOUNTQ for NUMBER OF JOBS

MDSCOUNTR for RESTART Queue

MDSCOUNTU for UNAVAILABLE Queue

MDSCOUNTV for VERIFY Queue

MDSCOUNTW for WAIT VOLUME Queue

MDSCOUNTSS for SYSTEM SELECT Queue

AOF839I • AOF850I

MDSCOUNTSV for SYSTEM VERIFY Queue or any user-defined type.

System action: JES3 recovery continues.

Operator response: Look at SPOOL or MDS usage. System automation may need some assistance to keep it in control.

System programmer response: Review your JES3 recovery commands.

Classes: 40 44

AOF839I CONTROL FILE SYNTAX ERROR THRESHOLD REACHED. CONSOLE REPORTING OF SYNTAX ERRORS SUSPENDED - PROCEEDING CONTINUES.

Explanation: Syntax errors that were detected when processing this automation control file have reached the reporting threshold. Subsequently detected syntax errors in this automation control file will be recorded in the netlog, but will not be reported on the console.

System action: Processing of the automation control file continues.

Operator response: Check the netlog to see automation control file syntax error messages.

System programmer response: None.

Classes: None.

AOF840E ERROR EXTRACTING RODM DATA FOR THE *type* OBJECT. RODM RETURN CODE *retcode*. RODM REASON CODE *reason*.

Explanation: The command cannot be executed because the RODM look-up failed and vital information could not be retrieved.

System action: The command is not executed.

The variable *type* is the RODM object type.

The variable *retcode* is the RODM return code.

The variable *reason* is the RODM reason code.

Operator response: Notify your system programmer.

System programmer response: The *retcode* and *reason* are the RODM return and reason codes, which are documented in the *NetView RODM Programming Guide*.

Classes: None.

AOF841E THE RODM NAME OR RODM USER ID COULD NOT BE DETERMINED.

Explanation: The command cannot be executed because the RODM name or user ID used to access RODM could not be determined.

System action: The command is not executed.

Operator response: Make sure that system automation focal point has initialized. If the focal point is running a satellite system automation, make sure the satellite has initialized.

System programmer response: None.

Classes: None.

AOF845I COMMAND REJECTED *command* COMMAND IS TOO LONG

Explanation: The command exceeds the maximum length of commands. The command cannot be executed.

The variable *command* shows the name of the command that could not be executed.

System action: None.

Operator response: Try to simplify the command and make it shorter.

System programmer response: None.

Classes: None.

**AOF850I ISSUED INGSEND
RESP=*resp*,OPER=*oper*,TO=*domain*,
CRM=*crm*,CORRWAIT=*corr*,
CMD="*command*"**

Explanation: This is to confirm the complete INGSEND command routed to the target system. This message is logged to the local and target netlog. This message is issued only by INGSEND when CRM=RMTCMD.

The variable *onet.odom.oopr* shows the origin network ID, domain ID and operator ID.

The variable *resp* shows what to do with the responses, if any.

The variable *oper* shows the target RMTCMD autotask ID.

The variable *domain* shows the target domain ID.

The variable *crm* shows the command routing mechanism used.

The variable *corr* shows the value used on the NetView Pipe command Corrwait stage.

The variable *command* shows the command to be executed.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

AOF851I **INVALID SYSPLEX DEFINITION -**
'data'. ENTRY IGNORED

Explanation: You have something coded on the ENVIRON SYSPLEX definition in your automation control file that system automation does not understand.

The variable *data* shows what is not being recognized.

System action: The entry is ignored. If the entry was for something important (such as the group name) this may have other consequences.

Operator response: Inform your system programmer.

System programmer response: If the automation control file was built with the customization dialogs, contact your IBM Support Center.

Classes: None.

AOF852I **SYSPLEX GROUP ENTRY MISSING -**
SYSPLEX WILL NOT BE SUPPORTED

Explanation:

System action:

Operator response:

System programmer response:

Classes: 40 46

AOF860I *instruction* **FOR** *entry type* **REJECTED -**
"reason"

Explanation: An automation policy processing instruction could not be completed, and was rejected for the reason given.

Note: ASSERT, MODIFY, and RETRACT are instructions used by the system automation automation policy processing to add, change, or delete data in the currently active policy definitions. The system automation policy has converted your request into one or more of these instructions.

The variable *instruction* shows the action being attempted for the entry-type pair. It can be ASSERT, MODIFY, or RETRACT.

The variable *entry* shows the entry field associated with the line of automation policy data being processed.

The variable *type* shows the type field associated with the line of automation policy data being processed.

The variable *reason* shows the reason why the action on the entry-type pair was rejected.

System action: The entry-type pair and associated data is ignored.

Operator response: Fix the problem if you can, using the information given in the message, or inform your system programmer.

System programmer response: Fix the problem using the information given in the message.

Classes: 40 46.

AOF867I **WARNING:** *entry type* - "*warning*"

Explanation: System automation has detected an inconsistency in the data that it has been given.

The variable *entry* shows the entry field associated with the line of automation policy data being processed.

The variable *type* shows the type field associated with the line of automation policy data being processed.

The variable *warning* shows the potential problem that has been found with the data.

System action: If the error is an incorrect data value it will be ignored, and it will not be reflected in the runtime parameters.

Operator response: Inform your system programmer.

System programmer response: Resolve the data inconsistency.

Classes: 40 46.

AOF868I **WARNING: THERE ARE NO JES**
SUBSYSTEMS DEFINED

Explanation: The automation cannot work correctly without a JES subsystem, and system automation did not define one.

System action: Processing continues. Some automation functions may not work properly.

Operator response: Inform your system programmer.

System programmer response: Determine whether JES was defined outside of system automation. If not, and you want your system to run with JES, you must define it.

Classes: 40 46.

AOF869I *nn* **ERRORS ENCOUNTERED**
PROCESSING MEMBER *member*

Explanation: This message indicates how many inconsistencies were found by system automation during the processing of this automation control file.

The variable *nn* shows the number of inconsistencies encountered during the checking of *member*.

The variable *member* shows the name of the automation control file member being processed.

| **System action:** Processing continues unless the advanced automation option AOFACFINIT is set to 0.

| **Operator response:** Review the NetView log for additional information given in one or more of the following messages: AOF206I, AOF701A, AOF784I, AOF785I, AOF860I, AOF867I, AOF868I, AOF871I, and fix the problem if you can, or inform your system programmer.

| **System programmer response:** Review the NetView log for additional information given in one or more of the following messages: AOF206I, AOF701A, AOF784I, AOF785I, AOF860I, AOF867I, AOF868I, AOF871I and fix the problem.

| **Classes:** 40.

AOF871I *time*: LESS THAN *n type entry* ARE DEFINED - AUTOMATION WILL CONTINUE WITH DEGRADED PERFORMANCE

Explanation: There are not enough definitions available in the automation control file for the entry type mentioned in the message.

The variable *time* shows the time when this message was generated.

The variable *type* shows the type specified in the automation control file.

The variable *entry* shows the entry specified in the automation control file.

System action: Processing continues with degraded performance.

Operator response: None.

System programmer response: Define the automation control file entries through the system automation customization dialogs.

Classes: None.

AOF890I THE COMMAND WILL BE ROUTED TO *domain_ID* USING THE DEFAULT OPERATOR ID *operator_ID*

Explanation: The command was issued in the form of a route command by operator *operator_ID* and executed in domain *domain_ID*.

The variable *domain_ID* shows the of the domain that the command was executed in.

The variable *operator_ID* shows the ID of the operator who issued the route command.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

AOF891I SSV ERROR OBTAINING SYSPLEX *item*

Explanation: This message indicates a non-zero return code has been returned from NetView when SA z/OS was trying to query a NetView CGLOBAL value. This is likely to have been caused by NetView and RODM being out of synchronization.

The variable *item* shows what SA z/OS was trying to resolve when the error occurred.

System action: The query is rejected by NetView and the SA z/OS request abends.

Operator response: None.

System programmer response: None.

Classes: None.

Classes:

AOF893I ENTERPRISE MONITORING IS NOT ENABLED. *reason*

Explanation: Enterprise monitoring is not enabled.

The variable *reason* gives more detail about the cause of the error.

System action: None.

Operator response: If you do not want to use enterprise monitoring then no action is required. Otherwise, contact your system programmer.

System programmer response: Use the reason text to help determining the problem.

Classes: None.

AOF900I AT LEAST ONE STRUCTURE IS STILL ALLOCATED IN THE CF *cfname*

Explanation: The process was started disconnecting the coupling facility *cfname* from its connected systems. However, at least one structure that is still allocated was detected by the DRAIN function. Or, one allocated structure that has a connection to an application was detected by the FORCE function.

System action: Processing terminates.

Operator response: Check the allocated structures of the coupling facility. Rebuild the structures to another coupling facility if possible before issuing the command again.

System programmer response: None.

Classes: 40, 43.

Modules: INGRX912, INGRX913.

AOF901I **PATH *chpid* FROM SYSTEM TO CF *cf_name* COULD NOT BE SET TO *status***

Explanation: The program received an unexpected message in response to the CF CHP command setting a sender path from a connected system to a coupling facility to either OFFLINE or ONLINE.

System action: Processing terminates.

Operator response: None.

System programmer response: Check the netlog for related messages to determine the cause of the error.

Classes: 40 43.

Modules: INGRX913, INGRX915.

AOF902I **DRAINING THE CF *cf_name* DID NOT COMPLETE SUCCESSFULLY**

Explanation: The program received an unexpected message in response to the CF CHP command setting a sender path from a connected system to a coupling facility to OFFLINE.

System action: Processing terminates.

Operator response: None.

System programmer response: Check the netlog for related messages to determine the cause of the error.

Classes: 40 43.

Modules: INGRX913.

AOF903I **EITHER THE SPECIFIED CF IS NOT DEFINED OR NO CF HAS BEEN DEFINED**

Explanation: Either the command was executed in a sysplex that does not use any coupling facility or the name of the coupling facility is not known in the sysplex.

System action: Processing terminates.

Operator response: Specify a valid coupling facility name.

System programmer response: None.

Modules: AOFRVX2A, AOFRVX22, AOFRX210, INGRX900, INGRX901.

AOF904I **A CF/CDS PROCESS IS STILL RUNNING. ACTION *action name* IS REJECTED**

Explanation: While the AUTXCF task is busy, new actions are rejected. This avoids performance degradation due to multiple rebuild processes, or unpredictable results due to multiple executions of an action.

The variable *action* shows the CF or CDS action that is being performed.

The variable *name* shows the name of the CF or the CDS.

System action: Processing terminates.

Operator response: Retry the action later.

System programmer response: None.

Modules: INGRVX3D, INGRVX90.

AOF905I ***action* OF STRUCTURE/CF *strname* FAILED**

Explanation: The program received an unexpected message in response to a SETXCF command. The action that is related to the structure or coupling facility did not complete successfully.

System action: Processing stops except when all the structures of a coupling facility are being rebuilt one by one. In this case processing continues with the next structure.

Operator response: None.

System programmer response: Check the netlog for related messages to determine the cause of the error.

Classes: 40, 43.

AOF906I **TIME-OUT OCCURRED ON COMMAND *command***

Explanation: The command did not respond in time. Reasons could be:

- XCF commands:
These can time out because they are processed by one single task.
- Commands that implement the Health Checker function:
These are INGRX352 and INGRVX35. They can time out because there is a problem with task AOFHC, for example, the task may not be active.

System action: Processing terminates except when all structures of a coupling facility are being rebuilt. In this case processing continues with the next structure.

Operator response: Before executing the action again, check the netlog and the system activities to determine the cause of the time-out.

System programmer response: None.

Classes: 40, 43.

AOF907I **STRUCTURE *strname* HAS CONNECTIONS OTHER THAN FAILED-PERSISTENT**

Explanation: The automation detected at least one connection to the structure that does not have the

AOF908I • AOF913I

status FAILED-PERSISTENT. Because XCF does not allow for the deletion of such a structure, the action is rejected.

System action: Processing terminates.

Operator response: Make sure that the structure has either no connections, or only FAILED-PERSISTENT connections before you force the deletion of the structure again.

System programmer response: None.

Classes: 40 43.

Modules: INGRX911.

AOF908I START OF AUTOTASK *taskname* FAILED. RC = *rc*

Explanation: Any action that is related to a sysplex resource, for example, a coupling facility, is performed on a dedicated autotask. If the task is not active, it is started via the AUTOTASK command. However, the command failed with the return code *rc*.

System action: Processing terminates.

Operator response: Inform your system programmer.

System programmer response: Analyze the return code and correct the error. Refer to the description of the AUTOTASK command help in the online help (HELP AUTOTASK).

Classes: 40 44.

Module: INGRVX90

AOF909I FUNCTION *function* REJECTED DUE TO A CF STATUS CHANGE

Explanation: You requested the function *function* to be performed. Meanwhile the status of the affected coupling facility has changed. The function is rejected because the current status of the coupling facility does not match the status required by the function.

System action: The program terminates.

Operator response: Press the refresh key to display the latest status.

System programmer response: None.

Classes: None.

Module: INGRVX90

AOF910I POLICY *policy* COULD NOT BE ACTIVATED

Explanation: During verification of the policy *policy* the automation detected that the policy does no longer fulfill the criteria to become active. The automation rejected the command to make the policy the active policy.

System action: Processing terminates.

Operator response: Refresh the screen. Then retry the operation.

System programmer response: None.

Classes: None.

Module: INGRVX90

AOF911I POLICY CHANGE(S) PENDING

Explanation: After the new policy has become active, the automation detected that one or more structures have a policy change pending.

System action: The command completed successfully.

Operator response: Use the command INGCF STR to find out what structures have a pending status. Rebuild each of these structures to remove its pending status.

System programmer response: None.

Classes: None.

Module: INGRVX90

AOF912I XCF COMMAND '*command*' IS NOT SUPPORTED BY THIS SYSTEM.

Explanation: The designated system does not support the indicated XCF command. START/STOP MAINTMODE requires z/OS version 1.9 or later.

System action: Processing stops.

Operator response: Reenter the command on a system that supports the command.

System programmer response: None

Classes: None.

AOF913I *itemname* VALUE OF CDS TYPE *type* HAS BEEN CHANGED FROM *oldvalue* TO *newvalue*

Explanation: The automation detected a situation where the size of a couple data set must be increased to prevent an outage situation. The value of the item *itemname* has been increased as stated.

System action: Processing continues.

Operator response: None.

System programmer response: Determine the reason for the change and take action accordingly.

Classes: 40, 43.

Module: INGRX711

**AOF914E RECOVERY OF THE CDS TYPE *type*
FAILED. RSN= *reason***

Explanation: The recovery of the couple data set of type *type* failed.

Reason Explanation

- | | |
|----|--|
| 10 | The command 'SETXCF COUPLE,PSWITCH' failed. |
| 19 | The LOGR couple data set(s) could not be extended. |
| 20 | The control card information of the primary couple data set could not be obtained. |
| 21 | A temporary couple data set with the new item values could not be formatted. |
| 22 | The temporary couple data set could not be allocated. |
| 23 | The old alternate couple data set could not be deleted. |
| 24 | The temporary couple data set could not be made the primary couple data set. |
| 25 | The old primary couple data set could not be deleted. |
| 26 | The old primary couple data set could not be reformatted. |
| 27 | The reformatted old primary couple data set could not be allocated as the new alternate couple data set. |
| 28 | The reformatted old primary couple data set could not be made the new primary couple data set. |
| 29 | The temporary couple data set could not be deleted. |
| 2A | The old alternate couple data set could not be reformatted. |
| 2B | Either the reformatted old alternate couple data set or a new spare couple data set could not be allocated as the alternate couple data set. |

System action: Processing terminates.

Operator response: Inform your system programmer.

System programmer response: Check the netlog and the system log for related system messages. Determine the reason and take action accordingly.

Reason Explanation

- | | |
|----|--|
| 10 | Check the system log to determine the reason why XCF did not switch the couple data sets. If the message was issued while extending the couple data sets, follow the comment for reason codes 20 through 29. |
|----|--|

19 Follow the description of reason codes 20 through 2B.

20-2B These codes relate to the process of extending the current couple data set(s) of type *type*. The codes reflect the progress of the process. Determine the current couple data sets and complete the process manually. Depending on the progress, you can determine the appropriate control card information either using the command INGPLEX CDS TYPE=*type* (reason > 24) or running the XCF utility against the temporary couple data set (reason > 21). If reason code 21 is issued, check the system log for an IXG257I or IXG261E message. If you find them, extend the couple data set(s) by increasing the DSEXTENT value by 1 or when not available by setting the value to 1. Otherwise and for reason code 20 determine the originator of the request to get the new item values.

Classes: 40, 43.

Module: INGRX305, INGRX711, INGRX720

**AOF915I NO SPARE VOLUMES DEFINED FOR
CDS TYPE *type*.**

Explanation: The recovery of couple data sets of type *type* has been triggered. However, the automation could not complete the recovery because no spare volumes have been defined where a couple data set could be allocated.

System action: Processing terminates.

Operator response: Inform your system programmer.

System programmer response: Review the netlog and the system log. If the message has not been issued during the initialization of the automation, complete the recovery manually. To prevent any future occurrence of this message, modify the customization by adding one or more spare volumes for the named type of couple data sets. Then recycle NetView.

Classes: 40, 43.

Module: INGRX720

**AOF916E NO MORE SPARE VOLUMES
AVAILABLE FOR CDS *type type*.**

Explanation: The recovery of couple data sets of type *type* has been triggered. However, the automation could not complete the recovery because all predefined spare volumes are either already in use, or they are unusable.

System action: Processing terminates.

Operator response: Inform your system programmer.

System programmer response: Review the netlog and the system log and complete the recovery manually.

Then check the predefined spare volumes of the named type of couple data sets. When less than three volumes have been defined and the corresponding couple data sets are allocated to XCF add at least one more spare volume and then recycle NetView. Otherwise, determine the reason why the spare couple data sets or volumes could not be used. One reason can be that there is not enough continuous space on the volume.

Classes: 40, 43.

Module: INGRX720

AOF917E THE HIGH-LEVEL QUALIFIER HAS NOT BEEN DEFINED.

Explanation: The recovery of couple data sets has been triggered. However, the automation could not complete the recovery because the high-level qualifier that was used to create a spare couple data set has not been defined.

System action: Processing terminates.

Operator response: Inform your system programmer.

System programmer response: Review the netlog and the system log and complete the recovery manually. To prevent any future occurrence of this message, disable the recovery for couple data sets. Or, modify the customization by defining the high-level qualifier and then recycle NetView.

Classes: 40, 43.

Module: INGRX720

AOF918I THE AUTOMATION IS GOING TO SWITCH THE COUPLE DATA SETS OF *type*

Explanation: The automation is going to switch the couple data sets of the specified type.

The variable *type* shows the type of the couple data sets that are being switched.

System action: Processing continues.

Operator response: None.

Classes: 40, 43.

Module: INGRX720

AOF919I *function* DETECTED STATUS *status* FOR CF *cfname*

Explanation: The function *function* detected a status change of the coupling facility inhibiting the function from further processing.

System action: Processing terminates.

Operator response: Press the REFRESH key and retry the function if possible.

System programmer response: None.

Classes: None.

Module: INGRX914

AOF920I NO SIGNALLING PATHS MATCH THE SPECIFIED CRITERIA *criteria*

Explanation: The program set up the commands MVS DISPLAY XCF,PATHIN,DEVICE=ALL and MVS DISPLAY XCF, PATHOUT,DEVICE=ALL. However, the message AOF920I was returned.

The variable *criteria* shows the path type (either PATHIN or PATHOUT).

System action: Processing terminates.

Operator response: None.

System programmer response: None.

AOF921I CONSOLE *console* IS INACTIVE

Explanation: The MVS command D R,LIST,T,CN=*console* was issued. However, no data was returned because the console is INACTIVE. Therefore, no requests can be displayed.

System action: Processing terminates.

Operator response: None.

System programmer response: None.

Classes: None.

Module: AOFRVX41

AOF922I REBUILD OR FORCE OF STRUCTURE *structure_name* NOT ALLOWED

Explanation: An attempt was made to REBUILD or FORCE a coupling facility structure. However, the REBUILD or FORCE command is not supported for this structure. This can have one of the following reasons:

- The structure has no connections allocated and does not support system-managed rebuild.
- The structure has no active connections allocated and does not support system-managed rebuild.
- One or more of the structure's active connections was created with REBUILD=NO and system-managed rebuild is not supported.
- No alternate coupling facility is defined, available or has sufficient space.

The variable *structure_name* shows the name of the structure that the REBUILD or FORCE command is not supported for.

System action: Processing terminates.

Operator response: None.

System programmer response: None.

**AOF923I NO ACTIVE TARGET SYSTEM
FOUND ON TARGET HARDWARE**
target (LU=LUNAME)

Explanation: The operator issued the command GETSPCFP *luname*. Either the target hardware has not been initialized yet, or the LU name of the Support Element has changed.

System action: Processing terminates.

Operator response: Use the processor operations command ISQXDST to verify that at least one target system is initialized. If not, issue the command ISQXIII *target* where *target* is one of the target system names in the list. When the command completes successfully, retry the GETSPCFP command.

System programmer response: If the luname has changed, correct the name in the processor operations control file and restart processor operations.

Classes: 40, 43.

**AOF924A AUTOMATION OF MESSAGE *ixc_msg*
TERMINATED. REASON: *reason*[,
{PROCESSOR OPERATIONS | BCPII}
COMMAND RC=*rc*]. SYSID: *sysname*.**

Explanation: The variable *ixc_msg* shows the message that triggered the automation. It can have the following values and explanations:

- | | |
|----------------|--|
| IXC102A | XCF is removing a system from the sysplex. |
| IXC402D | XCF determined that a system in the sysplex appears to be inoperative. |

XCF waits for the reply to proceed. However, the reply could not be automated.

The variable *reason* shows the reason code that was issued. It can have the following values:

- 10** The message being automated has not been formatted as expected. The message identifier could not be located.
- 11** A command was issued to the MVS system that reported that a system left the sysplex. A timeout occurred while waiting for a reply to this command.
- 12** The proxy resource name for the system leaving the sysplex could not be determined.
- 13** The Support Element of the system leaving the sysplex cannot be reached.
- 14** The target system name of the ISQ900I message could not be obtained or the target system that sent the ISQ900I message is not initialized to processor operations.
- 15** The replyid for the message being automated could not be determined.
- 16** The system name could not be located in the automated message.

- 17** The message that triggered the automation could not be retrieved for automation.
- 18** The automation requirements for the system leaving the sysplex could not be determined.
- 19** An error occurred while checking if the message was still outstanding.
- 30** A processor operations command failed. Refer to the appropriate command description.
- 31** A timeout occurred while waiting for the response of the Processor Management command.
- 32** The reply to the outstanding WTOR could not be sent.
- 33** An error occurred while determining the status of the local sysplex.

The variable *rc* shows the value of the return code.

System action: Processing terminates.

Operator response: Depending on which message triggered the automation, respond as follows:

- | | |
|----------------|--|
| IXC102A | Complete the shutdown of the system leaving the sysplex. Then reply DOWN to the outstanding WTOR. |
| IXC402D | Either reply 'INTERVAL= <i>sssss</i> ' (range 0 to 86400) to give the system the specified interval to become operative again. Or, complete the shutdown of the system leaving the sysplex. Then reply DOWN to the outstanding WTOR. |

System programmer response: Correct the problem. If reason code 11 was issued, no action is required.

Module: AOFRX700, AOFRX701, AOFRX702, INGRX705, INGRX706

Classes: 40, 43

TEC: YES

**AOF925I AUTOMATION OF MESSAGE *ixc_msg*
FOR *system* COMPLETED
SUCCESSFULLY**

Explanation: The automation of message IXC102A or IXC402D for system *system* completed successfully. The system is no longer part of the sysplex.

System action: None.

Operator response: None.

System programmer response: None.

Module: AOFRX700, AOFRX701, AOFRX702, INGRX705, INGRX706.

Classes: None.

TEC: NO

AOF926I **ERROR** *error* **DETECTED DURING
AUTOMATION OF MESSAGE** *ixc_msg*

Explanation: The routine responsible for the automation of messages IXC102A and IXC402D found an error.

The variable *error* shows the error condition. The following error conditions can occur:

- 10 The message is neither the IXC102A nor the IXC402D message.
- 11 The same message is being processed by another task.
- 12 Two or more commands were defined in the customization dialogs for message IXC102A. However, only one ISQCCMD can be issued.
- 13 The command defined for IXC102A message is not an ISQCCMD command.
- 14 The reply is no longer outstanding.
- 15 Incorrect call of a subsequent clist.
- 30 Using the supplied or default command, an attempt was made to deactivate the system leaving the sysplex. Another attempt will be made using the default command.

System action: Processing terminates for conditions 10, 11, 14, and 15. For conditions 12, 13 and 30 processing continues using the default processor management command SYSRESET CLEAR.

Operator response: None.

System programmer response: If error conditions 12, 13 or 30 occurred, correct the definitions and reload the automation control file.

Module: AOFRX700, AOFRX701, AOFRX702, INGRX705, INGRX706.

Classes: None.

TEC: NO

AOF927E **WTOR|WTOR BUFFER SHORTAGE
COULD NOT BE RESOLVED**

Explanation: The sysplex automation to recover from a WTO or WTOR buffer shortage did not complete successfully. One or multiple recovery stages signalled an error. This is a transient status of the recovery. The message IEA406I/IEA232I for WTO/WTOR buffer shortage relief may indicate that the buffer shortage has been resolved.

System action: Processing continues.

Operator response: Check the system log for message IEA406I/IEA232I indicating a buffer shortage relief. If the message could not be found, perform manual recovery actions.

System programmer response: Search the system log for message IEE889I issued by the operating system internally. Identify the buffering consoles or jobs. If possible, requeue console buffers to the system log (K Q,L=*conname*). If possible, increase MLIM. Cancel jobs

buffering messages or replies. After message IEA406I/IEA232I is issued, check whether the current CONSOLxx definitions may need to be adjusted to avoid future buffer shortages.

Module: INGRX730

Classes: 40, 43.

AOF928I **SYSPLEX** *sn bt* **RECOVERY INCREASES
THE** *in* **VALUE TO** *nv*

Explanation: The automation has detected a shortage of message or reply buffers. As a recovery result, the limit value is increased to avoid further buffer shortages. The changed value affects all active members in a sysplex, because the modified limit has sysplex scope.

The variable *sn* specifies the name of the sysplex.

The variable *bt* WTOR: write to operator with reply message buffers.

The variable *in* RMAX specifies the maximum buffer limit of write to operator with reply messages. If the system runs in SYSPLEX=LOCAL mode or the number of systems in the sysplex is greater than eight, the RMAX value is set to 9999. Otherwise it is not changed.

The variable *nv* shows a numeric number, representing the new limit value that is going to be active.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: This message may indicate that the current RMAX value of CONSOLxx member needs to be adjusted.

Classes: 40, 43.

Module: INGRX730

AOF929I **SYSPLEX** *sn bt* **RECOVERY SETS** *st* **ON
SYSTEM** *sys*

Explanation: The automation has detected a shortage of message or reply buffers. As a recovery result, a limit value or a console setting is changed to avoid further buffer shortages. The changed value or console setting affect the named system in the sysplex.

The variable *sn* specifies the name of the sysplex.

The variable *bt* WTOR: write to operator with reply message buffers.

The variable *st* shows the string representing the command that was issued to recover from the buffer shortage. The limit values or console settings stay in effect until they are changed manually. In case of an extended recovery situation, this message may be issued multiple times showing commands increasing MLIM or RLIM values.

The variable *sys* specifies the name of the system.

System action: Processing continues.

Operator response: If this message is displayed frequently, inform your system programmer.

System programmer response: Multiple occurrence of this message may indicate that the CONSOLxx member specified limits or console definitions need to be adjusted.

Classes: 40, 43.

Module: INGRX730

**AOF930I HARDWARE STATUS CHANGE
DETECTED**

Explanation: The hardware monitoring program detected a status change of one or more hardware components defined during customization.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

Module: INGRX051

**AOF931I VERIFICATION OF USER-DEFINED
CDS FAILED. RSN= *reason* DSN=*dsname***

Explanation:

The variable *reason* shows the value of the reason code. The following reason codes can occur:

- | | |
|----|---|
| 1 | The automation environment has not yet been initialized. |
| 2 | The data set name conflicts with the naming convention of the automation when creating a new couple data set. |
| 4 | The specified data set is not defined. |
| 8 | The specified data set is not available. |
| 12 | The specified data set is already allocated. |
| 16 | An unexpected error occurred. |
| 20 | The DYNALLOC macro failed. |
| 24 | The specified volume does not match the actual volume. |

The variable *dsname* shows the name of the data set.

System action: Processing terminates.

Operator response: Make sure that the name of the data set and volume (if specified) are correct.

System programmer response: Make sure that the data set is defined and available.

Classes: None.

Module: INGRX301

**AOF932I BOTH NAME AND VOLUME ARE
REQUIRED.**

Explanation: The user has only specified a user couple data set name or a volume.

System action: Processing stops.

Operator response: Specify both a data set name and a volume.

System programmer response: None.

Classes: None.

Module: INGRX301

AOF933I ACTION CODE IS NOT SUPPORTED.

Explanation: The action code is not supported for this entry for some program-related reason.

System action: Processing stops.

Operator response: See the panel help for the reason. Use a valid action code.

System programmer response: None.

Classes: None.

Module: INGRX300, INGRX904

AOF934I POLICY *policy* HAS BEEN STARTED.

Explanation: A 'SETXCF START,POLICY' command has been issued for the policy *policy*.

System action: Processing continues.

Operator response: Check that the policy becomes the active policy.

System programmer response: None.

Classes: None.

Module: INGRVX3A

**AOF935I ENTER THE COUPLE DATA SET AND
VOLUME.**

Explanation: The automation is waiting for a user-defined couple data set to be specified.

System action: None.

Operator response: Enter the required information.

System programmer response: None.

Classes: None.

Module: INGRX301

AOF936E **START OF POLICY** *policy* **FAILED.**

Explanation: A "SETXCF START,POLICY" command has been issued for the policy *policy*. The command failed.

System action: Processing stops.

Operator response: Review the netlog.

System programmer response: None.

Classes: None.

Module: INGRX301

AOF937E **SOME PENDING STRUCTURES CANNOT BE REBUILT AND REMAIN PENDING.**

Explanation: The automation function responsible for releasing the 'POLICY CHANGE PENDING' status of the structures detected that one or more structures remain in this status.

System action: Processing continues.

Operator response: Use the command 'D XCF,STR,STATUS=POLICYCHANGE' to list the structures whose status is pending. Determine the reason why these structures remain in the status. Manually perform the necessary actions for releasing the status.

System programmer response: None.

Classes: None.

Module: INGRX919

TEC: NO

AOF938E **DDNAME** *ddname* **IS PENDING.**

Explanation: An automation function could not complete successfully because a required data set is not available.

The variable *ddname* shows the name of the data set that is not available.

System action: Processing stops.

Operator response: Notify your system programmer.

System programmer response: Add the *ddname* to the NetView startup procedure. To prevent a NetView recycle, allocate the data set dynamically using the *ddname* above.

Classes: None.

Module: INGRVX20, INGRX200

TEC: NO

AOF939I **THERE IS NO ACTIVE POLICY.**

Explanation: The automation detected that no policy is active at the moment.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

Module: INGRX311, INGRX313, INGRX314

AOF940I *resource(s)* **AUTOMATION IS/ARE NOT ENABLED.**

Explanation: The selected function is not enabled for automation.

System action: Processing terminates.

Operator response: Enable the function by setting the RECOVERY flag of one of the named resources to 'Y'. Then, retry the function.

System programmer response: If the function should be enabled by default, update the customization values.

Classes: None.

AOF941I **CFRM COUPLE DATA SET IS NOT AVAILABLE.**

Explanation: The CFRM active policy could not be read because the couple data set supporting TYPE(CFRM) is not accessible from this system.

System action: Processing terminates.

Operator response: Issue the request from a system with access to a CFRM couple data set.

System programmer response: None.

Classes: None.

Module: INGRVX9Q

AOF942I **NO POLICIES DEFINED.**

Explanation: There are no policies defined for the selected couple data set type.

System action: None.

Operator response: Inform your system programmer.

System programmer response: Define policies as necessary.

Classes: None.

Module: INGRX311, INGRX313, INGRX314

AOF943I NO LOG STREAMS OR STRUCTURES DEFINED.

Explanation: There are no log streams or structures defined in the LOGR couple data set.

System action: None.

Operator response: Inform your system programmer.

System programmer response: Define log streams and structures as necessary.

Classes: None.

Module: INGRX312

AOF944I CDS TYPE *cdstype* IS NOT ENABLED ON SYSTEM *system*.

Explanation: The couple data set type cannot be displayed because it is not enabled on the target system.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

Module: INGRX310

AOF945I CDS TYPE *cdstype* IS NOT ACCESSIBLE ON SYSTEM *system*.

Explanation: The couple data set type cannot be displayed because it is not accessible on the target system.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

Module: INGRX310

AOF946I SOME SENDER PATH INFORMATION COULD NOT BE DETERMINED FOR *cfname*.

Explanation: Currently not all systems in the sysplex have access to the coupling facility *cfname*. However, the automation cannot determine the sender paths of some or all of those systems.

System action: Processing continues.

Operator response: If the local system does not show the control unit of the coupling facility, try the command from a system that shows the information when you enter the MVS command 'D CF,CFNM=*cfname*'.

System programmer response: None.

Classes: None.

Module: INGRVX9P

AOF947I *action* OF STRUCTURE *structure* IS NOT SUPPORTED.

Explanation: The automation detected that the action selected for the structure is either not supported or currently unavailable.

The variable *action* shows the action to be performed.

The variable *structure* shows the name of the structure.

In case of SYSTEM-MANAGED actions the cause of the problem could be that the CFRM couple data set is not formatted properly.

System action: The action is not performed.

Operator response: None.

System programmer response: None.

Classes: None.

Module: INGRX904

TEC: NO

AOF948I FUNCTION NO LONGER SUPPORTED.

Explanation: Due to some changes in a new version of this product this function has either been replaced or deleted.

System action: Processing terminates.

Operator response: If the function was replaced, use the new function.

System programmer response: None.

Classes: None.

Module: INGRVX90

AOF949I NO DUMP DATA SET AVAILABLE FOR JOB *jobname*

Explanation: The automation tries to take a dump before the job is canceled. Neither system dump data sets nor automatic allocated dump data sets were available.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: To fix the error do one of the following:

- Add more system dump data sets or volumes for automatically allocated dump data sets

AOF950I • AOF953E

- Clear full dump data sets to prevent further occurrences of this message.

Module: INGRX741

Classes: 40

TEC: NO

AOF950I **JOB** *jobname* **ON** *sysname* **HOLDS AN ENQ FOR** *nnn* **SECONDS ON RESOURCE** *resource*

Explanation: The automation found a long blocking ENQ on the resource held by the specified job. The policy defined during customization does not allow the cancelation of the job to free the resource.

The variable *jobname* shows the name of the job.

The variable *sysname* shows the system where the job is running.

The variable *nnn* shows the waiting time in seconds. A value of 999 is the maximum waiting time shown.

The variable *resource* shows the name of the resource where the ENQ was found.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: If the waiting time is feasible, remove the resource entry from the policy or increase the time if applicable. Otherwise allow the cancelation of the job if necessary.

Module: INGRX741

Classes: 40, 43.

TEC: NO

AOF951I **DUE TO A LONG ENQ DETECTION THE JOB** *jobname* **ON** *sysname* **IS BEING CANCELLED.**

Explanation: The automation detected a long blocking ENQ held by the specified job. The job is being canceled based on policy definitions.

The variable *jobname* shows the name of the job.

The variable *sysname* shows the system where the job is running.

System action: The job is canceled with the MVS CANCEL command.

Operator response: None.

System programmer response: None.

Module: INGRX741

Classes: 40

TEC: NO

AOF952I **DUE TO A AUXILIARY STORAGE SHORTAGE DETECTION THE JOB** *jobname* **IS BEING CANCELLED**

Explanation: The automation found an auxiliary storage shortage. The system issued a message indicating the job had rapidly increasing storage requests. The job is being canceled based on policy definitions.

The variable *jobname* shows the name of the job.

System action: The job is canceled with the MVS CANCEL command.

Operator response: None.

System programmer response: None.

Module: INGRX742

Classes: 40

TEC: NO

AOF953E **RECOVERY OF AUXILIARY STORAGE SHORTAGE COULD NOT COMPLETE SUCCESSFULLY. REASON=***reason*

Explanation: The automation could not recover an auxiliary storage shortage.

The variable *reason* shows the reason code. It can have the following values:

Reason	Description
--------	-------------

- | | |
|----|--|
| 4 | The recovery was enabled, but neither spare volumes for dynamically allocated page data sets nor preallocated page data sets were defined. |
| 8 | The spare volumes for dynamically allocated page data sets and the predefined local page data sets are used up. |
| 12 | The spare volumes for dynamically allocated page data sets and the predefined local page data sets are used up. Some of them have been marked unusable because the system rejected the data sets when the automation tried to make them available. |
| 16 | The IDCAMS service failed for the dynamic allocation of a page data set. Check the netlog for details. |
| 20 | The PAGTOTL value defined in IEASYSxx member used during IPL does not allow to add more local page data sets. |
| 24 | The system issued a message indicating a problem when the automation tried to make a local page data set available. |

System action: Processing terminates.

Operator response: Inform your system programmer.

System programmer response: If reason codes 4, 8, 12,

or 16 occurred, add more local page data sets manually. If reason code 20 occurred, follow the instructions given in messages IRA200E and IRA203I. For reason code 24 follow the instructions of the messages IEE78n issued before this message.

Module: INGRX742

Classes: 40, 43.

TEC: NO

AOF954I PAGE DATA SET *dsname* SUCCESSFULLY ADDED.

Explanation: Due to a recovery situation the automation successfully added a local page data set to the system.

The variable *dsname* shows the name of the data set.

System action: Processing continues.

Operator response: None.

System programmer response: Delete the page data set via the PAGEDEL command when the data set is no longer needed. The data set is then returned to the list of spare page data sets being used by the automation. If the data set was dynamically allocated you can also delete it physically when no other page data set is active on the volume where the data set is allocated.

Module: INGRX742

Classes: 40, 43.

TEC: NO

AOF955I NO DIFFERENCES FOUND

Explanation: A comparison of IPL information did not indicate any differences.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGRX200, INGRX201

Classes: None.

TEC: NO

AOF956I IPL INFORMATION OF *timestamp* FROM *sysname* HAS BEEN DELETED

Explanation: All IPL information collected for the system after it was IPLed at *timestamp* was deleted.

The variable *timestamp* shows the time of the IPL.

The variable *sysname* shows the name of the system.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGRVX20

Classes: None.

TEC: NO

AOF957I LOG ENTRY: *entry*

Explanation: The automation has logged an intermediate result of an automated process.

The variable *entry* shows the log entry.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGRVX92

Classes: None.

TEC: NO

AOF958I DUMP COULD NOT BE TAKEN. RC=*rc* RSN=*rsn*.

Explanation: The requested dump could not be taken.

The variable *rc* shows the return code. It can have the following values:

RC	Description
4	Only partial dump was taken.
8	SDUMPX macro error occurred.
12	An internal error occurred.
16	An internal error occurred.
20	An internal error occurred.
24	No private storage is available.
28	No common storage is available.

The variable *rsn* shows the reason code.

System action: Processing continues.

Operator response: Depending on the return code do the following:

RC	Description
4	Search the system log for message IEA911E for the reason of the partial dump.
8	Refer to the return and reason code description of the SDUMPX macro for the reason why the dump could not be taken.
12, 16, 20	Contact your IBM service representative.
24, 28	Retry the function later.

System programmer response: None.

Module: INGRX262

Classes: None.

TEC: NO

AOF959I SYSTEM *sysname* IS NOT REGISTERED.

Explanation: The requested system is not registered to the automation. However, this function requires the registration because it invokes system services locally.

System action: Processing continues.

Operator response: Override the system name with blanks to force the selection panel showing the system being registered. Or, start the automation on the requested system using the same XCF group as the local system.

System programmer response: None.

Module: INGRX260

Classes: None.

TEC: NO

AOF960E HARDWARE INFORMATION OF *system* COULD NOT BE VALIDATED.

Explanation: The automation detected that neither the hardware information of the indicated system has been defined, nor a connection to the Support Element of the indicated system has been made for any of the registered systems. If the system fails, the automation (if enabled) is not able to take the appropriate hardware actions to prevent possible hardware-related outages caused by the system.

The following variables are used:

system The name of a system or a coupling facility.

System action: Processing continues.

Operator response: None.

System programmer response: Add the hardware definitions and make them available to all registered systems using the ACF command. You can ignore this message if: the indicated system will run the automation, you have defined the necessary hardware information, and the system is able to contact the hardware.

Module: INGRX804, INGRX809

Classes: 40.

TEC: NO

AOF961I UNABLE TO CANCEL UN-NAMED JOB (SYSTEM=*sysname*/ASID=*asid*/TCB=*tcbaddr*). RESOURCE=*resource*.

Explanation: Automation detected a long running enqueue but is unable to cancel the job because the job name is unknown and it is running on an un-automated system.

The following variables are used:

sysname The name of the system running the address space.

asid The address space ID running the task.

tcbaddr The TCB address of the task holding the enqueue.

resource The enqueue resource major and minor name.

System action: None.

Operator response: None.

System programmer response: Ensure the enqueue is released and terminate the job if necessary.

Module: INGRX741

Classes: 40, 43.

TEC: YES

AOF962I UNABLE TO TERMINATE UN-NAMED JOB (SYSTEM=*sysname*/ASID=*asid*/TCB=*tcbaddr*). RESOURCE=*resource*.

Explanation: Automation detected a long running enqueue but is unable to cancel the job because the job name is unknown. An attempt to abend the task has also failed. Message AOF200I will detail why the abend has failed.

The following variables are used:

sysname The name of the system running the address space.

asid The address space ID running the task.

tcbaddr The TCB address of the task holding the enqueue.

resource The enqueue resource major and minor name.

System action: None.

Operator response: None.

System programmer response: Ensure the enqueue is released and terminate the job if necessary.

Module: INGRX741

Classes: 40, 43.

TEC: YES

AOF963I UN-NAMED JOB (SYSTEM=*sysname*/ASID=*asid*/TCB=*tcbaddr*) IS BEING TERMINATED. RESOURCE=*resource*.

Explanation: Automation detected a long running enqueue but is unable to cancel the job because the job name is unknown. Automation is attempting to abend the task.

The following variables are used:

sysname The name of the system running the address space.

asid The address space ID running the task.

tcbaddr The TCB address of the task holding the enqueue.

resource The enqueue resource major and minor name.

System action: None.

Operator response: None.

System programmer response: Check that the job has abended and ensure that the enqueue is released.

Module: INGRX741

Classes: 40, 43.

TEC: YES

AOF964I Due to the detection of a long *minor_res* lock the task *taskid* in address space *asid* on system *sysname* is being abended.

Explanation: The automation detected a lock on the indicated minor system resource being held for more than 10 seconds. To prevent the lockout of further commands the task holding the lock is being abended with the ability to do its own recovery.

The following variables are used:

minor_res The minor resource name of the lock. The major name is SYSIEFSD.

taskid The task ID holding the lock.

asid The address space ID running the task.

sysname The name of the system running the address space.

System action: The automation calls the Recovery Termination Manager to abend the indicated task.

Operator response: None.

System programmer response: None.

Module: INGRX743

Classes: 40, 43.

TEC: YES

AOF965I The command *command* issued by *jobname* is being purged due to a hung command detection.

Explanation: The automation detected a command that is still executing. The command is abended to avoid lockouts of other commands.

The following variables are used:

command The first two words of the command text when applicable.

jobname The job name issued the command.

System action: The automation abends the indicated command.

Operator response: None.

System programmer response: None.

Module: INGRX743

Classes: 40

TEC: NO

AOF966I Value *value* of type *type* for *sysname* could not be evaluated.

Explanation: The value for the indicated type could not be verified. The reason is that the BCP internal interface to the corresponding Support Elements could not be established on any of the registered systems in the sysplex.

The following variables are used:

sysname The name of the defined operating system.

value The defined value.

type The type in question. This can be one of the following:

CPC	The CPC name.
LPAR	The LPAR name.
SYSPLEX	The SYSLEX name.
TYPE	The operating system type, such as MVS or CF.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGRX809

Classes: 40

TEC: NO

AOF967E Value mismatch detected between *system1* and *system2* for *system3* and type *type*.

Explanation: The automation detected that the value of the indicated type could not be verified. The reason is that the BCP internal interface to the corresponding Support Elements wasn't established on any of the registered systems in the sysplex. In addition, at least two different definitions exist for the indicated type on different systems running SA z/OS.

As soon as the automation gets access to the Support Element, the value will be re-evaluated and automatically corrected.

The following variables are used:

<i>system1</i>	The name of a system running SA z/OS.
<i>system2</i>	The name of a system running SA z/OS.
<i>system3</i>	The name of the defined operating system.
<i>type</i>	The type in question. This can be one of the following:
CPC	The CPC name.
LPAR	The LPAR name.
SYSPLEX	The SYSLEX name.
TYPE	The operating system type, such as MVS or CF.

System action: Processing continues.

Operator response: None.

System programmer response: Check and correct the definitions before the next start-up of the affected systems.

Module: INGRX809

Classes: 40

TEC: NO

AOF968E Value *oldvalue* of type *type* for *sysname1* on *sysname2* has been replaced by *newvalue*.

Explanation: The automation detected that your hardware definition differs from the actual hardware. If the local system name is not shown in the message, a different setup other than the local system has been used. This should generally be avoided.

The following variables are used:

<i>sysname1</i>	The name of the defined operating system.
<i>newvalue</i>	The new value.

oldvalue The improper value. This may be (NULL) if a value has been found in the hardware configuration but there is no definition in the ACF.

sysname2 The name of the system where the improper setting was detected.

type The type in question. This can be one of the following:

CPC	The CPC name.
LPAR	The LPAR name.
SYSPLEX	The SYSLEX name.
TYPE	The operating system type, such as MVS or CF.

System action: The improper definition is temporarily replaced by the actual value to prevent any outage that could be caused by the old value.

Operator response: None.

System programmer response: Check and correct the definitions in the ACF before the next startup of the indicated system.

Module: INGRX809

Classes: 40

TEC: NO

AOF969I SESSION *luname* OF SMCS CONSOLE *consname* HAS BEEN TERMINATED DUE TO A BUFFER SHORTAGE CONDITION.

Explanation: Automation has detected a shortage of message or reply buffers caused by the indicated console. To prevent the shortage from turning into an outage condition, the console session has been terminated. This recovery action cannot be reversed on the shortage relief message.

The variable *luname* specifies the name of the session.

The variable *consname* specifies the name of the console.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGRX730

Classes: 40

TEC: NO

AOFS000I *aa*

Explanation: This message is given in conjunction with a subsequent error message that explains what is incorrect. It shows the SDF panel that is incorrect.

The variable *aa* identifies the buffer data in error.

System action: None.

Operator response: None.

System programmer response: Correct the panel definitions.

AOFS001I *aa* is active

Explanation: The task *aa* has been activated.

The variable *aa* shows the name of the task.

System action: None.

Operator response: None.

System programmer response: None.

AOFS002I *aa* is inactive

Explanation: The task *aa* is not active at this time.

The variable *aa* shows the name of the task number.

System action: None.

Operator response: Start the task by issuing START TASK=AOFTDDF.

System programmer response: None.

AOFS003I *aa* is terminating.

Explanation: The task *aa* is ending.

The variable *aa* shows the name of the task number.

System action: None.

Operator response: None.

System programmer response: None.

AOFS004I *aa* is terminated

Explanation: The task *aa* has ended.

The variable *aa* shows the name of the task number.

System action: None.

Operator response: None.

System programmer response: None.

AOFS005I LOGON to *aa* was unsuccessful

Explanation: The requested action could not be performed successfully.

The variable *aa* shows the name of the object that the action was performed on.

System action: None.

Operator response: None.

System programmer response: None.

AOFS006I LOGON denied, maximum users exceeded.

Explanation: The maximum number of users that can log on as SDF operators has been reached.

System action: SDF cannot be accessed.

Operator response: None.

System programmer response: Increase the value of MAXOPS in member AOFINIT in your NetView DSIPARM data set. Then restart SDF.

AOFS007I Request denied because NetView is terminating.

Explanation: SDF cannot be accessed because NetView is ending.

System action: None.

Operator response: None.

System programmer response: None.

AOFS008I Command SDF is not supported under a non-OST task.

Explanation: This message is a response to an attempt to access the SDF from a non-OST task. Non-OST tasks cannot access the SDF because they do not support full-screen function.

System action: None.

Operator response: None.

System programmer response: None.

AOFS009I Extended attribute support is required.

Explanation: This message is a response to a request to display a SDF panel. The panel could not be displayed because it requires extended console support.

System action: The SDF request is ended.

Operator response: None.

System programmer response: None.

AOFS010I Request "*aa*" was successful for "*bb*".

Explanation: The request has been completed successfully.

The variable *aa* shows the request that was performed.

The variable *bb* shows the component that was processed.

AOFS011I • AOFS023I

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **AOFS011I Request "aa cc" was successful for "bb".**

| **Explanation:** The request has been completed successfully.

| The variable *aa cc* shows the request that was performed.

| The variable *bb* shows the component that was processed.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **AOFS012I Panel aa has been added.**

| **Explanation:** The panel *aa* has been added successfully.

| The variable *aa* shows the name of the panel.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **AOFS013I Panel aa has been replaced.**

| **Explanation:** The panel *aa* has been replaced successfully.

| The variable *aa* shows the name of the panel.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **AOFS014I Panel aa has not been added.**

| **Explanation:** The panel *aa* has not been added because of a syntax error condition.

| The variable *aa* shows the name of the panel.

| **System action:** Processing continues.

| **Operator response:** None.

| **System programmer response:** Correct the panel definitions.

| **AOFS015I Panel aa has not been replaced.**

| **Explanation:** The panel *aa* has not been replaced because of a syntax error condition.

| The variable *aa* shows the name of the panel.

| **System action:** The former panel definitions are still valid.

| **Operator response:** None.

| **System programmer response:** Correct the panel definitions and reload the panel.

| **AOFS020I Too many columns defined for screen width.**

| **Explanation:** Dynamic composition of a SDF panel was requested with BODY or CELL statements but the number of columns specified exceeds the maximum panel width.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Correct the panel definitions.

| **AOFS021I Temporary screen read error.**

| **Explanation:** The number of lines allowed has been exceeded.

| **System action:** The system continues to attempt to display the screen data.

| **Operator response:** If this condition does not change without intervention, contact your system programmer.

| **System programmer response:** Review the adequacy of the screen display buffer for the expected screen activity levels.

| **AOFS022I Temporary screen read error - limit exceeded.**

| **Explanation:** The number of lines allowed has been exceeded.

| **System action:** The system terminates the function.

| **Operator response:** Contact your system programmer.

| **System programmer response:** Review the adequacy of the screen display buffer for the expected screen activity levels.

| **AOFS023I Buffer size specified is too small for screen size.**

| **Explanation:** The screen buffer size specified in AOFINIT is too small for the screen.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Increase the value of SCREENSZ in AOFINIT in your NetView DSIPARM data set. Then restart SDF.

AOFS030I Unable to find component "aa."

Explanation: SDF did not find an entry for the component *aa*.

The variable *aa* shows the name of the component.

System action: None.

Operator response: None.

System programmer response: None.

AOFS031I *bb* is unable to find component "aa."

Explanation: The command did not find an entry for the component *aa*.

The variable *aa* shows the name of the component.

The variable *bb* shows the name of the command.

System action: None.

Operator response: None.

System programmer response: None.

AOFS032I Unable to find component "aa bb(cc)".

Explanation: SDF did not find an entry for the component *aa*.

The variable *aa* shows the root name of the component.

The variable *bb* shows the name of the component.

The variable *cc* shows the alternate name of the component.

System action: None.

Operator response: None.

System programmer response: None.

AOFS090I *aa bb cc* status information

Explanation: A request to use status descriptor information has been processed for the status component identified in this message. Messages AOFS091I, AOFS092I, AOFS093I, and AOFS094I are generated after this one and show the contents of the status descriptors matching the request search arguments.

The variable *aa* shows the root (system) name of the requested status component.

The variable *bb* shows the status component name. If an alternate status component name was passed by SDF, it appears in parentheses following the primary status component name.

System action: Processing continues. However, the function may not operate as requested.

Operator response: None.

System programmer response: None.

AOFS091I *aa.bb(cc),PR=dd,RV=ee,CO=ff,HL=gg,DP=hh*

Explanation: A status descriptor has been found that matches the search arguments in a request to use status descriptor information. This message identifies the status descriptor and shows some of the information it contains. This message identifies the status descriptor and shows some of the information it contains. Messages AOFS092I, AOFS093I and AOFS094I show additional information contained in the status descriptor.

The variable *aa* shows the root (system) name of the requested status component.

The variable *bb* shows the status component name.

The variable *cc* shows the alternate status component name.

The variable *dd* shows the priority associated with the status descriptor. (This is normally specified in a request to add the status descriptor to the chain of descriptors for a status component).

The variable *ee* shows the reference value for the status descriptor.

The variable *ff* shows the color associated with the status descriptor. If this color is not specified on a request to add a status descriptor, the default color defined for the priority of the status descriptor is used. The color is presented in abbreviated form as follows:

- B (Blue)
- G (Green)
- P (Pink)
- R (Red)
- T (Turquoise)
- W (White)
- Y (Yellow)

The variable *gg* shows the highlighting option associated with the status descriptor. The highlighting option is presented in abbreviated form as follows:

- B (Blink)
- N (Normal)
- R (Reverse)
- U (Underscore)

The variable *hh* shows the number of existing duplicate status descriptors. Duplicate status descriptors have identical reference value, priority, info, color, highlight, and data values.

System action: Processing continues.

Operator response: None.

System programmer response: None.

AOFS092I DATE=*aa*,TIME=*bb*,REPORTER=*cc*(*dd*),
 PFP=*ee*, PU=*ff*,PD=*gg*,PLVU=*hh*,PLVD=*ii*

Explanation: A status descriptor has been found that matches the search arguments in a request to use status descriptor information. This message shows some of the information contained in the status descriptor. Messages AOFS091I, AOFS093I, and AOFS094I contain additional information contained in the status descriptor.

The variable *aa* shows the date that the status descriptor was created on in the format mm/dd/yy.

The variable *bb* shows the time that the status descriptor was created in the format hh:mm:ss.

The variable *cc* shows the NetView operator ID that made the request to create the status descriptor.

The variable *dd* shows the node of the reporter, if available, appears in parentheses immediately following the ID.

The variable *ee* indicates whether or not the status condition is propagated to the focal point. If propagation was requested, Y is shown, if not, N is shown.

The variable *ff* indicates whether or not the status condition is propagated upward through the SDF tree structure. If upward propagation was requested, Y is shown; if not, N is shown.

The variable *gg* indicates whether or not the status condition is propagated downward through the SDF tree structure. If downward propagation was requested, Y is shown; if not, N is shown.

The variable *hh* shows the tree node that upward propagation is not performed beyond. An asterisk(*) indicates that upward propagation ends at the root (system) node.

The variable *ii* shows the tree node that downward propagation is not performed beyond. An asterisk (*) indicates that downward propagation is performed to all subordinate nodes of the status component.

System action: Processing continues.

Operator response: None.

System programmer response: None.

AOFS093I IN=*aabbcc*

Explanation: A status descriptor has been found that matches the search arguments in a request to use status descriptor information. This message shows some of the information contained in the status descriptor. Messages AOFS091I, AOFS093I, and AOFS094I show additional information contained in the status descriptor.

The variable *aa* shows the value used to replace the default STATUSTEXT information when the status descriptor is used as the basis for the STATUSFIELD

and a status descriptor number other than 0 is specified on the STATUSFIELD definition.

System action: Processing continues.

Operator response: None.

System programmer response: None.

AOFS094I DA=*aabbcc*

Explanation: A status descriptor has been found that matches the search arguments in a request to use status descriptor information. This message shows some of the information contained in the status descriptor. Messages AOFS091I, AOFS092I, and AOFS093I show additional information contained in the status descriptor.

The variable *aa* shows the data displayed when the status descriptor is selected as the basis for a detail display.

System action: Processing continues.

Operator response: None.

System programmer response: None.

AOFS095I End of *aa bbcc* status information

Explanation: This message identifies the end of the status information:

The variable *aa* shows the root (system) name of the requested status component.

The variable *bb* shows the status component name. If an alternate status component name was passed by SDF, it appears in parentheses following the primary status component name.

System action: None.

Operator response: None.

System programmer response: None.

AOFS098I *aa*

Explanation: This message is generated by a CLIST. It provides the name of an SDF tree member for a selected message system.

The variable *aa* shows the SDF tree component name.

System action: Processing continues.

Operator response: None.

System programmer response: None.

AOFS099I End of *aa*

Explanation: This message indicates the end of the command *aa*.

The variable *aa* shows the name of the command.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **AOFS500I** **Maximum number of "aa-bb" operands exceeded.**

| **Explanation:** The maximum number of operands allowed for the keyword identified in this message was exceeded.

| The variable *aa* shows the name of the panel that contains the keyword.

| The variable *bb* shows the keyword that too many operands were specified for.

| **System action:** None.

| **Operator response:** Enter the command again, making sure that the correct number of operands is specified for the keyword you specify.

| **System programmer response:** None.

| **AOFS501I** **Specified operand "aa" invalid for "bb".**

| **Explanation:** An operand that is not valid was specified for this command.

| The variable *aa* shows the operand that is not valid.

| The variable *bb* shows the command was entered.

| **System action:** None.

| **Operator response:** Enter the command again using the correct operands.

| **System programmer response:** None.

| **AOFS502I** **Specified operand "aa" invalid for "bb cc".**

| **Explanation:** An operand that is not valid was specified for this command.

| The variable *aa* shows the operand that is not valid.

| The variable *bb cc* shows the command that was entered.

| **System action:** None.

| **Operator response:** Enter the command again using the correct operands.

| **System programmer response:** None.

| **AOFS503I** **Specified parameter "aa" invalid.**

| **Explanation:** The specified parameter is not valid for this command.

| The variable *aa* shows the parameter that is not valid.

| **System action:** None.

| **Operator response:** Enter the command again using the correct parameters.

| **System programmer response:** None.

| **AOFS504I** **Expected parameter(s) missing for request "aa".**

| **Explanation:** One or more command parameters are missing.

| The variable *aa* shows the parameter in error.

| **System action:** None.

| **Operator response:** Enter the command again using the correct parameter(s).

| **System programmer response:** None.

| **AOFS505I** **Syntax error for "aa".**

| **Explanation:** A syntax error was encountered in the command that was entered.

| The variable *aa* shows the parameter in error.

| **System action:** None.

| **Operator response:** Correct the command syntax and reissue the command.

| **System programmer response:** None.

| **AOFS506I** **First level of the tree must be a 1: aa.**

| **Explanation:** There was an attempt to define the structure of a tree with a number other than 1.

| The variable *aa* shows the first leaf of the tree.

| **System action:** The AOFTDDT task will not become active.

| **Operator response:** Contact your system programmer.

| **System programmer response:** Ensure your system tree definitions in AOFTREE start with a level number of 1. Then issue the START TASK=AOFTDDF command again.

| **AOFS507I** **Invalid level - less than 1: aa.**

| **Explanation:** There was an attempt to define the structure of a tree with a number that is less than 1.

| The variable *aa* shows the tree's leaf in error.

| **System action:** The AOFTDDF task will not become active.

| **Operator response:** Contact your system programmer.

| **System programmer response:** Make sure that the system tree definitions in AOFTREE have a level number of 1 or higher. Then issue the START TASK=AOFTDDF command again.

| **AOFS508I** **Status element in tree not found,**
| **Root=*aa* Element=*bb*.**

| **Explanation:** The root component or status component
| name, or both, that were specified in an SDF
| STATUSFIELD statement cannot be found in the active
| SDF tree structure. The information message is issued
| during SDF initialization, or while processing an
| SDFPANEL ADD command. If the function completes
| successfully, message AOFS011I is issued, indicating the
| panel in question.

| The variable *aa* shows the name of the root
| component.

| The variable *bb* shows the name of the status
| component.

| **System action:** Processing continues.

| **Operator response:** None.

| **System programmer response:** Ensure that the
| referenced root component or status component, or
| both, that were specified in the SDF STATUSFIELD
| statement for the loading panel exist in the active SDF
| tree structure.

| **AOFS509I** **Status element major error, Root=*aa***
| **Element=*bb*.**

| **Explanation:** An internal SDF status element chaining
| error occurred.

| The variable *aa* shows the name of the root
| component.

| The variable *bb* shows the name of the status
| component.

| **System action:** SDF request ended.

| **Operator response:** Contact your system programmer.

| **System programmer response:** If you cannot
| determine the cause of the problem using the
| information available to you, contact your IBM service
| representative.

| **AOFS510I** **Tree levels out of sequence: *aa*.**

| **Explanation:** An attempt was made to define the
| structure of a tree with the dependent subsystems
| numbered out of sequence.

| The variable *aa* shows the tree's leaf in error.

| **System action:** The AOFTDDF task will become
| active, but the status of the subsystems will not be
| determined.

| **Operator response:** None.

| **System programmer response:** Make sure that the
| system tree definitions in AOFTREE follow in a valid
| hierarchical sequence.

| **AOFS511I** **Member "*aa*" not found.**

| **Explanation:** SDF attempted to prepare the member
| named in the message but could not find it.

| The variable *aa* shows the name of the member that
| could not be found.

| **System action:** Processing continues.

| **Operator response:** None.

| **System programmer response:** Make sure that the
| member identified in the message is valid. Add the
| member in the DSIPARM concatenation chain or correct
| the SDF panel definitions that refer to the erroneous
| member.

| **AOFS512I** **Field overlapping occurred in *gg*:**
| ***aa(bb,cc) <-> dd(ee,ff).***

| **Explanation:** During the post-validation of all panels
| that have been defined SDF detected an overlay
| between two field definitions.

| The variables *aa* and *dd* show the name of the panel
| definition statement.

| The variables *bb* and *ee* show the actual row number
| where the field starts.

| The variables *cc* and *ff* show the actual column
| number where the field starts.

| The variables *gg* shows the name of the panel.

| **System action:** The panel is not processed.

| **Operator response:** None.

| **System programmer response:** Correct the panel
| definitions.

| **AOFS513I** **Terminating propagation level "*aa*" not**
| **found.**

| **Explanation:** The identified propagation level was not
| found the on the chain of predecessors (PropLvLU) or
| among the descendents of (PropLvLD) the status
| component identified as the target of the command.

| The variable *aa* shows the propagation level
| (PropLvLU or PropLvLD) as specified by SDF.

| **System action:** Propagation is performed; however, it
| ends at the root node or the descendent leaf nodes of
| the tree structure.

| **Operator response:** Check the propagation level value
| for accuracy.

| **System programmer response:** None.

| **AOFS514I** **Maximum number of *aa* keywords**
| **exceeded.**

| **Explanation:** The maximum number of occurrences of
| the keyword identified in this message was exceeded.

| The maximum number is normally a variable specified using another keyword.

| The variable *aa* shows the keyword that appears too many times.

| **System action:** Processing continues. However, the function may not operate as requested.

| **Operator response:** Note the message content and the command or function being attempted. Contact your system programmer.

| **System programmer response:** Increase the number of allowable keywords by changing or specifying the control variable.

| **AOFS515I** Keyword *aa* is out of sequence.

| **Explanation:** The identified keyword appears incorrectly in relation to other keyword parameters specified for the function being performed.

| The variable *aa* shows the keyword that appears incorrectly.

| **System action:** Processing continues. However, the function may not operate as requested.

| **Operator response:** Note the message content and the command or function being attempted. Contact your system programmer.

| **System programmer response:** Correct the keyword specification sequence and retry the operation.

| **AOFS516I** Keyword *aa(bb)* of *cc* is out of sequence.

| **Explanation:** The identified keyword appears incorrectly in relation to other keyword parameters specified for the function being performed.

| The variable *aa* shows the keyword that appears incorrectly.

| The variable *bb* shows the parameter that appears incorrectly.

| The variable *cc* shows name of the panel.

| **System action:** Processing continues. However, the function may not operate as requested.

| **Operator response:** Note the message content and the command or function being attempted. Contact your system programmer.

| **System programmer response:** Correct the keyword specification sequence and retry the operation.

| **AOFS517I** Expected parameter(s) missing for "*aa-bb*".

| **Explanation:** One or more parameters are missing.

| The variable *aa* shows the name of the panel.

| The variable *bb* shows the keyword that lacks the parameter(s).

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Correct the panel definitions. Then enter the command again.

| **AOFS518I** Expected parameter(s) missing for "*aa-bb(cc)*".

| **Explanation:** One or more parameters are missing:

| The variable *aa* shows the name of the panel.

| The variable *bb* shows the keyword that lacks the parameter(s).

| The variable *cc* shows the parameter that is missing.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Correct the panel definitions. Then enter the command again.

| **AOFS519I** Specified operand "*aabb*" invalid for "*cc-dd(ee)*".

| **Explanation:** An operand that is not valid was specified for this command.

| The variable *aa* shows the operand that is not valid.

| The variable *bb* indicates if the operand is too long.

| The variable *cc* shows the name of the panel.

| The variable *dd* shows the keyword in error.

| The variable *ee* shows the parameter in error.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Correct the panel definitions. Then enter the command again.

| **AOFS520I** Specified parameter "*aa-bb*" invalid.

| **Explanation:** The specified parameter is not valid for the keyword.

| The variable *aa* shows the name of the panel.

| The variable *bb* shows the keyword in error.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Correct the panel definitions. Then enter the command again.

| **AOFS521I** CELL definitions do not fit in *aa*: BODY(*bb,cc*).

| **Explanation:** During the post-validation of all panels that have been defined SDF detected a BODY section with CELL definitions that will not fit. Either a single CELL definition is out of range or the total length of all

| CELL definitions is too large. The total length is
 | computed as follows:
 | #_cols (BODY) * rightmost end_pos (CELL) +
 | (#_cols (BODY) - 1) * distance (BODY) + 1

| The length of the body section is calculated as follows:
 | end_pos - start_pos + 1

| If the BODY defines an undefined end_pos the panel
 | width is used. In case the PANEL defines an undefined
 | width the screen width defined in member AOFINIT or
 | on the command SDFPANEL is used.

| The variable *aa* shows the name of the panel.
 | The variable *bb* shows the actual row number where
 | the BODY field starts.
 | The variable *cc* shows the actual columns number
 | where the BODY field starts.

| **System action:** The panel is not processed.
 | **System programmer response:** Correct the panel
 | definitions.
 | **Operator response:** None.

| **AOFS800E Internal error: Request "aa" invalid.**
 | **Explanation:** An internal programming error occurred.
 | The variable *aa* shows the name of the request.
 | **System action:** The request is rejected.
 | **Operator response:** Record the type of the request,
 | and contact your system programmer.
 | **System programmer response:** Contact your IBM
 | service representative.

| **AOFS801E Internal error: Invalid descriptor point.**
 | **Explanation:** An internal programming error has
 | occurred involving the propagation of status through
 | the component tree as maintained by SDF. A request
 | was made to propagate status without supplying the
 | status to be propagated.
 | **System action:** Other messages related to this problem
 | might be generated.
 | **Operator response:** Contact your system programmer.
 | **System programmer response:** Report this problem to
 | your IBM service representative. Attempt to identify
 | the tree being used, and whether a status was being
 | added or deleted when the condition was detected.

| **AOFS802E Internal error: Invalid state "aa" action
 | "bb".**
 | **Explanation:** An internal error occurred during
 | processing of a SDF panel definition statement. There is
 | a syntax error in the DSIPARM member that contains
 | the panel definition.

| The variable *aa* shows the value of the state that is
 | not valid.
 | The variable *bb* shows the value of the action that is
 | not valid.

| **System action:** None.
 | **Operator response:** Contact your system programmer.
 | **System programmer response:** Review the SDF panel
 | definition and correct the syntax that is in error.

| **AOFS810I Macro *aa* type *bb* request was
 | unsuccessful, RC=*cc*.**
 | **Explanation:** The program called the indicated macro.
 | The macro was not able to satisfy the request and
 | returned the indicated return code for use in problem
 | determination.
 | The variable *aa* shows the name of the macro.
 | The variable *bb* shows the type of the request.
 | The variable *cc* shows the return code of the macro.
 | **System action:** None.
 | **Operator response:** Contact your system programmer.
 | **System programmer response:** Check the return code
 | to determine the cause of the problem.

| **AOFS811I *aa* request was unsuccessful, RC=*bb*.**
 | **Explanation:** The program called the indicated macro.
 | The macro was not able to satisfy the request and
 | returned the indicated return code for use in problem
 | determination.
 | The variable *aa* shows the name of the macro.
 | The variable *bb* shows the return code of the macro.
 | **System action:** None.
 | **Operator response:** Contact your system programmer.
 | **System programmer response:** Check the return code
 | to determine the cause of the problem.

| **AOFS812I Macro *aa* request was unsuccessful,
 | RC=*bb*.**
 | **Explanation:** The program called the indicated macro.
 | The macro was not able to satisfy the request and
 | returned the indicated return code for use in problem
 | determination.
 | The variable *aa* shows the name of the macro.
 | The variable *bb* shows the return code of the macro.
 | **System action:** None.
 | **Operator response:** Contact your system programmer.
 | **System programmer response:** Check the return code
 | to determine the cause of the problem.

| AOFS813I **Macro *aa* request was unsuccessful for**
| *bb*, RC=*cc*.

| **Explanation:** The program called the indicated macro.
| The macro was not able to satisfy the request and
| returned the indicated return code for use in problem
| determination.

| The variable *aa* shows the name of the macro.

| The variable *bb* shows the name of the control block.

| The variable *cc* shows the return code of the macro.

| **System action:** None.

| **Operator response:** Contact your system programmer.

| **System programmer response:** Check the return code
| to determine the cause of the problem.

| AOFS820I **I/O error occurred while processing**
| member "*aa*".

| **Explanation:** An I/O error occurred during processing
| of the member *aa*.

| The variable *aa* shows the name of the member.

| **System action:** Processing terminates.

| **Operator response:** Make sure the member identified
| in the message is valid. If the member identified in the
| message is valid, contact your system programmer for
| further analysis.

| **System programmer response:** Verify that the member
| name exists in the file pointed to by the procedure in
| use at the time of the error. Review the Netlog being
| used at the time of the error to see whether there was a
| system error that could have prevented the member
| from being found.

Messages AOF000I to AOF820I

Chapter 3. Messages EVE120I to EVE842E (CICS Automation)

This chapter contains CICS Automation messages. CICS Automation also uses the NetView HELPMMSG facility to provide message information online. To view a help panel for a specific message, enter on the CICS Automation command line:

HM EVE nnn

Where nnn is the message ID number.

EVE120I **Command accepted for** *subsystem*,
 APPLID = *applid*.

Explanation: An EVESNCCI command has been validated and is accepted.

Operator response: None.

EVE121E **Error on** *DSLxxx* **request in** *progrname*, **RC**
 = *retcode*.

Explanation: CICS Automation found an error on a *DSLxxx* macro request issued in the program identified by *progrname*. The register 15 return code is *retcode*.

Operator response: None.

System programmer response: Analyze the netlog and determine whether there is a logic error. If the problem is a CICS Automation logic error, contact your IBM Support Center.

Problem determination: Refer to *Customization: Using Assembler*, for return codes from the *DSLxxx* macro requests.

EVE122E **EVENTASK NOT ACTIVE**

Explanation: A DSIMQS macro request failed because the addressed task identified by *taskid* was not active. This error can occur when any automated operator specified in the EVENTASK initialization member is not active or when the EVENTASK optional task is not active.

Operator response: If the EVENTASK task is not active, start the task by issuing the "START TASK =EVENTASK" or by using option 9.3 from the operator interface. If one of the automated operators is not active, start the autotask by issuing the "AUTOTASK OPID=*taskid*" command.

System action: When the error is met, a return code of 8 is passed to the issue of the EVESNCCI command.

System programmer response: Determine why *taskid* was not active.

EVE125E **No storage available on** *DSLxxxx* **request**
 in *progrname*.

Explanation: An error has been met on a *DSLxxx* request in the specified program, indicating that no NetView storage is available.

Operator response: None.

System programmer response: Investigate NetView storage use. If you cannot locate the source of the storage problem, contact your IBM Support Center.

EVE128I **Positive acknowledgement.**

Explanation: A positive response (ACK) has been received on an EVESNCCI Converse request.

Operator response: None.

EVE129E *msgtext*

Explanation: A negative response (NACK) has been received on an EVESNCCI Converse request or an error has been detected during the processing of the request. In the latter case, *msgtext* contains the EVE1 nnn E message text explaining the error.

Operator response: If an error condition is indicated, browse the netlog to determine the cause of the error. NACK can be a "normal" response from a CICS transaction. CSMT log may contain useful information as well in case of an error.

EVE130I **EVENTASK ready.**

Explanation: EVENTASK OPT initialization has completed successfully.

Operator response: None.

EVE131I **EVENTASK terminated.**

Explanation: EVENTASK OPT has completed termination.

Operator response: None.

EVE132E **Initialization member** *memname* **not**
 found.

Explanation: The EVENTASK initialization member

EVE133E • EVE148I

memname could not be found in one of the DSIPARM data sets.

Operator response: None.

System programmer response: Ensure that the initialization member specified by MEM= keyword on the TASK statement for EVENTASK is contained in one of the DSIPARM data sets. Restart the task by issuing the "START TASK = EVENTASK" command or option 9.3 from the operator interface.

EVE133E Error in *memname* at line *linenum*, code = *retcode*.

Explanation: An error of type *retcode* is on line *linenum* of the EVENTASK initialization member *memname*.

- 01 Unrecognized keyword.
- 02 Duplicate specification.
- 03 Operand specification error.
- 04 Too many server keywords.
- 05 No valid servers specified.

Operator response: None.

System programmer response: Correct the failing statement in the EVENTASK initialization member *memname* and restart the task with the "START TASK = EVENTASK" command or option 9.3 from the operator interface.

EVE136I Error on PPI request *reqid*, RC = *retcode*.

Explanation: The program-to-program interface request *reqid* received a non-zero return code. This is generally an error, however REQUEST 002 RC=014 indicates the PPI receiver is active; this is not an error condition.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Review the NetView netlog and investigate the cause of the error. Program-to-program interface request types *reqid* and request return codes *retcode* are documented in the appendix "Program-to-Program Interface Return Codes" in *NetView Application Programming Guide*. *reqid* identifies the PPI request type and *retcode* details what the problem is in issuing the PPI request.

EVE137E NetView subsystem not available.

Explanation: An error has been encountered to a program-to-program interface request indicating that no NetView subsystem was available to act as a program-to-program interface server.

Operator response: After the NetView subsystem is restarted, start the EVENTASK optional task by issuing the "START TASK=EVENTASK" command or option 9.3 from the operator interface.

System programmer response: Determine why no NetView subsystem was active.

EVE140E Incorrect PPI buffer received.

Explanation: The program-to-program interface buffer sent to the EVENTASK optional task was not correct. Possible causes include:

- Program-to-program interface buffer header was incomplete.
- Request type was incorrect (not C, S, R, A, or N).
- Length specification was inconsistent.

Operator response: None.

System programmer response: Correct the routine that created the incorrect program-to-program interface buffer. If the error is caused by a CICS Automation logic error, contact your IBM Support Center.

EVE141E Incorrect MQS buffer received.

Explanation: The MQS message buffer, containing a program-to-program interface request that was sent to the EVENTASK optional task, was not correct. Possible causes include:

- Program-to-program interface buffer header was incomplete.
- Request type was incorrect (not C, S, R, A, or N).
- Length specification was inconsistent.

Operator response: This problem is caused by CICS Automation logic error. Contact your IBM Support Center.

EVE142E Function *function* not found in *memname*.

Explanation: A program-to-program interface buffer sent to the EVENTASK optional task contained a function and type specification that no entry in the EVENTASK initialization member *memname* was found for.

Operator response: None.

Programmer response: Correct the routine that created the incorrect program-to-program interface buffer if the function name *function* was incorrect.

System programmer response: Add a request or response server for *function* in the EVENTASK initialization member *memname* if the function name was correct.

EVE148I Segment-chain *id* canceled.

Explanation: An EVESNCCI cancel request successfully freed all saved segments for the segment identifier *id*.

Operator response: None.

EVE149I Segment-chain *id* not found.

Explanation: An EVESNCCI cancel request (TYPE=C) was issued. The specified segment-chain *id* could not be found.

Operator response: None.

EVE171E *procname* : Error in *progrname* (*transid*), *reason* = *reason*.

Explanation: CICS Automation has detected an error in the program: *progrname* in the transaction *transid*. The type of error is *reason*.

In all CICS messages *procname* is the JOB name or STC name that identifies the CICS system that issued the message.

System action: Depending on *reason* a transaction dump is created in the active CICS dump data set.

If the error occurred in:	Then:
EVESPPIC, EVESPPIP or EVESPPIS,	The transaction is terminated.
EVESCCCI	A non-zero return code is passed to the caller.
EVESHPTH, and EVESHPTH was invoked by a converse request (TYPE=C)	A NACK response is passed to NetView.
EVESTISP	Control is relinquished to allow CICS initialization to continue, but the program-to-program interface is not started.
EVESPERR or EVESPMMSG	Control is returned to the calling module.

Operator response: None.

System programmer response: Refer to “Error Codes” on page 117 and if necessary analyze the transaction dump to determine the cause of the error, then correct the problem. If the problem is caused by a CICS Automation logic error, contact your IBM Support Center.

EVE172I *applid* : PPI active.

Explanation: This message confirms that the program-to-program interface successfully initialized.

The variable *applid* is the CICS application Identifier, which is used as the PPI receiver name for this CICS.

Operator response: None required.

EVE173I *applid* : PPI inactive.

Explanation: This message confirms that the program-to-program interface normally terminated.

The variable *applid* is the CICS application Identifier, which is used as the PPI receiver name for this CICS.

Operator response: None required.

EVE174E *applid* : PPI not active.

Explanation: The transaction to terminate the program-to-program interface has been invoked, but the program-to-program interface was not active.

The variable *applid* is the CICS application Identifier, which is used as the PPI receiver name for this CICS.

Operator response: None.

EVE175E *applid* : Function *function* not found in EVESPINM.

Explanation: A program-to-program interface buffer sent from NetView to CICS contained a function *function* that no entry in EVESPINM was found for.

The variable *applid* is the CICS application Identifier, which is used as the PPI receiver name for this CICS.

System action: For a converse request (TYPE=C), a NACK response containing the error message text is returned to the operator at the domain specified in the program-to-program interface buffer header. This is the *operid* in *domainid* of the originating EVESNCCI command.

For all other requests (TYPE=S, R, A or N): none.

Operator response: None.

Programmer response: Correct the routine that created the incorrect program-to-program interface buffer if the function name *function* was incorrect.

System programmer response: Add a request server for *function* in EVESPINM if the function name was correct.

EVE176E *procname* : No response expected for identifier *id*.

Explanation: A response identified by *id* was received from NetView. No transaction is awaiting that response. This may be caused by:

- An incorrect *id* in the response
- The transaction awaiting the response timing out
- A C961 or C971 error occurring.

Operator response: None.

Programmer response: Have the NetView command list or command processor that issues the unexpected

EVE177E • EVE657E

PPI RESPONSE corrected, or increase the timeout value on the converse request.

EVE177E *procname* : **NetView subsystem is not available.**

Explanation: A program-to-program interface request has been issued, but not NetView subsystem was available to act as a program-to-program interface server.

System action: The program-to-program interface request is ignored. The program-to-program interface is terminated.

Operator response: After the NetView subsystem that serves program-to-program interface requests has been (re)started, restart the program-to-program interface: "MODIFY *procname*,COPS". COPS can also be restarted via support panel option 9.3.

System programmer response: Determine why the NetView subsystem was not available.

EVE178E *procname* : **PPI already active.**

Explanation: The transaction to start the program-to-program interface was invoked. The program-to-program interface was not inactive, however.

System action: The program-to-program interface start-up request is ignored.

Operator response: None.

EVE179E *procname* : **Receiver program *progname* problem, reason = *reason*.**

Explanation: An error has been encountered on a program-to-program interface request indicating that the specified receiver program *progname* was "not defined" or "not active."

System action: Return code 8 and an error code (C226 or C204) identifying the precise cause of the failure are set for the caller.

Operator response: None.

System programmer response: Analyze the cause of the failure, then correct the problem. Ensure that RECEIVERID= operand in the EVESPINM initialization member is identical to RECEIVERID= operand in the EVENTASK initialization member.

EVE182E *procname* : **Console has not been defined to CICS.**

Explanation: EVESPINM module specifies a console definition where COPC transaction should be started. That console name is not defined in CICS.

System programmer response: Either correct EVESPINM, or define the console to CICS.

EVE209I **WARNING: ACF ENTRY *entrytype*, HAS INVALID DATA "*data*" FOR "*entry*", *reason*.**

Explanation: The entry-type pair named in the message was found by the automation routine to contain errors. This message is issued when building the CICS Message Exit policy from the ACF data.

The variable *entrytype* is the ENTRY-TYPE pair.

The variable *data* is the data that caused the error.

The variable *entry* is:

- "INSERT=" for messages beginning with DFH.
- "TOKEN=" for TD Queue messages.
- "MSGDISP=" for both.

It refers to the statement that contained the error.

The variable *reason* is an explanation of the error.

Invalid number or invalid choice of keyword.

System action: The message policy is loaded without the offending data.

In the case of MSGDISP= this might mean the entire message is not loaded into the policy.

In the case of TOKEN= or INSERT= the offending TOKEN= or INSERT= statement is ignored.

Operator response: None required.

System programmer response: Correct the entry-type pair in your automation control file and reload it using the INGAMS command.

EVE654E **No INCLUDE/EXCLUDE entries exist in LISTSHUT table for *subsystem*.**

Explanation: The LISTSHUT table must contain either INCLUDE or EXCLUDE entries for the specified subsystem in order for SHUTDOWN to occur.

System action: No transactions will be purged.

Operator response: Change the table and reload the control file.

Classes: 40.

EVE657E **Message received by EVEET050: *msgtext*.**

Explanation: EVEET050 received an unexpected message.

System action: Processing is terminated.

Operator response: Notify your system programmer.

System programmer response: Contact your IBM Support Center.

Classes: 40.

EVE841I *procname : text.*

Explanation: The CEMT command listed in *text* has been requested from NetView through the program-to-program interface.

System action: None.

Explanation: The CEMT command listed in the preceding EVE841I message is not supported by EVESCEMT, for example, CEMT PERFORM SHUTDOWN.

System action: Ignore CEMT command. If the CEMT command was requested with a converse request (TYPE=C), return a NACK response with message EVE842E as NACK text.

Operator response: None.

EVE842E *procname : Unsupported CEMT command.*

Error Codes

CICS error codes start with the following letters:

- A** These are standard CICS error codes. Refer to *CICS Messages and Codes*.
- C** These are CICS Automation program-to-program interface error codes. See below.

C001

Module: EVESCEMT, EVESPPIC, EVESPPIP, and EVESPPIS.

Explanation: Someone attempted to invoke a CEMT command or to start or stop the program-to-program interface by entering the corresponding transaction name from a terminal, or someone entered the transaction name for the long-running program-to-program interface transaction from a terminal.

System action: Terminate the transaction abnormally.

Operator response: None.

System programmer response: Educate your users that they should not alter these transactions.

System action: Issue message EVE171E. Set return code 8 for caller.

Operator response: None.

System programmer response: Start PPI TASK.

C002

Module: EVESPPIP and EVESPPIS.

Explanation: The address of the program-to-program interface Intercommunication Area (ICA) was corrupted, probably because some other program altered unprotected storage in the CICS address space.

System action: Issue message EVE171E. Dump CICS and transaction storage.

Operator response: None.

System programmer response: Analyze the dump to find the cause of the error, then correct the problem.

C004

Module: EVESCCCI.

Explanation: EVESCCCI has not been invoked via EXEC CICS LINK.

System action: Issue message EVE171E. Terminate transaction abnormally.

Operator response: None.

Programmer response: Use EXEC CICS LINK to invoke EVESCCCI.

System programmer response: None.

C003

Module: EVESCCCI.

Explanation: A program-to-program interface request was issued while the program-to-program interface was not active.

C005

Module: EVESCCCI.

Explanation: Incorrect parameter list passed to EVESCCCI.

System action: Issue message EVE171E. If the length of the parameter list is incorrect, terminate transaction abnormally; otherwise return to caller with return code 12.

Operator response: None.

Programmer response: Correct the parameter list passed via the COMMAREA on the LINK to EVESCCCI.

System programmer response: None.

C006

Module: EVESPPIC.

Explanation: The buffer containing a program-to-program interface request that was sent from NetView to CICS was incorrect. This is caused by:

- An incomplete program-to-program interface buffer header.
- An incorrect request type (not C, S, R, A, N).
- An inconsistent length specification.
- An incorrect program-to-program interface sender identification. This is caused by a mismatch between the RECEIVERID specifications in the EVENTASK and EVESPINM initialization members.

System action: Issue message EVE171E. Dump transaction storage.

Operator response: None.

System programmer response: If caused by a mismatch between EVENTASK and EVESPINM, correct the mismatch error. Otherwise, contact your IBM Support Center because this problem is caused by a CICS Automation logic error.

C007

Module: EVESCEMT.

Explanation: A CEMT request was sent over the program-to-program interface from NetView to CICS. The request server found that the INTDSECT block was incorrect.

System action: Issue message EVE171E. Dump transaction storage. Terminate transaction abnormally.

Operator response: None.

System programmer response: This problem is caused by a CICS Automation logic error. Contact your IBM Support Center.

C008

Module: EVESCEMT, and EVESPPIC.

Explanation: Return code 12 was received from EVESCCCI because the OUTDSECT block was incorrect.

System action: Issue message EVE171E. Dump transaction storage. Terminate transaction abnormally.

Operator response: None.

System programmer response: This problem is caused by a CICS Automation logic error. Contact your IBM Support Center.

C009

Module: EVESPPIC

Explanation: Transaction could not be started successfully.

System action: Issue message EVE181E. Dump transaction storage.

Operator response: None.

System programmer response: If non-terminal transaction security is not active in CICS, ensure that this function is disabled in CICS Automation by specifying USERID=NO in the CICS PPI initialization member EVESPINM. For further information on this member, refer to *IBM Tivoli System Automation for z/OS Product Automation Programmer's Reference and Operator's Guide*.

In addition, make sure that there are LMT definitions for this CICS system.

If these requirements are fulfilled, contact your IBM Support Center.

C011

Module: EVESTISP and EVESPPIC.

Explanation: CICS WRITEQ TS failed.

System action: Issue message EVE171E. Dump transaction storage. Terminate transaction abnormally.

Operator response: None.

System programmer response: This problem is caused by a CICS Automation logic error. Contact your IBM Support Center.

C012

Module: EVESCCCI, EVESPPIC, EVESPPIP, and EVESPPIS.

Explanation: CICS READQ TS failed.

System action: Issue message EVE171E. Dump transaction storage. Terminate transaction abnormally.

Operator response: None.

System programmer response: This problem is caused by a CICS Automation logic error. Ensure that the PLTPI program EVESTISP has been executed successfully. Contact your IBM Support Center.

C013

Module: EVESTISP, EVESPPIC, and EVESPPIS.

Explanation: CICS START failed.

System action: Issue message EVE171E. Dump CICS tables.

Operator response: None.

System programmer response: Analyze the dump to find the cause of the failure and correct the problem. Ensure that the transaction that failed to start and the program to be invoked when the transaction starts have been defined and enabled. Also make sure that

the program resides in a DFHRPL library.

C014

Module: EVESCEMT.

Explanation: CICS RETRIEVE failed.

System action: Issue message EVE171E. Dump transaction storage. If the CEMT command was requested through a converse request (TYPE=C), return a NACK response with message EVE842E as NACK text. Terminate transaction abnormally.

Operator response: None.

System programmer response: This problem is caused by a CICS Automation logic error. Contact your IBM Support Center.

C015

Module: EVESCCCI, EVESCEMT, EVESPERR, EVESPPIC, and EVESPPIS.

Explanation: CICS LINK or LOAD failed.

System action: Issue message EVE171E. Dump transaction storage and CICS tables. Terminate transaction abnormally.

Operator response: None.

System programmer response: Analyze the dump to find the cause of the failure, then correct the problem. Ensure programs are defined and enabled and available in a DFHRPL library.

C017

Module: EVESCCCI, EVESCEMT, and EVESPPIC.

Explanation: CICS GETMAIN failed.

System action: Issue message EVE171E. Terminate transaction abnormally.

Operator response: None.

System programmer response: Increase the region size.

C018

Module: EVESCCCI.

Explanation: CICS FREEMAIN failed, or, possibly, a storage violation has occurred.

System action: Issue message EVE171E. Dump transaction storage.

Operator response: None.

System programmer response: Analyze the dump to find the cause of the failure, then correct the problem.

C019

Module: EVESCCCI.

Explanation: CICS DELAY failed.

System action: Issue message EVE171E. Dump transaction storage. Set return code 16 for caller.

Operator response: None.

System programmer response: Analyze the dump to find the cause of the failure, then correct the problem.

C020

Module: EVESPPIC.

Explanation: CICS CANCEL failed.

System action: Issue message EVE171E. Dump transaction storage. Terminate transaction abnormally.

Operator response: None.

System programmer response: Analyze the dump to find the cause of the failure, then correct the problem.

C117

Module: EVESTISP.

Explanation: OS GETMAIN failed.

System action: Issue message EVE171E. Dump transaction storage.

Operator response: None.

System programmer response: Analyze the dump to find the cause of the failure. then correct the problem. Increase the region size or amend SIT parameters to allow for sufficient OSCOR.

C2nn

Module: EVESCCCI and EVESPPIC.

Explanation: A program-to-program interface problem has occurred. *nn* identifies the program-to-program interface return code.

Problem determination: Program-to-program interface request return codes *retcode* are documented in *Application Programming Guide: Program-to-Program Interface*.

System action: For *nn* = 20, 22, 23, 25, 31, 33, 36, 40, and 90, issue message EVE171E and dump transaction storage. For all other values of *nn*, no message is issued.

Operator response: None.

System programmer response: Determine the cause of the error. If the error is caused by a CICS Automation logic error, contact your IBM Support Center.

C950

Module: EVESCEMT.

Explanation: The TWA size of the COMT transaction is too small.

System action: Issue message EVE171E. Terminate transaction abnormally.

Operator response: None.

System programmer response: Ensure that the TWO size of the COMT transaction is greater than or equal to the TWA size of the CEMT transaction (512 bytes).

C960

Module: EVESCCCI.

Explanation: A TS item passed to EVESCCCI by EVESPPIC when a response is received from NetView has an incorrect length

System action: Issue message EVE171E. Dump transaction storage. Set return code 16 for the caller.

Operator response: None.

System programmer response: This problem is caused by a CICS Automation logic error. Contact your IBM Support Center.

C961

Module: EVESCCCI.

Explanation: RQE chain corrupted.

System action: Issue message EVE171E. Dump transaction storage. Set return code 16 for the caller.

Operator response: None.

System programmer response: This problem is caused by a transaction that altered storage in the CICS address space, or by a CICS Automation logic problem. Determine the cause of the problem. If the problem is caused by a CICS Automation logic problem, contact your IBM Support Center.

C971

Module: EVESPPIC.

Explanation: RQE chain corrupted.

System action: Issue message EVE171E. Dump transaction storage. Ignore all RQEs that currently exist, which may result in several EVE176E messages later.

Operator response: None.

System programmer response: This problem is caused by a transaction that altered storage in the CICS address space, or by a CICS Automation logic problem. Determine the cause of the problem. If the problem is

caused by a CICS Automation logic problem, contact your IBM Support Center.

C980

Module: EVESPERR.

Explanation: Undefined error code.

System action: Issue message EVE171E. Dump transaction storage.

Operator response: None.

System programmer response: This problem is caused by a CICS Automation logic error. Contact your IBM Support Center.

C990

Module: EVESPMMSG.

Explanation: Undefined message ID.

System action: Issue message EVE171E. Dump transaction storage.

Operator response: None.

System programmer response: This problem is caused by a CICS Automation logic error. Contact your IBM Support Center.

Chapter 4. Messages EVI022I to EVI844A (IMS automation)

Messages generated by IMS automation are built from the DFS, and DXR messages issued by IMS/VS, in addition to the AVM, IOS, and IEF messages issued by the MVS Availability Manager, the Input/Output Supervisor, and the MVS Job Scheduler respectively.

IMS automation supports the NetView Help Message command (HM). By entering HM plus the message ID on a NetView or IMS automation command line, you can access the message documentation online.

Important note for system programmers:

The IMS messages use message classes 0, 40, 43, 44, 46, 47 and 60. This means that the notify operator entries in the control file need to have these message classes defined in the CLASS= parameter.

EVI022I **Bad return code** *retcode* from *routine*.

Explanation: IMS automation issued an internal command or macro call and received a non-zero return code. This may indicate an internal error.

The variable *retcode* shows the return code.

The variable *routine* shows the command, module or macro name that issued the return code.

System action: Processing is terminated.

Operator response: Contact your system programmer.

System programmer response: Check the return code of the indicated routine and any associated messages on the system log and NetLog. Contact your IBM Support Center for further assistance.

Classes: None.

EVI030I *resource* not defined to System Automation

Explanation: One of the SA z/OS commands was entered against a dependent region that is not defined to SA z/OS.

The variable *resource* is the name of the resource that is unknown to SA z/OS.

System action: The system does not process this action and stops processing multiple actions.

Operator response: Correct the action character if it was entered incorrectly or do not request SA z/OS actions for non-SA z/OS regions. The exceptions are START and STOP, which will issue the /START REGION and /STOP REGION commands for the dependent region.

Classes: None.

Module: EVIRYDPO

EVI031I *resource* cannot process action &2

Explanation: This message may be issued for the following reasons:

1. One of the actions entered against a dependent region cannot be processed by that dependent region. This is usually due to an /ASSIGN against a dependent region that has no classes.
2. One of the actions entered against a resource in the TCO list cannot be processed by the resource type. This is usually due to issuing a LOAD request against the LTERM, or issuing a START or STOP request against a MEMBER resource.

The variable *resource* is the name of the resource.

The variable &2 is the action.

System action: The system does not process this action and stops processing multiple actions.

Operator response: Use the appropriate line command against the appropriate resource type.

Classes: None.

Module: EVIRYTCO

EVI326I **No TCO definitions found.**

Explanation: When attempting to issue the requested command(s) for the TCO option selected by the operator, the process was unable to find any entries in the control file for this request.

System action: The request is denied.

Operator response: Notify your system programmer

EVI350I • EVI844A

that the function selected is not defined in the control file.

System programmer response: Verify that the control file contains the required entries for the function selected by the operator.

Operator response: Review the netlog for errors. Report the problem to your system programmer.

System programmer response: Browse the netlog to determine the cause of the error.

Classes: 0, 40, 60.

EVI350I TCO HAS NOT BEEN INITIALIZED, REQUEST REJECTED

Explanation: A TCO function has been attempted and TCO is not supported by this IMS at this time.

System action: The request function is not executed.

Operator response: Ensure that the IMS that is associated with the request is active. Use option 1, inquiry, from the IMS automation panels to find out. If it is active, check with your system programmer to ensure that this IMS has been setup to include IMS TCO support.

System programmer response: Ensure that the IMS associated with the TCO request has been setup to include IMS TCO support. For example, IMS uses 2 logical terminals, as well as a DD statement in the IMS startup PROC to support TCO requests to start, stop, and load a TCO member.

EVI421I Request ignored, 'IMS' is not up.

Explanation: The requested action is invalid because IMS is not up.

Operator response: Startup the IMS subsystem. After a successful startup, retry the request.

EVI706E Syntax error - parameter "*parmtype*" = "*parmvalue*". *msgtext*

Explanation: Invalid value specified in the Automation control file member for keyword (refer to PARAMETER value in EVI706E message).

System action: The CLIST is terminated.

Operator response: Notify your system programmer.

| **System programmer response:** Correct the value for
| the *parmtype* keyword in the policy and refresh the
| configuration.

Classes: 40, 47, 60.

EVI844A *subsystem* RESYNC FAILURE - UNABLE TO COMPLETE *command* COMMAND

Explanation: This message indicates that a problem occurred during IMS resynchronization processing.

The variable *subsystem* shows the name of the IMS subsystem.

The variable *command* shows the failed command.

System action: The resynchronization processing may be incomplete.

Chapter 5. Messages EVJ000I to EVJ440I (TWS Automation)

EVJ000I *date time module* ENTRY/EXIT*parms/*
 RC=*rc*

Explanation: When the trace option is activated, TWS Automation writes this data to the log for entry and exit tracing. The date is shown in the format mm/dd/yyyy.

Operator response: None.

EVJ001I *command:* STARTED, PARMs=*parameters*

Explanation: Command processor has been entered with the parameters shown. This is an informational message issued for audit purposes.

Destination: Command processor name, parameters.

System action: None.

Operator response: None.

System programmer response: None.

EVJ002I *command:* COMPLETED
 SUCCESSFULLY, PARMs=*parms*

Explanation: Command processor has completed successfully. This is an informational message issued for audit purposes.

Destination: Command processor name, parameters.

System action: None.

Operator response: None.

System programmer response: None.

EVJ003E *command* UNABLE TO PROCESS
 PARMs=(*parameters*), REASON=*reason*

Explanation: OPC has initiated a request, but it cannot be processed due to the reason shown. This message will normally be preceded by a message giving specific details of the failure.

Destination: Request parameters, reason text.

System action: The request is canceled.

Operator response: Review log for other messages.

System programmer response: Resolve any system or definition problems, then reset the request.

EVJ005E *module* UNABLE TO OBTAIN *parameter*

Explanation: The module indicated tried to obtain the value of a parameter from product globals, but the value was null or not as expected.

Destination: Module name and failing parameter

System action: The module ceases processing and returns a non-zero return code to its calling routine.

Operator response: Review log for other messages and notify your system programmer.

Determine the source of the failure, typically an incorrect control file value or failure of the product initialization routines (beginning with EVJEAIC) to run, and correct it. Call IBM Service if you require assistance.

EVJ006E *cmdname* COMMAND FAILED FOR
 module: RETURN CODE=*rc*
 PARMs=*parms*

Explanation: A command issued within this command processor received an invalid return code.

The variable *cmdname* shows the name of the command that failed.

The variable *module* shows the name of the calling command processor.

The variable *rc* shows the return code from the failing command.

The variable *parms* shows the parameters passed to the failing command.

System action: The command processor terminates and logs an error message.

Operator response: Review the error messages in the NetView and MVS logs.

System programmer response: Review the associated error messages in the NetView and MVS logs. If necessary, correct definition problems and retry the operation.

EVJ007E SUBSYSTEM *subsys* IN INVALID
 STATUS *stat* FOR REQUEST (*request*)

Explanation: TWS Automation has attempted to execute the request for the indicated subsystem, but was unable to complete because the subsystem was in an invalid status. For example, trying to start a subsystem that is already up, or stop a subsystem that is already down.

Destination: Command processor name, subsystem, subsystem status, TWS Automation request type.

System action: The operation in OPC is posted in error with a U001 status.

Operator response: Review NetView log for prior activity for the subsystem.

EVJ008E • EVJ041E

Check for manual intervention with the subsystem, such as starts and stops performed outside of automation control. Also, for shutdowns, specify RESTART=CTL so that subsystems will not be restarted by NetView initializations.

EVJ008E OPERATION ALREADY IN PROGRESS FOR *subsys*, REQUEST=(*reqparms*)

Explanation: TWS Automation has attempted to execute the request for the indicated subsystem, but was unable to process because a previous request is still outstanding for the subsystem, or the previous request may not have completed successfully.

The variable *subsys* shows the name of the subsystem involved.

The variable *reqparms* shows the details of the previous request.

| **System action:** The new operation, indicated by
| additional message EVJ003E is posted in error with a
| U005 status.

Operator response: Review NetView log for prior activity for the subsystem.

| Check for previous errors in the log, or review TWS
| and workstation definitions. When problems are
| resolved, reset the TWS status file record using
| OPCAQRY and rerun the operation.

EVJ009E NO RESPONSE FROM SYSTEM *domainid* WHILE SENDING REQUEST (*request*)

Explanation: TWS Automation has attempted to forward the request to the remote domain, but did not receive an acknowledgment from that system.

Destination: Domain ID, request parameters.

System action: The operation in OPC is posted in error with an S999 status.

Operator response: Check status and connectivity of the system in question. If the problem can be resolved, rerun the failing operation.

Your gateway to the remote NetView is blocked, or the remote system may be extremely busy. If the former, you should resolve the gateway problem. If the latter you should schedule your operations on this remote NetView farther apart.

EVJ010I *variable data*

Explanation: This message is used internally to transfer data between systems.

Destination: Date, time, message text.

System action: None, this is a normal message.

Operator response: None.

EVJ011I *date time OPC message text*

Explanation: While attempting to communicate with OPC, an operator initiated request to display or modify data has failed. The message text is the OPC message received. This message is normally preceded by another message, indicating the TWS Automation module that was attempting to request data from OPC.

Destination: Date, time, message text.

System action: None.

Operator response: Review message in the appropriate OPC/ESA or OPC/A manual.

Correct problems such as incorrect definitions of OPC subsystem name.

EVJ025E INVALID VALUE "*value*" FOR *keyword*

Explanation: The parameter you have supplied is not valid.

System action: None.

Operator response: Try again using the correct parameters.

EVJ026E MISSING PARAMETER *parameter*

Explanation: The parameter is required, but was omitted.

System action: None.

Operator response: Try again using the correct parameters.

EVJ029I *command* ENCOUNTERED UNEXPECTED EVENT *event_name*

Explanation: The indicated command was issued, but the response received was not as desired. For example, the command may have timed out, or the procedure may have been canceled while waiting for a message.

System action: None.

Operator response: Investigate the cause of the failure.

EVJ040I PROCESSING COMMANDS

Explanation: The TWS Batch Command Processor is starting.

System action: None.

Operator response: None.

EVJ041E POSITIONAL PARAMETER *name* MUST BE SPECIFIED

Explanation: The parameter mentioned in the message is required and must be specified.

System action: Batch command processing aborts and

the batch job ends with RC=16.

Operator response: Correct the EVJRYCMD command and retry.

EVJ042E **PARAMETER** *name* **ERROR, VALUE NOT NUMERIC**

Explanation: The value specified in parameter *name* is not numeric. The value must be a number.

System action: Batch command processing aborts and the batch job ends with RC=16.

Operator response: Correct the EVJRYCMD command and retry.

EVJ043E **PARAMETER** *name* **ERROR, information**

Explanation: The value of the named parameter is incorrect.

The variable *name* represents the parameter in error.

The variable *information* contains the detail of the error.

System action: Batch command processing aborts and the batch job ends with RC=16.

Operator response: Contact your system programmer.

System programmer response: Ensure that the PPI destination named is 1 to 8 characters in length and consists of alphanumeric characters.

EVJ044E **PARAMETER TRACE ERROR, INVALID VALUE - SEE THE REXX TRACE COMMAND**

Explanation: Specification of the TRACE= parameter requires a value consistent with the REXX Trace command.

System action: Batch command processing aborts and the batch job ends with RC=16.

Operator response: Contact your system programmer.

System programmer response: Check with your IBM Support Center. Re-specify the TRACE= parameter with the correct value.

EVJ045I **OPENING PPI INTERFACE FROM**
source TO destination

Explanation: Command Processing is about to send commands to the SA z/OS PPI command processor. This message is issued when the PPI interface is being opened.

The variable *source* is the PPI Sender id used for this batch job.

The variable *destination* is the PPI receiver name of the command sever. It is specified with the SERVER= parameter.

System action: Processing continues.

Operator response: None.

System programmer response: None.

EVJ046E **UNABLE TO OPEN PPI FOR**
receiver_name

Explanation: An attempt to open the PPI receiver named *receiver_name* failed. This message is followed by EVJ047I that describes why the attempt to open failed.

System action: Command processing is aborted and the batch job will terminate with return code 20.

Operator response: Contact your system programmer.

System programmer response: Check message EVJ047I to find the probable cause of the error.

EVJ047I *dsiphone_error* [, **PPI reason** *ppi_error*]

Explanation: This message provides details of the NetView DSIPHONE error that caused processing to fail. It also gives details of the PPI error reason if that is feasible.

The DSIPHONE error responses are:

- DSIPHONE called without arguments.
- Too many parameters for this request type.
- Too few parameters for this request type.
- Invalid request type.
- REXX stem or variable name too long.
- REXX variable operation failed.
- PPI receiver name is too long.
- PPI request failed.
- Stem's .0 element invalid.
- Too many elements in stem.
- First line of MLWTO not control line.
- Invalid WAIT interval specified.
- Argument 6 is not 'APPEND' or blank.
- PPI RECEIVE timed out.
- Invalid MLWTO line type attribute.
- Invalid MLWTO attribute length.
- Unable to obtain storage.
- Invalid attempt to call DSIPHONE from NetView.
- Invalid attempt to call DSIPHONE.
- Internal error.
- An unknown error occurred.

The PPI error responses are:

- The specified receiver is not active.
- The PPI is available.
- The receiver program is active.
- The receiver program is inactive.

- The receiver program is already active.
- The receiver ECB is not zero.
- The request code is not defined.
- The program is not in Primary Addressing mode.
- The program is not authorized.
- The PPI interface is not active.
- The ASCB address is not correct.
- The receiver program is not defined.
- No active SSI for PPI found.
- No data buffer in receiver buffer queue.
- Receiver buffer is not large enough.
- No NetView storage available.
- Buffer length is not valid.
- Receiver queue is full.
- ESTAE recovery cannot be established.
- Sender or Receiver ID not valid.
- Processing error has occurred.
- System abend occurred.
- User abend occurred.
- AIFR or the input length was not valid.
- Could not identify the data as a message.
- Incomplete multiline message was discarded.
- Illegal alert forwarding loop was detected.
- Specified target type does not support the data.
- User is not authorized.
- An unknown error occurred.

Numeric codes may also be present for either the DSIPHONE code or the PPI code.

System action: Command processing is aborted and the batch job terminates with return code 20.

Operator response: Contact your system programmer.

System programmer response: For some of the PPI error responses action can be taken to correct the error:

- “PPI receiver not defined” is usually due to an incorrect SERVER= specification.
- “PPI interface not active” or “No active SSI” is usually due to no NetView SSI subsystem having started.

Most of the other reasons will require you to contact your IBM Support Center.

EVJ048I EXECUTING COMMAND

Explanation: Command processing is about to send a command to SA z/OS for processing.

System action: Processing continues.

Operator response: None.

System programmer response: None.

EVJ049E UNABLE TO SEND TO *destination* FOR *source* EVJ049E UNABLE TO RECEIVE FROM *destination*

Explanation: PPI processing failed to send data from the source to the destination PPI receivers, or receive data from the destination PPI receiver.

The variable *source* is the PPI Sender Id for this batch job.

The variable *destination* is the PPI receiver name of the command server and is specified with the SERVER= parameter.

This message is followed by EVJ047I to describe the problem in more detail.

System action: Processing is aborted and the batch job is terminated with return code 20.

Operator response: Contact your system programmer.

System programmer response: Refer to the information in the following EVJ047I message and take any appropriate action.

EVJ052E PPI INTERFACE FAILED TO CLOSE FOR *source*

Explanation: The PPI interface did not successfully close after command processing was completed.

The variable *source* is both the PPI receiver name of the batch job and the job name of the batch job.

System action: Command processing is aborted and the batch job is terminated with return code 20.

Operator response: None.

System programmer response: Possibly no action. Check whether commands that were issued by the batch job executed correctly on the NetView by examining the NetView log.

EVJ053I BATCH COMMAND RECEIVED FROM *userid* JOBNAME *jobname*

Explanation: An OPC batch command has been received from a batch job and is to be executed on this NetView.

The variable *userid* shows the user ID of the user who submitted the batch job.

The variable *jobname* is the job name of the batch job.

System action: The command specified in message EVJ054I is processed.

Operator response: None.

System programmer response: None.

EVJ054I BATCH COMMAND IS *command-text*

Explanation: An OPC batch command has been received from a batch job and is to be executed on this NetView. This message logs the command to be executed.

The variable *command-text* shows the batch command.

System action: The command is processed.

Operator response: None.

System programmer response: None.

EVJ055I PIPE PPI FOR *receiver* FAILED.
RC=*returncode* - *description*

Explanation: A PIPE PPI command failed when attempting to communicate with the receiver.

The variable *returncode* shows either the return code from the PIPE command or the return code from the PPI stage.

The variable *description* shows a description of the error.

System action: The output from the current command received from the batch job will not be sent to the batch job. However, the command may have run successfully. The batch job will probably time out or produce some other error.

Operator response: For recoverable errors such as errors with the receiver, resubmit the batch job if you are allowed to rerun the failed batch job. For all other errors notify your system programmer.

System programmer response: Report the error to your service representative.

EVJ056E Command return code *nn* greater than MAXRC *mmm* - Command Processing Aborted.

Explanation: A command sent to the SA z/OS agent for processing has executed and returned with a code greater than the maximum acceptable return code.

The variable *nn* shows the return code from the command.

The variable *mmm* shows the maximum acceptable return code from the MAXRC parameter of the EVJRYCMD command.

System action: The batch interface will stop sending commands to the SA z/OS Agent and will terminate. The return code for the batch job step will be determined by the HIGHRC parameter.

Operator response: For recoverable errors, such as errors with the receiver, resubmit the batch job if you are allowed to rerun the failed batch job. For all other errors notify your system programmer.

System programmer response: If this is not a

user-defined situation, report the fault to your service representative.

EVJ060I Executing command for ADID=*name*
WSID=*identifier* OPNUM=*nnn*
IATIME=*timestamp* >> *command_text*

Explanation: A command has been received from TWS and is to be executed on this NetView. This message logs the command to be executed. Note that the command might be truncated if it is too long.

The variable *name* shows the application description identifier as defined in TWS.

The variable *identifier* shows the workstation identifier.

The variable *nnn* shows the TWS operation number.

The variable *timestamp* shows the input arrival time as determined by TWS.

The variable *command_text* shows the command.

System action: The command is processed.

Operator response: None.

System programmer response: None.

EVJ061I Command for ADID=*name*
WSID=*identifier* OPNUM=*nnn* failed
IATIME=*timestamp* RC=*nn*

Explanation: A command received via the TWS interface was executed but reported a non-zero return code. See the preceding message EVJ060I for the command in error.

The variable *name* shows the application description identifier as defined in TWS.

The variable *identifier* shows the workstation identifier.

The variable *nnn* shows the TWS operation number.

The variable *timestamp* shows the input arrival time as determined by TWS.

The variable *nn* shows return code from the command.

System action: None.

Operator response: None.

System programmer response: None.

EVJ062I REQUEST for *wsname* *adname* *opnum*
iatime purged

Explanation: While performing a configuration refresh, System automation detected that a deleted resource was involved in TWS request. Since the resource is no longer under control of SA z/OS the request is removed from the internal control structure.

Variable *wsname* identifies the workstation name involved.

Variable *adname* identifies the application name involved as defined in TWS

Variable *opnum* specifies the operation number

Variable *iatime* specifies the input arrival time assigned to the request.

Operator response: None.

System action: Processing continues.

EVJ063I **Command for ADID=*name*
WSID=*identifier* OPNUM=*nnn*
IATIME=*timestamp* finished.
ERRCODE=*errcode***

Explanation: A command received via the TWS interface was executed and finished. See the preceding message EVJ060I for command that was executed.

The variable *name* shows the application description identifier as defined in TWS.

The variable *identifier* shows the workstation identifier.

The variable *nnn* shows the TWS operation number.

The variable *timestamp* shows the input arrival time as determined by TWS.

The variable *errcode* shows the error code for the operation. The error code is only present when the operation failed.

System action: None.

Operator response: None.

System programmer response: None.

EVJ100I **EVJ07001: ENTERING SA z/OS
TWS/OPC EXIT**

Explanation: TWS has now called the EVJ07001 exit.

Destination: None.

System action: None.

Operator response: Information message.

System programmer response: Information message.

EVJ101I **EVJ07001: DATA SENT=*data***

Explanation: This message logs the data sent to NetView across the NetView PPI in the system log.

Destination: Data passed.

System action: None.

Operator response: Information message.

System programmer response: Information message.

EVJ102I **EVJ07001: PPI REQUEST *nm* FAILED,
RC=*rc***

Explanation: EVJ07001 was unable to complete a NetView PPI request. See the NetView PPI manual for more information on this request type and return code.

Destination: Request type and return code.

System action: TWS operation will be set to E status.

Operator response: Review error message in log.

System programmer response: Review error message in log.

EVJ103I **EVJ07001: GETMAIN FOR WORKAREA
FAILED**

Explanation: EVJ07001 was unable to do a GETMAIN for work area storage.

Destination: None.

System action: EVJ07001 will put the TWS operation in E status.

Operator response: Report the problem to your system programmer.

System programmer response: Review NetView storage allocation and usage.

EVJ104I **EVJ07001: OPC BAD OPERDESC: *data***

Explanation: EVJ07001 has found the contents of the operation description field to be invalid. Message EVJ105I will list the actual error data.

Destination: TWS application name.

System action: EVJ07001 will put the TWS operation in E status.

Operator response: Inform TWS Scheduler of possible definition error.

System programmer response: None.

EVJ105I **EVJ07001: OPC DATA *data***

Explanation: EVJ07001 has found this data to be invalid.

Destination: Data.

System action: EVJ07001 will put the TWS operation in E status.

Operator response: Inform TWS Scheduler of possible definition error.

System programmer response: None.

EVJ106I *modname1: modname2* STATUS UPDATE
FAILED, RC=*rc*

Explanation: The module listed was unable to update operations status using DRK/EQQUSINT.

Destination: Issuing module and status update module.

System action: WTO error message is issued.

Operator response: Inform TWS schedulers and system programmers.

System programmer response: Review system and TWS/OPC message logs.

EVJ107I **EVJ07001: TWS/OPC FEATURE EXIT ENDED**

Explanation: EVJ07001 has successfully ended.

Destination: None.

System action: None.

Operator response: Information message.

System programmer response: Information message.

EVJ108I *module-name* UNABLE TO LOAD
MODULE CNMCNETV

Explanation: This message is issued from the TWS status change exit, (EQQUX007/DRKUX007), when it cannot load the NetView PPI interface.

Destination: Issuing module name.

System action: Processing for this request halts. The operation will be posted in error with a return code of UNTV.

Operator response: Notify your system programmer.

System programmer response: This message most likely indicates an installation error. You must make the NetView PPI module accessible to TWS, either by STEPLIB or putting the NetView load library, CNMLINK, in LINKLST.

EVJ109I **EVJ07001: LEAVING SA Z/OS TWS/OPC EXIT**

Explanation: EVJ07001 has successfully ended.

Destination: Module name and successful command name.

Operator response: Information message.

System programmer response: Information message.

EVJ110I *modulename* UNABLE TO LOAD
MODULE *target-module-name*

Explanation: This message is issued from an TWS command processor, which is unable to find or load an TWS interface module.

Destination: Issuing module name and target module name.

System action: Processing for this request halts.

Operator response: Notify your system programmer.

System programmer response: This message most likely indicates an installation error. You must make the TWS interface module accessible to NetView, either by STEPLIB (preferred) or by putting the TWS load library in LINKLST.

EVJ120I *applid iatime opnum job status wsname*
errcode abcode usrcode job#

| **Explanation:** A TWS operation has been put into TWS
| error status or has been reset from TWS error status.
| This causes a Status Display Facility update and an
| NMC update to occur. For an operation changing to
| error status the update will add an entry to SDF and
| NMC while an operation changing from error status
| will remove an entry from SDF and NMC. SDF entries
| are added to the 'OPC Automation Application in
| Error' panel (OPCERR). When enabled, alerting for the
| specified event notification targets will occur.

The variable *applid* shows the TWS application name.

The variable *iatime* shows the TWS input arrival time.

The variable *opnum* shows the TWS operation number.

The variable *job* shows the TWS job name.

The variable *status* shows the TWS status.

The variable *wsname* shows the TWS workstation for this operation.

The variable *errcode* shows the TWS error code.

The variable *abcode* shows the TWS abend code.

The variable *usrcode* shows the TWS user code.

The variable *job#* shows the TWS job number.

| **System action:** SDF and NMC are updated according
| to the data displayed in the message and when
| enabled, an alert for event notification targets will be
| generated.

Operator response: None.

System programmer response: None.

EVJ121I **Application** *jobname opno* **NOT RESET
BECAUSE TIME LIMIT EXCEEDED**

Explanation: This operation was found to be in error status in OPC with error code, UNTV, indicating that the NetView interface was unavailable at the time it came READY. Now that the interface is available, the status was not reset because more time has elapsed since the failure than is permitted by the OPRESET value in the control file (see ENVIRON OPCA0).

Destination: Application name, Job name, Operation number.

System action: Operation remains in error status.

Operator response: Manually reset operation when appropriate. OPCACMD may be used to do this.

System programmer response: Determine whether the OPRESET value should be increased.

EVJ122E **UNKNOWN TWS API SEGMENT
segment ENCOUNTERED**

Explanation: When the INGTWS command is decoding data returned from TWS via the API, an unknown segment name was encountered.

The variable *segment* shows the name of the segment that is unknown to the command.

System action: Data from the TWS API is not successfully decoded. The command will present data that has been decoded up to the point of the failure.

Operator response: Notify your system programmer.

System programmer response: Report the error to your service representative.

EVJ123I **[DEFAULT SAFE|UPDATE
PARAMETER] INVALID
MODIFICATION PARAMETERS
FOLLOW:**

Explanation: For an INGTWS REQ=MOD request an invalid update parameter was encountered that the command could not process. The source of the parameter in error is either the Default Safe for the DEFAULT SAFE version of the message or the UPDATE command parameter for the UPDATE PARAMETER version of the message. Message EVJ124I follows this message with a list of up to five parameter names that are in error.

System action: Processing stops.

Operator response: Correct the incorrect parameter and reissue the command.

System programmer response: None.

EVJ124I *parm1 parm2 parm3 parm4 parm5*

Explanation: This message follows message EVJ123I and specifies up to five parameter names that are in error.

System action: None.

Operator response: Correct the incorrect parameter/s and reissue the command.

System programmer response: None.

EVJ125I *subsystem* **IS NOT AN ACTIVE
CONTROLLER OR A TRACKER**

Explanation: This message is issued when INGOPC cannot resolve the resource specification to an active controller or to an available tracker.

System action: None.

Operator response: Correct the incorrect parameter (or parameters) and reissue the command. Make sure that the resource specification will resolve to an active Available Controller or an Available Tracker.

System programmer response: None.

EVJ201I **EVJTOPPI: TASK IS ACTIVE**

Explanation: NetView optional subtask initialization has completed successfully.

Destination: None.

System action: None.

Operator response: Information message.

System programmer response: Information message.

EVJ202I **EVJTOPPI: PPI TO EVJTOPPI
INITIALIZED**

Explanation: NetView optional subtask has initialized its NetView PPI queue.

Destination: None.

System action: None.

Operator response: Information message.

System programmer response: Information message.

EVJ203I **EVJTOPPI: PPI TO EVJTOPPI
DEACTIVATED**

Explanation: NetView optional subtask has deactivated the NetView PPI queue.

Destination: None.

System action: None.

Operator response: Information message.

System programmer response: Information message.

EVJ204I **EVJTOPPI: TASK IS TERMINATED**

Explanation: NetView optional subtask has terminated successfully.

Destination: None.

System action: None.

Operator response: Information message.

System programmer response: Information message.

EVJ205E **EVJTOPPI: PPI REQUEST *mm* FAILED, RC=*rc***

Explanation: EVJTOPPI was unable to complete a NetView PPI request. Refer to the *NetView Application Programming Guide: PPI* for more information on the request type and return code.

Destination: Request type and return code.

System action: None.

Operator response: Notify your system programmer.

System programmer response: Review log for messages that may indicate the reason for the failure. A possible reason is that NetView is being shutdown.

Note: The OPC operation status will be unchanged. User intervention may be required.

EVJ206I **EVJTOPPI: PPI DATA FROM EVJTOPPI DISCARDED**

Explanation: EVJTOPPI received request data from a non-OPC subsystem. The data is logged in the NetView log, but is not passed to EVJESPVY.

Destination: None.

System action: None.

Operator response: Informational. Review message EVJ214I for data.

System programmer response: Informational.

EVJ207I **EVJTOPPI: DSIFRE FAILED FOR USER STORAGE**

Explanation: EVJTOPPI was not able to FREE user storage before terminating.

Destination: None.

System action: None.

Operator response: Report problem to system programmer.

System programmer response: Review error.

EVJ208I **EVJTOPPI: DSIFRE FAILED FOR QUEUED STORAGE**

Explanation: EVJTOPPI was not able to FREE queued storage before terminating.

Destination: None.

System action: None.

Operator response: Report problem to system programmer.

System programmer response: Review error.

EVJ209I **EVJTOPPI: DSIGET FAILED FOR USER STORAGE**

Explanation: EVJTOPPI was not able to GET user storage at initialization.

Destination: None.

System action: None.

Operator response: Report problem to system programmer.

System programmer response: Review NetView storage allocation and utilization.

EVJ210I **EVJTOPPI: DSILCS FAILED TRYING TO GET A SWB**

Explanation: EVJTOPPI was not able to GET a service work block.

Destination: None.

System action: None.

Operator response: Report problem to system programmer.

System programmer response: Review NetView storage allocation and utilization.

EVJ211I **EVJTOPPI: ENQ ERROR**

Explanation: EVJTOPPI was not able to perform an ENQUEUE.

Destination: None.

System action: None.

Operator response: Report problem to system programmer.

System programmer response: Check for multiple EVJTOPPI tasks or possible deadlock condition.

EVJ212I **EVJTOPPI: DEQ ERROR**

Explanation: EVJTOPPI was not able to perform a dequeue.

Destination: None.

EVJ213I • EVJ424E

System action: None.

Operator response: Report problem to system programmer.

System programmer response: Check for multiple EVJTOPPI tasks started, or a deadlock condition.

EVJ213I **EVJTOPPI: DSILCS FAILED TRYING TO FREE SWB**

Explanation: EVJTOPPI was not able to FREE a service work block.

Destination: None.

System action: None.

Operator response: Report problem to system programmer.

System programmer response: Contact support center.

EVJ214I **EVJTOPPI: xxxxxxxxxxxxxxxxxxxxxxxxx**

Explanation: This message is sent to the NetView log to document the discarded data.

Destination: Data.

System action: None.

Operator response: Informational.

System programmer response: Informational.

EVJ300I **EVJSAOPS: DRK/EQQUSINT POST WAS SUCCESSFUL**

Explanation: Post to OPC using DRK/EQQUSINT was successful.

Destination: None.

System action: None.

Operator response: Informational.

System programmer response: Informational.

EVJ302I *cmdname:* **INVALID KEYWORD SUPPLIED**

Explanation: Invalid KEYWORD was issued when invoking this command. Message EVJ303I will follow listing the KEYWORD and DATA in error.

Destination: Command name.

Operator response: Correct parameters used to invoke the command.

System programmer response: Correct the command and retry.

EVJ303I *cmdname:* **KEYWORD ERROR, KEYWORD(keyword) DATA(data)**

Explanation: This message gives the keywords and data that are invalid when this command was invoked.

Destination: Command name, keyword and data.

Operator response: Review parameters used to invoke the command.

System programmer response: Modify parameters accordingly.

EVJ410I **ADID JOBNAME WS OPNO S ERRC IA OPTXT**

Explanation: This is the first line of a multiple-line message. It displays the titles for message EVJ411I.

EVJ411I *applid jobname workstation opid status errcode IA_time text.*

Explanation: This is the second line of a multiple-line message. It displays the corresponding information from message EVJ410.

EVJ412I **END OF REQUEST.**

Explanation: This message indicates the end of the data.

EVJ413I **UNABLE TO FIND DATA FOR REQUESTED LIST**

Explanation: This message is issued from OPCALIST or OPCACMD and indicates that no application or operation in the current plan meets the combination of input parameters specified.

Destination: None.

System action: Processing for this request halts.

Operator response: Correct the input to the OPCALIST command or the OPCACMD panel. If it is not obvious which parameter is error specify fewer parameters and issue the command again, so that OPC may return a list of valid application occurrences in the current plan. If there are none go to the OPC dialog under ISPF and add an occurrence.

System programmer response: None.

EVJ424E **COULD NOT LOCATE OPC CONTROLLER - reason**

Explanation: The OPC Controller as defined in the OPCA PCS automation control file statement could not be located in either the local system or any of the systems in the local sysplex.

The *reason* variable (optional) shows the reason code.

System action: Processing is terminated.

Operator response: Investigate the cause of the failure. Further diagnostic messages may be available in the NetView log. Possible causes include:

- An error in the OPCA PCS statement in the automation control file.
- The OPCA PCS defines the controller to be running on another system.
- No OPC controller is running (UP status in SA z/OS) where specified.

EVJ425I NO SRSTAT ENTRY FOUND IN OPCA PCS DEFINITION FOR OPC AUTOMATION.

Explanation: The SRSTAT entry for the TWS Automation is not set.

```
OPCA PCS,
SRSTAT=YES|NO|ALL
```

System action: None.

Operator response: Report to your system programmer that the OPCA PCS statement in the automation control file for setting special resources status was not defined. This entry allows TWS Automation to set OPC special resources for the named job name in the OPCA OPCSSTAT definition.

System programmer response: Refer to *IBM Tivoli System Automation for z/OS TWS Automation Programmer's Reference and Operator's Guide* for further information.

EVJ430I ADID JOBNAME WS OPNO S ERRC IA OPTTEXT

Explanation: This is the first line of a multiple-line message. It displays the titles for message EVJ431E, as follows:

ADID	OPC Application ID
JOBNAME	MVS Job Name
WS	OPC Workstation
OPNO	OPC Operation Number
S	OPC Status
ERRC	OPC Error Return Code
IA[®]	OPC Input Arrival Time
OPTTEXT	OPC Operations Text

EVJ431E *adid jobname ws opno s errc ia optext*

Explanation: This is the continuation of a multiple-line message begun with EVJ430I.

Destination: The inserts are as follows:

adid	OPC Application ID
jobname	MVS Job Name
ws	OPC Workstation
opno	OPC Operation Number
s	OPC Status
errc	OPC Error Return Code
ia	OPC Input Arrival Time
optext	OPC Operations Text

EVJ440I *date day* WORK|FREE.

Explanation: This message tells you whether today is a work day or a free day in the OPC calendar. The date is shown in the format mm/dd/yyyy.

Codes Posted to TWS by TWS Automation

OAUT

Explanation: Indicates that the EQQUXSAZ exit was not able to pass the request to the local SA z/OS using the NetView PPI.

Rnnn

Explanation: The command passed to SA z/OS for execution failed. *nnn* is the return code from the command. RXXX is used when the return code is >999.

S998

Explanation: Job not defined to any active system in sysplex. A request to run an operation from TWS was received by TWS Automation. The control file mapped the workstation (NVxx) to SYSPLEX. All active SA z/OS domains in the local sysplex were queried

but no definition was found for the job name specified in the TWS request.

This may be the result of one or more systems (or their SA z/OS) being inactive or it may be a definition error. If all systems where the job is defined are offline then the operation will be retried when SA z/OS gateway connectivity is re-established.

S999

Explanation: Gateway to remote NetView was unavailable. A request to run an operation from TWS was received by the automation platform (NetView) address space, but it was destined for another NetView domain. An attempt was made to forward this operation to that domain, designated by the workstation name in the request, but this attempt failed. Either the gateway from the local system to the remote domain was down, or no response was received

Codes Posted to TWS by TWS Automation

from the remote system after waiting for one minute.

UNTV

Explanation: NetView unavailable. The TWS exit in TWS was unable to transmit this request to the automation platform (NetView) address space. Either the NetView in the system running the TWS/ESA Controller was down at the time this request was initiated or the NetView SSI was down.

U001

Explanation: Invalid status for the subsystem. The subsystem failed to start or terminate in the interval, or has changed to an unexpected status.

U002

Explanation: Invalid flag combination detected. This may indicate autotask definition errors. Review the NetView log for additional information.

U003

Explanation: Missing or incorrect TWS Automation definitions in the Automation control file or errors occurred reading the status file or executing an SA z/OS command such as INGREQ. Review the TWS policy definitions and check the netlog for additional messages.

U004

Explanation: Configuration error: The EVTOPER is not defined. This task is used to process status change notifications from the automation manager.

U005

Explanation: Sequence Error. This is usually an indication of two or more requests in progress for the same subsystem. Review the NetView log for additional information.

U006

Explanation: Configuration Error. An Automation Configuration File statement of the type OPCA DOMAINID, CODE= specifies SYSPLEX. The TWS Controller is not running in a system where SA z/OS Release 3 or later is installed. Use of the SYSPLEX keyword on this statement requires SA z/OS Release 3 or later to be installed where the TWS Controller runs.

U007

Explanation: Execution of the command is rejected by the installation exit AOFEXC20.

U008

Explanation: Timeout condition occurred while processing the command.

U009

Explanation: An invalid parameter was found while processing the command passed from TWS.

| U010

| **Explanation:** The completion information provided by
| exit AOFEX20 is incorrect. See the Netlog for more
| details on the reasons.

Chapter 6. Messages HSAH001I to HSAL6288I

HSAH001I HSAET32 SERVICES *verb* COMPLETE.

Explanation: Processing of the command completed successfully.

The variable *verb* shows the name of the HSAET32 command verb that completed successfully.

System action: HSAET service processing continues as requested by the command verb.

Operator response: None.

System programmer response: None.

HSAH002I INVALID HSAET32 SERVICES REQUEST *request*.

Explanation: The identified HSAET32 command is not valid.

The variable *request* shows the name of the HSAET32 command that is not valid.

System action: None.

Operator response: Correct and reissue the command as necessary. Valid services are REBUILD,START,STATUS,DISABLE,ENABLE.

System programmer response: None.

HSAH003I HSAET32 SERVICES REQUEST FAILED, RC=*rc*, REASON *rsn*.

Explanation: An error occurred during an attempt to process a valid HSAET32 command.

The variable *rc* shows the return code that was issued.

The variable *rsn* shows the reason code that was issued.

System action: None.

Operator response: Issue an HSAET32 STATUS command and verify that the status reported by message HSAH004I is as expected. Then reissue the command as appropriate. If the problem persists, notify your system programmer.

System programmer response: This message is issued with return code 8 and reason code 20 (14x) as a result of an HSAET32 START command if ET32 services were started previously (usually done during NetView initialization). If other problems occurred, contact your IBM Support Center.

HSAH004I HSAET32 SERVICES ARE *status*.

Explanation:

The variable *status* shows the current status of ET32 services. The following values can occur:

INACTIVE

HSAET32 services were not initialized and are unavailable.

ENABLED

HSAET32 services were initialized and are available.

DISABLED

HSAET32 services were initialized but were disabled from use.

System action: None.

Operator response: Verify that the status indicated is as currently required.

System programmer response: If the status value is INACTIVE, SA z/OS Startup HSAET32 initialization is in progress. Message HSAH001I must follow, indicating that HSAET32 SERVICES START is complete.

If this is not the case, check the netlog for additional messages indicating HSAET32 command problems, such as insufficient NetView operator command authorization. After the problem is corrected you can try to restart the HSAET32 services manually, issuing the IBM service command HSAET32 START, or recycle SA z/OS.

HSAL1071E CANNOT REGISTER OMVS SHUTDOWN OPTIONS, ERRNO=*code1*, ERRNO2=*code2*.

Explanation: The automation manager called the C-function `__shutdown_registration`, which failed.

The variable *code1* is the error return code.

The variable *code2* is the error return code.

System action: The SA z/OS automation manager continues with initialization.

Operator response: None.

System programmer response: Examine the error codes to determine why the function call failed. In most cases registration for OMVS shutdown fails because the automation manager does not have superuser or equivalent authority. Alternatively you may want to give the automation manager read permission to the BPX.SHUTDOWN facility class profile.

HSAL1072I THE AUTOMATION MANAGER IS BLOCKING OMVS SHUTDOWN.

Explanation: If an OMVS shutdown command is in progress the automation manager will block the

shutdown until the automation manager is terminated with a STOP,DEFER or a manual STOP command.

This is because the parmlib option BLOCKOMVS is set to YES.

System action: The automation manager continues processing.

Operator response: None.

System programmer response: None.

HSAL1081I SWITCHING COMMUNICATION FROM MQ TO XCF IS *action*.

Explanation: The WebSphere MQ subsystem was stopped or an error occurred in WebSphere MQ. The automation manager runs on the last system within the XCF group. The automation manager continues running with XCF communication only rather than terminating and thereby forcing a takeover.

The variable *action* may contain one of the following values:

STARTED

The switch from WebSphere MQ to XCF started.

COMPLETED

The switch from WebSphere MQ to XCF completed.

System action: The automation manager continues processing, but uses XCF for the communication between the automation agent and the automation manager.

Operator response: None.

System programmer response: None.

HSAL1082I SWITCHING COMMUNICATION FROM XCF TO MQ IS *action*.

Explanation: The automation manager was started with XCF communication. WebSphere MQ is available and the automation manager switches from XCF to WebSphere MQ communication.

The variable *action* may contain one of the following values:

STARTED

The switch from XCF to WebSphere MQ started.

COMPLETED

The switch from XCF to WebSphere MQ completed.

System action: The automation manager continues processing, using WebSphere MQ communication.

Operator response: None.

System programmer response: None.

HSAL1083I AUTOMATION MANAGER HAS CREATED THE INITIAL POSIX THREAD WITH PID @1

Explanation: The automation manager has successfully created the initial POSIX thread. The OMVS kernel has assigned the specified process ID (PID) to the initial POSIX thread. If you do not see this message then the automation manager is still waiting for OMVS to come up. If the automation manager is running with superuser authority, the initial POSIX thread will be created before OMVS initialization is completed. Otherwise the automation manager waits until OMVS initialization is completed and message BPXI004I comes up. This may require JES to be up.

System action: None.

Operator response: None.

System programmer response: None.

HSAL1091E INTERNAL ERROR TYPE *error type* DETECTED. DIAGNOSTIC INFO MAY FOLLOW: P1=*parm1* P2=*parm2* P3=*parm3*

Explanation: An internal error was found. Up to three additional diagnostic information fields may be provided, depending on the error type.

The variable *parm1* shows what could not be processed.

The variable *parm2* shows the value of the command, action or termination type.

The variable *parm3* (optional) shows the first parameter of the command.

The variable *error type* shows a decimal number that defines the type of the error. The following values can occur:

- 101 The thread manager got an invalid command, action or termination type that cannot be processed.
- 102 Cannot create thread specific data object for the main thread.
- 103 The class HsaSemaphore supports event and mutex semaphores. A method was applied for a semaphore type, that does not support this method.
The variable *parm1* shows the semaphore name.
- 104 The single MTIB object has not yet been created.
- 105 The single MTIB object must not be created with a MTIBptr=zero.
- 106 The factory tries to create a derived request handler but the function doCreateHandler returns zero.

- The variable *parm1* shows the handler name.
- 107 It was attempted to add a factory into the factory registry. However, a factory with the same name has already been added.
The variable *parm1* shows the factory name.
 - 108 A null pointer was detected. Processing stops.
 - 109 It was attempted to find a factory within the factory registry. However, the factory name is not contained in the registry.
The variable *parm1* shows the search argument (=factory name).
 - 110 It was attempted to access the factory registry. However, the factory registry has not yet been created and the static pointer to it is zero.
 - 111 Invalid attempt to reply to a RPC Handle that does NOT have the response flag ON.
 - 112 Dtor of Request Handler Factory detects that number of handlers is not zero.
The variable *parm1* shows the factory name.
The variable *parm2* shows the number of handlers.
The variable *parm3* shows the ivCtorError.
 - 113 Request Handler Factory attempts to create a handler but the factory error code ivCtorError is not zero.
The variable *parm1* shows the factory name.
The variable *parm2* shows the ivCtorError.

System action: The SA z/OS automation manager terminates.

Operator response: None.

System programmer response: Contact your IBM Support Center.

HSAL1092E INTERNAL ERROR TYPE *error type*
 WITHIN THREAD *thread name*
 DETECTED. DIAGNOSTIC INFO MAY
 FOLLOW: P1=*parm1* P2=*parm2*

Explanation: An internal error was detected. Up to two additional diagnostic information fields may be provided, depending on the type of the error.

The variable *thread name* shows the name of the thread where the error occurred.

The variable *error type* shows a decimal number that defines the type of the error. The following values can occur:

- 202 The thread manager does not support to restart a thread after a thread has been terminated.
The variable *parm1* shows the maximum retry count of threads.
- 203 A thread has been terminated and a restart is either not possible or not allowed.
- 204 The default implementation of a virtual function was called unexpectedly.
If specified the variable *parm1* shows the current timer.
- 205 An attempt was made to add a thread with a duplicate thread name to the Registry.
- 206 The system tried to find a collection class element but the given key (=thread name) could not be found.
- 207 Timer thread call handleExpiredTimer() after control was returned the timer has not been reset correctly, for example, previous timer >= new timer.
The variable *parm1* shows the previous timer specified.
The variable *parm2* shows the new timer specified.
- 208 This exception is used if a thread has received a request of class HsaRequest with a request type that the function handleRequest cannot process.
The variable *parm1* shows the request type.
- 209 Function completeRequest() attempted to send a request back to the creator thread. However, the thread name is NULL.
- 211 LogonId not yet set or not valid.
The variable *parm1* shows the logonID that is not valid.
- 251 The program detected an internal error that probably leads to thread termination.
- 252 The program detected an error that leads to thread termination and takeover. Examine other messages that may have been issued together with this message, for example, INGY110, to determine the cause of the error.

System action: The SA z/OS automation manager terminates.

Operator response: None.

System programmer response: For error type 252 try

HSAL1093E • HSAL1096I

to fix the problem. For all other error types, contact your IBM Support Center.

HSAL1093E COMMUNICATION OR SYSTEM SERVICE FAILED, ERROR TYPE=*error type* DIAGNOSTIC INFO MAY FOLLOW: P1=*parm1* P2=*parm2* P3=*parm3* P4=*parm4* P5=*parm5*

Explanation: A system error was detected. Up to five additional diagnostic information fields may be provided, depending on the type of the error.

The variable *error type* shows a decimal number that defines the type of the error. The following values can occur:

- 301** A z/OS UNIX pthread service failed.
The variable *parm1* shows the return code.
The variable *parm2* shows the error number.
The variable *parm3* shows the pthread service.
The variable *parm4* shows the current thread.
The variable *parm5* shows the semaphore name.
- 302** An attempt to get storage failed.
The variable *parm1* shows the amount of storage in bytes.
- 303** An attempt to perform an MVS system service failed.
The variable *parm1* shows the name of the HSAAPIMV service.
The variable *parm2* shows the return code of the service.
The variable *parm3* shows the current thread.
The variable *parm4* (optional) shows the API return code.
The variable *parm5* (optional) shows the API reason code.
- 304** An attempt to perform a Communication Service failed.
The variable *parm1* shows the name of the HSAAPICO service.
The variable *parm2* shows the return code of the assembler stub.
The variable *parm3* shows the communication return code of the service.
The variable *parm4* (optional) shows the name of the current thread.

System action: The SA z/OS automation manager terminates.

Operator response: None

System programmer response: Contact your IBM Support Center.

HSAL1094I THREAD *thread name* HAS RECEIVED A Z/OS UNIX SIGNAL *signal*.

Explanation: The signal handler got control due to a z/OS UNIX signal. Any received z/OS UNIX signal that a signal handler was registered for will terminate the automation manager.

The variable *thread name* shows the name of the thread that received the signal.

The variable *signal* shows the z/OS UNIX signal.

System action: The SA z/OS automation manager terminates.

Operator response: None.

System programmer response: Analyze the z/OS UNIX signal. If necessary, contact your IBM Support Center.

HSAL1095I SUSPENDED THREAD *thread name* HAS BEEN INTERRUPTED DUE TO A STOP REQUEST.

Explanation: The threads have been suspended, for example, during CONFIG load or REFRESH.

The variable *thread name* shows the name of the thread where the error occurred.

System action: The SA z/OS automation manager terminates abnormally.

Operator response: None.

System programmer response: None.

HSAL1096I AUTOMATION MANAGER TERMINATION IN PROGRESS DUE TO THREAD TERMINATION OR EXCEPTION BY THREAD *thread name*

Explanation: A thread of the automation manager ends abnormally. Analyze the error message HSAL1093E or HSAL1904I issued with this message.

The variable *thread name* shows the name of the thread where the error occurred.

System action: The SA z/OS automation manager terminates.

Operator response: None.

System programmer response: None.

HSAL1101I DIAGINFO SNAPSHOT *dsn* HAS BEEN CREATED

Explanation: A snapshot of all in-storage objects (state image queue) maintained by the automation manager has been written into the named data set.

The variable *dsn* shows the name of the data set that contains the snapshot.

System action: None.

Operator response: None.

System programmer response: None.

HSAL1102I DIAGINFO RECORDING INTO *dsn* HAS BEEN STARTED

Explanation: The automation manager has started recording all received workitems into the named data set. This includes writing the current state image queue into the data set.

The variable *dsn* shows the name of the data set.

System action: None.

Operator response: None.

System programmer response: None.

HSAL1103I DIAGINFO RECORDING INTO *dsn* HAS BEEN COMPLETED

Explanation: The automation manager has stopped recording all received workitems into the named data set.

The variable *dsn* shows the name of the data set.

System action: None.

Operator response: None.

System programmer response: None.

HSAL1104E DIAGINFO RECORDING INTO *dsn* HAS FAILED. REASON: *reason*

Explanation: The automation manager detected an error while recording the received workitems into the named data set. The cause of the error is indicated by the reason code.

The variable *dsn* shows the name of the data set.

The variable *reason* shows the cause of the error.

System action: None.

Operator response: Check logs for other messages associated with this event and data set.

If the data set experienced an out-of-space condition, consider increasing its size.

Restart the recording.

If the error cannot be corrected, contact your IBM Support Center.

System programmer response: None.

HSAL1105E IO OPERATION OF THE TAKEOVER FILE DSN=*dataset_name* FAILED. REASON: *diag_info*.

Explanation: The automation manager tried to write or update the takeover file. A file operation function failed.

The variable *dataset_name* is the name of the takeover file.

The variable *diag_info* provides the following information about the error:

C library function

The file operation function that failed, for example, `fopen`.

C library error code

The error code issued by the file operation function (errno).

C library error message

The error message text provided by the failing file operation function.

System action: The automation manager continues running.

Operator response: Check for I/O error messages and notify your system programmer.

System programmer response: Perform a cold or warm start. Note that the automation manager *must not* perform a hot-start or takeover, because the takeover file may be damaged or contain old data.

HSAL1106I MQ OPERATION ON STATE QUEUE FAILED. REASON: *reason*.

Explanation: An error occurred while accessing the MQ state queue.

The variable *reason* contains internal debugging information and the MQ action performed.

System action: The automation manager stops updating the MQ state queue but continues to maintain the resource status in-storage and in the Takeover file.

Operator response: If message HSAL1109I is previously issued, stop the MQ manager. This is necessary to avoid message flooding because the automation manager attempts to access the MQ State queue periodically.

System programmer response: Examine the syslog for additional messages giving more details about the cause of the error.

HSAL1107I REFRESH OF TAKEOVER FILE
DSN=*dataset_name* WITH CURRENT
AUTOMATION.

Explanation: The automation manager starts updating the takeover file with the in-storage data maintained for each resource. The first step is erasing the current data in the takeover file. The second step is writing the in-storage data into the takeover file.

The variable *dataset_name* is the name of the takeover file.

System action: None.

Operator response: None.

System programmer response: None.

HSAL1108I REFRESH OF TAKEOVER FILE
DSN=*dataset_name* COMPLETED.

Explanation: The automation manager has completed writing the in-storage resource data to the takeover file. The content of the takeover file is now identical with the in-storage resource data.

The variable *dataset_name* is the name of the takeover file.

System action: None.

Operator response: None.

System programmer response: None.

HSAL1109I REFRESH OF AUTOMATION
MANAGER DATA TO MQ STATE
QUEUE STARTED

Explanation: The automation manager started refreshing the MQ State queue with the in-storage data maintained for each resource. This process consists of the following two steps:

1. Purging the MQSeries State queue
2. Updating the MQSeries State queue with the resource data

System action: None.

Operator response: None.

System programmer response: None.

HSAL1110E R15 VALUE = *rc* REASON CODE = *rs*
RBA = *rba* LAST OP = *lastop* EXPECTED
RECORD SIZE = *rsize* REPORTED IO
RESULT = *asize*

Explanation: The automation manager has had a VSAM I/O error.

The variable *rc* shows the VSAM return code

The variable *rs* shows the VSAM reason code.

The variable *rba* shows the VSAM relative byte address.

The variable *lastop* shows the last I/O operation executed.

The variable *rsize* shows the expected number of bytes to be read in the I/O operation.

The variable *asize* shows the actual number of bytes read in the I/O operation.

System action: Additional error message will be issued.

Operator response: Contact your system programmer.

System programmer response: Contact IBM for support.

HSAL1111I DISABLING OF TAKEOVER FILE
DSN=*dataset_name* COMPLETED.

Explanation: The automation manager has freed the takeover file and stopped writing to it.

The variable *dataset_name* is the name of the takeover file.

System action: None.

Operator response: None.

System programmer response: None.

Classes: None.

HSAL1113I WORKITEM EXPIRED: *thread_name*
DATE:*exp_date* TIME:*exp_time*
SIG:*wi_signature*

Explanation: The workitem has passed its expiry time.

The variable *thread_name* is the name of the thread.

The variable *exp_date* is the expiry date of the workitem.

The variable *exp_time* is the expiry time of the workitem.

The variable *wi_signature* is the workitem signature.

System action: None.

Operator response: Notify your system programmer.

System programmer response: Try to determine reason for expiry, for example, high system or automation manager activity. Examine the netlog for additional information.

HSAL1300E TIMER REQUEST IS INVALID: RC=*rc*,
TYPE=*type*, CREATOR=*creator*

Explanation: The validation of an internal timer request failed with a return code.

The variable *rc* shows the value of the return code.

The variable *type* shows the type of the request.

The variable *creator* shows the creator or the request.

System action: The request is rejected.

Operator response: None.

System programmer response: None.

Module: HSACMTMR

HSAL1301E INVALID REQUEST RECEIVED:
TYPE=type, CREATOR=creator.

Explanation: An LE component received a request of a type that could not be processed.

The variable *type* shows the type of the request.

The variable *creator* shows the creator or the request.

System action: The request is rejected.

Operator response: Contact your IBM Support Center.

System programmer response: None.

Module: HSACMTMR , HSAHNDLS

HSAL1400E OVERRIDE FILE NOT AVAILABLE.
FUNC=function, CODE=errorcode.

Explanation:

The variable *function* shows the function that could not be performed.

The variable *errorcode* shows the value of the error code that was issued (return code bytes 1-4 and reason code bytes 5-8).

System action: The override file is marked unavailable. All requests querying or manipulating any overrides are rejected. Existing base schedules are still working as defined in the customization dialogs.

Operator response: Notify your system programmer.

System programmer response: Analyze the function code by referring to the OS/390 LE environment STDIO.H file. The errorcode can be analyzed by referring to *OS/390 DFSMS Macroinstructions for Data Sets* using the return and reason codes from the error code. If the problem is solved, start a secondary automation manager and initiate a takeover. If you cannot resolve the problem contact your IBM Support Center.

Module: HSACMSST

HSAL1401E TIMER REQUEST FAILED. RC=rc,
ORIGINATOR=name, REQUEST=req.

Explanation: A timer request has been returned with an unexpected return code to the LE component that is responsible for schedules.

The variable *rc* shows the return code that was issued.

The variable *name* shows the name of the originator.

The variable *req* shows the name of the request.

System action: The request is ignored. This may lead to an unexpected behaviour of the affected schedule.

Operator response: Notify your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSAHNDLS

HSAL1402E WORKITEM COULD NOT BE
QUEUED. RC=rc.

Explanation: The LE component responsible for schedules could not inform the component managing the resources about the begin or the end of a service period.

The variable *rc* shows the return code that was issued.

System action: Processing continues. However, this may lead to an unexpected behaviour of the affected resource(s).

Operator response: Initiate[®] a takeover to a secondary automation manager. If the problem persists notify your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSACMSST, HSACMTMR

HSAL1403I A TIMER EXPIRED FOR resource THAT
IS NO LONGER AVAILABLE.

Explanation: The current and all future overrides of the resource have been deleted. Also, the current timer of the resource expired before it could be purged.

The variable *resource* shows the name of the resource.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: HSACMSST

HSAL1404E A VSAM ERROR OCCURRED. RC=rc,
RSNreason FUNCTION=function

Explanation: A function operation on the override file failed with a return code and a reason code.

The variable *rc* shows the value of the return code.

The variable *reason* shows the value of the reason code.

The variable *function* shows the name of the function.

System action: The override file is marked unavailable. All requests querying or manipulating any overrides are rejected. Existing base schedules are still working as defined in the customization dialogs.

Operator response: Notify your system programmer.

System programmer response: Analyze the VSAM return code and reason code and correct the error

HSAL1405I • HSAL6012E

condition. When the problem is solved, start a secondary automation manager and initiate a takeover.

Module: HSACMSST

HSAL1405I WORKITEM REQUEST FAILED. RC=*rc*, RSN=*reason*

Explanation: A function operation on the override file failed with a return code and a reason code.

The LE component responsible for schedules received an unexpected response from a workitem request that signalled the beginning or the end of a service period.

The variable *rc* shows the value of the return code.

The variable *reason* shows the value of the reason code.

System action: Processing continues. However, this may lead to an unexpected behavior of the affected resources.

Operator response: Notify your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSAHNDLS

HSAL1406I SERVICE PERIOD TIME SLOT HAS BEEN MISSED FOR *resource*.

Explanation: The LE component that is responsible for schedules received an event indicating the beginning or the end of a time slot. However, the current status of the resource is equal to the status of the resource before the event was generated. This indicates the loss of at least one time slot and can only have occurred when the automation manager or one of its components was suspended for this time frame.

The variable *resource* shows the name of the affected resource.

System action: Processing continues. However, this may lead to unexpected behavior for the affected resource.

Operator response: Check the service period of the indicated resource. Perform the missed action manually if necessary.

System programmer response: None.

Module: HSACMSST

HSAL5100E ENVIRONMENT SETUP FOR WORKITEM CREATION FAILED. RC=*rc* RSN=*rsn*.

Explanation: The setup function to create a work item failed. The internal communication based on work items is no longer guaranteed.

The variable *rc* shows the value of the return code.

The variable *rsn* shows the value of the reason code.

System action: The automation manager terminates.

Operator response: Inform your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSACLWI

HSAL6010A *resource*; INTERVENTION REQUIRED; BEYOND AUTOMATION

Explanation: A problem with the resource has been detected that automation cannot handle.

The variable *resource* shows the name of the resource where an intervention is required.

System action: Automation for the resource is stopped.

Operator response: Fix the problem or notify your system programmer.

System programmer response: Try to fix the problem.

Classes: 80, 81.

HSAL6011E *resource*; INTERVENTION REQUIRED; AUTOMATION DENIED

Explanation: The resource could not be automated because the automation flag is turned off.

The variable *resource* shows the name of the resource that could not be automated.

System action: Automation for the resource is stopped.

Operator response: Check why the automation flag is turned off and decide whether to turn it on.

System programmer response: None.

Classes: 80, 82.

HSAL6012E *resource*; INTERVENTION REQUIRED; AUTOMATION INHIBITED

Explanation: Automation of the resource is not possible due to an intervention required on another resource. The immediate problem with this resource is only the symptom of a problem with another resource.

The variable *resource* shows the name of the resource where an intervention is required.

System action: Automation for the resource is stopped.

Operator response: Find the resource that is impacting this one and fix the problem.

System programmer response: None.

Classes: 80, 83.

HSAL6020I *resource*; AWAITING AUTOMATION

Explanation: The resource is automatable, but is waiting for prerequisites to be satisfied. These could be either dependencies or triggers.

The variable *resource* shows the name of the resource.

System action: Automation waits until the prerequisites for automation are satisfied and then starts processing.

Operator response: If the prerequisites involve operator action, for example, manually setting a trigger, check that this was done.

System programmer response: None.

Classes: 80, 84.

HSAL6030I *resource* IN AUTOMATION

Explanation: Commands concerning the resource have been sent to the agent responsible for it.

The variable *resource* shows the name of the resource.

System action: Automation for the resource continues.

Operator response: None.

System programmer response: None.

Classes: 80, 85.

HSAL6040E *resource*; DEGRADED

Explanation: The resource is the target of a MakeAvailable request and is currently degraded.

The variable *resource* shows the name of the resource that is degraded.

System action: Automation continues. If an opportunity arises, the resource will be restored to Satisfactory.

Operator response: Investigate why the resource is degraded.

System programmer response: Assist with operator action.

Messages HSAL6101I through HSAL6155I are a set of self-explanatory messages that appear in the history logger.

HSAL6156I Prepare Unavailable order forgotten (cancel sent)

Explanation: The automation manager told the automation agent to issue the SHUTINIT commands for the resource and it has done so. The Automation Manager has now decided that the resource should not be stopped and has cleared its record that this has happened. The automation agent has been told to clear the shutdown process it has in progress for the subsystem.

Classes: 80, 86, 87.

HSAL6041E *resource*; DEGRADED

Explanation: The resource is currently degraded.

The variable *resource* shows the name of the resource that is degraded.

System action: Automation continues. If possible, the resource will be restored to Satisfactory.

Operator response: Investigate why the resource is degraded.

System programmer response: Assist with operator action.

Classes: 80, 87.

HSAL6050I *resource*; SATISFACTORY

Explanation: The resource is the target of either a MakeAvailable or MakeUnavailable request and that status has been achieved.

System action: Automation proceeds normally.

The variable *resource* shows the name of the resource.

Operator response: None.

System programmer response: None.

Classes: 80, 88, 89.

HSAL6051I *resource*; SATISFACTORY

Explanation: The resource has achieved its desired status.

The variable *resource* shows the name of the resource.

System action: Automation continues.

Operator response: None.

System programmer response: None.

Classes: 80, 89.

Note that this message is not guaranteed to be issued whenever such a situation occurs. If, for example, the automation agent were to disconnect, the record that the SHUTINIT had been issued would be cleared, but this message (and the cancel order to the agent) would not be sent.

System action: Automation continues.

Operator response: None.

System programmer response: None.

HSAL6230A • HSAL6288I

Classes: 80, 89.

Messages HSAL6157I through HSAL6199I are a set of self-explanatory messages and are not documented here.

Message HSAL6220I is a self-explanatory message and is not documented here.

HSAL6230A *resource*; AGENT ORDERED TIME-OUT; LOST AGENT STATUS UPDATE?

Explanation: The automation manager did not receive the expected response within 5 minutes for an order that it sent to the resource's automation agent.

The variable *resource* shows the name of the resource.

System action: Automation for the resource is suspended, its automation state will be set to Problem. In the event that the expected response should subsequently arrive, automation will be resumed.

Operator response: Check the state of the resource

with INGLIST. If it shows an outstanding operation, use Update to change the resource's automation status to Idle. This should resume automation for the resource. If INGLIST does not show an outstanding operation, you need to first set the resource's observed state to HardDown, then set its automation state to Idle, and finally set its observed state to SoftDown. This should cause the automation manager to resend the order to the agent.

System programmer response: Check your service levels for XCF and SA z/OS. Contact IBM service for SA z/OS.

Classes: 80, 81.

Messages HSAL6250I through HSAL6283I are a set of self-explanatory messages and are not documented here.

HSAL6284I REQUEST FOR *requesttype action* FROM *source* FOR *resource* UNSATISFIED AFTER TIME OUT

Explanation: Automation returns that the request has not been satisfied within the specified time.

The variable *requesttype action* shows the type of the request and the action.

The variable *source* shows the name of the source.

The variable *resource* shows the name of the resource.

System action: Automation continues.

Operator response: None.

System programmer response: None.

Classes: 80, 81.

The variable *source* shows the name of the source.
The variable *resource* shows the name of the resource.

System action: Automation continues.

Operator response: None.

System programmer response: None.

Classes: 80, 81.

HSAL6286I REQUEST FOR *requesttype action* FROM *source* FOR *resource* SATISFIED

Explanation: Automation returns that the request was satisfied.

The variable *requesttype action* shows the type of the request and the action.

The variable *source* shows the name of the source.

The variable *resource* shows the name of the resource.

System action: Automation continues.

Operator response: None.

System programmer response: None.

Classes: 80, 88, 89.

HSAL6285I REQUEST FOR *requesttype action* FROM *source* FOR *resource* CANCELLED AFTER TIME OUT

Explanation: Automation returns that the request has not been satisfied within the specified time and therefore was removed.

The variable *requesttype action* shows the type of the request and the action.

Message HSAL6287I is a self-explanatory message and is not documented here.

HSAL6288I REQUEST FOR *requesttype action* FROM *source* FOR *resource* EXPIRED

Explanation: Automation returns that the request was satisfied after the specified time period.

The variable *requesttype action* shows the type of the request and the action.

The variable *source* shows the name of the source.

The variable *resource* shows the name of the resource.

Messages HSAH001I to HSAL6288I

System action: Automation continues.

System programmer response: None.

Operator response: None.

Classes: 80, 81.

Messages HSAL6289I through HSAL6999I are a set of self-explanatory messages and are not documented here.

Chapter 7. Messages HSAM1000I to HSAM5411I

HSAM1000I AUTOMATION MANAGER SUBTASKS ARE BEING STARTED.

Explanation: The basic initialization step of the SA z/OS automation manager has successfully completed. All subtasks are being attached and the main task will wait observing the subtasks.

System action: The SA z/OS automation manager continues with the initialization.

Operator response: None.

System programmer response: None.

HSAM1001I THE KEY *keyword* CANNOT BE RETRIEVED FROM PKVT, CONTINUE WITH DEFAULT VALUE: *default value*.

Explanation: A keyword cannot be resolved. The key value specified in the parmlib member will be ignored.

The variable *keyword* shows the start parameter that can be modified in the parmlib member.

The variable *default value* shows the default value specified for the keyword. If the default value is blank, this field may be blank.

System action: The SA z/OS automation manager continues with the initialization.

Operator response: None.

System programmer response: Contact your IBM Support Center.

HSAM1002I AUTOMATION MANAGER INITIALIZATION DETECTED THAT THE SUBSYSTEM *subsystem* IS MISSING OR CONFLICT DETECTED WITH PARMLIB PARAMETERS COMM/MQM.

Explanation: The automation manager needs the specified system component for initialization.

The variable *component* shows the required system component, for example, the OMVS kernel or the WebSphere MQ Manager.

System action: The SA z/OS automation manager terminates.

Operator response: Inform your system programmer.

System programmer response: Start the required system component. Examine the automation manager PARMLIB member for correct specification of the COMM and MQM parameters. If COMM=XCF, the MQM parameter must not be specified. If COMM=MQ, a valid

WebSphere MQ subsystem name must be specified for the MQM parameter.

HSAM1003I ARM *function* WAS SUCCESSFUL FOR ELEMENT=*element* TYPE=*type* restart.

Explanation: An ARM function (either register or deregister) could be successfully completed.

The variable *function* shows the ARM function. It can be either register or deregister.

The variable *element* shows the ARM element, representing the SA z/OS automation manager.

The variable *type* shows the type of the ARM element SA z/OS automation manager.

The variable *restart* shows whether the automation was restarted by ARM or not (RESTART=YES or RESTART=NO.)

System action: The SA z/OS automation manager continues initializing being ARM enabled.

Operator response: None.

System programmer response: None.

HSAM1004I ARM *function* FAILED FOR ELEMENT=*element* TYPE=*type*, IXCARM RET=RC RSN=RSN restart.

Explanation: An ARM function (either register or deregister) could not be completed successfully.

The variable *function* shows the ARM function. It can be either register or deregister.

The variable *element* shows the ARM element, representing the SA z/OS automation manager.

The variable *type* shows the type of the ARM element "SA z/OS automation manager" and optionally the restart type which informs you whether the automation was restarted by ARM or not (RESTART=YES or RESTART=NO or RESTART=ERR).

The variable RC shows the return code of the ARM function.

The variable RSN shows the reason code of the ARM function. If the Automation Manager retries ARM registration due to parameter ARMWAIT=nnn in parmlib member HSAPRMxx then the variable RSN has a suffix ",RETRY...". This informs you that the Automation Manager is waiting for ARM to be up.

System action: The SA z/OS automation manager continues initializing without being ARM enabled. If RESTART=ERR then the SA z/OS automation manager was restarted on a different MVS system. In this case,

HSAM1005I • HSAM1010I

the SA z/OS automation manager terminates.

Operator response: None.

System programmer response: Verify if ARM was correctly customized for the SA z/OS automation manager. If necessary, contact your IBM Support Center.

HSAM1005I COMMUNICATION IS XCF ONLY. THE TAKEOVER FILE WILL BE USED FOR HOT START AND RECOVERY PURPOSES.

Explanation: The automation manager does not use MQ Series, neither for the communication with the automation agents nor for recording the current status and settings of the resources. Instead XCF is used for the communication with the automation agents.

The automation manager records all changes made to the resources in the takeover file identified by DD-name HSATKOV. In case of an abend or any normal takeover (hot start), the automation manager reads the takeover file to restore the most current status of the resources.

Note that any request that has not yet been processed by the automation manager will be lost and cannot be recovered in case of an abend or any regular takeover.

System action: None.

Operator response: None.

System programmer response: None.

HSAM1006E PARAMETER *option* CANNOT BE RETRIEVED FROM PARMLIB.

Explanation: The named parameter is mandatory but not specified in the HSAPRMxx parmlib member that is used at the startup time of the automation manager.

The variable *option* is the name of the parameter that is missing.

System action: The SA z/OS automation manager terminates.

Operator response: None.

System programmer response: Add the parameter in the HSAPRMxx parmlib member that is used at the startup time of the automation manager.

HSAM1010I AUTOMATION MANAGER TERMINATION COMPLETE, REASON CODE = *reason code*

Explanation: The automation manager main task terminates. If the *reason code* is zero it terminated successfully. The reason code can have the following values:

0 Successful completion

- 1 Implicit normal termination due to directed takeover (changed operation mode from SAM to PAM)
- 4 Stop Immed
- 8 Stop Force
- 20 Initialization failed because MTIB could not be created
- 24 Initialization failed because MTIB Vector could not be created
- 28 Initialization failed because a task could not be created
- 32 Initialization failed because a task could not be attached
- 33 Initialization stopped but no takeover is performed
- 34 Either the initial storage check failed, or a GETMAIN service abended after initialization due to a storage shortage. The automation manager needs a minimum of 40 MB free storage.
- 35 ARM (De)Reg failed
- 36 z/OS UNIX System Services has canceled the job
- 39 An I/O error occurred for the takeover file
- 40 Implicit normal or forced termination due to error detected by observer
- 44 The subsystem is missing
- 45 z/OS does not meet the prerequisites
- 48, 52 C++ task failed. This is probably an exception
- 56 LE task abended
- 60 Task abended
- 64 LE user abend
- 72 Bad return code from subtask shell function
- 76 Event handler detected an error
- 80 The job was canceled

Note that for all return codes < 40 no ARM restart is performed. It will be suppressed even if ARM is enabled, except for /F jobname,STOP,...,RESTART.

System action: The SA z/OS automation manager terminates.

Operator response: None.

System programmer response: Analyze the *reason code* and, if issued, the HSAM1050E error message.

HSAM1011I CURRENT Z/OS SYSTEM DOES NOT MEET AUTOMATION MANAGER PREREQUISITES: xxxxx

Explanation: The automation manager requires a z/OS prerequisite that is not met by the current z/OS system. The following prerequisites may be detected:

z/OS < 1.9

The release of z/OS is earlier than V1.9.

System action: The SA z/OS automation manager terminates.

Operator response: None.

System programmer response: Run the automation manager on a z/OS system that meets the specified prerequisites.

See Table 3 on page 38 for diagnostic information about combinations of *error type* and *error code*.

The variable *service* shows the name of the service that caused the error.

The variable *return code* shows the return code of the failing service (hex number). This token is specified together with *service*.

The variable *reason code* shows the reason code of the failing service (hex number). This token is specified together with *service*.

The variable *diagnostic* shows diagnostic information in a string of tokens. The first token specifies the name of the failing module. All remaining tokens vary depending on the error code. Data provided by *diagnostic* is intended for IBM service personnel.

HSAM1050E AUTOMATION MANAGER ERROR DIAGNOSIS: SYSTEM=*system* TASK=*task name* ERRTYPE=*error type* ERRCODE=*error code* SERVICE=*service* RC=*return code* RSN=*reason code* DIAG=*diagnostic*

Explanation: This is a generic error message that provides a detailed problem description of an error that was detected during automation manager processing.

The variable *system* shows the name of the MVS system where the error was detected.

The variable *task name* shows the name of the automation manager task that causes the error.

The variable *error type* shows the type of the error. The following values can occur:

- 4 Input checking failed.
- 8 Initialization error.
- 12 Termination error.
- 16 An internal sysplex communication pipe service provided by an SA z/OS system failed.
- 20 RPC Command Handler failed.
- 24 A system service named by SERVICE, such as the MVS request IXCJOIN, failed.
- 32 Storage problem.
- 36 Internal error.

Note: An internal error 36, combined with RC=0000EC6 and RSN=0000FF09 may indicate automation manager termination due to OMVS shutdown.

- 40 Environment error.

The variable *error code* shows a unique number to determine the location in the code where the error was detected.

For further information about the combinations of error types and error codes and their meaning, refer to Table 3 on page 38.

System action: None.

Operator response: None.

System programmer response: Look into the table for combinations of error type/codes for a detailed explanation of the error. Examine the return code and reason code of the failing service if available. If necessary, contact your IBM Support Center.

HSAM1051E SYSPLEX COMMUNICATION MANAGER INITIALIZATION FAILED.

Explanation: An error was detected during initialization of the sysplex communication manager task in the SA z/OS automation manager.

System action: The SA z/OS automation manager terminates.

Operator response: None.

System programmer response: Look for additional HSAM1050E messages that are issued with this message.

HSAM1052I SYSPLEX COMMUNICATION MANAGER INITIALIZATION COMPLETE.

Explanation: The sysplex communication manager task of the SA z/OS automation manager could be successfully initialized and is ready to receive.

System action: The SA z/OS automation manager continues with the initialization.

Operator response: None.

System programmer response: None.

HSAM1059I DIAGNOSTIC INFO: MOD=*module*
INCIDENT=*incident diagnostic-text*

Explanation: The purpose of this message is to provide diagnostic information to IBM service staff.

The variable *module* shows name of the module that detected the incident.

The variable *incident* shows a unique ID for an internally detected incident.

The variable *diagnostic-text* shows useful text in free format. The text depends on the incident that occurred. For example, for INCIDENT=100, this is the duplicate handle + seq-number.

System action: Processing continues.

Operator response: None.

System programmer response: None.

**HSAM1301I SA z/OS AUTOMATION MANAGER
WAITING FOR INITIALIZATION
DELAY COMPLETION.**

Explanation: An automation manager initialization delay of greater than 30 seconds was specified. The specified delay allows another automation manager to assume priority control of the automation sysplex if started within the delay period.

System action: The SA z/OS automation manager suspends initialization until the specified delay expires or is canceled through an operator command. This message is deleted when the specified delay is no longer in effect.

Operator response: The delay may be canceled by routing an MVS Modify(F) <jobname>,GO command to the issuing system image.

System programmer response: None.

**HSAM1302A SPECIFY SA z/OS AUTOMATION
CONFIGURATION DATA SET NAME.**

Explanation: The SA z/OS automation manager has been started with PROMPT=YES in effect. If requested, this message will appear on the first SA z/OS automation manager instance to assume control of the automation sysplex following a sysplexwide IPL or similar automation shutdown.

System action: The SA z/OS automation manager waits for a response to this request.

Operator response: A null or asterisk (*) response may be used to indicate that the default data set as determined by the start type is to be used. For a warm start, this default is the data set that was in use when the previous primary automation manager instance terminated. For a cold start, the default is the data set or generation data group (GDG) member as specified for the installation through the selected parmlib member.

The response may also be a fully qualified data set or GDG member name. If the default configuration data set is a GDG member, a different member of the same GDG may be specified using an asterisk followed by the relative GDG number in parentheses, for example, *(0) or *(-1).

System programmer response: None.

**HSAM1303I SA z/OS AUTOMATION MANAGER
IS WAITING FOR MQSERIES
CONNECTION**

Explanation: The SA z/OS automation manager is checking whether MQSeries is ready.

System action: If MQSeries is not ready the SA z/OS automation manager retries until the TAKEOVERTIMEOUT defined in the parmlib is expired. If it is expired the SA z/OS automation manager switches from hot to warm start.

Operator response: None.

System programmer response: None.

**HSAM1304A SPECIFY SA z/OS AUTOMATION
CONFIGURATION DATA SET NAME.**

Explanation: The SA z/OS automation manager was unable to determine or otherwise process the configuration data set as specified through the HSAM1302A response, or defaulted from prior use. Additional messages will indicate any errors encountered while attempting to process the specified or default data set.

System action: The SA z/OS automation manager waits for a response to this request.

Operator response: The response must be the fully qualified MVS data set or GDG member name containing the SA z/OS configuration data. If the default configuration data set is a GDG member, a different member of the same GDG may be specified using an asterisk followed by the relative GDG number in parentheses, for example, *(0) or *(-1). The automation manager may also be stopped if no valid configuration data is available.

System programmer response: None.

**HSAM1305I UNABLE TO PROCESS SA z/OS
AUTOMATION CONFIGURATION
DATA SET** *dataset*

Explanation: The SA z/OS automation manager was unable to dynamically allocate the named configuration data set. The named data set name was successfully allocated, but cannot be used by the issuing automation manager instance. Additional messages will indicate any errors encountered while attempting to process the specified or default data set.

The variable *dataset* shows the name of the configuration data set that could not be allocated dynamically.

System action: The SA z/OS automation manager instance prompts for a new configuration data set name via message HSAM1304A.

Operator response: Check that the data set has not been incorrectly modified prior to starting the affected automation manager instance. Also check whether all copy operations from the configuration dialog output completed successfully. Respond to the HSAM1304A prompt with a new or backup data set name as described in the installation procedures.

System programmer response: None.

**HSAM1306I UNABLE TO ALLOCATE SA z/OS
AUTOMATION CONFIGURATION
DATA SET *dataset***

Explanation: The SA z/OS automation manager was unable to dynamically allocate the named configuration data set. The data set name could not be allocated or is not a partitioned data set. Additional dynamic allocation messages will indicate any errors encountered while attempting to allocate the data set.

The variable *dataset* shows the name of the configuration data set that could not be allocated dynamically.

System action: The SA z/OS automation manager instance prompts for a new configuration data set name via message HSAM1304A.

Operator response: Check that the data set name is correct and that it is a correctly catalogued partitioned data set. Respond to the HSAM1034A prompt with a new or backup data set name described in the installation procedures.

System programmer response: None.

**HSAM1307I SA z/OS SECONDARY AUTOMATION
MANAGER INITIALIZATION
COMPLETE.**

Explanation: The SA z/OS automation manager instance has completed initialization as a secondary manager, and is ready to assume automation sysplex control with minimum delay in case the primary manager terminates.

System action: The SA z/OS automation manager instance monitors for changes in the configuration status, and attempts to assume control of the automation sysplex in case the primary automation manager terminates.

Operator response: None.

System programmer response: None.

**HSAM1308I SA z/OS PRIMARY AUTOMATION
MANAGER INITIALIZATION
COMPLETE, TYPE=*type***

Explanation: The SA z/OS automation manager instance has completed initialization as the primary automation manager, and has assumed control of the automation sysplex.

The variable *type* shows the start type. It can be COLD, WARM, or HOT.

System action: The SA z/OS automation manager instance monitors for changes in the configuration status, and is ready to service requests from all automation agents defined to the automation sysplex.

Operator response: None.

System programmer response: None.

**HSAM1309I SA z/OS PRIMARY AUTOMATION
MANAGER HOT TAKEOVER
COMPLETE.**

Explanation: The SA z/OS automation manager instance has successfully completed a takeover operation and is now the primary automation manager. This message may also appear when a START=HOT is used to manually restart the manager.

System action: The SA z/OS automation manager instance monitors for changes in the configuration status, and is ready to service requests from all automation agents defined to the automation sysplex.

Operator response: None.

System programmer response: None.

**HSAM1310I RC=*rc* REAS=*reason* OPCODE=@5
(@3.@4) @6.**

Explanation: This is an internal message for diagnostic purposes only.

Ignore messages with RC=2.

System action: None.

Operator response: None.

System programmer response: Contact your IBM Support Center.

HSAM1311I @1

Explanation: This is an internal message for diagnostic purposes only.

System action: None.

Operator response: None.

System programmer response: None.

HSAM1314I TAKEOVER WILL BE DELAYED FOR
timeout SECONDS.

Explanation: Takeover will be delayed for a number of seconds if it is done from the VSAM takeover file.

The variable *timeout* is taken from the TAKEOVERTIMEOUT parameter of the HSAPRMxx parmlib member.

System action: Processing continues. Any stop command is accepted but does not cause termination until the time that is specified in TAKEOVERTIMEOUT has expired.

Operator response: None.

System programmer response: None.

Classes: None.

HSAM1315I SUCCESSFUL ALLOCATION OF
TAKEOVER FILE *dataset_name*.

Explanation: The automation manager issues this message when the takeover file has been allocated successfully.

The variable *dataset_name* is the name of the takeover file as specified in the HSAPRMxx parmlib member.

System action: The SA z/OS automation manager continues with initialization.

Operator response: None.

System programmer response: None.

HSAM1316I ALLOCATION OF TAKEOVER FILE
dataset_name FAILED WITH RC=*code*;
DIAGNOSTIC INFO RETCODE=*retcode*
REASON=*reason*.

Explanation: The automation manager tried to allocate the specified takeover file but this failed.

The variable *dataset_name* is the name of the takeover file as specified in the HSAPRMxx parmlib member.

The variable *code* explains why the allocation failed. It can have the following values:

- 98** No more memory
- 99** Invalid input parameter
- 10nn**
Allocation failed
- 20nn**
Deallocation failed

where *nn* is one of the following:

- 8** Invalid data set name, member name, or DD-name.

- 12** Allocation error. *retcode* and *reason* are the return code and reason code from the SVC 99 (DynAlloc).

The variable *retcode* gives more details about why the allocation failed.

The variable *reason* is the reason code.

System action: The automation manager continues and issues message HSAM1317A, asking the operator how to continue.

Operator response: None.

System programmer response: Analyze why the allocation failed. Look for additional messages (HSAL1105E in case of an I/O error) that are issued with this message.

HSAM1317A REPLY "HOT" TO RETRY HOT-START
OR "WARM" TO CONTINUE WITH
WARM-START OR "CANCEL".

Explanation: During the initialization of the automation manager performing a hot-start, an error occurred while accessing the takeover file. Either the allocation failed or a VSAM read/write error occurred or the takeover file was disabled by the previous PAM.

System action: The automation manager waits for an operator response.

Operator response: Specify:

HOT To retry hot-start processing by allocating the takeover file again.

WARM To continue with a warm start.

CANCEL
Results in HSAM1390E with the option to stop the automation manager.

System programmer response: None.

HSAM1320E SA z/OS AUTOMATION MANAGER
request REQUEST TIMED OUT.

Explanation: The named internal automation manager request did not complete within the time limit associated with the request.

The variable *request* shows the name of the request that timed out.

This may be due to temporary system performance factors, delays caused by an unusual amount of automation activity, or problems processing the named request.

System action: The SA z/OS automation manager instance continues monitoring for commands and configuration changes, and resumes normal operation when the named request completes. The message will also be deleted when the request has completed.

Operator response: If this message appears often or remains outstanding for an extended period, advise

your system programmer of the condition.

System programmer response: Investigate the following request specific possibilities:

- AMS_READ_CONFIGURATION

This request is issued during Primary automation manager startup, automation manager takeover, and when a configuration refresh is requested. If the message appears during a hot start or takeover operation, the delay is probably MQSeries related. If it appears during a Warm or Cold start, or configuration refresh operation, the delay is probably I/O related.

Another reason might be that the automation manager does not run as super user for UNIX System Services or one of your HFSs is not mounted with SUB=MSTR. See also message "HSAL1083I" on page 136 for more information.

- AMS_APPLY_CONFIGURATION

While the timeout value for most automation manager requests is determined by IBM, the AMS_APPLY_CONFIGURATION timeout value may be specified through the HSAPRMxx parameter member. The time necessary to complete this function depends on the number of resources defined, and the temporary appearance of the HSAM1320E message may indicate that the specified or default timeout value is too low.

HSAM1321I SA z/OS AUTOMATION AGENT *request* REQUEST TIMED OUT.

Explanation: An automation agent request did not complete within the time limit. This may be due to temporary system performance factors, delays caused by an unusual amount of automation activity, or problems processing the named request.

The variable *request* shows the name of the request that timed out.

For a configuration refresh it is required that check ACF is complete on all agents. Note that a configuration refresh operation is considered complete when the automation manager configuration has been updated, and does not require that all automation agents have completed their corresponding configuration update activity.

System action: The SA z/OS automation manager instance continues monitoring for commands and configuration changes, and resumes normal operation when the named request completes. The message will be deleted when a response was received from all affected agents.

Operator response: Verify that all active automation agents appear to be working correctly and are communicating with the automation manager. If the request is Check_ACF, an Fjobname,GO command issued to the Primary automation manager will cause it

to continue a configuration refresh operation without the check.

If this message appears often or remains outstanding for an extended period, advise your system programmer of the condition.

System programmer response: None.

HSAM1323I UNABLE TO OBTAIN PREVIOUS CONFIGURATION INFORMATION, RC=*rc*, Reason=*reason*.

Explanation: The SA z/OS automation manager was unable to read the data set defined by the HSACFGIN DD statement.

The variable *rc* shows the value of the return code.

The variable *reason* shows the value of the reason code.

System action: The SA z/OS automation manager instance terminates.

Operator response: Check that the data set defined by the HSACFGIN DD statement of the automation manager start procedure is available and correctly initialized. Restart the automation manager using an alternate data set if available, or contact your systems programmer.

System programmer response: All instances of the automation manager within the automation sysplex must point to the same CFGINFO data set. A new data set can be initialized by starting the automation manager with TYPE=COLD.

HSAM1324E UNABLE TO UPDATE CONFIGURATION INFORMATION, RC=*rc*, REASON=*reason*

Explanation: The SA z/OS automation manager could not update the data set defined by the HSACFGIN DD statement.

The variable *rc* shows the value of the return code.

The variable *reason* shows the value of the reason code.

System action: If this message appears during initial automation manager startup, the instance continues to run normally. However, a subsequent warm may not be possible. If the error occurs during a configuration refresh operation, the configuration refresh will be aborted.

Operator response: Check that the data set defined by the HSACFGIN DD statement of the automation manager start procedure is available and correctly allocated. Also check that the automation manager has the necessary authorization to update it. When the problem is resolved, perform a configuration refresh using the correct data set name.

System programmer response: All instances of the

automation manager within the automation sysplex must point to the same CFGINFO data set. Also, all automation manager instances must have the correct SAF authority to allow it to be updated. If the instance was started with TYPE=COLD, this may be the initial access to the data set. Otherwise, the data set is opened and read during automation manager initialization.

HSAM1330I *request* **REQUEST COMPLETED SUCCESSFULLY ON** *agent*.

Explanation: An automation manager request has successfully completed on an automation agent.

The variable *request* shows the request that was completed.

The variable *agent* shows the name of the automation agent.

System action: The SA z/OS automation manager instance continues normal processing relative to the identified automation agent.

Operator response: None.

System programmer response: None.

HSAM1331I *request* **REQUEST FOR** *agent* **FAILED**

Explanation: An automation manager request failed on an automation agent.

The variable *request* shows the request that failed.

The variable *agent* shows the name of the automation agent.

System action: The automation manager action depends on the request that failed and on the reason why the request was issued. In general however, the actions are as follows:

- Load_ACF Request

The affected agent is unable to initialize or perform automation until the cause of the failure is identified and resolved. The most common cause of this failure is a mismatch between the ACF available to the agent and the current automation manager configuration. Additional automation agent messages may provide more detailed information about the reason for the failure.

- Check_ACF request

This request is normally issued during a configuration refresh operation to verify that the new configuration is available to all automation agents. The configuration refresh operation is terminated as a result of this condition.

- Refresh_ACF

This request is normally issued at the end of a configuration refresh operation to activate the revised ACF. The affected agent should continue to operate in a compatible manner with the new configuration using the previously active ACF data. Thus, newly

added resources will not be automated and existing resources will be automated using the previously defined automation policy.

Operator response: Contact your system programmer. When the problem has been resolved, reissue a configuration refresh command to retry the failed request.

System programmer response: The most common cause of Load_ACF failure is that the ACF available to the automation agent is incompatible with the current manager configuration. Similarly, a Check_ACF failure often indicates an incompatibility between the available ACF and the new automation manager configuration. Other possible causes include corruption of the ACF content and various other problems verifying its content. The reporting automation agent should provide additional information about the reason for the failure.

The failure of a Refresh_ACF request indicates problems were encountered while activating the changes contained in the new configuration. Any resources that were deleted or added by the new configuration will not be automated, and existing resources will be automated using the previous agent policy.

HSAM1332I *response* **INVALID RESPONSE FROM** *agent*, **REASON=***reason*.

Explanation: An invalid response for a request was received from an automation agent.

The variable *response* shows the invalid response that the automation agent issued.

The variable *agent* shows the name of the automation agent that issued the response.

The variable *reason* shows the value of the reason code that the automation agent issued.

System action: If the request is Check_ACF, the associated configuration refresh operation will be terminated. Otherwise, the automation manager operation continues as if the return code had been zero.

Operator response: Report the message to your system programmer. Check whether the identified agent operates correctly. Also check whether any error messages are issued that indicate problems when processing the named request.

System programmer response: This message indicates a software problem. Report the problem to your IBM Support Center.

HSAM1333I *request* **REQUEST PROCESSED ON** *agent*.

Explanation: The automation manager has completed processing of a request.

The variable *request* shows the name of the request.

The variable *agent* shows the name of the automation agent where the request was processed.

Normally, the automation manager then goes back to ready state. However, this message does not confirm that the request was processed successfully. It may also have failed. In case of a failure, an additional message may be issued, for example, if a refresh request fails, message HSAM1337I is issued.

System action: The SA z/OS automation manager instance continues processing normally. Additional messages, for example message HSAM1330I, may be issued due to events that are received later on.

Operator response: None.

System programmer response: None.

HSAM1337I CONFIGURATION REFRESH REQUEST TERMINATED.

Explanation: A configuration refresh request has been terminated due to errors or automation agent incompatibilities. No changes to the automation manager configuration have been implemented and no automation agent updates have been scheduled as a result of the request.

System action: Automation continues using the configuration in effect when the request was issued.

Operator response: Check for other messages indicating the reason for the termination of the request. Determine the cause of the error and correct your data. Then reissue the request.

System programmer response: None.

HSAM1338I CONFIGURATION REFRESH REQUEST COMPLETED WITH AGENT UPDATE ERROR INDICATIONS.

Explanation: A configuration refresh request has completed successfully on the automation manager. However, one or more automation agents have reported problems when implementing the ACF changes. Message HSAM1331I contains more information about the affected agent(s), and the individual agents should have issued messages indicating the problem details.

System action: Automation continues using the new automation manager configuration. The affected agent(s) should continue to operate in a compatible manner with the new configuration using the previously active ACF data. Thus, newly added resources will not be automated. Existing resources will be automated using the previously defined automation policy.

Operator response: Contact your system programmer. When the cause has been determined, issue a configuration refresh command to retry the failed request.

System programmer response: Check for automation agent error messages indicating the cause of the ACF refresh failure and correct your data accordingly.

The failure of a Refresh_ACF request indicates problems were encountered while activating the changes contained in the new configuration. Any resources that were deleted or added by the new configuration will not be automated. Existing resources will be automated using the previous agent policy.

HSAM1339I CONFIGURATION REFRESH REQUEST COMPLETED SUCCESSFULLY.

Explanation: This message is issued when a configuration refresh request has been completed successfully. It indicates that the primary automation manager and all SA z/OS automation agents within the automation sysplex have processed the request and have indicated normal completion of the update process.

System action: Automation continues using the requested configuration.

Operator response: None.

HSAM1360I UNABLE TO RETRIEVE RECOVERY INFORMATION - WARM START RECOVERY IN PROGRESS.

Explanation: A takeover or startup with TYPE=HOT cannot complete because the recovery information needed to do so is either unavailable or has been invalidated.

System action: The automation manager performs a warm start to reach an operational state. Automation continues after the warm start operation completes. This operation may take some time to complete because each automation agent must reconnect with the automation manager and present status for all resources under its control.

Operator response: Any automation manager requests that were incomplete when the previously active PAM terminated may need to be resubmitted.

System programmer response: None.

HSAM1361I UNABLE TO PROCESS RECOVERY INFORMATION - ABNORMAL TERMINATION IN PROGRESS

Explanation: A takeover, ARM restart, or restart with TYPE=HOT cannot complete because the recovery information needed to do so is invalid.

System action: The automation manager instance will be abnormally terminated after invalidating the recovery information. This prevents a subsequent takeover or restart from attempting a retry using the invalid recovery data.

Operator response: If another automation manager instance is available, it should attempt a takeover operation, detect the invalidated restart information, and recover by attempting a warm start (see message HSAM1360I). Otherwise, the current automation manager instance will be restarted by ARM if defined by the installation as eligible for restart. If neither of these two conditions apply, manually restart the automation manager instance specifying either TYPE=WARM or TYPE=HOT.

HSAM1370I TAKEOVER WILL BE DONE FROM TAKEOVER FILE:*dsn*.

Explanation: The automation manager performs a hot start or a takeover from the takeover file with the name specified in *dsn*.

System action: None.

Operator response: None.

System programmer response: None.

HSAM1371I TAKEOVER FILE CANNOT BE READ OR IS CORRUPTED.

Explanation: The automation manager tried to read the takeover file. However, the data set is either not readable or is not a valid takeover file (does not contain a valid identifier).

System action: The automation manager continues with a warm start.

Operator response: None.

System programmer response: Check that the data set has a valid takeover file identifier. The first four bytes must contain the binary value X'12345678'.

HSAM1372I FORCE WARM START DUE TO TAKEOVER FILE OR OPERATOR RESPONSE.

Explanation: The automation manager tried to perform a hot start or a takeover from the takeover file. This action failed and the hot start was degraded to a warm start. The reason for the failure is that the operation code at offset 16 was previously set to 'W' or that it is not valid. This may occur when the previous automation manager instance has invalidated the takeover file, for example, because of an I/O problem or an internal problem.

System action: The automation manager continues with a warm start.

Operator response: None.

System programmer response: None.

HSAM1373I PROBLEM WITH TAKEOVER FILE DETECTED. REASON=*rr*, INFO=*yyy*

Explanation: The automation manager detected one of the following problems regarding the takeover:

- The reason code *rr* describes the problem:

Disabled

The previous PAM has disabled the takeover file. The takeover file might therefore be outdated.

IOError

The previous PAM has detected an I/O error for the takeover file. The takeover file might therefore be outdated.

Mismatch

During takeover a mismatch was detected between the takeover file name in the HSAPRMxx parmlib member and the name that was used by the previous PAM.

Possible Mismatch

During startup of a SAM, a mismatch is detected between the takeover file name in the HSAPRMxx parmlib member and the name that was used by the previous PAM. This message might be an indication that the SAM and PAM do not use the same parmlib member HSAPRMxx. However, there is a time window where the SAM and the PAM are started concurrently where this possible mismatch can occur and can possibly be ignored. This is the case when the pending PAM has not yet finished reading the configuration file and has not yet updated the HSACFGIN accordingly. In any case it is recommended to check that the parmlib members are still valid.

- The variable *yyy*='tkovfile_dsn1, tkovfile_dsn2' where *tkovfile_dsn1* is taken from HSAPRMxx and *tkovfile_dsn2* is the takeover file that was used by the previous PAM.

System action: The automation manager also issues message HSAM1317A and waits for a response. This is not the case for reason 'Possible Mismatch'.

Operator response: Depending on the reason code, proceed as follows:

- If *rr*=Disabled or IOError, check that the takeover file is valid.
- If *rr*=Mismatch, cancel the automation manager and correct the HSAPRMxx parmlib member. See also message HSAM1317A.
- If *rr*=Mismatch and you decide to enter HOT, the automation manager uses the same takeover file as was used by the previous PAM rather than the takeover file name that is specified in the HSAPRMxx parmlib member.
- If *rr*=Mismatch and you decide to enter WARM, the automation manager that is specified in the HSAPRMxx parmlib member.

System action: The automation manager also issues

message HSAM1317A and waits for response.

HSAM1380I *request* **REQUEST ACCEPTED.**

Explanation: A request has been received and validated by the automation manager.

The variable *request* shows which request was received and validated.

System action: The automation manager instance processes the request.

Operator response: None.

System programmer response: None.

HSAM1381I *INVALID verb* **REQUEST PARAMETER,**
request **REQUEST IGNORED.**

Explanation: A parameter specified with a request is invalid for the request verb.

The variable *verb* shows the request verb that one of the parameters specified on the request is invalid for.

The variable *request* shows the request that could not be processed.

System action: The automation manager instance ignores the request.

Operator response: Specify valid parameters and reissue the request.

System programmer response: None.

HSAM1382I **UNSUPPORTED COMMAND VERB,**
request **REQUEST IGNORED.**

Explanation: An unsupported command verb was used to route a request to the automation manager. The automation manager supports only the MVS MODIFY (F) command for request input.

The variable *request* shows the request that could not be processed.

System action: The automation manager instance ignores the request.

Operator response: Contact your system programmer.

System programmer response: The automation manager command interface received the identified request with an MVS command request code other than that used for the MVS Modify command.

HSAM1383I *verb* **REQUEST REJECTED:**
INCORRECT AUTOMATION
MANAGER ROLE OR STATUS.
INVALID verb **REQUEST PARAMETER,**
request **REQUEST IGNORED.**

Explanation: An MVS MODIFY (F) command was routed to an automation manager instance that is not in a role or state compatible with the requested request

verb. Because most automation manager requests run only on the primary automation manager (PAM), it is likely that the command was routed to a secondary instance. Other requests require that the target automation manager be in a "ready" state to be accepted.

The variable *verb* shows the specified request verb.

The variable *request* shows the request that could not be processed.

System action: The automation manager instance ignores the request.

Operator response: Verify that the command was routed to the correct automation manager instance and that the instance is in a ready state. Reissue the command accordingly.

System programmer response: None.

HSAM1384I **INVALID REQUEST** *verb, request*
REQUEST IGNORED. INVALID *verb*
REQUEST PARAMETER, *request*
REQUEST IGNORED.

Explanation: An MVS MODIFY (F) command was received with an invalid request verb.

System action: The automation manager instance ignores the request.

Operator response: Correct the verb and parameter syntax and reissue the request.

HSAM1389I **BUILD ID FOR HSAPIEHM** *xxx yyy*

Explanation: This is an information message for IBM Service. The fields *xxx* and *yyy* provide the compilation date of the event handler, for example, 2009154 152228.

System action: None.

Operator response: None.

HSAM1390E **REPLY "YES" TO CONFIRM OR "NO"**
TO CANCEL SA z/OS AUTOMATION
MANAGER STOP REQUEST.

Explanation: A conditional request has been made to stop an SA z/OS automation manager. No other SA z/OS automation manager instances are available and ready to assume automation sysplex control if this request is allowed to continue. This will result in an interruption of automation services.

System action: The SA z/OS automation manager waits for a response to this request, and continues automation sysplex control until confirmation is received.

Operator response: Reply 'YES' to terminate SA z/OS automation, 'NO' to cancel the stop request. Another SA z/OS automation manager instance may be started

HSAM1391E • HSAM5101I

prior to confirming the request to avoid an interruption of automation services.

System programmer response: None.

HSAM1391E INVALID RESPONSE: REPLY "YES" TO CONFIRM OR "NO" TO CANCEL STOP REQUEST.

Explanation: An invalid response has been given to message HSAP1390E. Refer to the explanation of that message for additional information.

System action: The SA z/OS automation manager waits for a response to this request, and continues automation sysplex control until confirmation is received.

Operator response: Reply 'YES' to terminate SA z/OS automation, 'NO' to cancel the stop request. Another SA z/OS automation manager instance may be started prior to confirming the request to avoid an interruption of automation services.

System programmer response: None.

HSAM1392I AUTOMATION MANAGER STOP REQUEST CANCELLED.

Explanation: A request to confirm an automation manager stop request has been denied. Refer to the explanation of message HSAM1390E for additional information.

System action: The issuing SA z/OS automation manager continues processing.

Operator response: None.

System programmer response: None.

HSAM1393I DEFERRED STOP REQUEST WAITING FOR AUTOMATION AGENT TERMINATION.

Explanation: A deferred stop request was issued to the primary automation manager. Upon the expiration of the stop delay, one or more automation agents were active and no secondary automation manager was available to accomplish a takeover.

System action: The SA z/OS automation manager waits until all automation agents are disconnected or another automation manager instance is started.

Operator response: A deferred stop cannot be canceled. However, starting a new automation manager instance will result in the termination of the current primary manager with a takeover by that new instance.

System programmer response: None.

HSAM1396I SA z/OS AUTOMATION MANAGER SHUTDOWN IN PROGRESS, TYPE=*type*

Explanation: A request has been received to stop an SA z/OS automation manager instance.

The variable *type* indicates the type of shutdown as requested by the initiating command.

System action: The SA z/OS automation manager instance terminates after performing cleanup activities.

Operator response: None

System programmer response: None.

HSAM1399I @1

Explanation: This is an internal message for diagnostic purposes only.

System action: None.

Operator response: None.

System programmer response: None.

HSAM5100E NO SQA STORAGE AVAILABLE.

Explanation: The first automation manager running after an IPL allocates storage in the system queue area (SQA) to hold information shared by all automation managers running on the single image and the component trace itself. This area could not be allocated.

System action: The component trace of all automation managers is disabled.

Operator response: Inform your system programmer.

System programmer response: Increase the system queue area size.

Module: HSAPICB7

HSAM5101I NUMBER OF ACTIVE AMS EXCEEDS THE NUMBER OF ASIDS SUPPORTED FOR THE COMPONENT TRACE.

Explanation: The number of active automation managers exceeds the number of address space IDs supported for the component trace. Therefore, the automation manager issuing this message did not find a free entry in the system queue area (SQA) to store its information.

System action: The automation manager's component trace is disabled.

Operator response: None.

System programmer response: None.

Module: HSAPICB7

HSAM5102E NAME/TOKEN SERVICE FAILED.
RC=*rc*.

Explanation: The name/token pair could not be created or deleted during initialization or termination of the automation manager.

The variable *rc* shows the value of the return code that was issued.

System action: The automation manager's component trace is disabled.

Operator response: None.

System programmer response: Analyze the return code (see IEANTCR/IEANTDL in *z/OS MVS Assembler Service Reference*).

Module: HSAPICB7, HSAPICB8

HSAM5103E MODULE *module* NOT FOUND.

Explanation: The automation manager's component trace initialization routine could not load the required module.

The variable *module* shows the name of the module that could not be loaded.

System action: If this message is followed by message HSAM5109I the component trace is disabled. Otherwise, the automation manager is terminated.

Operator response: Inform your system programmer.

System programmer response: Make the required module available to the automation manager.

Module: HSAPICB7

HSAM5104I SQA ENTRY NOT FOUND.

Explanation: The automation manager's component trace termination routine could not clear its system queue area information.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: HSAPICB8

**HSAM5105E DE-/REGISTRATION TO THE COMPONENT TRACE FAILED. RC=*rc*
RSN=*rsn*.**

Explanation: Registration to or deregistration from the component trace failed.

The variable *rc* shows the return code that was issued.

The variable *rsn* shows the reason code that was issued.

System action: The automation manager's component trace is disabled on registration.

Operator response: Inform your system programmer.

System programmer response: Analyze the return code and the reason code (see CTRACE in the *z/OS Authorized Assembler Service Reference*).

Module: HSAPICB7, HSAPICB8

**HSAM5106I ERROR OCCURRED WHEN WRITING A TRACE RECORD. POS=*position*,
SYSTEM CC=*cc*.**

Explanation: The PC routine writing the trace records abended with the system completion code *cc*.

The variable *position* shows the position inside the module where the abend occurred.

The variable *cc* shows the completion code.

System action: The automation manager's component trace is disabled.

Operator response: Inform your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSAPSTAR

**HSAM5107I ERROR OCCURRED IN START/STOP ROUTINE. POS=*position*, SYSTEM
CC=*cc*.**

Explanation: The automation manager's start/stop routine of the component trace abended with the system completion code *cc*.

The variable *position* shows the position inside the module where the abend occurred.

The variable *completion_code* shows the completion code.

System action: The TRACE command is not processed.

Operator response: Inform your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSAPSTSS

**HSAM5108I COMPONENT *component_name*
ALREADY DEFINED TO THE
COMPONENT TRACE.**

Explanation: An automation manager could not be registered to the component trace because a previous automation manager did not deregister itself.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: HSAPICB7

HSAM5109E NAME/TOKEN SERVICE IS UNAVAILABLE.

Explanation: The MVS name/token services IEANTCR/RT/DL are not available.

System action: The automation manager's component trace is disabled.

Operator response: None.

System programmer response: Make the MVS name/token service available.

Module: HSAPICB7

HSAM5110I Error occurred in display routine. Pos=*position*, System CC=*cc*.

Explanation: The automation manager's display routine of the component trace abended.

The variable *position* shows the position in the module where the abend occurred.

The variable *cc* shows the system completion code that the display routine abended with.

System action: The TRACE command is not processed.

Operator response: Notify your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSAPSCTD

HSAM5120E LISTENER USER EXIT INITIALIZATION FAILED. RC=*rc*.

Explanation: The activation of the System Logger user exit failed with a return code.

The variable *rc* shows the value of the return code.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: If the return code is greater than 4, refer to the return codes of the macro ENFREQ in *z/OS MVS Authorized Assembler Service Reference* for further information.

Module: HSAPICB7, HSAPNCTI

HSAM5121E LISTENER USER EXIT DELETION FAILED. RC=*rc*.

Explanation: The deactivation of the System Logger user exit failed with a return code.

The variable *rc* shows the value of the return code.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: If the return code is greater than 4, refer to the return codes of the macro

ENFREQ in *z/OS MVS Authorized Assembler Service Reference* for further information.

Module: HSAPICB8, HSAPNCTI

HSAM5160I VSAM *request* ERROR OCCURRED. RC=*rc* REASON=*reason*

Explanation: The VSAM macro instruction that corresponds to the request failed.

The variable *request* shows the request that failed.

The variable *rc* shows the return code that was issued.

The variable *reason* shows the reason code that was issued.

System action: Processing terminates.

Operator response: Inform your system programmer.

System programmer response: Check the return and reason codes in *DFSMS/MVS Macro Instructions for Data Sets*, in the chapter about *VSAM Macro Return and Reason Codes*. Correct the error accordingly. Then rerun the program with the PARM parameter 'RERUN'.

Module: HSAPSIPL

Destination: None.

TEC: NO

HSAM5170I UNKNOWN TRACE OPTION FOUND: *option*

Explanation:

The variable *option* shows the trace option specified on the OPTIONS parameter of the IPCS CTRACE command. The following are valid values:

- Communication
- Event
- Instruction
- Lock
- Schedule
- Workitem

System action: Processing continues.

Operator response: Correct or delete the option.

System programmer response: None.

Module: HSAPSTUE

HSAM5171I UNKNOWN KEYWORD FOUND: *keyword*.

Explanation: The parser found a keyword that is not valid.

The variable *keyword* shows the specified filter option.

The following are valid values:

- AM

- CALLID
- ITEM
- KEY
- METHOD
- MODULE
- TASKID

System action: The keyword is ignored.

Operator response: Specify a correct keyword.

System programmer response: None.

Module: HSAPSTUE

HSAM5172I TOO MANY VALUES SPECIFIED FOR KEYWORD *keyword*.

Explanation: All keyword options except *AM* on the *OPTIONS* parameter support up to five different values. However, the parser found at least one additional value for keyword option *keyword*.

The variable *keyword* shows the specified filter option.

System action: The values that exceed the number of five are ignored.

Operator response: Correct the keyword option.

System programmer response: None.

Module: HSAPSTUE

HSAM5173I *N* (ST/ND/RD/TH) VALUE OF KEYWORD *keyword* IS EMPTY.

Explanation: The parser expects a string of at least one character for the keyword option *keyword* on the *OPTIONS* parameter. However, it found no string, or a string representing the NULL string.

The variable *N* shows the number of missing values.

The variable *keyword* shows the specified filter option.

System action: The value is ignored.

Operator response: Correct the value.

System programmer response: None.

Module: HSAPSTUE

HSAM5174I VALUE *value* OF KEYWORD *keyword* IS NOT NUMERIC.

Explanation: The parser expects an integer for the keyword option *keyword* on the *OPTIONS* parameter. However, it found a non-numeric value.

The variable *value* shows the value that is not numeric.

The variable *keyword* shows the specified filter option.

System action: The value is ignored.

Operator response: Correct the value.

System programmer response: None.

Module: HSAPSTUE

HSAM5175I PARSING OF THE OPTIONS PARAMETER STOPPED AT COLUMN *col*.

Explanation: A column value greater than 0 indicates one of the following:

- A keyword is not spelled correctly.
- A keyword is incorrectly delimited.
- A trace option is incorrectly delimited.

A column value of 0 can be caused by one of the following:

- A parameter consisting of blanks and commas only
- A parameter length greater than 32K
- An allocation error of temporary storage

The variable *col* shows which column the parser stopped at.

System action: Processing terminates.

Operator response: Correct the *OPTION* parameter, or increase the region size.

System programmer response: None.

Module: HSAPSTUE

HSAM5185I THE EXTERNAL WRITER HAS BEEN STARTED AUTOMATICALLY.

Explanation: The start/stop routine detected that the external writer is not active. Because the automation manager's component trace requires an active external writer, the start/stop routine issued the appropriate *TRACE* command.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: HSAPSTSS

HSAM5186I ENVIRONMENTAL ERROR OCCURRED DURING POST OF THE LAST ACTIVE AM.

Explanation: *POST* of the automation manager waiting for the completion of the start/stop routine failed.

System action: Processing continues. But the automation manager is still waiting.

Operator response: Stop the automation manager manually.

System programmer response: None.

Module: HSAPSTSS

HSAM5187I PARAMETER WTR= IS MISSING.

Explanation: The start/stop routine detected that the WTR parameter is missing in the reply to the preceding TRACE command.

System action: The TRACE command is discarded.

Operator response: Re-issue the TRACE command and add the WTR parameter to the reply.

System programmer response: None.

Module: HSAPSTSS

HSAM5188I PARSING OF TRACE OPTIONS FAILED. RC=*rc*.

Explanation: The start/stop routine scheduled in the *MASTER* address space encountered a storage problem when parsing the OPTION parameter.

The variable *rc* shows the value of the return code.

System action: The TRACE command is discarded.

Operator response: Try the command later. If the problem persists, contact your IBM Support Center.

System programmer response: None.

Module: HSAPSTSS

HSAM5189E ACCESS TO DATA SPACE FAILED. RC=*rc*.

Explanation: The data space where the trace records are temporarily stored could not be accessed.

The variable *rc* shows the value of the return code.

System action: At least the last trace buffer is lost when the external writer is stopped.

Operator response: Inform your system programmer.

System programmer response: Analyze the return code (see ALESERV in *z/OS Authorized Assembler Service Reference*).

Module: HSAPSTSS

HSAM5190E TCB TOKEN OF PARENT TASK COULD NOT BE OBTAINED. RC=*rc*.

Explanation: The TCB token of the parent task of the start/stop routine could not be obtained.

The variable *rc* shows the value of the return code.

System action: The TRACE command is discarded.

Operator response: Inform your system programmer.

System programmer response: Analyze the return code (see TCBTOKEN in the *z/OS Authorized Assembler Service Reference*).

Module: HSAPSTSS

HSAM5191I PARAMETER WTR=DISCONNECT IS NOT SUPPORTED ON A START TRACE COMMAND.

Explanation: The start/stop routine expects the name of the startup procedure of the external writer to be used. But it found the reserved word DISCONNECT.

System action: The TRACE command is discarded.

Operator response: Reissue the TRACE command and correct the WTR parameter.

System programmer response: None.

Module: HSAPSTSS

HSAM5192I INVALID TRACE OPTION SPECIFIED.

Explanation: The specified trace options could not be determined because of a parsing error.

System action: The TRACE command is rejected.

Operator response: Reissue the TRACE command and correct the OPTION parameter. Valid options are *ALL* or a combination of :

- [C]ommunication
- [E]vent
- [I]nstruction
- [L]ock
- [S]chedule
- [W]orkitem

The options must be separated by a comma.

System programmer response: None.

Module: HSAPSTSS

HSAM5193I NO TRACE OPTION SPECIFIED.

Explanation: Either the OPTION parameter was omitted, or no value was specified for it.

System action: The TRACE command is rejected.

Operator response: See message HSAM5192I for further information.

System programmer response: None.

Module: HSAPSTSS

HSAM5194I UNKNOWN TRACE OPTION FOUND: *option*.

Explanation: The start/stop routine found an unsupported trace option.

The variable *option* shows the specified trace option.

System action: The TRACE command is rejected.

Operator response: See message HSAM5192I.

System programmer response: None.

Module: HSAPSTSS

HSAM5195E SQA HAS BEEN CORRUPTED.
EYE-CATCHER=*string*.

Explanation: The start/stop routine detected that the system queue area has been overwritten by another task or address space.

System action: The TRACE command is discarded.

Operator response: Inform your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSAPSTSS

HSAM5196E NAME/TOKEN *token_name* COULD NOT BE FOUND.

Explanation: The start/stop routine could not obtain the address of the system queue area (SQA). If the message is issued as a response to a STOP command, the data space that holds the trace buffers before they are passed to the external writer, can no longer be deleted.

The variable *token_name* shows the specified token name.

System action: The TRACE command is discarded.

Operator response: Inform your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSAPSTSS

HSAM5197E RETRIEVING NAME/TOKEN *token_name* FAILED. RC=*rc*.

Explanation: The start/stop routine encountered an error when calling the MVS service IEANTRT. If the message is issued as a response to a STOP command, the data space that holds the trace buffers before they are passed to the external writer, can no longer be deleted.

The variable *token_name* shows the specified token name.

The variable *rc* shows the value of the return code.

System action: The TRACE command is discarded.

Operator response: Inform your system programmer.

System programmer response: Analyze the return code (see IEANTRT in *z/OS MVS Authorized Assembler Service Reference*).

Module: HSAPSTSS

HSAM5198E DATA SPACE CREATION FAILED.
RC=*rc* RSN=*rsn*.

Explanation: The start/stop routine could not create the data space holding the trace buffers before they are passed to the external writer.

The variable *rc* shows the value of the return code.

The variable *rsn* shows the value of the reason code.

System action: The TRACE command is discarded.

Operator response: Inform your system programmer.

System programmer response: Analyze the return code and the reason code (see DSPSERV in the *z/OS MVS Authorized Assembler Service Reference*).

Module: HSAPSTSS

HSAM5199I LAST TRACE BUFFER COULD NOT BE WRITTEN. RC=*rc* RSN=*rsn*.

Explanation: The start/stop routine encountered an error when writing the last trace buffer to the external writer synchronously.

The variable *rc* shows the value of the return code.

The variable *rsn* shows the value of the reason code.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: Analyze the return code and the reason code (see CTRACEWR in *z/OS Authorized Assembler Reference*).

Module: HSAPSTSS

HSAM5200E UNKNOWN KEYWORD FOUND:
keyword.

Explanation: A library member (see the preceding IEE252I message) or the PARM parameter of the startup procedure contains a keyword that is not known to the automation manager.

The variable *keyword* shows the value of the unknown keyword.

System action: The keyword is ignored.

Operator response: Inform your system programmer.

System programmer response: Correct or delete the keyword depending on where it is specified.

Module: HSAPSPLM

HSAM5201E INVALID KEYWORD FOUND: *keyword*.

Explanation: A library member (see the preceding IEE252I message) or the PARM parameter of the automation manager's startup procedure contains a keyword that was not spelt correctly.

The variable *keyword* shows the value of the invalid keyword.

HSAM5202E • HSAM5207I

System action: The keyword is ignored.

Operator response: Inform your system programmer.

System programmer response: Correct the keyword depending on where it is specified.

Module: HSAPSPLM

**HSAM5202E INVALID KEYWORD VALUE
DETECTED: KEYWORD=*keyword*
Value=*value*.**

Explanation: A library member (see the preceding IEE252I message) or the PARM parameter of the automation manager's startup procedure contains an invalid keyword.

The variable *keyword* shows the name of the keyword.

The variable *value* shows the value of the invalid keyword.

System action: The keyword is ignored.

Operator response: Inform your system programmer.

System programmer response: Correct the value of the keyword depending on where it is specified.

Module: HSAPSPLM

**HSAM5203E VALIDATION ROUTINE *module* NOT
FOUND.**

Explanation: A keyword validation routine could not be loaded.

The variable *module* shows the name of the module.

System action: The keyword is ignored.

Operator response: Inform your system programmer.

System programmer response: Make sure that the SA z/OS library SINGMOD1 is in the STEPLIB concatenation of the automation manager startup procedure and that the module is catalogued in this library.

Module: HSAPSPLM

**HSAM5204E INVALID KEYWORD DETECTED AT
column IN *string*.**

Explanation: The parse service detected an invalid delimiter of a keyword. The invalid delimiter was found either in a library member (see the preceding IEE252I message) or in the PARM parameter of the automation manager's startup procedure.

The variable *column* shows the column where the invalid delimiter was detected.

The variable *string* shows the string that contains the invalid delimiter.

System action: All keywords of the string *string* are ignored.

Operator response: Inform your system programmer.

System programmer response: Correct the delimiter depending on where it is specified.

Module: HSAPSPLM

HSAM5205I PROCESSING MEMBER *member*.

Explanation: The specified library member (see the preceding IEE252I message) is being processed.

The variable *member* shows the name of the member being processed.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: HSAPSPLM

HSAM5206E MEMBER *member* NOT FOUND.

Explanation: The specified library member resides neither in the library concatenation of DD name HSAPLIB when this message is preceded by message IEF761I nor in the library of the PARMLIB concatenation.

The variable *member* shows the name of the member that could not be found.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: Either make sure that the member is available or change the value of the keyword *member* to a suffix that is available. If the default member HSAPRM00 cannot be found, copy it from the SA z/OS library SINGSAMP to a library of the appropriate concatenation.

Module: HSAPSPLM

**HSAM5207I PARMLIB SERVICE FAILED WHEN
READING *member*, RC=*rc* Reason=*rsn*.**

Explanation: During initialization of the automation manager the MVS PARMLIB service failed.

The variable *member* shows the name of the member being processed when the MVS PARMLIB service failed.

The variable *rc* shows the value of the return code.

The variable *rsn* shows the value of the reason code.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: Refer to z/OS MVS Assembler Services Reference for information about the reason and return codes.

Module: HSAPSPLM

**HSAM5208I INVALID STRING FOUND IN *member*:
string.**

Explanation: The parser service detected more than one keyword in a single line of a member.

The variable *member* shows the name of the member where the error occurred.

System action: All keywords specified in the string are ignored.

Operator response: Inform your system programmer.

System programmer response: Correct the line in the library member *member* (see the preceding IEE252I message)

Module: HSAPSPLM

HSAM5209I MEMBER *member* TOO LARGE:

Explanation: The indicated member contains more than 10240 lines of data.

The variable *member* shows the name of the member being processed using the MVS PARMLIB service.

System action: The data is truncated after line 10239.

Operator response: None.

System programmer response: None.

Module: HSAPSIPL

HSAM5210I MEMBER *member* EMPTY:

Explanation: The indicated member has no data.

The variable *member* shows the name of the member being processed using the MVS PARMLIB service.

System action: None.

Operator response: None.

System programmer response: None.

Module: HSAPSIPL

HSAM5211I MEMBER *member* FOUND IN *dsn*.

Explanation: The indicated member has been found in the indicated data set of the PARMLIB concatenation.

The variable *member* shows the name of the member being processed using the MVS PARMLIB service.

The variable *dsn* shows the data set name that the member has been read from.

System action: None.

Operator response: None.

System programmer response: None.

Module: HSAPSIPL

**HSAM5212E I/O ERROR READING MEMBER
member.**

Explanation: An I/O error occurred when trying to read the indicated PARMLIB member.

The variable *member* shows the name of the member being processed using the MVS PARMLIB service.

System action: None.

Operator response: Inform your system programmer.

System programmer response: Correct the problem and re-run the program.

Module: HSAPSIPL

**HSAM5213E VALUE MISSING FOR KEYWORD
keyword.**

Explanation: A library member (see the preceding IEE252I message) or the PARM parameter of the automation manager's startup procedure contains a keyword that has no value defined.

The variable *keyword* shows the name of the keyword.

System action: The keyword is ignored.

Operator response: Inform your system programmer.

System programmer response: Correct the value of the keyword depending on where it is specified.

Module: HSAPSPLM

**HSAM5214I THE NEXT MESSAGES APPLY TO
MEMBER *member*.**

Explanation: The library member (see the preceding IEE252I message) contains at least one improper specification. This message is followed by further HSAM52nn messages.

The variable *member* shows the name of the member being processed using the MVS parmlib service.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: HSAPSPLM

**HSAM5215I THE NEXT MESSAGES APPLY TO
PARM STRING.**

Explanation: The PARM parameter of the startup procedure contains at least one improper specification. This message is followed by further HSAM52nn messages.

System action: Processing continues.

Operator response: None.

System programmer response: None.

HSAM5300E • HSAM5401E

Module: HSAPSPLM

HSAM5300E MESSAGE MODULE *module* NOT FOUND.

Explanation: The automation manager initialization routine could not load the required message module.

The variable *module* shows the name of the module that could not be found.

System action: The automation manager terminates.

Operator response: Inform your system programmer.

System programmer response: Make sure that the SA z/OS library SINGMOD1 is the STEPLIB concatenation of the automation manager startup procedure and that the module is catalogued in this library. Then, restart the automation manager.

Module: HSAPICB9

HSAM5301E MESSAGE *msgno* NOT FOUND IN MODULE *module*.

Explanation: The message service of SA z/OS could not find the indicated message number in the message module.

The variable *msgno* shows the message number that could not be found in the indicated message module.

The variable *module* shows the name of the module that does not contain the indicated message number.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: Make sure that the indicated module is in the correct SA z/OS library.

Module	Library	Concatenation
HSAAMME	SINGMOD2	LNKLST
HSAAMLE	SINGMOD1	STEPLIB
INGAMYE	SINGMOD2	LNKLST

Verify that this library is part of the corresponding concatenation. In case two versions of SA z/OS exist in the concatenation ensure that the latest version precedes the older version.

Module: HSAPSMMSG

HSAM5400I *logname* LOG REQUEST *request* FAILED. RC=*rc*, RSN=*reason*.

Explanation: The request *request* to the log *logname* ended with the unexpected reason code *reason*.

System action: Processing continues. Depending on the reason code the affected log is marked either permanently unavailable:

- 0409

- 040A
- 080C
- 080D
- 0813
- 081A
- 081F
- 0820
- 082E
- 0843
- 0850
- 0851
- 0853
- 085D
- 0890
- 08E2
- 08E3

or temporarily unavailable:

- 0406
- 0860
- 0861
- 0862
- 0863
- 0864
- 0865
- 0867
- 0868
- 0891

Operator response: Inform your system programmer.

System programmer response: See “return and reason codes” of the related IXG... macro in z/OS MVS *Authorized Assembler Service Reference* for further analysis. In case the log is marked permanently unavailable and the problem can be solved, start a secondary automation manager and initiate a takeover. In addition, recycle the affected NetView(s).

Module: HSAPSSLG

HSAM5401E SYSTEM LOGGER HAS INSUFFICIENT ACCESS AUTHORITY TO *log*. COMPONENT TRACE.

Explanation: The system logger address space does not have access authority to the coupling facility structure associated with the log stream *log*.

System action: Processing continues. The log is marked permanently unavailable.

Operator response: Inform your system programmer.

System programmer response: Make sure the system logger address space has SAF access to the structure. Then, start a secondary automation manager and initiate a takeover. In addition, recycle the affected NetView(s).

Module: HSAPSSLG

HSAM5402I STRUCTURE *STRNAME* IS NOT DEFINED IN THE CFRM POLICY.

Explanation: The structure *strname* is not defined in the currently active CFRM policy.

System action: Processing continues. The log is marked permanently unavailable.

Operator response: Inform your system programmer.

System programmer response: Define the structure *strname* in the policy and activate the policy. Then, start a secondary automation manager and initiate a takeover. In addition, recycle the affected NetView(s).

Module: HSAPSSLG

HSAM5403I STREAM *streamname* IS NOT DEFINED IN THE LOGR POLICY.

Explanation: The log stream name *streamname* has not been defined in the LOGR policy.

System action: Processing continues. The log is marked permanently unavailable.

Operator response: Inform your system programmer.

System programmer response: Define the log stream *strname* in the LOGR policy. Then, start a secondary automation manager and initiate a takeover. In addition, recycle the affected NetView(s).

Module: HSAPSSLG

HSAM5404I XES FAILED. DIAG1=*XES_RC*, DIAG2=*XES_RSN*.

Explanation: A severe cross-system extended services (XES) error has occurred.

System action: Processing continues. The affected log is marked permanently unavailable.

Operator response: Inform your system programmer.

System programmer response: Analyze the XES return code and the XES reason code.

Module: HSAPSSLG

HSAM5405I *text*

Explanation: The message is issued in conjunction with message HSAM5400I and reason code 081F. The reason code indicates that the system logger encountered an internal problem while processing the LOGR couple data set. The two or more messages represent the content of the 'answer area' that must be provided by each logger request.

System action: None.

Operator response: Inform your system programmer.

System programmer response: Contact your IBM Support Center.

Module: HSAPSSLG

HSAM5406I Some blocks of log *logname* have been deleted.

Explanation: The message is issued in conjunction with message HSAM5400I and the reason codes 0408 and 085C. To satisfy a write log request, log blocks older than six days have been deleted. The deletion is repeated day by day until the free space is large enough to satisfy the write request.

The variable *logname* shows the name of the log that blocks have been deleted from.

System action: None.

Operator response: None.

System programmer response: None.

Module: HSAPSSLG

HSAM5407I *logname* LOG IS NOT AVAILABLE.

Explanation: The system logger address space is not available for the remainder of this IPL.

System action: Processing continues. All logs are marked permanently unavailable.

Operator response: None.

System programmer response: None.

Module: HSAPSSLG

HSAM5408I *logname* LOG WAS TEMPORARILY UNAVAILABLE FROM *from_time* TO *to_time*.

Explanation: While the specified log was temporarily unavailable, a WRITE request to the log had to be rejected. After the log has become available again, this message is written to the log documenting that one or more WRITE requests were not logged.

The variable *logname* shows the name of the log that was not available.

The variables *from_time* and *to_time* show the time span during which the log was not available.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: HSAPSSLG

HSAM5409I *logname* LOG HAS BEEN DELETED FROM THE FIRST GAP.

Explanation: The message is issued in conjunction with message HSAM5400I and the reason codes 0402, 0403, 0836, and 084B. The log has been deleted from the first gap being detected when reading the log from

HSAM5410E • HSAM5411I

the most recent data to the earliest data. One reason could be that the log stream has been defined with an incorrect VSAM share option, for example, SHR(1 3).

System action: None.

Operator response: Inform your system programmer.

System programmer response: Check and correct the VSAM definitions of the affected log stream.

Module: HSAPSSLG

HSAM5410E *logname* LOG REQUEST *request* FAILED.
RC=*rc*, RSN=*reason*, Type=*type*.

Explanation: The request *request* to the log *logname* ended with the unexpected reason code *reason*.

System action: Processing continues. The log *logname* is marked permanently unavailable.

Operator response: Inform your system programmer.

System programmer response: Check the system log for associated IXG prefixed messages. Check the relevant zOS MVS System Messages manual for the IXG message details. If you cannot resolve the problem, contact your IBM Support Center.

Module: HSAPSSLG

HSAM5411I *logname* LOG HAS BEEN DELETED.

Explanation: The message is issued in conjunction with message HSAM5400I and the reason codes 0405, 0407, 0408, and 085C. The whole log must be deleted to satisfy a write log request.

System action: None.

Operator response: None.

System programmer response: None.

Module: HSAPSSLG

Chapter 8. Messages IHVA100I to IHVZ760

IHVA100I **Cannot UNALLOCATE/DETACH device number *bb* (LSN *aa*).**

Explanation: I/O operations sent an UNALLOCATE (MVS) or DETACH (VM) for the specified switch, which failed. This occurs when an attempt is made to close a switch that does not have a specified DDNAME allocated, or the switch was never opened.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

User response: This message is informational. Actions proceed. Check whether the DDNAME associated with the switch device number is still allocated.

IHVA200I **Error detected on switch *aa*.**

Explanation: I/O operations has detected an error on the indicated switch. This only appears on the message log.

The variable *aa* shows the switch device number.

User response: None.

IHVA201I **Switch *aa* Port *bb* state changed.**

Explanation: I/O operations detected a change in the status of the indicated port. This appears only on the message log.

The variable *aa* shows the switch number.

The variable *bb* shows the port address.

User response: None.

IHVB001I **Elapsed times**

| IHVCnnn Messages without a message type indicate that these messages are displayed on console
| without its message ID.

IHVC000I ***aa* command successful.**

Explanation: The command completed successfully.

The variable *aa* shows the name of the command.

User response: None.

IHVC001I ***aa* command successful, RC=*bb* RSN=*cc*.**

Explanation: The command completed, but some condition was detected that caused a nonzero return code. For example, if the FORCE option was used and a host voted no, the command would still be processed.

The variable *aa* shows the name of the command.

The variable *bb* shows the return code that was issued.

Explanation: This message is the header message of a series of IHVB002I messages.

User response: None.

IHVB002I ***aa bb:cc:dd.ee***

Explanation: This message shows the elapsed time of a particular module that the module has consumed during the startup process of I/O operations.

The variable *aa* shows the module's short name.

The variables *bb*, *cc*, *dd*, and *ee* show the elapsed time.

User response: None.

IHVB900I **STORAGE OBTAIN failed in *aa*.**

Explanation: There has been a STORAGE OBTAIN failure. This could have been caused by an operating system problem.

The variable *aa* shows the name of the module.

System programmer response: Determine why there was not enough storage and correct the condition. Restart I/O operations.

IHVB990I **STORAGE RELEASE failed in *aa*.**

Explanation: There has been a STORAGE RELEASE failure. The error indicates a system storage problem.

The variable *aa* shows the name of the module.

User response: Report the problem to your IBM Support Center.

The variable *cc* shows the reason code that was issued.

User response: Refer to the subsequent message, or use the DISPLAY RESULTS command to determine what condition was detected that caused the nonzero return code.

IHVC002I ***aa* command failed, RC=*bb* RSN=*cc*.**

Explanation: The command failed.

The variable *aa* shows the name of the command.

The variable *bb* shows the return code.

The variable *cc* shows the reason code.

User response: Use the DISPLAY RESULTS command to determine why the command failed. Correct the

condition and reenter the command.

IHVC080I **Command not supported by this host's *aa* level**

Explanation: The command failed because it is not supported at this level of operating environment.

The variable *aa* shows the MVS release.

User response: None.

IHVC100I *aa* failed *bb* authority for *cc*.

Explanation: The specified user does not have the specified access authority for the command

The variable *aa* shows the user ID.

The variable *bb* shows the authorization level (read, update, control).

The variable *cc* shows the RACF entity name.

User response: Contact your security administrator for the proper authorization.

IHVC200I *aa* *bb* *cc*

Explanation: This message is the output of Display commands.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

The variable *cc* shows the switch unique identifier.

User response: None.

IHVC201I *aalbb* *cc* *dd*

Explanation: This message is the output of a DISPLAY HOST command.

The variable *aa* shows the VTAM name.

The variable *bb* shows the TCP/IP host name.

The variable *cc* shows the name of the sysplex.

The variable *dd* shows the name of the system.

User response: None.

IHVC202I *aa* *bb* *cc* *dd* *ee* *ff*

Explanation: This message is the output of a Display command.

The variable *aa* shows the VTAM name.

The variable *bb* shows the Version of I/O operations on this host (format Vx).

The variable *cc* shows the Release of I/O operations on this host (format Ry).

The variable *dd* shows the physical ID of this host.

The variable *ee* shows the name of the sysplex.

The variable *ff* shows the name of the system.

User response: None.

IHVC204I **BACKOUT**

VARYs: *aa* *bb* (*cc,dd*)(*ee,ff*)(*gg,hh*)

Explanation: This message is the output of a DISPLAY HOST command.

The variable *aa* shows the number of attempted Varys.

The variable *bb* shows the number of failed Varys.

The variables *cc*, *ee* and *gg* show the device number.

The variables *dd*, *ff* and *hh* show the CHPID.

User response: None.

IHVC205I **Cannot CHAIN a port to itself**

Explanation: An attempt was made to chain a port to itself. The CHAIN command was specified with the same port on the same switch. This is not allowed.

User response: Check the parameters on the CHAIN command.

IHVC206I *aa* *bb*

Explanation: This message is the output of display commands.

The variable *aa* shows the TCP/IP host name.

The variable *bb* shows the IP address of this host, either in IPv4 format or IPv6 format.

User response: None.

IHVC210 **Connected to:** *aa* *bb* *cc*

Explanation: This message is the output of a DISPLAY CHP command.

The variable *aa* shows the port.

The variable *bb* shows the port status.

The variable *cc* shows the port name.

User response: None.

IHVC211 **Chained to:** *aa* *bb* *cc* *dd* *ee*

Explanation: This message is the output of a DISPLAY CHP command.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

The variable *cc* shows the port.

The variable *dd* shows the port status.

The variable *ee* shows the port name.

User response: None.

IHVC212

(aa,bb)(cc,dd)(ee,ff)(gg,hh)

Explanation: This message is the output of a DISPLAY VARY command.

The variables *aa*, *cc*, *ee* and *gg* show the device number.

The variables *bb*, *dd*, *ff* and *hh* show the CHPID.

User response: None.

IHVC213 *aa caused VARYs by bb on cc*

Explanation: This message is a header line for the DISPLAY VARY command, which gives information about the status of the previous operation that caused a configuration change. This message gives information regarding the last command entered and the user who entered it.

The variable *aa* shows the last command entered.

The variable *bb* shows the user ID of the user who entered the command.

The variable *cc* shows the system ID where the last command was entered.

User response: None.

IHVC214 ONLINE *aa bb (cc,dd,"ee")*

Explanation: This message is the output of a DISPLAY VARY command.

The variable *aa* shows the number of attempted Varys.

The variable *bb* shows the number of failed Varys.

The variable *cc* shows the device number.

The variable *dd* shows the CHPID.

The variable *ee* shows the reason why VARY failed.

User response: None.

IHVC215 BACKOUT *aa bb (cc,dd,"ee")*

Explanation: This message is the output of a DISPLAY VARY command.

The variable *aa* shows number of attempted Varys.

The variable *bb* shows the number of failed Varys.

The variable *cc* shows the device number.

The variable *dd* shows the CHPID.

The variable *ee* shows the reason why VARY failed.

User response: None.

IHVC220 Cntl unit data:*aa bb cc dd ee*

Explanation: This message is the output of a DISPLAY DEV command.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

The variable *cc* shows the port.

The variable *dd* shows the port status.

The variable *ee* shows the port name.

User response: None.

IHVC221

Port

Explanation: This message is the output of a DISPLAY CHP command.

User response: None.

IHVC222 *aa connected to: bb cc dd ee*

Explanation: This message is the output of a DISPLAY NAME or DISPLAY PORT command.

The variable *aa* shows the midport of chain.

The variable *bb* shows the port name.

The variable *cc* shows the port.

The variable *dd* shows the port status.

The variable *ee* shows the port I/O definition.

User response: None.

IHVC223 *aa chained to: bb
cc dd ee ff*

Explanation: This message is the output of a DISPLAY NAME or DISPLAY PORT command.

The variable *aa* shows the midport of chain.

The variable *bb* shows the switch device number.

The variable *cc* shows the logical switch number.

The variable *dd* shows the port.

The variable *ee* shows the port status.

The variable *ff* shows the port I/O definition.

User response: None.

IHVC224 *aa connected to: bb cc dd*

Explanation: This message is the output of a DISPLAY CHP command.

The variable *aa* shows the midport of chain.

The variable *bb* shows the port.

The variable *cc* shows the port status.

The variable *dd* shows the port name.

User response: None.

IHVC225 *aa chained to:bb cc dd ee ff*

Explanation: This message is the output of a DISPLAY CHP command.

The variable *aa* shows the midport of chain.

The variable *bb* shows the switch device number.

The variable *cc* shows the logical switch number.

IHVC230I • IHVC243

The variable *dd* shows the port.

The variable *ee* shows the port status.

The variable *ff* shows the port name.

User response: None.

IHVC230I **Switch(es) reported an error, display data may be inaccurate.**

Explanation: This message is the output of a DISPLAY command when a switch reported an error.

User response: Enter **DISPLAY SWITCH *** to display the status of all switches.

IHVC231 *aa* caused CONFIGs by *bb* on *cc*

Explanation: This message is a header for the DISPLAY VARY command.

The variable *aa* shows the last command entered.

The variable *bb* shows the user ID of the user who entered the command.

The variable *cc* shows the system ID where the last command was entered.

User response: None.

IHVC232 **Config type Tried Failed 1st *aa* failures (Chp)**

Explanation: This is a header for the DISPLAY VARY command.

The variable *aa* shows the number of Configs displayed.

User response: None.

IHVC233 **OFFLINE** *aa* *bb* (*cc,dd,ee,ff,gg,hh,ii,jj*)

Explanation: This message is the output of the DISPLAY VARY command.

The variable *aa* shows the number of attempted Configs.

The variable *bb* shows the number of failed Configs.

The variables *cc, dd, ee, ff, gg, hh, ii, and jj* show the CHPIDs.

User response: None.

IHVC234 **BACKOUT** *aa* *bb* (*cc,dd,ee,ff,gg,hh,ii,jj*)

Explanation: This message is the output of the DISPLAY VARY command.

The variable *aa* shows the number of attempted Configs.

The variable *bb* shows the number of failed Configs.

The variables *cc, dd, ee, ff, gg, hh, ii, and jj* show the CHPIDs.

User response: None.

IHVC235 **ONLINE** *aa* *bb* (*cc,dd,ee,ff,gg,hh,ii,jj*)

Explanation: This message is the output of the DISPLAY VARY command.

The variable *aa* shows the number of attempted Configs.

The variable *bb* shows the number of failed Configs.

The variables *cc, dd, ee, ff, gg, hh, ii, and jj* show the CHPIDs.

User response: None.

IHVC240 **A/P Port I/O-Def Port name**

Explanation: This is a header for the DISPLAY NAME or DISPLAY PORT command.

User response: None.

IHVC241 *aa* *bb* *dd* *cc*

Explanation: This message is the output of a DISPLAY NAME or DISPLAY PORT command.

The variable *aa* shows either A or P (allowed or prohibited).

The variable *bb* shows the port.

The variable *cc* shows the port name.

The variable *dd* shows the I/O-defined type.

User response: None.

IHVC242I **There are no port names on switch *aa*.**

Explanation: There are no port names on the specified switch.

The variable *aa* shows the switch device number.

User response: None.

IHVC243 **Chained to:** *aa*
bb *cc* *dd* *ee*

Explanation: This message is the output of a DISPLAY command.

The variable *aa* shows the switch device number.

The variable *bb* shows logical switch number.

The variable *cc* shows the port.

The variable *dd* shows port status.

The variable *ee* shows the port I/O definition.

User response: None.

IHVC244I

(aa,bb,cc,dd,ee,ff,gg,hh)

Explanation: This message is the output of the DISPLAY VARY command.

The variables *aa*, *bb*, *cc*, *dd*, *ee*, *ff*, *gg*, and *hh* show the CHPIDs.

User response: None.

IHVC270I **No FAILED VARYs or CONFIGs detected on previous command.**

Explanation: This message is the output of a DISPLAY command.

User response: None.

IHVC271 **SYNC SWITCH detected VARYs by *aa* on *bb***

Explanation: This message is a header for the DISPLAY VARY command. It informs the user that the SYNC SWITCH command was performed with the NOPROCESS option and Varys were detected.

The variable *aa* shows the user ID of the user who entered the command.

The variable *bb* shows the system ID where the last command was entered.

User response: None.

IHVC272 **SYNC SWITCH detected CONFIGs by *aa* on *bb***

Explanation: This message is a header for the DISPLAY VARY command. It informs the user that the SYNC SWITCH command was performed with the NOPROCESS option and ESCON Converter Model 1 CHPIDs were detected.

The variable *aa* shows the user ID of the user who entered the command.

The variable *bb* shows the system ID where the last command was entered.

User response: None.

IHVC273 **Swch LSN Unique id**
 Tag Status SC Flags

Explanation: This message is the first header line for the DISPLAY SWITCH * DETAIL command.

Swch	Switch device number
LSN	Logical switch number
Unique id	Node element descriptor
Tag	NED unique identifier
Status	Status information
SC	Status code
Flags	Internal flags

User response: None.

IHVC274 **LIC EC level Sense bytes**
 0-15

Explanation: This message is the second header line for the DISPLAY SWITCH * DETAIL command.

LIC	LIC version and release
EC level	Engineering change level
Sense	Switch sense information

User response: None.

IHVC275 **Ins Imp 1st LP Sense bytes**
 16-31

Explanation: This message is the third header line for the DISPLAY SWITCH * DETAIL command.

Ins	Number of installed ports
Imp	Number of implemented ports
1st	Number of first implemented port
LP	Number of logical paths
Sense	Switch sense information

User response: None.

IHVC276 *aa bb cc dd ee ff gg*

Explanation: This message is the output of a DISPLAY SWITCH * DETAIL command.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number if available.

The variable *cc* shows the node element descriptor (NED).

The variable *dd* shows the NED's unique identifier.

The variable *ee* shows the switch status information.

The variable *ff* shows the switch status code.

The variable *gg* shows the internal status flags.

User response: None.

IHVC277 *aa bb cc dd ee ff*

Explanation: This message is the output of a DISPLAY SWITCH * DETAIL command.

The variable *aa* shows the LIC version and release.

The variable *bb* shows the EC level.

The variables *cc*, *dd*, *ee*, and *ff* show 4 bytes of sense data in hexadecimal format.

User response: None.

IHVC278 *aa bb cc dd ee ff gg hh*

Explanation: This message is the output of a DISPLAY SWITCH * DETAIL command.

The variable *aa* shows the number of installed ports.

The variable *bb* shows the number of implemented ports.

The variable *cc* shows the first implemented port number.

The variable *dd* shows the number of logical paths.

The variables *ee*, *ff*, *gg*, and *hh* show 4 bytes of sense data in hexadecimal format.

User response: None.

IHVC301 Port name *aa* already on this switch.

Explanation: The specified port name is already defined on the switch.

The variable *aa* shows the port name.

User response: Do not use duplicate names on the same switch.

IHVC400I No switches known to I/O-OPS.

Explanation: There are no switches defined in the IOCP data. This means there are no switches available to receive commands. I/O operations cannot be used to send commands to switches. However, some I/O operations display capabilities are still available.

User response: None.

IHVC401I Host(s) unable to process the *aa* command.

Explanation: One or more hosts running I/O operations were unable to process the command. This may be due to the Varys that were requested, VTAM or TCP/IP communication problems, or other conditions preventing the hosts from processing the command.

The variable *aa* shows the name of the command that could not be processed.

User response: Use the DISPLAY RESULTS and DISPLAY HOST commands to determine why the hosts could not process the command and to determine the status of the hosts.

IHVC402I *aa* operand(s) do not apply to any switch.

Explanation: Asterisk (*) was specified as the switch device number, indicating that the command should be sent to all switches. The operands given in the command are not found on any switch. For example, if the command was **BLOCK DA ***, there was no switch that had the name *DA* defined.

The variable *aa* shows the name of the command.

User response: Use the DISPLAY RESULTS command for more information. Determine the correct operands and reenter the command.

IHVC403I *aa* operand(s) do not apply to switch *bb*.

Explanation: A switch device number was specified with the command, but that switch was not affected by the command. For example, the command was **BLOCK Printer_1 0123**, but the switch device number *0123* does not have a port with the name *Printer_1*.

The variable *aa* shows the name of the command.

The variable *bb* shows the switch device number.

User response: Use the DISPLAY RESULTS command for more information. Determine why the switch was not affected by the operands. Reenter the command with the correct operands.

IHVC404 Current switch data unavailable

Explanation: I/O operations did not have an up-to-date copy of the switch data and was unable to refresh its own database. The I/O operations data for the switch is not correct, and commands cannot be sent to the switch.

User response: Enter the RESET SWITCH command, which will attempt to synchronize the database. If the RESET SWITCH command cannot synchronize the database, contact your system administrator.

IHVC405 Switch data has been changed

Explanation: I/O operations did not have an up-to-date copy of the switch data for this switch but was able to normally refresh its own database.

User response: Use the Display commands to determine what might have changed and then reenter the command.

IHVC406I Some switches unable to process *aa*.

Explanation: The command was sent to the switches, but one or more switches could not successfully process the command.

The variable *aa* shows the name of the command.

User response: Use the DISPLAY RESULTS and the DISPLAY SWITCH commands to determine which switches could not process the command and what conditions were detected that prevented them from processing the command.

IHVC407I Host(s) and switch(es) unable to process *aa*.

Explanation: Some hosts and some switches were unable to successfully process the command.

The variable *aa* shows the name of the command.

User response: Use the DISPLAY RESULTS command to determine which hosts and which switches could not process the command.

IHVC409 **Port *aa* does not exist**

Explanation: A port was specified with a command, but the port is outside the valid range of ports defined by the hardware.

The variable *aa* shows the name of the port that does not exist.

User response: Use the DISPLAY RESULTS command for more information. Determine the correct port to be used and reenter the command.

IHVC410I **Commands cannot affect port *aa*
(switch control unit)**

Explanation: A port was specified with a command, but the port is the control unit port of the switch. Commands cannot affect the switch control unit port.

The variable *aa* shows the control unit port.

User response: Determine the correct port to be used and reenter the command.

IHVC411 **Unable to return this switch to
original state**

Explanation: A command was sent to multiple switches and processed, but there was a reason to stop the command and return this switch to its original state (the state it was in before the command was sent). This message was issued because I/O operations was unable to return this switch.

User response: Enter the command that returns the switch to its previous state. For example, if you entered **BLOCK (C0)**, the command to return the switch is **UNBLOCK (C0)**.

IHVC413I **I/O-OPS internal error**

Explanation: I/O operations was unable to update its database to include the new port name that was defined.

User response: Enter a DISPLAY NAME command for the port name and determine whether I/O operations processed the command correctly. If not, reenter the command. If the same error recurs, notify your system programmer.

IHVC414 **Attempted to CONNECT port *aa* to
itself**

Explanation: An attempt was made to connect a port to itself. This is not a valid operation.

The variable *aa* shows the name of the port.

User response: Reenter the CONNECT command with 2 unique ports.

IHVC415 **Port *aa* already CONNECTed**

Explanation: An attempt to connect port *aa* failed because port *aa* is already connected to another port.

The variable *aa* shows the name of the port.

User response: Determine whether the existing connection should be disconnected. If so, enter the DISCONNECT command. Then reenter the CONNECT command.

IHVC416 **Port *aa* not installed, cannot
CONNECT**

Explanation: An attempt to connect port address *aa* failed because port address *aa* is not installed.

The variable *aa* shows the name of the port address.

User response: Determine whether the correct port address was specified with the CONNECT command. If so, contact your system administrator to determine why the port is not installed (physically not in the switch).

IHVC417 **No available paths to the switch for
the command**

Explanation: There are no available paths for I/O operations to send the command to the switch. The command would remove the last existing path. This could happen if there is only one path defined to the switch and the command attempts to block or connect this path. Furthermore, if more than one path is defined but all the other paths have been blocked or connected, there is only one path left, which cannot be removed.

User response: Verify that this command is necessary. Then reenter the command with the FORCE option from another I/O operations that has multiple paths to the switch.

IHVC418I **Requested change not allowed at the
switch.**

Explanation: The command requested a change that would have been rejected by the switch. For example, the BLOCK command attempted to block the control unit port of the switch.

User response: Enter the DISPLAY RESULTS command to determine what was incorrect.

IHVC419 **No port names exist which can be
affected by this command**

Explanation: Either no port names are defined or the only port name is for the control unit port.

User response: Refer to the command section to determine the correct use of the command.

IHVC420 Attempted to ALLOW port *aa* to itself

Explanation: The Allow command was specified with 2 identical ports.

The variable *aa* shows the name of the port.

IHVC421I Switch(es) data has been changed.

Explanation: Changes have been made to one or more switches since you entered the last command or status of a host connected to the switch changed.

User response: Use the Display commands to determine what might have changed and then reenter the command.

IHVC422I A user obtained the I/O-OPS lock and processed commands.

Explanation: Another user took control of I/O operations command processing, processed I/O operations commands, and then released control of command processing with an UNLOCK command.

User response: Use the Display commands to determine what might have changed. Then reenter the command.

IHVC423I Channel subsystem configuration data has changed.

Explanation: During the processing of an I/O operations command, a change was detected in the channel subsystem configuration data. This could be the result of configuration changes made in PR/SM mode or through MVS Dynamic I/O. I/O operations refreshes its view of the configuration when such changes are made. During this refresh, user commands cannot be processed.

User response: Use the display commands (for example, DISPLAY CHP) to determine the change, or consult the system operator's console log for detailed messages. If the I/O operations command failed and it is determined that the configuration data changes do not affect the command, reenter the command.

IHVC424I Switch and channel subsystem configuration data has changed.

Explanation: During the processing of a I/O operations command, a change was detected in the channel subsystem configuration data and the data associated with a switch.

User response: Use the Display commands (such as DISPLAY SWITCH or DISPLAY CHP) to determine what might have changed or consult the system operator's console log for detailed messages. If the I/O operations command failed and it is determined that the switch or configuration changes do not affect the

command, reenter the command.

IHVC425I Refresh of I/O-OPS configuration data failed.

Explanation: A failure was detected during the update of the I/O operations database. This may have been caused by a problem obtaining channel subsystem configuration information. If Vary Paths were performed, these requests may have been based on old database information.

User response: Consult the system operator's console log to determine the cause of the system error that resulted in the database refresh failure.

IHVC426I Current switch and channel configuration data unavailable.

Explanation: During the processing of a I/O operations command, a change was detected in the channel subsystem configuration data, and the current switch data was not obtained.

User response: Use the Display commands (such as DISPLAY SWITCH or DISPLAY CHP) to determine what might have changed or consult the system operator's console log for detailed messages. If the I/O operations command failed and it is determined that the switch or configuration changes do not affect the command, reenter the command.

IHVC427I *aa* does not exist on switch *bb*.

Explanation: The specified port name does not exist on the specified switch.

The variable *aa* shows the name of the port that does not exist.

The variable *bb* shows the switch device number.

User response: Use the DISPLAY NAME command to see whether the port name exists on any other switch.

IHVC428I Current switch data unavailable.

Explanation: I/O operations did not have an up-to-date copy of the switch data and was unable to refresh its own database. The I/O operations data for the switch is not correct and commands cannot be sent to the switch.

User response: Enter the RESET SWITCH command, which will attempt to synchronize the database. If the Reset Switch command cannot synchronize the database, contact your system administrator.

IHVC429I Disable function for switch *aa* not in effect.

Explanation: This switch is not enabled to support the disable function of the REMOVE SWITCH command.

The variable *aa* shows the switch device number.

IHVC430I **Switch(es) have been REMOVED by another user.**

Explanation: One or more switches have been removed from your system. This was the result of a user on another system entering the REMOVE SWITCH command.

User response: Use the DISPLAY RESULTS command to determine which switches have been removed.

IHVC431 **Switch has been REMOVED**

Explanation: This switch has been removed from the system.

User response: None.

IHVC432I **Switch(es) have been RESTORED by another user.**

Explanation: One or more switches have been restored to your system. This was the result of a user on another system entering the RESTORE Switch command.

User response: Use the DISPLAY RESULTS command to determine which switches have been restored.

IHVC433 **Switch has been RESTORED**

Explanation: This switch has been restored to the system.

User response: None.

IHVC434 **Name *aa* not valid**

Explanation: The name specified contains characters that are not supported by the current switch configuration.

The variable *aa* shows the name of the port.

User response: Change the port name.

IHVC435 **Attempted to ALLOW FC port *aa* to FC port *bb***

Explanation: An attempt to allow a dynamic connection between port *aa* and another fibre channel port *bb* failed because both are fibre channel bridge ports.

The variable *aa* shows the name of the first fibre channel bridge port.

The variable *bb* shows the name of the second fibre channel bridge port.

User response: Prohibit the dynamic connection between the fibre channel bridge ports.

IHVC436 **Cannot connect *aa*, because it is a fibre channel port**

Explanation: An attempt to connect a port to the fibre channel port *aa* failed because a fibre channel port cannot be part of a static connection.

The variable *aa* shows the name of the fibre channel bridge port.

User response: Remove the static connection from the fibre channel port.

IHVC437 **Settings on switch prevent configuration changes**

Explanation: Someone has selected a management setting at the switch's local console that prevents control of the switch via the interfaces that I/O operations uses. I/O operations can continue to provide configuration information about this switch. However, I/O operations cannot change any configuration setting, for example, port connectivity hierarchy attributes or port names.

User response: Check whether the switch setup was done correctly. Some switch models support multiple types of remote management but require a customer setup choice to designate exactly one management agent to be able to change the switch configuration. This can prevent uncoordinated changes. I/O operations can provide its system integrated switching protection only when switch settings allow it to control the switches.

IHVC500 **Swch devn LSN Unique
id
Status Code**

Explanation: This is a header for the DISPLAY RESULTS command that shows the switch device number, logical switch number, and the switch unique identifier.

User response: None.

IHVC501 *aa bb cc dd ee*

Explanation: This is a header for the DISPLAY RESULTS command:

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

The variable *cc* shows the switch unique identifier.

The variable *dd* shows the switch status.

The variable *ee* shows the status code.

User response: None.

IHVC502 Switch not defined as a device

Explanation: The switch in the previous command has not been defined to IOCP as a device. I/O operations cannot communicate with this switch.

System programmer response: Determine whether this switch should be defined to IOCP and do so if needed.

IHVC505I No failures detected by this host

Explanation: The host did not detect any problems with sending the command.

User response: None.

IHVC506 No failures detected by this switch

Explanation: No problems were detected by the specified switch.

User response: None.

IHVC510I I/O-OPS logical/physical component not working

Explanation: I/O operations's logical-to-physical coordination component is not working on the specified host. This could be because the Vary concentrator is not working properly.

User response: Report the problem to your IBM Support Center.

IHVC511I VTAM not running, command not broadcast

Explanation: VTAM is not running. The command could not be broadcast to I/O operations on other systems. Message IHVI003, which was sent previously, contains the VTAM OPEN ACB error return code.

User response: Check with your system operator to see whether VTAM has been installed and is active.

IHVC512I Backout failed

Explanation: While trying to return the system to its initial state, some or all of the VARY commands that were done could not be restored.

User response: Enter the DISPLAY VARY command and use the list of failed Backout Varys to resend the requests.

IHVC513I Another user has the I/O-OPS lock

Explanation: This host has detected that an I/O operations user on another system holds the I/O operations lock.

User response: From the host issuing this message, enter I/O operations commands to determine which

user on which system has the lock. Contact that user to determine when the user will enter the Unlock command to release the lock.

IHVC514I Another user is processing an I/O-OPS command

Explanation: This host has detected that an I/O operations user on another system is processing I/O operations commands.

User response: From the host issuing this message, enter I/O operations commands to determine which user on which system is processing I/O operations commands. Contact that user to determine when processing is done and, if required the UNLOCK command has been entered to release the lock.

IHVC515I VTAM or TCP/IP error while broadcasting the command

Explanation: There was a VTAM or TCP/IP error while I/O operations was trying to communicate with another host.

User response: Check with your system operator to see whether VTAM and TCP/IP are active and functioning properly.

IHVC516I VTAM or TCP/IP session NOT established with this host

Explanation: A VTAM or TCP/IP session could not be established with the host. Broadcasts to that system will not be successful.

User response: Determine why a VTAM or TCP/IP session could not be established and correct the problem.

IHVC517I No response to the command

Explanation: This host did not respond to the command it received from the primary host.

User response: Determine why there was no response and correct the problem.

IHVC518I Vote of 'NO' - critical resource protected

Explanation: A host failed Vary Path requests.

User response: Enter the DISPLAY VARY command. Then use the list of failed VARY requests to determine which paths could not be varied. Free the devices and reenter the command. If necessary, use the FORCE option.

IHVC519I Backout processing in progress

Explanation: This host is still in the process of backing out the Varys from a previous command.

User response: None.

IHVC520 Command not sent to this switch

Explanation: The command was not sent to the switch. Either the command specified a switch other than this one, or the command specified ports or port names that were not defined for this switch.

User response: None.

IHVC522 Could not communicate with the switch

Explanation: The switch did not respond (or responded with a failure status) to the command that was entered. Two of the probable causes are that a prerequisite EC (engineering change) has not been installed on the switch, or the operating system is not at the minimum level required to support the command.

System programmer response: Determine why the command could not be sent to the switch and correct the problem.

IHVC523 Could not write host names to the switch

Explanation: I/O operations was unable to write its VTAM and TCP/IP host names to the switch. This means that other I/O operations programs will not be able to find out about this host, and, therefore, will not be able to start a VTAM or TCP/IP session with this host.

System programmer response: Determine why the command could not be sent to the switch and correct the problem.

IHVC524 Switch data has been changed during command execution

Explanation: After I/O operations verified that it had an up-to-date database, the switch data changed. When I/O operations tried to process the command, it no longer had the correct data.

User response: Determine what data was changed on the switch.

IHVC525I This host has been RESET off

Explanation: This message is issued following some Display commands if this host has been reset off.

User response: Either reset the host on or be aware that Display command data might not be complete,

because the host is removed from participating in consensus processing.

IHVC526I Refresh of I/O-OPS configuration data failed

Explanation: A failure was detected during the update of the I/O operations database. This may have been caused by a problem obtaining channel subsystem configuration information. If Vary Paths were performed, these requests may have been based on old database information.

User response: Consult the system operator's console log to determine the cause of the system error that resulted in the database refresh failure.

IHVC527I Channel subsystem configuration data has changed

Explanation: During the processing of an I/O operations command, a change was detected in the channel subsystem configuration data. This could be the result of configuration changes made in PR/SM mode or through MVS Dynamic I/O. I/O operations refreshes its view of the configuration when such changes are made. During this refresh, user commands cannot be processed.

User response: Use the Display commands (such as DISPLAY CHP) to determine the change or consult the system operator's console log for detailed messages. If the I/O operations command failed and it is determined that the configuration data changes do not affect the command, reenter the command.

IHVC528I aa CHPID bb is online

Explanation: An attempt was made to affect a port communicating with an ESCON channel converter. When the ESCON channel in converter mode is online, a BLOCK, DISCONNECT, UNCHAIN, or REMOVE command causes a channel error.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID number.

User response: Configure the ESCON channel in converter mode offline and reenter the command.

IHVC530I Failed VARYs Offline: Tried Failed 1st 3 failures (Dev,Chp)

Explanation: This is a header for the DISPLAY RESULTS command that shows the number of Varys tried, the number of Varys that failed, and the first three device number/CHPID pairs that failed.

User response: None.

IHVC531I Failed backout VARYs: Tried Failed 1st 3 Failures (Dev,Chp)

Explanation: This is a header for the DISPLAY RESULTS command that shows the number of Varys tried, the number of Varys that failed, and the first three device number/CHPID pairs that failed.

User response: None.

IHVC532I

aa bb (cc,dd)(ee,ff)(gg,hh)

Explanation: This is Display command data that shows the number of Varys attempted by the host, the number of Varys that failed, and the first three device number/CHPID pairs that failed.

The variable *aa* shows the number of attempted Varys.

The variable *bb* shows the number of failed Varys.

The variable *cc* shows the device number of the first CHPID.

The variable *dd* shows the first CHPID.

The variable *ee* shows the device number for the second CHPID.

The variable *ff* shows the second CHPID.

The variable *gg* shows the device number of the second CHPID.

The variable *hh* shows the third CHPID.

User response: None.

IHVC533 Port *aa* cannot be CONNECTed while in maintenance mode

Explanation: Port *aa* has been placed in maintenance mode by a service representative. A port cannot be connected while in maintenance mode.

The variable *aa* shows the name of the port.

User response: Wait for the service representative to complete maintenance.

IHVC534I Failed VARYs Online : Tried Failed 1st 3 Failures (Dev,Chp)

Explanation: This is a header for the DISPLAY RESULTS command that shows the number of Varys tried, the number of Varys that failed, and the first three device number/CHPID pairs that failed.

User response: None.

IHVC535I *aa* session established with this host

Explanation: A VTAM or TCP/IP session has been established with this host, and broadcasts to this system should be successful.

User response: None.

IHVC536I NO communication initialized on this host

Explanation: This message is issued for the primary host when I/O operations is unable to establish any communication with either VTAM or TCP/IP, or both, on this host.

User response: Verify that either VTAM or TCP/IP, or both, are operational and that all I/O operations installation requirements are met.

IHVC537I *aa* communication is initialized on this host

Explanation: This message is issued for the primary host when I/O operations is operational and communicating with either VTAM or TCP/IP, or both.

The variable *aa* shows the type of the communication, either VTAM or TCP/IP.

User response: None.

IHVC538I *aa bb* command not supported by this host

Explanation: The specified command is not supported by the target host indicated in the command. The release level of I/O operations on that host must be upgraded to support the requested command.

The variable *aa* shows the name of the I/O operations command.

The variable *bb* shows the command operand.

User response: Use non-I/O operations commands to perform the required task.

IHVC539I Cannot *aa bb cc* that is type *dd*

Explanation: The specified command cannot be processed for the CHPID because I/O operations does not support that command for the CHPID type indicated.

The variable *aa* shows the name of the I/O operations command.

The variable *bb* shows the CHPID.

The variable *cc* shows the CHPID number.

The variable *dd* shows the CHPID type.

User response: Use non-I/O operations commands to perform the required task.

IHVC540I Host name *aa* is reset off.

Explanation: The specified VTAM application name or TCP/IP host name is reset off, so the command entered for that host cannot be processed.

The variable *aa* shows the VTAM application name or the TCP/IP host name of the host that is reset off.

User response: Determine why the host is reset off. If possible, reset the host on by using the RESET HOST (ON) command and reissue your original command.

IHVC602I WRITESWCH command request bits incorrect.

Explanation: The requested combination of bits specified for the WRITESWCH command are not valid. For example, only one of the bits for the Modify, And, or Or mask can be specified for a port on a switch.

User response: Check the WRITESWCH command block and determine the correct setting of the request bits.

IHVC603I Host path status differs from switch settings

Explanation: As a result of the SYNC SWITCH command with the NOPROCESS option, I/O operations determined that the host's path status differed from the connectivity settings at the switch.

User response: Enter the DISPLAY VARY SYNC command, and if you want the listed Varys performed, reenter the command with the PROCESS option.

IHVC604I SYNC VARY(s) failed

Explanation: As a result of the SYNC SWITCH command with the PROCESS option, some of the required Varys failed.

User response: Enter the DISPLAY VARY command. Then use the list of failed Vary requests to determine which paths could not be varied and take appropriate action.

IHVC605I I/O-OPS unable to perform Config

Explanation: As a result of a connectivity command, CHPIDs require either Config Offline or Online, but the named I/O operations does not have the capability to perform the Config. This could be caused by an incorrect level of either I/O operations or the System Control Program.

User response: Manually enter the necessary commands to configure the CHPIDs to their desired status. Then reenter the connectivity command if appropriate.

IHVC606I Config(s) failed

Explanation: I/O operations attempted to perform Configs that were unsuccessful.

User response: Use operator commands to DISPLAY CHPID status to determine the reason for the failures and take appropriate action.

IHVC607I Failed CONFIGs OFFLINE: Tried Failed Failures (Chp)

Explanation: This is a header for the DISPLAY RESULTS command that shows the number of Configs tried, the number of Configs that failed, and the first eight Configs that failed.

User response: None.

IHVC608I Failed CONFIGs ONLINE: Tried Failed Failures (Chp)

Explanation: This is a header for the DISPLAY RESULTS command that shows the number of Configs tried, the number of Configs that failed, and the first eight Configs that failed.

User response: None.

IHVC609I Failed CONFIGs Backout: Tried Failed Failures (Chp)

Explanation: This is a header for the DISPLAY RESULTS command that shows the number of Configs tried, the number of Configs that failed, and the first eight Configs that failed.

User response: None.

IHVC610I

aa bb (cc,dd, ee,ff,gg,hh,ii,jj)

Explanation: This message is the output of a DISPLAY RESULTS command that shows the number of Configs tried, the number of Configs that failed, and the first eight Configs that failed.

The variable *aa* shows the number of attempted Configs.

The variable *bb* shows the number of failed Configs.

The variables *cc, dd, ee, ff, gg, hh, ii, and jj* show the CHPIDs of the first eight Configs that failed.

User response: None.

IHVC611I

(aa,bb,cc,dd,ee,ff,gg,hh)

Explanation: This message is the output of a DISPLAY RESULTS command.

The variables *cc, dd, ee, ff, gg, and hh* show the CHPIDs.

User response: None.

IHVC612I SYNC SWITCH command not fully supported

Explanation: This host is not equipped with the correct level of either I/O operations or the System Control Program to perform the SYNC SWITCH command. If the PROCESS option is specified with the

IHVC613I • IHVC621I

command, all required Varys will be attempted. If the NOPROCESS option is specified with the command, the list of required Varys is not created.

User response: None.

IHVC613I **Timeout occurred configuring *aa* CHPID(s) *bb***

Explanation: An attempt was made to configure ESCON converter CHPIDs but the current time-out value was exceeded.

The variable *aa* shows the number of CHPIDs attempted.

The variable *bb* shows whether the CHPIDs are online or offline.

User response: Enter the Display Timeout command to determine the current time-out value. Then enter the Reset Timeout command to increase the time-out value, and reenter the command.

IHVC614I ***aa* CHPID *bb* status can not be verified**

Explanation: An attempt was made to prevent a port communicating through an ESCON Converter Model 1, and there is no device associated with the converter. I/O operations cannot determine whether the CHPID is online or offline.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID.

User response: Determine the status of the CHPID. If it is online, configure the CHPID offline. Then reenter the command with the Force option.

IHVC615I **Host(s) issued WARNING(s) for the *aa* command.**

Explanation: I/O operations hosts detected one or more of the following conditions:

- Failed Vary Path Online requests
- Failed ESCON Converter Model 1 Config Online requests
- ESCON Converter Model 2 may need enablement
- VTAM or TCP/IP communication problems
- Other conditions that could cause a warning

The variable *aa* shows the name of the command that caused the warning.

User response: Enter the DISPLAY RESULTS and DISPLAY HOST commands to determine why the hosts responded with a warning.

IHVC616I **VARY ONLINE(s) failed**

Explanation: A host failed Vary Path Online requests.

User response: Enter the DISPLAY VARY command. Then use the list of failed Vary Online requests to determine which paths could not be varied online.

IHVC617I **Some switches not affected by the command.**

Explanation: All switches were specified for the switch device number, but the I/O operations command did not affect all switches known to this host.

User response: Enter the DISPLAY RESULTS command to determine which switches were affected by the command.

IHVC618I **More than one asterisk specified in port name.**

Explanation: Only one asterisk (*) can be specified in the port name. This asterisk signifies a generic port name.

User response: Reenter the command, specifying only one asterisk in the port name.

IHVC619I **More than 512 ports affected by the command.**

Explanation: The generic port name that was specified affected more than the allowable maximum of 512 ports.

User response: Reenter the command with a smaller subset of the generic port name.

IHVC620I **ESCON Converter Model 2 processing not supported**

Explanation: This message is the output of a DISPLAY RESULTS command and indicates that the host does not support ESCON Converter Model 2 processing.

User response: If it is determined that the command should proceed, reenter the command with the Nocheck option.

IHVC621I **ESCON Converter Model 2(s) detected**

Explanation: This message is the output of a DISPLAY RESULTS command and is followed by message IHVC623I. As a result of a connectivity command, ESCON Converter Model 2s were detected on paths affected by that command.

User response: If it is determined that the command should proceed, reenter the command with the Nocheck option.

IHVC622 **ESCON Converter Model 2 paths
may require *aa*:**

Explanation: This message is the output of a DISPLAY RESULTS command and is followed by message IHVC623I.

The variable *aa* specifies disablement or enablement.

User response: Take the appropriate action to disable or enable ESCON Converter Model 2 paths.

IHVC623 **Id Type Mod Serial number Port
Logical name / Unique id**

Explanation: This is a header for the messages IHVC626I and IHVC627I.

User response: None.

IHVC624 **Chained path associated with: *aa bb***

Explanation: This message is the output of a DISPLAY RESULTS command and is followed by message IHVC625I. An ESCON Converter Model 2 was detected in a chained path.

The variable *aa* shows the port.

The variable *bb* shows the name of the port.

User response: None.

IHVC625 **Data follows for chained switch: *aa***

Explanation: This message gives information for the switch involved in the chained path referred to in message IHVC624I.

The variable *aa* shows the switch unique identifier.

User response: None.

IHVC626 **CU *aa bb cc dd ee***

Explanation: This is the data associated with message IHVC623I and defines the ESCON Converter Model 2 control unit.

The variable *aa* shows the control unit type.

The variable *bb* shows the control unit model.

The variable *cc* shows the control unit serial number.

The variable *dd* shows the port.

The variable *ee* shows the name of the port.

User response: None.

IHVC627 **CV *aa bb cc dd ee***

Explanation: This is the data associated with message IHVC623I and defines the ESCON converter.

The variable *aa* shows the ESCON converter type.

The variable *bb* shows the ESCON converter model.

The variable *cc* shows the ESCON converter serial number.

The variable *dd* shows the port.

The variable *ee* shows the name of the port.

User response: None.

IHVC628I **No WRITESWCH command request bits set.**

Explanation: The requested combination of bits specified for the WRITESWCH command was all 0's.

User response: Check the WRITESWCH command block and determine the correct setting of the request bits.

IHVC630I ***aa on bb* has the I/O-OPS lock.**

Explanation: Another user has locked I/O operations command processing. If that user is performing connectivity changes involving any of the components you are attempting to obtain information about, that information might not be correct.

The variable *aa* shows the user ID.

The variable *bb* shows the I/O operations VTAM name or TCP/IP host name.

User response: To obtain current data, reenter the Display or Query command when the specified user releases the lock. If you need to obtain the command lock, enter the GETLOCK command.

IHVC631I ***aa on bb* has the I/O-OPS lock, data overflowed.**

Explanation: Another user has locked I/O operations command processing. If that user is performing connectivity changes involving any of the components you are attempting to obtain information about, that information might not be correct. Also, an overflow condition occurred that was caused by the Display or Query command generating more information than can be returned in a single display.

The variable *aa* shows the user ID.

The variable *bb* shows the I/O operations VTAM name or TCP/IP host name.

User response: Reenter the Display or Query command with the appropriate range to return the desired information. To obtain current data, reenter the Display or Query command when the specified user releases the lock. If you need to obtain the command lock, enter the Getlock command.

IHVC640I **Switch *aa* does not support the requested function.**

Explanation: You attempted a configuration file function on a switch that does not support the function.

The variable *aa* shows the switch device number.

User response: Check to ensure that you entered the correct switch number. If so, the level of microcode on that switch may need to be upgraded to support the attempted function.

IHVC641I Switch does not support the requested function

Explanation: You attempted a configuration file function on a switch that does not support the function.

User response: Check to ensure that you entered the correct switch number. If so, the level of microcode on that switch may need to be upgraded to support the attempted function.

IHVC642I File name *aa* not valid.

Explanation: You attempted a WRITEFILE command with a file name that is not supported at the switch.

The variable *aa* shows the file name.

User response: Reenter the command with a valid file name.

IHVC643I File descriptor *aa* not valid.

Explanation: You attempted a WRITEFILE command with a file descriptor that is not supported at the switch.

The variable *aa* shows the file descriptor.

User response: Reenter the command with a valid file descriptor.

IHVC644I Switch file in use, no access.

Explanation: You attempted to perform a configuration file function, but the file was in use. This could also occur if you are trying to change the IPL file, and the current designation is ACTIVE=SAVED.

User response: Verify that the file name entered is correct. If so, reenter the command. If not, correct the file name and reenter.

IHVC645 Switch file in use, no access

Explanation: You attempted to perform a configuration file function, but the file was in use. This could also occur if you are trying to change the IPL file, and the current designation is ACTIVE=SAVED.

User response: Verify that the file name entered is correct. If so, reenter the command. If not, correct the file name and reenter.

IHVC646 Maximum number of files saved at the switch

Explanation: You attempted a WRITEFILE command that will create more than the maximum number of allowable files at the switch.

User response: Verify that you want to perform the requested function. If so, delete an existing file, then reenter the command.

IHVC647 File name *aa* does not exist on switch

Explanation: You attempted to perform a file function and the file name does not exist on the switch.

The variable *aa* shows the file name.

User response: View the current list of files at the switch, then reenter the command with the desired file name.

IHVC648I File name *aa* does not exist on switch *bb*.

Explanation: You attempted to perform a file function and the file name does not exist on the switch.

The variable *aa* shows the file name.

The variable *bb* shows the switch device number.

User response: Verify that the switch device number entered is correct, and that the file name entered is on the current list of files at that switch. Reenter the command with the correct file name and switch device number.

IHVC649 Required resources not available to access file

Explanation: You attempted to perform a configuration file function and the required resources at the switch were not available. For example, the buffer space used by the switch is currently being used by other host programs.

User response: Retry the command. If possible, determine why the resources were not available, and reenter the command once they become available.

IHVC650I Error in WRITEFILE datablock.

Explanation: You entered a WRITEFILE command with a data block that contained errors.

User response: Issue the DISPLAY RESULTS command to determine the errors, then correct the errors and reenter the command.

IHVC651 Connectivity attributes for port *aa* conflict with port *bb*

Explanation: Conflicting port connectivity attributes were detected on the indicated WRITEFILE command. Either the mask or connect status for these ports do not

match. For example, if you specified that port C0 should be connected to port C1, but you also specified that port C1 should be connected to port C2 the connectivity attributes for ports C0 and C1 would be in conflict.

The variable *aa* shows the first port.

The variable *bb* shows the second port.

User response: Check the Writefile data block for the two ports, correct the error, and reenter the command.

IHVC653 Incorrect number of implemented ports for switch

Explanation: You entered a WRITEFILE command with a data block that did not contain the same number of implemented ports as the specified switch.

User response: Check the data block and reenter the command.

IHVC654I Maximum number of files saved at switch *aa*.

Explanation: You attempted to perform a configuration file function that will create more than the maximum number of allowable files at the switch.

The variable *aa* shows the switch device number.

User response: Verify that you want to perform the requested function. If so, issue a delete request to access additional space at the switch.

IHVC655I Required resources not available to access file.

Explanation: You attempted to perform a configuration file function and the required resources at the switch were not available. Possible causes are:

- The I/O operations console is not available
- File Access Facility is not available
- The specified file is locked, for example, by another user
- The file key buffer is full
- No data buffer is available on File access Facility
- The file was not saved because the maximum number of saved files is reached
- Saved file directory information is required and initialization was not complete
- The buffer space that is used by the switch is currently being used by other host programs

User response: If possible, determine why the resources were not available, and reenter the command once they become available.

IHVC656I Cannot delete IPL file at the switch.

Explanation: You attempted to delete the IPL file at the switch. This is not allowed.

User response: None.

IHVC657 Cannot delete IPL file at the switch

Explanation: You attempted to delete the IPL file at the switch. This is not allowed.

User response: None.

IHVC658I Switch initialization not complete

Explanation: A configuration file function was attempted while the switch was in the process of initializing.

User response: Wait until the switch completes its initialization and reenter the command.

IHVC659 Switch initialization not complete

Explanation: A configuration file function was attempted while the switch was in the process of initializing.

User response: Wait until the switch completes its initialization and reenter the command.

IHVC660 File name *aa* not valid

Explanation: You attempted a WRITEFILE command with a file name that is not supported at the switch.

The variable *aa* shows the file name.

User response: Reenter the command with a valid file name.

IHVC661 File descriptor *aa* not valid

Explanation: You attempted a WRITEFILE command with a file descriptor that is not supported at the switch.

The variable *aa* shows the file descriptor.

User response: Reenter the command with a valid file descriptor.

IHVC720I MVS internal error

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC725I Path not operational

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC730I Last path to device

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC735I Last path (reserved)

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC737I Last path (allocated)

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC740I Reserved path

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC745I Kept off by other source

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC750I Path validation failed

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC755I Device not connected

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC760I Permanent error

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC763I Coupling facility not accessible

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC764I Coupling facility path does not exist

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC765I Last path to coupling facility

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC766I Last path to active coupling facility

Explanation: This is the reason why the referenced VARY command failed.

User response: None.

IHVC800

Explanation: Separator line.

User response: None.

IHVC802I VTAM or TCP/IP name *aa* not known to this host.

Explanation: The specified VTAM application name or TCP/IP host name is not known to the host issuing the command.

The variable *aa* shows the VTAM application name or the TCP/IP host name that is unknown.

User response: Check the name for entry errors, then retry.

IHVC803I This Host: *aa/bb* Sysplex/System: *cc/dd*

Explanation: This message is the output of a DISPLAY HOST command.

The variable *aa* shows the VTAM name.

The variable *bb* shows the TCP/IP host name.

The variable *cc* shows the name of the sysplex.

The variable *dd* shows the name of the system.

User response: None.

IHVC804I	Name identifier	Level Sysplex	Physical System
----------	--------------------	------------------	--------------------

Explanation: This is a header for display commands.

User response: None.

IHVC805I **The following switches are attached:**

Explanation: This is a header for display commands.

User response: None.

IHVC806I	Swch
----------	------

Explanation: This message is the output of display commands.

User response: None.

IHVC807I	Devn	LSN	Unique id
----------	------	-----	-----------

Explanation: This is a header for display commands.

User response: None.

IHVC808I	VTAMname/TCP host System	Sysplex
----------	-----------------------------	---------

Explanation: This is a header for the DISPLAY RESULTS commands.

User response: None.

IHVC809I	OFFLINE
VARYs:	aa bb (cc,dd)(ee,ff)(gg,hh)

Explanation: This message is the output of a DISPLAY HOST command.

The variable *aa* shows the number of attempted Varys.

The variable *bb* shows the number of failed Varys.

The variables *cc*, *ee*, and *gg* show the device number.

The variables *dd*, *ff*, and *hh* show the CHPID.

User response: None.

IHVC810I	CHPID <i>aa</i> not known to this host
----------	--

Explanation: The specified CHPID is not known to the host processing the command.

The variable *aa* shows the CHPID.

User response: Enter the **DISPLAY CHP *** command to identify all known CHPIDs.

IHVC811	Vary type	Tried	Failed	1st <i>aa</i> failures
	(Dev,Chp,Msg)			

Explanation: This is a header for the DISPLAY VARY command.

The variable *aa* shows the number of Varys displayed.

User response: None.

IHVC812	Chp	Swch	Status
---------	-----	------	--------

Explanation: This is a header for the DISPLAY CHP command.

User response: None.

IHVC813	Chp Type	Devn	LSN Port	H S
	C P	Port name		

Explanation: This is a header for the DISPLAY CHP command.

User response: None.

IHVC814	<i>aa bb cc dd ee ff gg</i>
---------	-----------------------------

Explanation: This message is the output of a DISPLAY CHP command.

The variable *aa* shows the CHPID.

The variable *bb* shows CHPID type.

The variable *cc* shows switch device number.

The variable *dd* shows the logical switch number.

The variable *ee* shows the port.

The variable *ff* shows the port status.

The variable *gg* shows the port name.

User response: None.

IHVC815	Port
---------	------

Explanation: This message is the output of a DISPLAY CHP command.

User response: None.

IHVC816	Chp	Swch	Status
---------	-----	------	--------

Explanation: This is a header for the DISPLAY CHP command.

User response: None.

IHVC817	Chp Type	Devn	LSN Port	H S
	C P	Port name		

Explanation: This is a header for the DISPLAY CHP command.

User response: None.

IHVC818	<i>aa bb cc dd ee ff gg</i>
---------	-----------------------------

Explanation: This message is the output of a DISPLAY CHP command.

The variable *aa* shows the CHPID.

The variable *bb* shows the CHPID type.
 The variable *cc* shows the switch device number.
 The variable *dd* shows the logical switch number.
 The variable *ee* shows the port.
 The variable *ff* shows the port status.
 The variable *gg* shows the port name.

User response: None.

IHVC819 OFFLINE *aa bb (cc,dd,"ee")*

Explanation: This message is the output of a DISPLAY VARY command.

The variable *aa* shows the number of attempted Varys.
 The variable *bb* shows the number of failed Varys.
 The variable *cc* shows the device number.
 The variable *dd* shows the CHPID.
 The variable *ee* shows the reason why this vary failed.

User response: None.

IHVC820I Device *aa* not known to this host.

Explanation: The specified device is not known to the host sending the command.

The variable *aa* shows the device number.

User response: Verify that the correct device number was entered. If so, enter DISPLAY DEV * *sw* for each switch to find the correct device number. Then reenter the command.

IHVC821I Switch device *aa* not known to this host.

Explanation: The specified switch was not known to the host sending the command.

The variable *aa* shows the switch device number.

User response: Enter DISPLAY SWITCH * to display all switch device numbers known to the host. Then reenter the command with the correct switch device number.

IHVC822 (*aa,bb,"cc"*)

Explanation: This message is the output of a DISPLAY VARY command.

The variable *aa* shows the device number.
 The variable *bb* shows the CHPID.
 The variable *cc* shows the reason why this vary failed.

User response: None.

IHVC823I Device *aa* not found on switch *bb*.

Explanation: The specified device was not found on the specified switch.

The variable *aa* shows the device number.

The variable *bb* shows the switch device number.

User response: Enter DISPLAY DEV *de* * to find out which switch the device is on. Then reenter the command with the correct switch device number.

IHVC824 **Port**

Explanation: This message is the output of a DISPLAY DEV command.

User response: None.

IHVC825 **Chp Swch Status**

Explanation: This is a header for the DISPLAY DEV command.

User response: None.

IHVC826 **Devn Chp Type Devn LSN Port H S C P Port name**

Explanation: This is a header for the DISPLAY DEV command.

User response: None.

IHVC827 *aa bb cc dd ee ff gg hh*

Explanation: This message is the output of a DISPLAY DEV command.

The variable *aa* shows the device number.
 The variable *bb* shows the CHPID.
 The variable *cc* shows the CHPID type.
 The variable *dd* shows the switch device number.
 The variable *ee* shows the logical switch number.
 The variable *ff* shows the port.
 The variable *gg* shows the port status.
 The variable *hh* shows the port name.

User response: None.

IHVC828 **Connected to:** *aa bb cc*

Explanation: This message is the output of a DISPLAY DEV command.

The variable *aa* shows the port.
 The variable *bb* shows the port status.
 The variable *cc* shows the port name.

User response: None.

IHVC829 **Chained to:** *aa bb cc dd ee*

Explanation: This message is the output of a DISPLAY DEV command.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

The variable *cc* shows the port.

The variable *dd* shows the port status.

The variable *ee* shows the port name.

User response: None.

IHVC830 *aa connected to:* *bb cc dd*

Explanation: This message is the output of a DISPLAY DEV command.

The variable *aa* shows the midport of chain.

The variable *bb* shows the port.

The variable *cc* shows the port status.

The variable *dd* shows the port name.

User response: None.

IHVC831 *aa chained to:* *bb cc dd ee ff*

Explanation: This message is the output of a DISPLAY DEV command.

The variable *aa* shows the midport of chain.

The variable *bb* shows the switch device number.

The variable *cc* shows the logical switch number.

The variable *dd* shows the port.

The variable *ee* shows the port status.

The variable *ff* shows the port name.

User response: None.

IHVC834 **Swch LSN Unique id**
 Ins Status **SC**

Explanation: This is the header for the DISPLAY SWITCH command. SWITCH * DETAIL command.

Swch Switch device number

LSN Logical switch number

Unique id Node element descriptor

Ins Number of installed ports

Status Status information

SC Status code

User response: None.

IHVC835 **Port**

Explanation: This message is the output of a DISPLAY SWITCH command.

User response: None.

IHVC836 **Status**

Explanation: This is a header for the DISPLAY SWITCH command.

User response: None.

IHVC837 **Port H S C P I/O-Def Port**
 name

Explanation: This is a header for the DISPLAY SWITCH command.

User response: None.

IHVC838 *aa bb cc dd ee ff*

Explanation: This message is the output of a DISPLAY SWITCH command.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

The variable *cc* shows the switch unique identifier.

The variable *dd* shows the number of ports installed.

The variable *ee* shows the switch status information.

The variable *ff* shows the switch status code.

User response: Enter the DISPLAY RESULTS command to get more information about the switch.

IHVC839 *aa bb dd cc*

Explanation: This message is the output of a DISPLAY SWITCH command.

The variable *aa* shows the port.

The variable *bb* shows the port status.

The variable *cc* shows the port name.

The variable *dd* shows the port I/O definition.

User response: None.

IHVC840I *aa unknown to this host.*

Explanation: The specified port name is not known to this host.

The variable *aa* shows the port name.

User response: None.

IHVC841I **No BLOCKed port names on switch *aa*.**

Explanation: There are no blocked port names on the specified switch.

The variable *aa* shows the switch device number.

User response: None.

IHVC842I No CONNECTed port names on switch *aa*.

Explanation: There are no connected port names on the specified switch.

The variable *aa* shows the switch device number.

User response: None.

IHVC843 *aa* does not exist on switch *bb*

Explanation: The specified port name does not exist on the specified switch.

The variable *aa* shows the port name.

The variable *bb* shows the switch device number.

User response: Use the DISPLAY NAME command to see whether the port name exists on any other switch.

IHVC844 Port

Explanation: This is a header for the DISPLAY NAME or DISPLAY PORT command.

User response: None.

IHVC845 Swch Status I/O

Explanation: This is a header for the DISPLAY NAME or DISPLAY PORT command.

User response: None.

IHVC846 Port Name Devn
LSN Port H S C P Def

Explanation: This is a header for the DISPLAY NAME or DISPLAY PORT command.

User response: None.

IHVC847 *aa bb cc dd ee ff*

Explanation: This message is the output of a DISPLAY NAME or DISPLAY PORT command.

The variable *aa* shows the port name.

The variable *bb* shows the switch device number.

The variable *cc* shows the logical switch number.

The variable *dd* shows the port.

The variable *ee* shows the port status.

The variable *ff* shows the port I/O definition.

User response: None.

IHVC848 Connected to: *aa bb cc dd*

Explanation: This message is the output of a DISPLAY NAME or DISPLAY PORT command.

The variable *aa* shows the port name.

The variable *bb* shows the port.

The variable *cc* shows the port status.

The variable *dd* shows the port I/O definition.

User response: None.

IHVC849 Allow/Prohibit mask for this port:

Explanation: This is a header for the DISPLAY NAME or DISPLAY PORT command.

User response: None.

IHVC850I Port *aa* does not exist on switch *bb*.

Explanation: The specified port does not exist on the specified switch.

The variable *aa* shows the port.

The variable *bb* shows the switch device number.

User response: Enter DISPLAY PORT (*) *sw_dev_number* to display all ports on the switch.

IHVC851 No BLOCKed ports on switch *aa*.

Explanation: There are no blocked ports on the specified switch.

The variable *aa* shows the switch device number.

User response: None.

IHVC852 No CONNECTed ports on switch *aa*.

Explanation: There are no connected ports on the specified switch.

The variable *aa* shows the switch device number.

User response: None.

IHVC853 Config type Number 1st *aa*
CONFIGs (Chp)

Explanation: This is a header for the DISPLAY VARY command.

The variable *aa* shows the number of Configs displayed.

User response: None.

IHVC854 OFFLINE *aa*
(*bb,cc,dd,ee,ff,gg,hh,ii*)

Explanation: This message is the output of the DISPLAY VARY command.

The variable *aa* shows the number of Config Offlines.

The variables *bb, cc, dd, ee, ff, gg, hh, and ii* show the CHPIDs.

User response: None.

IHVC855 **ONLINE** *aa*
 (*bb,cc,dd,ee,ff,gg,hh,ii*)

Explanation: This message is the output of the DISPLAY VARY command.

The variable *aa* shows the number of Config Onlines.

The variables *bb, cc, dd, ee, ff, gg, hh, and ii* show the CHPIDs.

User response: None.

IHVC856I **CVC CHPIDs may require**
ENABLEMENT: *aa,bb,cc,dd,ee,ff,gg,hh*

Explanation: This message is the first line of the output of a DISPLAY RESULTS command that shows the CHPID numbers of CVC CHPIDs that might need to be configured online.

The variables *aa, bb, cc, dd, ee, ff, gg, and hh* show the CHPID numbers.

User response: Determine whether the CHPID must be configured online to make it operational, and take the appropriate action.

IHVC857I *aa,bb,cc,dd,ee,ff,gg, hh*

Explanation: This message is the second line of the output of a DISPLAY RESULTS command that shows the CHPID numbers of CVC CHPIDs that might need to be configured online.

The variables *aa, bb, cc, dd, ee, ff, gg, and hh* show the CHPID numbers.

User response: Determine whether the CHPID must be configured online to make it operational, and take the appropriate action.

IHVC860I **Current timeout value is *aa* seconds.**

Explanation: This message is the result of a DISPLAY TIMEOUT command.

The variable *aa* shows the timeout value in seconds.

User response: None.

IHVC870I **Number of VARYs specified must be**
between 1 and 1500.

Explanation: I/O operations can only display the first 1500 Varys. This message is the output of a DISPLAY command.

User response: None.

IHVC871 **No VARYs failed on previous command**

Explanation: This message is the output of a DISPLAY command.

User response: None.

IHVC872 **SYNC type** **Number** **1st *aa***
VARYs (Dev,Chp)

Explanation: This is a header for the DISPLAY VARY command.

The variable *aa* shows the number of Sync Varys displayed.

User response: None.

IHVC873 **OFFLINE** *aa*
 (*bb,cc*)(*dd,ee*)(*ff,gg*)(*hh,ii*)

Explanation: This message is the output of a DISPLAY VARY command.

The variable *aa* shows the number of Sync Offline Varys to be done.

The variables *bb, dd, ff, and hh* show the device number.

The variables *cc, ee, gg, and ii* show the CHPID.

User response: None.

IHVC874 **ONLINE** *aa* (*bb,cc*)(
 dd,ee)(*ff,gg*)(*hh,ii*)

Explanation: This message is the output of a DISPLAY VARY command.

The variable *aa* shows the number of Sync Online Varys to be done.

The variables *bb, dd, ff, and hh* show the device number.

The variables *cc, ee, gg, and ii* show the CHPID.

User response: None.

IHVC875 **No SYNC VARYs detected on previous**
command

Explanation: This message is the output of a DISPLAY command.

User response: None.

IHVC876I **No VARYs or CONFIGs detected on**
previous command.

Explanation: This message is the output of a DISPLAY command.

User response: None.

IHVC877 **No SYNC CONFIGs detected on**
previous command

Explanation: This message is the output of a DISPLAY command.

User response: None.

IHVC878I No SYNC VARYs or CONFIGs detected on previous command

Explanation: This message is the output of a DISPLAY command.

User response: None.

IHVC879 No failed CONFIGs on previous command

Explanation: This message is the output of a DISPLAY command.

User response: None.

IHVC880I ONLINE VARYs: aa bb
(cc,dd)(ee,ff)(gg,hh)

Explanation: This message is the output of a DISPLAY command. For an example of how the command output appears when it is formatted, refer to the appropriate command section.

- The variable *aa* shows the number of Varys.
- The variable *bb* shows the number of failed Varys.
- The variables *cc*, *ee* and *gg* show the device number.
- The variables *dd*, *ff* and *hh* show the CHPID.

User response: None.

IHVC881I Display command caused data overflow.

Explanation: The command completed, but an overflow condition occurred that resulted in a non-zero return code. This overflow condition was caused by the DISPLAY command generating more information than can be returned in a single display.

User response: Reenter several DISPLAY commands with smaller ranges.

IHVC882I Starting number must be less than or equal to ending number.

Explanation: The DISPLAY command failed because the user entered a value for the starting number that was greater than the ending number. The starting number must contain a value that is less than or equal to the value of the ending number.

User response: Reenter the DISPLAY command with the appropriate range to return the desired information.

IHVC883I No devices found in the specified range on switch *aa*.

Explanation: The DISPLAY command failed because no devices were found in the specified range on the specified switch.

The variable *aa* shows the switch device number.

User response: Reissue the DISPLAY command with

the appropriate range to return the desired information.

IHVC884I Sysplex system(s) not associated with a host name detected:

Explanation: This message is the result of a DISPLAY HOST * command. Following this message is a list of systems that are not associated with a VTAM name or TCP/IP host name.

User response: None.

IHVC885I Sysplex <- Systems
----->

Explanation: This is a header for the DISPLAY HOST * command. Following this message is a list of systems that are not associated with a VTAM name or TCP/IP host name.

User response: None.

IHVC886I aa bb, cc, dd, ee, ff,

Explanation: This message is the output of DISPLAY HOST * command. It displays a list of systems that are not associated with a VTAM name or a TCP/IP host name.

- The variable *aa* shows the name of the sysplex.
- The variables *bb*, *cc*, *dd*, *ee*, and *ff* show system names.

User response: None.

IHVC887I bb, cc, dd, ee, ff,

Explanation: This message is the output of DISPLAY HOST * command, and is a continuation of message IHVC886I. It displays a list of systems that are not associated with a VTAM name or a TCP/IP host name.

The variables *aa*, *bb*, *cc*, *dd*, *ee*, and *ff* show system names.

User response: None.

IHVC890I Device is not known to I/O-OPS.

Explanation: The VARY command failed because the device was not found in the I/O operations database.

User response: Verify the device number and reissue the command with the correct number.

IHVC891I Device is a coupling facility.

Explanation: The Vary command failed because the device attempted to be varied is a coupling facility.

User response: Verify the device number and reissue the command with the correct number.

IHVC892I No device for *aa*.

Explanation: The VARY command failed because no devices were found for the input token.

The variable *aa* shows the token.

User response: Verify the token number and reissue the command with the correct number.

IHVC893I No device for *aa*.

Explanation: The VARY command failed because no devices were found for the input token.

The variable *aa* shows the token.

User response: Verify the token number and reissue the command with the correct number.

IHVC902I Switch *aa* not opened.

Explanation: The specified switch has not been opened prior to receiving a command.

The variable *aa* shows the switch device number.

User response: Enter the **Reset Switch** command to try to open the switch. The switch may not be varied online. Vary it online and reissue the command.

IHVC903 Port *aa* already CHAINED

Explanation: On a CHAIN command, one of the specified ports was already part of a chained connection.

The variable *aa* shows the port.

User response: Enter **Display Switch** *sw_device_number* to get the status of each port on the switch.

IHVC905I Not allowed to purge resident host.

Explanation: The RESET HOST command with the PURGE option was specified for the resident host. This is not allowed.

User response: Reevaluate which host is to be purged.

IHVC906 Cannot *aa* CHAINED port *bb*

Explanation: The DISCONNECT command cannot be used to remove the dedicated connection in a chained path. The CONNECT command cannot be used to establish a dedicated connection to a port defined to be part of a chain.

The variable *aa* shows the name of the command.

The variable *bb* shows the port name.

User response: Enter the UNCHAIN command to remove the dedicated connection. Enter the CHAIN command to establish the dedicated connection.

IHVC908 Ports *aa* and *bb* are not CONNECTed to each other

Explanation: The user tried to disconnect two ports that are not connected to each other. However, one of the ports is involved in a dedicated connection with a different port.

The variable *aa* shows the first port.

The variable *bb* shows the second port.

User response: Reenter the command with the correct ports.

IHVC909 Execution of the CHAIN command will form a closed loop

Explanation: The user entered a CHAIN command that will cause the chained path to be a closed loop.

User response: Reenter the command with the correct operands.

IHVC910 Port *aa* is not part of a CHAINED path

Explanation: The port that was specified in the UNCHAIN command is not part of a chained path.

The variable *aa* shows the port.

User response: Reenter the UNCHAIN command with the correct command parameters.

IHVC911 Ports *aa* and *bb* are not part of the same CHAINED path

Explanation: The specified ports in the UNCHAIN command are not part of the same chained path.

The variable *aa* shows the first port.

The variable *bb* shows the second port.

User response: Reenter the UNCHAIN command with the correct command parameters.

IHVC912 Ports not in the correct CHAINED path order

Explanation: The ports specified in the UNCHAIN command were defined in a different order when specified in the CHAIN command.

User response: Reenter the UNCHAIN command with the correct command parameters.

IHVC913I Command affects Inter-Switch-Link port *aa* on *bb*.

Explanation: The port specified is implemented as a FICON port for an Inter-Switch-Link (E_Port). A BLOCK command is not allowed for this type of port. This is also true for UNBLOCK, ALLOW, PROHIBIT, or WRITEPORT unless the user specifies the option IGNORE along with the command. The reason for

requesting the additional option is that I/O operations can no longer guarantee "safe-switching" when an E_Port is involved in the command being performed.

"Safe-switching" sets the paths and devices online or offline when the path from a chipid to a device either becomes valid or is no longer valid because of a port manipulation. The option makes the operator aware of this fact.

The variable *aa* shows the port address of the first or only port.

The variable *bb* shows the switch device number of the port.

User response: For UNBLOCK, ALLOW, PROHIBIT, or WRITEPORT reenter the command using the option IGNORE.

IHVC914 Command affects Inter-Switch-Link port *aa*

Explanation: This message occurs as the third line of information for a particular switch device in the output of a DISPLAY RESULTS command where a previous command has been rejected. The port specified is implemented as a FICON port for an Inter-Switch-Link (E_Port). A BLOCK command is not allowed for this type of port. This is also true for UNBLOCK, ALLOW, PROHIBIT, or WRITEPORT unless the user specifies the option IGNORE along with the command.

The reason for requesting the additional option is that I/O operations can no longer guarantee "safe-switching" when an E_Port is involved in the command being performed.

The variable *aa* shows the port address of the first or only port.

User response: For UNBLOCK, ALLOW, PROHIBIT, or WRITEPORT reenter the command using the option IGNORE.

IHVC998I I/O-OPS command response

Explanation: This is the control line header for a multiple line display.

User response: None.

IHVC999I I/O-OPS display

Explanation: This is the control line header for a multiple line display.

User response: None.

IHVD000I I/O-OPS *Vaa.Rbb.Mcc* START in progress.

Explanation: The system operator has entered a START command. I/O operations sends this message to show that it has been loaded by the system but is not ready to accept commands.

The variable *aa* shows the version level of I/O operations.

The variable *bb* shows the release level of I/O operations.

The variable *cc* shows the modification level of I/O operations.

User response: None.

IHVD001I I/O-OPS START is complete.

Explanation: This message follows IHVD000I when I/O operations has successfully started and is ready to accept commands.

User response: None.

IHVD002I I/O-OPS STOP in progress.

Explanation: The system operator has entered a command to stop I/O operations or a severe error has caused I/O operations to stop.

User response: None.

IHVD003I I/O-OPS STOP is complete.

Explanation: This message follows IHVD002I when I/O operations has successfully stopped on this host.

User response: None.

IHVD004I *aa* failed for *bb*, RC=*cc*.

Explanation: A system service or instruction failed. This service or instruction is required for I/O operations to perform its functions. This may also indicate a previous error.

The variable *aa* shows the ATTACH, LOAD, or STARTUP.

The variable *bb* shows the module or function

The variable *rc* shows the return code.

User response: Before continuing you should look for any previous I/O operations error messages. For an explanation of the return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. Use the return code to determine the cause of the error. Correct the condition and restart I/O operations.

IHVD005I *aa* failed for *bb cc*, RSN=*dd*.

Explanation: A system service or instruction failed. This service or instruction is required for I/O operations to perform its functions.

The variable *aa* shows the system service or instruction, for example, UCBSKAN, that failed.

The variable *bb* shows the module, function, or system service target identifier.

The variable *cc* shows the module, function, or system service target name.

The variable *dd* shows the reason code.

User response: For an explanation of the reason code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. Report the problem to your IBM Support Center. Stop and restart I/O operations.

IHVD006I Data Handling system error, RSN=*aa*.

Explanation: A unrecoverable system error was detected during I/O operations database processing.

The variable *aa* shows the reason code.

User response: For an explanation of the reason code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. Report the problem to your IBM Support Center. Stop and restart I/O operations.

IHVD010I Switch *aa* associated with device number *bb*.

Explanation: I/O operations procedures to associate the specified switch with a device number were successfully completed.

The variable *aa* shows the logical switch number.

The variable *bb* shows the switch device number.

User response: None.

IHVD011I Configuration has no switch.

Explanation: I/O operations detected that there are no logical switch numbers associated with the CHPIDs in the I/O configuration definition. I/O operations is unable to recognize any switches in the configuration.

System programmer response: Correct any errors found in the I/O configuration definition. Then restart I/O operations.

IHVD012I Switch *aa* cannot be associated with a device.

Explanation: I/O operations procedures to associate the specified switch as a device were unsuccessful. Some of the conditions that may cause this are:

- I/O configuration definitions for the control unit or I/O device for the switch were missing or incorrectly defined.
- The switch is not operational or does not exist.
- All defined paths from the CHPID to the switch control unit were not operational or were blocked.

The variable *aa* shows the logical switch number.

User response: Determine the cause of the problem and take appropriate corrective action.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD013E Channel subsystem configuration data not valid.

Explanation: The information returned by the channel subsystem was not valid because there are no valid CHPIDs, devices, or paths defined.

System programmer response: Report the problem to your IBM Support Center.

IHVD014E Unable to obtain channel subsystem configuration data.

Explanation: I/O operations was unable to obtain channel subsystem configuration data due to hardware or software incompatibility.

System programmer response: Report the problem to your IBM Support Center.

IHVD102I System page error accessing switch *bb* (LSN *aa*).

Explanation: When attempting to access the specified switch, the operating system detected a system paging problem.

The variable *aa* shows the logical switch number.

The variable *bb* shows the switch device number.

User response: Report the problem to your IBM Support Center.

IHVD103I System I/O error accessing switch *bb* (LSN *aa*).

Explanation: When attempting to access the specified switch, the operating system detected a system I/O error.

The variable *aa* shows the logical switch number.

The variable *bb* shows the switch device number.

User response: Report the problem to your IBM Support Center.

IHVD104I Switch *bb* (LSN *aa*) issued a unit check.

Explanation: The specified switch sent a unit check. This condition is caused by an I/O operations programming problem.

The variable *aa* shows the logical switch number.

The variable *bb* shows the switch device number.

User response: Report the problem to your IBM Support Center.

IHVD105I Cannot ALLOCATE/ATTACH switch *bb* (LSN *aa*).

Explanation: When attempting to allocate (MVS) or attach (VM) the specified switch, the operating system detected an error.

The variable *aa* shows the logical switch number.

The variable *bb* shows the switch device number.

User response: This message is informational; I/O operations will proceed. Check whether the DDNAME associated with the switch device number is allocated to another job or user. If so, report the problem to your IBM Support Center.

IHVD106I Name *aa* invalid for port *bb* on switch *cc*.

Explanation: The name that has been assigned to the specified port on the specified switch is not a valid port name.

The variable *aa* shows the port name.

The variable *bb* shows the port number.

The variable *cc* shows the switch device number.

User response: Change the port name. If port FE is specified for ESCON Director Models 1 and 2, this is the control unit port (CUP).

IHVD107I Cannot set user alert mode at switch *aa* (LSN *bb*).

Explanation: During open processing I/O operations determined that it has a connection to this switch. Therefore it issued an ENABLE USER ALERT MODE command. However, the switch did not execute the command successfully.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

User response: None. However, you should keep in mind that users working directly at the director's switch do not get a warning that a host program is in control of this switch.

IHVD108I Settings on switch *aa* (LSN *bb*) prevent configuration changes.

Explanation: Someone has selected a management setting at the switch's local console that prevents control of the switch via the interfaces that I/O operations uses. I/O operations can continue to provide configuration information about this switch. However, I/O operations cannot change any configuration setting such as a port connectivity hierarchy attributes or port names.

The variable *aa* shows the switch device number.

The variable *bb* shows the logical switch number.

User response: Check whether the switch setup was done correctly. Some switch models support multiple

types of remote management but require a customer setup choice to designate exactly one management agent to be able to change the switch configuration. This can prevent uncoordinated changes. I/O operations can provide its system integrated switching protection only when switch settings allow it to control the switches.

IHVD201I System swapping problem: No I/O to switch(es) performed.

Explanation: When attempting to access the specified switches, the operating system detected an address space swapping request failure. This caused all I/O to switches to be canceled.

User response: Report the problem to your IBM Support Center.

IHVD202I Cannot use CHAINED switch *aa*.

Explanation: A switch contains a defined chained relationship to the specified switch that is either not known to the host or is not opened.

The variable *aa* shows the switch unique identifier.

User response: If the specified switch is not known to the host, no action is necessary. If the switch is known but is not opened, the cause of the unsuccessful open should be identified and corrected. A Reset Switch command should then be entered for the affected switch.

IHVD205I CHPID *aa* not valid but has device paths assigned.

Explanation: During I/O operations startup or after a configuration change, I/O operations obtained channel subsystem data indicating that the specified CHPID number is not valid. This CHPID has paths to one or more devices.

The variable *aa* shows the CHPID number.

User response: If the CHPID is operational and defined in the I/O configuration definition, report the problem to your IBM Support Center.

System programmer response: If the CHPID was incorrectly defined, correct any errors found in the I/O configuration definition.

IHVD306I Port *aa* on LSN *bb* incorrectly defined as supporting a CHAIN.

Explanation: During the refresh of the database, I/O operations determined that a chain was defined but the dedicated connection to support that chain did not exist. The user must reenter the CHAIN command to define the chained path and establish the dedicated connection, or enter the UNCHAIN command to remove the defined chain.

The variable *aa* shows the port.

The variable *bb* shows the logical switch number.

User response: Enter the CHAIN or UNCHAIN command.

IHVD307I Chain *aa bb cc dd ee* broken.

Explanation: I/O operations has determined that the specified chain is broken. The required dedicated connection might be missing or incorrect.

The variable *aa* shows the first port.

The variable *bb* shows the first switch device number.

The variable *cc* shows the second port.

The variable *dd* shows the third port.

The variable *ee* shows the second switch device number.

User response: Enter the appropriate CHAIN or UNCHAIN command.

IHVD308I Multiple switches assigned unique id *aa*.

Explanation: During verification of the switch, I/O operations determined that more than one switch is associated with the same unique identifier. This situation could occur when a dedicated connection is incorrectly specified on chained paths when I/O operations opens a switch or when errors exist in the I/O configuration definition. This condition should be corrected immediately. Otherwise display and connectivity commands could affect the wrong switch.

The variable *aa* shows the switch unique identifier.

User response: Enter **Display Switch *** to determine which switches are assigned to the specified unique identifier. If errors in the I/O configuration definition are determined to be the problem, record the condition and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD309I Chain redefined to *aa bb cc dd ee*.

Explanation: I/O operations has redefined a chain to correspond to the physical connections at the switch.

The variable *aa* shows the first port.

The variable *bb* shows the first switch device number.

The variable *cc* shows the second port.

The variable *dd* shows the third port.

The variable *ee* shows the second switch device number.

User response: None.

IHVD310I *aa* CHPID *bb* determined to be on port *cc* LSN *dd*.

Explanation: I/O operations procedures have identified the port associated with the CHPID.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID.

The variable *cc* shows the port for CHPID.

The variable *dd* shows the logical switch number.

User response: None.

IHVD320I *aa* CHPID *bb* reassigned to port *cc* on LSN *dd*.

Explanation: During verification of CHPID ports, I/O operations determined that a CHPID port was incorrectly specified and corrected it in the I/O operations database. This occurs if the I/O configuration definition for the CHPID was incorrect, or if a chained path was not specified at I/O operations startup.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID.

The variable *cc* shows the port for CHPID.

The variable *dd* shows the logical switch number.

User response: If a chained path was not specified, enter the CHAIN command for the appropriate ports.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD330I Device paths to *aa* CHPID *bb* reassigned to LSN *cc*.

Explanation: During verification of device paths to CHPIDs, I/O operations determined that a device path was incorrectly specified and corrected it in the I/O operations database. This occurs if the I/O configuration definition for the CHPID was incorrect or if a chained path was not declared at I/O operations startup. Because devices with paths to a CHPID must share the same logical switch association, these paths have been reassigned to the specified switch. This message is sent following message IHVD320I.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID number.

The variable *cc* shows the logical switch number.

User response: If a chained path was not specified, enter the CHAIN command for the appropriate ports.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD331I *aa bb* added to I/O-OPS database.

Explanation: As a result of the refresh of the channel subsystem configuration data, the specified channel path or switch has been added to the I/O operations database.

The variable *aa* shows the CHPID or LSN.

The variable *bb* shows the channel port ID or logical switch number.

User response: None.

IHVD332I *aa bb* deleted from I/O-OPS database.

Explanation: As a result of the refresh of the channel subsystem configuration data, the specified channel path or switch has been deleted from the I/O operations database.

The variable *aa* shows the CHPID or LSN.

The variable *bb* shows the channel port ID or logical switch number.

User response: None.

IHVD333I *aa bb* added to I/O-OPS database.

Explanation: I/O operations added a resource to its internal database. This appears only on the message log.

The variable *aa* shows the CNTL or DEVN.

The variable *bb* shows the control unit number or device number.

User response: None.

IHVD334I *aa bb* deleted from I/O-OPS database.

Explanation: I/O operations deleted a resource from its internal database. appears only on the message log.

The variable *aa* shows the CNTL or DEVN.

The variable *bb* shows the control unit number or device number.

User response: None.

IHVD335I Switch *aa* Port *bb* added to I/O-OPS database.

Explanation: I/O operations has added a port to its internal database. This appears only on the message log.

The variable *aa* shows the switch device number.

The variable *bb* shows the port number.

User response: None.

IHVD336I Switch *aa* Port *bb* deleted from I/O-OPS database.

Explanation: I/O operations has deleted a port from its internal database. This appears only on the message log.

The variable *aa* shows the switch device number.

The variable *bb* shows the port number.

User response: None.

IHVD337I Switch *aa* added to I/O-OPS database.

Explanation: I/O operations has added a switch to its internal database. This appears only on the message log.

The variable *aa* shows the switch device number.

User response: None.

IHVD338I Switch *aa* deleted from I/O-OPS database.

Explanation: I/O operations has deleted a switch from its internal database. This appears only on the message log.

The variable *aa* shows the switch device number.

User response: None.

IHVD339I DEVN *aa* deleted from and added to the I/O-OPS database

Explanation: I/O operations deleted a resource from its internal database and then added it to its internal database. This appears only on the message log.

The variable *aa* shows the device number.

User response: None.

IHVD340I CHPID *aa* and its devices are no longer associated with LSN *bb*.

Explanation: During verification of channel subsystem data, it was determined that the CHPID and its paths are no longer associated with a switch.

The variable *aa* shows the CHPID number.

The variable *bb* shows the logical switch number.

User response: None.

IHVD341I Device *aa* state changed to online.

Explanation: Message issued to SYSLOG only for RMC automation. Operator varied a device online and the state change must be monitored by RMC. The IEE450I or IEE457I message issued by MVS is displayed when an operator issues a DISPLAY command (d u,,,A80) and when an operator varies a range of devices online or offline. In the case of the VARY with a range, MVS issues the DISPLAY

command (d u,,,xxx) that shows up in the SYSLOG.

The variable *aa* shows the device number.

User response: None.

IHVD342I Device *aa* state changed to offline.

Explanation: Message issued to SYSLOG only for RMC automation. Operator varied a device offline and the state change must be monitored by RMC. The IEE450I or IEE457I message issued by MVS is displayed when an operator issues a DISPLAY command (d u,,,A80) and when an operator varies a range of devices online or offline. In the case of the VARY with a range, MVS issues the DISPLAY command (d u,,,xxx) that shows up in the SYSLOG.

The variable *aa* shows the device number.

User response: None.

IHVD344I Control unit *aa* status update.

Explanation: I/O operations has detected a status change for the indicated control unit. This appears only on the message log.

The variable *aa* shows the control unit number.

User response: None.

IHVD347I Device paths to *aa* CHPID *bb* not assigned to a port.

Explanation: During verification of device paths, I/O operations determined that the port associated with devices that have paths to the specified CHPID could not be assigned. This occurs if:

1. There are I/O configuration definition errors for a CTC_P-type CHPID, and an expected dedicated connection is not found
2. The CHPID is described as CTC_?, indicating that the control unit port for the paths associated with the CHPID can not be determined
3. Other I/O configuration definition errors have been detected for the specified CHPID

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID number.

User response: Record the condition and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition or make the appropriate connectivity or system changes to the specified CHPID.

IHVD348I Device paths to *aa* CHPID *bb* not assigned to port *cc* LSN *dd*.

Explanation: During verification of device paths, I/O operations determined that the port associated with devices with paths to the specified CHPID could not be

assigned. This occurs if there are I/O configuration definition errors for the control unit or CNC-type CHPID, and an expected dedicated connection is not found.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID number.

The variable *cc* shows the port.

The variable *dd* shows the logical switch number.

User response: Record the condition and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD349I Device paths to *aa* CHPID *bb* assigned to port *cc* LSN *dd*.

Explanation: I/O operations procedures have identified the port for the devices associated with the specified CHPID.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID number.

The variable *cc* shows the port.

The variable *dd* shows the logical switch number.

User response: None.

IHVD350I Port *aa* LSN *bb* has more than CHPID *cc* assigned.

Explanation: During verification of switch ports, I/O operations determined that there was more than one CHPID logically associated with the specified port. This condition is physically impossible and may occur if:

- The wrong switch is specified in the I/O configuration definition for the CHPID.
- The CHPID is a member of a channel-to-channel relationship, where the complete path is defined in the I/O configuration definition. In this case, the message is informational and might not represent an error condition.

The variable *aa* shows the port.

The variable *bb* shows the logical switch number.

The variable *cc* shows the CHPID number.

User response: Record the condition and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD360I Port *aa* LSN *bb* has CHPID and device paths assigned.

Explanation: During verification of switch ports, I/O operations determined that both CHPIDs and device paths were logically associated with the same port. This condition is physically impossible and may occur if:

- There are errors in the I/O configuration definition for the CHPID or control units.
- The CHPID and control unit is a member of a channel-to-channel relationship, where the complete path is defined in the I/O configuration definition. In this case, the message is informational and might not represent an error condition.

The variable *aa* shows the port.

The variable *bb* shows the logical switch number.

User response: Record the condition and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD368I Cannot assign devices to unimplemented port *aa* LSN *bb*.

Explanation: During verification of device paths, I/O operations discovered that the control unit was assigned to an unimplemented port. This condition is physically impossible and occurs if there are errors in the I/O configuration definition for the CHPID (incorrect SWITCH=*parameter*), or control unit (incorrect LINK=*parameter*). This will also occur if ports associated with a control unit are defined in the I/O configuration definition but were not implemented on the specified switch.

The variable *aa* shows the port.

The variable *bb* shows the logical switch number.

User response: Record the condition and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD370I Cannot determine port for *aa* CHPID *bb* on LSN *cc*.

Explanation: During verification of CHPID ports, I/O operations could not determine the port for the specified CHPID on the specified switch. Possible causes of this condition are:

- I/O configuration definition errors for the CHPID or control unit.
- The required dedicated connection between ports on the switch has not been established to support the switch, or the switch paths are offline.
- The CHPID or the port associated with the CHPID is not operational, or the switch port is blocked.
- The required chain between switches has not been established.
- In the case of a converter-type CHPID, none of the devices defined to that CHPID could be brought online.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID number.

The variable *dd* shows the logical switch number.

User response: Determine the cause of the problem and take the appropriate action. If the I/O configuration definition is in error, record the problem and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD378I *aa* CHPID *bb* not assigned to port *cc* LSN *dd*.

Explanation: During verification of CHPID ports, I/O operations determined that the port associated with the specified CHPID could not be assigned. This occurs if there are I/O configuration definition errors for a converter-type CHPID, and an expected dedicated connection is not found.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID number.

The variable *cc* shows the port.

The variable *dd* shows the logical switch number.

User response: Record the condition and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD380I Multiple LSNs incorrectly assigned to switch *aa*. LSN *bb* chosen.

Explanation: During verification of the device number associated with the switch, multiple logical switch numbers were indicated for this switch in IOCP data. Because there can be only one logical switch number for a switch, I/O operations selected the indicated logical switch number.

The variable *aa* shows the device number.

The variable *bb* shows the logical switch number.

User response: Record the condition and report it to your system programmer.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD382I Multiple ports incorrectly assigned to switch *aa*. Port *bb* used.

Explanation: During verification of the control unit port defined in IOCP data, inconsistent port assignments were detected. This could represent errors in the I/O configuration definition.

The variable *aa* shows the device number.

The variable *bb* shows the port number.

User response: Record the condition and report it to your system programmer.

System programmer response: Correct any errors

found in the I/O configuration definition.

IHVD385I Multiple IDs for CU *aa*, ID *bb* chosen.

Explanation: During verification of the control unit, I/O operations determined that more than one ID is associated with the control unit. There are two situations that create this message:

- Devices in the same control unit return node descriptor information according to “best practice”. In this case the message might indicate an I/O configuration definition error or a cabling problem for the control unit.
- Devices in the same control unit do not return node descriptor information according to “best practice”, which occurs when their LCUs have defined different SDCs. In this case the message does not indicate a problem.

Best Practice: The node ID - the rightmost part of a node descriptor (ND) - consists of two parts:

- A 26-byte EBCDIC value that represents the SDC ID that describes the type, model, manufacturer, and serial number.
- A 2-byte hexadecimal interface ID that makes the node ID unique.

The node ID must be worldwide unique. The “best practice” defines the SDC ID to be unique for all node IDs of a processor, switch, or control unit, which almost all manufacturers conform to. One known exception is EMC Disk Library for Mainframe model 4080. For example, the I/O operations API command "Q E CU V cu_number" returns different values in the fields PD and CU_PTOK for the system that issues this message but the same value for the system that does not report the message.

The specified ID is selected to represent the control unit.

The variable *aa* shows the control unit number.

The variable *bb* shows the unique identifier.

User response: If errors in the I/O configuration definition are determined to be the problem, record the condition and report it to your system programmer. If you determine a cabling problem, take appropriate action.

System programmer response: Correct any errors that are found in the I/O configuration definition.

IHVD387I *aa bb* misconfigured, NED=*cc*.

Explanation: When verifying the specified resource, I/O operations determined that more than one node element descriptor (NED) is associated with the resource. This problem was probably caused by an I/O configuration definition error, a cabling problem for the resource, or an incorrect customization and setup of the I/O resource. The indicated NED is selected to represent the control unit or device.

The variable *aa* shows CU or DEVICE.

The variable *bb* shows the control unit or device number.

The variable *cc* shows NED (NED selected by I/O operations to represent the resource.)

User response: If errors in the I/O configuration definition are determined to be the problem, record the condition and report it to your system programmer. If a cabling problem is determined, correct the error.

System programmer response: Correct any errors found in the I/O configuration definition.

IHVD390I Type for CHPID *aa* respecified to *bb*.

Explanation: I/O operations respecified this CHPID's type to reflect information obtained during verification. This can occur if:

- There was a change in switch connectivity, such as after any connectivity command was performed.
- There was a change in the operational state of the CHPID, which could occur following a CHPID reconfiguration or the detection of a port or cable failure.

The variable *aa* shows the CHPID number.

The variable *bb* shows the CHPID type.

User response: None.

IHVD395I *aa* CHPID *bb* path is misdefined.

Explanation: During verification of paths associated with the specified CHPID, I/O operations detected a mismatch between what is defined in the channel subsystem configuration data and what is described in the ESCON/FICON Director configuration. This condition could occur if a chained path was incorrectly defined, an error was made in the I/O configuration data for the CHPID, or a cable was incorrectly installed.

The variable *aa* shows the CHPID type.

The variable *bb* shows the CHPID number.

User response: Determine the cause of the path mismatch and take appropriate action.

IHVD398I CHPID status could not be verified.

Explanation: An error was encountered during I/O operations verification of CHPID status. This could occur during I/O operations starting or command processing. Possible reasons for the error could be a GETMAIN failure or a VM Diagnose FC error.

System programmer response: Report the problem to your IBM Support Center.

IHVD400I Host and CHAIN data for switch *aa* has been lost.

Explanation: The VTAM names or TCP/IP host names of some I/O operations base programs and any chain definitions associated with the switch *aa* have been overlaid. Intersystem communication may not be possible.

The variable *aa* shows the switch device number.

User response: Enter a Reset Switch command on all I/O operations that have this switch allocated to them to restore the VTAM names and TCP/IP host names. If you suspect a chaining definition is lost, reenter the CHAIN command.

IHVD410I Path test was successful.

Explanation: Self-explanatory.

User response: None.

IHVD411I Unable to perform path test, RC=*aa* Reason=*bb*.

Explanation: The path test service failed.

The variable *aa* shows the service return code.

The variable *bb* shows the reason code.

User response: Report the problem to your IBM Support Center, supplying this message, the return code, and the reason code, if provided.

System programmer response: Refer to the description of the IOSPTHV macro in *MVS Programming: Authorized Assembler Services Reference, Volume 2 (ENFREQ-IXGWRITE)*.

IHVD412I Unable to determine the ND for path *aa* of CU *bb*.

Explanation: I/O operations could not determine the node descriptor for this path of the control unit. This message is only issued for control units that are attached to cascaded switches. A missing node descriptor can lead to unpredictable results when manipulating the entry port or destination port of the path whose node descriptor is missing. For example, the paths of the affected devices are not varied ONLINE or OFFLINE.

The variable *aa* shows the CHPID number.

The variable *bb* shows the control unit number.

User response: Use the MVS command D M=CU(*uuuu*) to determine the reason why the path is not available. Make the path available, if needed.

IHVD420I *aa* and *bb* no longer share a common IODF.

Explanation: I/O operations has detected that the active IODF on the first host, which is the primary host and whose VTAM application name or TCP/IP host name replaces *aa*, is no longer synchronized with the second, or peer, host, whose VTAM application name or TCP/IP host name replaces *bb*. This condition can impair I/O operations' ability to process its multisystem commands, such as queries, REMOVE DEV, and RESTORE DEV, and to display unified system images.

User response: Perform a dynamic I/O activation on the primary host using the same IODF that has been activated on the peer host.

IHVD421I IODF activation caused loss of common IODF with hosts:

Explanation: This header message precedes one or more IHVD422I messages that list the hosts that I/O operations detected that the I/O definition file (IODF) was activated on, and which has caused IODF synchronization to be lost with the peer hosts. This condition can impair I/O operations' ability to process its multisystem commands, such as queries, REMOVE DEV, and RESTORE DEV, and to display unified system images.

User response: Perform a dynamic I/O activation on the host systems that are specified in the subsequent IHVD422I messages, using the currently active IODF.

IHVD422I *aa, bb, cc, dd, ee, ff*

Explanation: See message IHVD421I.

The variables *aa, bb, cc, dd, ee, and ff* show the VTAM application or TCP/IP host names of the hosts.

User response: See message IHVD421I.

IHVD830I Configuration change:

| **Explanation:** This header message precedes one or
| more IHVD333I, IHVD334I, IHVD339I, IHVD833I,
| IHVD834I, and IHVD839I messages.

| **User response:** None.

| **IHVD833I DEVN *aa-bb* added to the I/O-OPS database.**

| **Explanation:** I/O operations added a continuous
| range of resources to its internal database. This appears
| only on the message log.

- | • The variable *aa* shows the first device number.
 - | • The variable *bb* shows the last device number.
- | **User response:** None.

| **IHVD834I** **DEVN *aa-bb* deleted from the I/O-OPS database**

- | **Explanation:** I/O operations deleted a continuous range of resources from its internal database. This appears only on the message log.
- | The variable *aa* shows the first device number.
 - | The variable *bb* shows the last device number.
- | **User response:** None.

| **IHVD839I** **DEVN *aa-bb* deleted from and added to the I/O-OPS database.**

- | **Explanation:** I/O operations deleted a continuous range of resources from its internal database and then added the same range to its internal database. This appears only on the message log.
- | • The variable *aa* shows the first device number.
 - | • The variable *bb* shows the last device number.
- | **User response:** None.

IHVD950I **ESTAE failed.**

Explanation: Owing to an ESTAE failure, the I/O operations ESTAE environment could not be established during program startup.

System programmer response: Identify and fix the problem. Restart I/O operations.

IHVD998I **I/O-OPS is not operational.**

Explanation: I/O operations terminated normally or abnormally.

User response: Try to restart I/O operations.

IHVD999E **I/O-OPS Abend *aa*, CSECT=*bb*, PSW=*cc dd*.**

Explanation: A system service or instruction failed and I/O operations cannot recover. An I/O operations dump was generated and I/O operations stopped.

 The variable *aa* shows the system abend code.

 The variable *bb* shows the failing CSECT name.

 The variable *cc* shows the failing program status word (PSW) left half.

 The variable *dd* shows the failing program status word (PSW) right half.

System programmer response: Save the I/O operations dump and the console log. Report the problem to your IBM Support Center. Try to restart I/O operations.

IHVI001E **No session with *aa*. OPNDST failed, RC=*bb* FB=*cc* Loc=*dd*.**

Explanation: A VTAM session setup request between this host and the I/O operations host with the VTAM name *aa* has failed.

 The variable *aa* shows the I/O operations VTAM name.

 The variable *bb* shows the OPNDST return code.

 The variable *cc* shows the OPNDST feedback code.

 The variable *dd* shows M or X depending on where the failure was detected, either in the mainline code or in the exit code.

User response: Any subsequent I/O operations command attempts to establish this session. If the attempt is successful, message IHVI002I is sent.

Lack of a session is considered a “vote of NO” in consensus processing.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take the appropriate action.

IHVI002I **Session set up between *aa* and *bb*.**

Explanation: A VTAM OPNDST successfully set up a session between the I/O operations hosts with the VTAM names *aa* and *bb*.

 The variables *aa* and *bb* show the I/O operations VTAM names.

User response: None.

IHVI003E **No communication with VTAM. OPEN ACB failed, RC=*aa*.**

Explanation: The VTAM OPEN ACB request failed with return code *aa*. This host is not able to communicate with any other host and is running in the minimal operating environment unless TCP/IP communication could be established.

 The variable *aa* shows the VTAM OPEN ACB error return code.

User response: Any subsequent I/O operations command issues the OPEN ACB. If the OPEN ACB is successful, I/O operations attempts to set up sessions with other hosts. If the OPEN ACB fails, this host remains in the minimal operating environment in case a TCP/IP communication could also not be established and requires the FORCE option to be entered with connectivity commands.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take appropriate action.

IHVI004E Communication with *aa* has been lost.

Explanation: The VTAM session between this host and the I/O operations host with the VTAM name *aa* has been lost. There is no communication between these two hosts until the session is reestablished.

The variable *aa* shows the I/O operations VTAM name.

User response: Any subsequent I/O operations command attempts to establish this session. If the attempt is successful, message IHVI002I is sent. If the attempt fails, message IHVI001E is issued.

Lack of a session is considered a “vote of NO” in consensus processing. If the command must be processed, there are the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- **Reset Host *aa*OFF** can be entered to temporarily ignore the I/O operations host with the VTAM name *aa*.
- **Reset Host *aa*PURGE** can be entered to permanent ignore the I/O operations host with the VTAM name *aa*.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take the appropriate action.

IHVI005E Unable to receive data. RECEIVE ANY failed, RC=*aa* FB=*bb*.

Explanation: The VTAM RECEIVE ANY that was sent to get data from other I/O operations hosts has failed.

The variable *aa* shows the VTAM return code.

The variable *bb* shows the VTAM feedback code.

User response: Any subsequent I/O operations command sends RECEIVE ANY. If the attempt is successful, normal communication between hosts continues.

If the RECEIVE ANY continues to fail, no data can be retrieved from other I/O operations hosts. This is considered a “vote of NO” in consensus processing.

If a command must be processed, reenter the command with the FORCE option. This causes all “votes of NO” to be ignored.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take appropriate action.

IHVI006E Unable to send to *aa*. SEND failed, RC=*bb* FB=*cc* Loc=*dd*.

Explanation: During a SEND to the I/O operations host with the VTAM name *aa*, the VTAM SEND failed with return code *bb* and feedback code *cc*. This is considered a “vote of NO” in consensus processing.

The variable *aa* shows the I/O operations VTAM name.

The variable *bb* shows the VTAM return code.

The variable *cc* shows the VTAM feedback code.

The variable *dd* shows M or X depending on where the failure was detected, either in the mainline code or in the exit code.

User response: Any subsequent I/O operations command attempts to send data. If the SEND is successful, normal processing continues.

Failure of the SEND results in a lack of response from the named host. This is considered a “vote of NO” in consensus processing. If the command must be processed, there are the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- **Reset Host *aa*OFF** can be entered to temporary ignore the I/O operations host with the VTAM name *aa*.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take the appropriate action.

IHVI007E Unable to receive from *aa*. RECEIVE failed, RC=*bb* FB=*cc*.

Explanation: While attempting to get data from the I/O operations host with the VTAM name *aa*, the VTAM RECEIVE failed with return code *bb* and feedback code *cc*. This is considered a “vote of NO” in consensus processing.

The variable *aa* shows the I/O operations VTAM name.

The variable *bb* shows the VTAM return code.

The variable *cc* shows the VTAM feedback code.

User response: Any subsequent I/O operations command attempts to receive data. If the RECEIVE is successful, normal processing continues.

Failure of the RECEIVE results in a lack of response from the named host. This is considered a “vote of NO” in consensus processing. If the command must be processed, there are the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- **Reset Host *aa*OFF** can be entered to temporary ignore the I/O operations host with the VTAM name *aa*.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take appropriate action.

If return code zero indicates that the RECEIVE has been completed successfully by VTAM, I/O operations has detected an error within the data received, for example, invalid record length. In this case check your VTAM definitions for I/O operations host-to-host communications. Check your logmode definitions for a

RUSIZE of zero as recommended on the VBUILD TYPE APPL example. Refer to *IBM Tivoli System Automation for z/OS Planning and Installation* for further information about the VTAM definitions.

IHVI008E SETLOGON failed, RC=aa FB=bb.

Explanation: The VTAM SETLOGON request has failed with return code *aa* and feedback code *bb*. This host is not notified of any new I/O operations host because the SCIP exit cannot be driven.

The variable *aa* shows the VTAM return code.

The variable *bb* shows the VTAM feedback code.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take appropriate action.

IHVI009E Session not set up with aa. GENCB failed, RC=bb FB=cc.

Explanation: The VTAM GENCB failed with return code *bb* and feedback code *cc*. The session was not set up between this host and the I/O operations host with the VTAM name *aa*.

The variable *aa* shows the I/O operations VTAM name.

The variable *bb* shows the VTAM return code.

The variable *cc* shows the VTAM feedback code.

User response: Any subsequent I/O operations command attempts to establish this session. If the attempt is successful, message IHVI002I is sent.

Lack of a session is considered a “vote of NO” in consensus processing. If the command must be processed, there are the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- **Reset Host aaOFF** can be entered to temporarily ignore the I/O operations host with the VTAM name *aa*.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take appropriate action.

IHVI010E No session with aa. OPNSEC failed, RC=bb FB=cc.

Explanation: A request to establish a session from this host to the I/O operations host with the VTAM name *aa* failed with return code *bb* and feedback code *cc*.

The variable *aa* shows the I/O operations VTAM name.

The variable *bb* shows the OPNSEC return code.

The variable *cc* shows the OPNSEC feedback code.

User response: Any subsequent I/O operations command attempts to establish this session. If the

attempt is successful, message IHVI002I is sent.

Lack of a session is considered a “vote of NO” in consensus processing. If the command must be processed, there are the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- **Reset Host aaOFF** can be entered to temporarily ignore the I/O operations host with the VTAM name *aa*.

System programmer response: Refer to *VTAM Messages and Codes* for an explanation of the return codes and take appropriate action.

IHVI011E Unable to write host data at switch.

Explanation: An error occurred when trying to send **WRITE HOST DATA** to one or more switches. This data is the I/O operations VTAM name and TCP/IP name of this host. If the names do not appear in the switch host data area, other I/O operations hosts do not know of this host and no communication occurs with this host.

User response: Use the DISPLAY SWITCH command to determine the cause of the failure. Enter **Reset Switch** to have the I/O operations VTAM name written to the switch host data area.

IHVI012I Cannot disable switch operator alert.

Explanation: An error occurred when trying to send **DISABLE LOCAL OPERATOR ALERT** to one or more switches. This causes a warning message to be sent to any local switch operator when making changes to the switch.

User response: None.

IHVI013E Unable to erase host data at switch.

Explanation: An error occurred when trying to send **WRITE HOST DATA** to one or more switches. This host was attempting to erase its I/O operations VTAM name and TCP/IP host name from the host data area as I/O operations was stopping. By leaving the name in the host data area, any new I/O operations hosts attempt to create a VTAM or TCP/IP session with this host.

User response: The **Reset Host Purge** command can be sent from any other I/O operations host to either erase that name from the host data area or ignore that host in consensus processing.

IHVI101I No TCP session with aa. bb failed, RSN=cc.

Explanation: A TCP session setup request between this host and the I/O operations host with the TCP/IP host name *aa* has failed.

The variable *aa* shows the TCP/IP host name.

The variable *bb* shows the API function that failed.

The variable *cc* shows the error code.

User response: Any subsequent I/O operations command attempts to establish this session. If the attempt is successful, message IHVI102I is sent. Lack of a session is considered a “vote of NO” in consensus processing.

System programmer response: Refer to “Appendix B. Return codes” in *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for an explanation of the error code and take appropriate action.

IHVI102I Outbound connection set up between *aa* and *bb*.

Explanation: A TCP/IP outbound connection has been successfully established between the I/O operations hosts with TCP/IP host names *aa* and *bb*.

The variables *aa* and *bb* show TCP/IP host names.

User response: None.

IHVI103E No communication with TCP/IP. *aa* failed, RSN=*bb*.

Explanation: The socket of the server port could not be bound. This host is not able to communicate with any other host via TCP/IP. If VTAM communication is also unavailable, this host is running in the minimal operating environment.

The variable *aa* shows the API function that failed.

The variable *bb* shows the error code.

User response: Any subsequent I/O operations command issues the bind request. If the bind is successful, I/O operations will attempt to set up sessions with other hosts that do not have a VTAM session with this host. If the bind fails and VTAM communication is also unavailable, this host remains in the minimal operating environment and requires the FORCE option to be entered with connectivity commands.

System programmer response: Refer to “Appendix B. Return codes” in *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for an explanation of the error code and take appropriate action.

IHVI104E Communication with *aa* has been lost.

Explanation: The TCP/IP connection between this host and the I/O operations host with the TCP/IP host name *aa* has been lost. There is no communication between these two hosts until the connection is reestablished.

The variable *aa* shows the TCP/IP host name.

User response: Any subsequent I/O operations command attempts to establish this connection. If the

attempt is successful, message IHVI102I is sent. If the attempt fails, message IHVI101E is issued. Lack of a connection is considered a “vote of NO” in consensus processing. If the command must be processed, you have the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- Enter Reset Host *aa* OFF to temporarily ignore the I/O operations host with the TCP/IP host name *aa*.
- Enter Reset Host *aa* PURGE to permanently ignore the I/O operations host with the TCP/IP host name *aa*.

IHVI105E Unable to receive data. SELECT failed. RSN=*aa*.

Explanation: The TCP/IP SELECT that was issued to get data from other I/O operations hosts has failed.

The variable *rs* shows the error code.

User response: Any subsequent I/O operations command issues the SELECT. If the attempt is successful, normal communication between hosts continues. If the SELECT continues to fail, no data can be received from other I/O operations hosts. This is considered a “vote of NO” in consensus processing. If a command must be processed, reenter the command with the FORCE option. This causes all “votes of NO” to be ignored.

System programmer response: Refer to “Appendix B. Return codes” in *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for an explanation of the error code and take appropriate action.

IHVI106E Unable to send data to *aa*. RSN=*bb*.

Explanation: During a SEND to the I/O operations host with the TCP/IP host name *aa*, the TCP/IP SEND failed with error code *bb*. This is considered a “vote of NO” in consensus processing.

The variable *ho* shows the TCP/IP host name.

The variable *rs* shows the error code.

User response: Any subsequent I/O operations command attempts to send data. If the SEND is successful, normal processing continues. Failure of the SEND results in a lack of response from the named host. This is considered a “vote of NO” in consensus processing. If the command must be processed, you have the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- Enter Reset Host *aa* OFF to temporary ignore the I/O operations host with the TCP/IP host name *aa*.

System programmer response: Refer to “Appendix B. Return codes” in *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for

an explanation of the error code and take appropriate action.

IHVI107E Unable to receive data from *aa*. RSN=*bb*.

Explanation: While attempting to get data from the I/O operations host with the TCP/IP host name *aa*, the TCP/IP RECV failed with error code *bb*. This is considered a “vote of NO” in consensus processing.

The variable *ho* shows the TCP/IP host name.

The variable *rs* shows the error code.

User response: Any subsequent I/O operations command attempts to receive data. If the RECV is successful, normal processing continues. Failure of the RECV results in a lack of response from the named host. This is considered a “vote of NO” in consensus processing. If the command must be processed, you have the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- Enter Reset Host *ho* OFF to temporary ignore the I/O operations host with the TCP/IP host name *aa*.

System programmer response: Refer to “Appendix B. Return codes” in *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for an explanation of the error code and take appropriate action.

IHVI109E Connection not set up with *aa*. RSN=*bb*.

Explanation: TCP/IP failed to connect the I/O operations host with the TCP/IP host name *aa*. The connection was not set up between this host and the I/O operations host with the TCP/IP host name *aa*.

The variable *ho* shows the TCP/IP host name.

The variable *rs* shows the error code.

User response: Any subsequent I/O operations command attempts to establish this session. If the attempt is successful, message IHVI102I is sent. Lack of a session is considered a “vote of NO” in consensus processing. If the command must be processed, you have the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- Enter Reset Host *aa* OFF to temporary ignore the I/O operations host with the TCP/IP host name *aa*.

System programmer response: Refer to “Appendix B. Return codes” in *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for an explanation of the error code and take the appropriate action.

IHVI114I TCP/IP communication suppressed by user.

Explanation: The communication via TCP/IP has been suppressed by the user.

User response: I/O operations does not attempt to set up sessions with other hosts using TCP/IP. If VTAM is also unavailable, this host remains in the minimal operating environment and requires the FORCE option to be entered with connectivity commands.

System programmer response: If you want to communicate via TCP/IP the next time, make sure that the server port is predefined in the *hlq.ETC.SERVICES* data set. See the step “Define Host-to-Host Communications” in chapter “Installing SA z/OS on Host Systems” in *IBM Tivoli System Automation for z/OS Planning and Installation*.

IHVI115E TCP/IP host name *aa* too long.

Explanation: I/O operations could not resolve the TCP/IP node name into an alias name of up to 8 characters. I/O operations closes the connection to the remote host.

The variable *aa* shows the TCP/IP host name.

User response: None.

System programmer response: If you want to communicate via TCP/IP the next time, make sure that you provide an alias name of up to 8 characters for the named host.

IHVI116E TCP/IP communication has terminated.

Explanation: I/O operations receives the condition that TCP/IP has terminated.

User response: When TCP/IP becomes available again I/O operations tries to reestablish the connections that have been closed. These connections are currently returning a “vote of NO” in consensus processing. If the command must be processed, you have the following options:

- Reenter the command with the FORCE option. This causes all “votes of NO” from all hosts to be ignored.
- Enter Reset Host *aa* OFF to temporary ignore the I/O operations host with the TCP/IP host name *aa*.
- Enter Reset Host *aa* PURGE to permanent ignore the I/O operations host with the TCP/IP host name *aa*.

IHVI121I Connection rejected. Function *aa* failed, RSN=*bb*.

Explanation: A remote host whose name was unknown up to this point in time sent a connection request. The connection was rejected due to an error condition in the named function.

The variable *aa* shows the API function that failed.

The variable *bb* shows the error code.

User response: None.

System programmer response: Refer to “Appendix B. Return codes” in *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for an explanation of the error code and take appropriate action.

IHVI122I Inbound connection set up between *aa* and *bb*.

Explanation: A TCP/IP inbound connection has been established successfully between the I/O operations hosts with TCP/IP host names *aa* and *bb*.

The variables *aa* and *bb* show TCP/IP host names.

User response: None.

IHVI130I OMVS segment of *aa* not found, SAF/RACF RC=*bb/cc* RSN=*dd*

Explanation: I/O operations could not find an OMVS segment that has been defined for its user ID. Because some TCP/IP macros require this segment, the TCP/IP communication is ignored for the lifetime of I/O Operations.

The variable *aa* shows the user ID of I/O operations.

The variable *bb* shows the SAF return code of the RACROUTE EXTRACT request.

The variable *cc* shows the RACF return code of the RACROUTE EXTRACT request.

The variable *dd* shows the reason code of the RACROUTE EXTRACT request.

User response: If you want I/O operations to use TCP/IP, refer to the installation step “Customizing I/O Operations” in *IBM Tivoli System Automation for z/OS Planning and Installation* for information about how to define an OMVS segment. After you have defined the OMVS segment you need to restart I/O operations.

IHVL001I Message *aabb* not found in module *cc*.

Explanation: I/O operations requested a message that could not be found in the indicated message module.

The variable *aa* shows the component ID and the sub-component ID.

The variable *bb* shows the message number.

The variable *cc* shows the name of the message module.

System programmer response: Record the sequence of events leading up to the message, including the message number. Report the problem to your IBM Support Center.

IHVL005I Command caused data overflow.

Explanation: The command completed, but an overflow condition occurred that resulted in a non-zero return code. This overflow condition was caused by the command generating more information than can be returned in a single display.

User response: Reenter several commands with smaller ranges.

IHVO000I I/O-OPS is ready to process operator commands.

Explanation: Self-explanatory.

User response: Enter a I/O operations command.

IHVO001I No I/O-OPS command was specified.

Explanation: An MVS MODIFY or a VM CP SEND command was entered to call I/O operations but it did not contain an I/O operations command. This could have been entered from the system console or sent from an application program.

User response: Enter the MODIFY or SEND command again with a I/O operations command and any required operands. Or, if applicable, correct the code statement that calls the I/O operations API so that the first parameter is an I/O operations command.

IHVO002I *aa* is not a valid I/O-OPS command.

Explanation: The specified command is not a valid I/O operations command.

The variable *aa* shows the incorrect command.

User response: Correct the command and reenter it.

IHVO003I No operands were specified for the command.

Explanation: The command that was entered requires at least one operand.

User response: Enter the command with the required operands.

IHVO004I *aa* exceeds 24 characters.

Explanation: The specified operand is longer than the maximum length of 24 characters.

The variable *aa* shows the first 24 characters of the operand.

User response: Correct the operand and reenter the command.

IHVO005I *aa* is not a valid option.

Explanation: The option specified is not a valid I/O operations command option.

The variable *aa* shows the option that is not valid.

User response: Correct the option and reenter the command.

IHVO006I *aa* and *bb* are opposing options.

Explanation: You cannot specify the two options that replace *aa* and *bb* in the same command. For example, you cannot specify F and NOF in the same command.

The variable *aa* and *bb* show the conflicting options.

User response: Correct one or both options and reenter the command.

IHVO007I Too many operands were specified.

Explanation: The number of operands specified exceeds the maximum number that I/O operations can process.

User response: Correct the parameters and reenter the command.

IHVO008I Combination of '*' operands not valid.

Explanation: Specifying the all operand (*) for both the third and fourth operands on the DISPLAY DEV or the DISPLAY NAME command is not allowed. For example, neither DISPLAY DEV * * nor DISPLAY NAME * * is a valid I/O operations command.

User response: Correct the operands and reenter the command.

IHVO009I *aa* is not a valid port address.

Explanation: The port address specified is not a valid port address.

The variable *aa* shows the port address that is not valid.

User response: Correct the port address and reenter the command.

IHVO010I *aa* is not a valid operand.

Explanation: The operand specified as *aa* is not valid. For a REMOVE/RESTORE CHP command, an asterisk (*) is not valid as an I/O operations host name.

The variable *aa* shows the operand that is not valid.

User response: Correct the operand and reenter the command.

IHVO011I One or more operands are missing.

Explanation: Self-explanatory.

User response: Reenter the command with the correct number of operands.

IHVO012I A port name of '*' is not valid on the WRITE command.

Explanation: A single asterisk (*) cannot be assigned as the name of a port.

User response: Use a valid port name.

IHVO014I Specified I/O-OPS command is not valid on this interface.

Explanation: The specified I/O operations command cannot be entered from the interface you are using.

User response: Reenter the command from the appropriate interface.

IHVO016I *aa* option is limited to connectivity commands.

Explanation: The specified option can be used with connectivity commands only.

The variable *aa* shows the connectivity command option.

User response: Correct the option and reenter the command.

IHVO017I NOFORCE and NOVARY are conflicting options.

Explanation: The NOForce and NOVary options cannot be used together in the same I/O operations command.

User response: Correct the options and reenter the command.

IHVO018I FORCE, VARY and BACKOUT are conflicting options.

Explanation: The Force, Vary, and Backout options cannot be used together in the same I/O operations command.

User response: Correct the options and reenter the command.

IHVO019I ONLY cannot be used with a second operand of *aa*.

Explanation: The ONLY option cannot be used with a second operand of * or (*).

The variable *aa* is either * or (*).

IHVO020I • IHVO033I

For example, the following Allow commands are not valid:

```
Allow PRT1 * 123 ONLY
Allow PRT2 (*) 123 ONLY
```

The ONLY option is limited to the Allow command. Therefore, this error condition applies to the ALLOW command only.

User response: Replace with port or port name.

IHVO020I *aa* option is limited to the ALLOW command.

Explanation: The ONLY and ALSO options are limited to the ALLOW command.

The variable *aa* is either ONLY or ALSO.

For example, the following I/O operations commands are not valid:

```
BLOCK PRT1 123 ONLY
DISPLAY CHP * ALSO
```

User response: Remove the ONLY or ALSO option and reenter the command.

IHVO021I *aa* option is limited to the REMOVE SWITCH command.

Explanation: The NODisable and Disable options are limited to the REMOVE SWITCH command.

The variable *aa* is either NODISABLE or DISABLE.

For example, the following I/O operations commands are not valid:

```
BLOCK PRT1 123 NODisable
DISPLAY CHP * Disable
```

User response: Remove the NODISABLE or DISABLE option and reenter the command.

IHVO022I NOCHECK and NOVARY are conflicting options.

Explanation: The NOCheck and NOVary options cannot be used together in the same I/O operations command.

User response: Correct the options and reenter the command.

IHVO023I *aa* option is limited to the SYNC SWITCH command.

Explanation: The Process and NOProcess options are limited to the Sync Switch command.

The variable *aa* is either PROCESS or NOPROCESS.

For example, the following I/O operations commands are not valid:

```
BLOCK PRT1 123 PROCESS
DISPLAY CHP * NOPROCESS
```

User response: Remove the PROCESS or NOPROCESS option and reenter the command.

IHVO024I *aa* option is limited to the DISPLAY VARY command.

Explanation: The SYNC and FAIL options are limited to the DISPLAY VARY command.

The variable *aa* is either SYNC or FAIL.

For example, the following I/O operations commands are not valid:

```
BLOCK PRT1 123 SYNC
DISPLAY CHP * FAIL
```

User response: Remove the SYNC or FAIL option and reenter the command.

IHVO025I FORCE, NOVARY and BACKOUT are conflicting options.

Explanation: The Force, NOVary, and Backout options cannot be used together in the same I/O operations command.

User response: Correct the options and reenter the command.

IHVO032I *aa* option is not valid with specified command.

Explanation: When used as options, Block and Connect are limited to the DISPLAY PORT, DISPLAY NAME, and DISPLAY SWITCH commands.

The variable *aa* is either BLOCK or CONNECT.

For example, the following I/O operations commands are not valid:

```
DISPLAY CHP * BLOCK
Allow TAPE_1 123 CONNECT
```

User response: Remove the BLOCK or CONNECT option and reenter the command.

IHVO033I *aa* option is not valid with specified command.

Explanation: The specified option is not valid for the command that was entered.

The variable *aa* shows the option that is not valid.

User response: Reenter the command with valid command options.

IHVO048I *aa* option is not valid with specified command.

Explanation: The ON, OFF, and PURGE options are limited to the RESET HOST command.

The variable *aa* is either ON, OFF, or PURGE.

User response: Reenter the command with a valid option.

IHVO064I I/O-OPS is already running.

Explanation: While I/O operations was already running on this host, an attempt was made to start another I/O operations. The attempt to start another I/O operations was rejected by the I/O operations program that is already running. The I/O operations that is running sent this message and continues processing.

User response: None.

IHVO065I I/O-OPS is not running.

Explanation: A STOP command was entered. However, I/O operations was either not started or was already stopped.

User response: None.

IHVO100I The PARM parameter *aa* has been accepted.

Explanation: The indicated PARM parameter overrides the corresponding default value.

The variable *aa* is the name of the parameter.

User response: None.

IHVO101I PARM parameter *aa* incorrectly specified.

Explanation: The indicated PARM parameter has been incorrectly specified. Either the value is missing, or the value is incorrect, or a value has been specified for a positional parameter.

The variable *aa* is the name of the parameter.

User response: Correct the value of the indicated parameter and restart I/O operations.

IHVO102I Unknown PARM parameter *aa*.

Explanation: The indicated PARM parameter is not known by I/O operations.

The variable *aa* is the name of the parameter.

User response: Correct or remove the parameter and restart I/O operations.

IHVO252E ESTAE failed, RC=*aa*.

Explanation: The I/O operations ESTAE environment could not be established.

The variable *aa* shows the return code.

User response: For an explanation of the return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. Restart I/O operations. If this error recurs, contact your system programmer.

System programmer response: Determine why the ESTAE environment could not be established.

IHVO253E I/O-OPS MVS Communications Input Buffer is not valid.

Explanation: The I/O operations MVS Communications Input Buffer (CIB) is not valid.

System programmer response: Determine why the CIB is not valid.

IHVO255E Unrecoverable I/O-OPS error.

Explanation: There was an unrecoverable I/O operations error.

System programmer response: Check the console log and the I/O operations dump to determine why this error occurred.

IHVQ036I Display command caused data overflow, RC=*aa* RSN=*bb*.

Explanation: The command completed, but an overflow condition occurred that caused a non-zero return code to be generated. This overflow condition was caused by the DISPLAY command generating more information than can be returned in a single display.

User response: The command has returned as much of the data that you requested as possible on the interface that you are using.

IHVR001I I/O-OPS command

Explanation: This is the control line header for a multiple line display.

User response: None.

IHVR010I Command entered by *aa* on *bb*:

Explanation: This is the first message of the audit trail in the system log.

The variable *aa* shows the user ID that the command was entered from.

The variable *bb* shows the system ID where the command was entered.

User response: None.

IHVR011I *aa*

Explanation: This is the second message of the audit trail in the system log. It is a variable length message, with a maximum of 120 characters, that contains the command name, command operands, and command options that I/O operations is about to process.

The long form of the command name, operands, and options is used for all commands except the WRITEPORT command. On the WRITEPORT command, the abbreviated form of the connectivity commands, specified as operands, and the command options are used. The following abbreviations are used for auditing the WRITEPORT command:

- Block = B
- Unblock = U
- Connect = C
- Disconnect = N
- Noforce = NOF
- Force = F
- Vary = V
- Novary = NOV
- Backout = BA
- Nobackout = NOB

On a WRITEPORT command, the 6th operand (if specified) contains 256 characters defining the allow or prohibit status for the specified port. For this message, the 256 characters are converted to 8 groups of 8 hex characters each, with each bit representing a port's allow or prohibit status. If the bit is 0, that port is allowed. If it is 1, it is prohibited. The groups are separated by blanks. Some examples of this message are:

```
IHVR011I GETLOCK
IHVR011I DISPLAY HOST* 0FF0
IHVR011I ALLOW (C1) (C3) 01E4 ALSO NOFORCE VARY BACKOUT
IHVR011I WRITEPORT TEST1 0FD0 UNBLOCK CONNECT 120C
FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFF 60000000 00000000
NOFORCE VARY BACKOUT
```

User response: None.

IHVR012 *aa bb cc dd ee ff*

Explanation: This is the first line of a 4-line message block of the audit trail in the system log that records WRITEFILE and WRITESWCH commands.

It shows the first 20 bytes of the WTFL (WRITEFILE) or WSWT (WRITESWCH) control block of each port that is affected by the command.

Because the control blocks do not fit into a single output line, each control block is split into 4 output lines. The first line of these lines is prefixed by the control block's port number in hex, for example:

```
IHVR011I WRITESWCH [NO]FORCE [NO]VARY [NO]BACKOUT
pn _____ (IHVR012)
_____ (IHVR013)
_____ (IHVR013)
<_____> _____ (IHVR014)
pn _____ (IHVR012)
_____ (IHVR013)
_____ (IHVR013)
<_____> _____ (IHVR014)
:
```

User response: None.

IHVR013 *aa bb cc dd*

Explanation: This is the second and third message of a 4-line message block of the audit trail in the system log that records WRITEFILE and WRITESWCH commands.

It shows the prohibit dynamic connection mask (PDCM) that is part of the WTFL (WRITEFILE) or WSWT (WRITESWCH) control block of each port that is affected by the command, for example:

```
IHVR011I WRITESWCH [NO]FORCE [NO]VARY [NO]BACKOUT
pn _____ (IHVR012)
_____ (IHVR013)
_____ (IHVR013)
<_____> _____ (IHVR014)
pn _____ (IHVR012)
_____ (IHVR013)
_____ (IHVR013)
<_____> _____ (IHVR014)
:
```

User response: None.

IHVR014 *<aa> bb*

Explanation: This is the last message of a 4-line message block of the audit trail in the system log that records WRITEFILE and WRITESWCH commands.

It shows the port logical name and chain information that are part of the WTFL (WRITEFILE) or WSWT (WRITESWCH) control block of each port that is affected by the command, for example:

```
IHVR011I WRITESWCH [NO]FORCE [NO]VARY [NO]BACKOUT
pn _____ (IHVR012)
_____ (IHVR013)
_____ (IHVR013)
<_____> _____ (IHVR014)
pn _____ (IHVR012)
_____ (IHVR013)
_____ (IHVR013)
<_____> _____ (IHVR014)
:
```

User response: None.

IHVR020I *aa* command successful.

Explanation: The command completed successfully. This is an audit trail message.

The variable *aa* shows the name of the command.

User response: None.

IHVR021I *aa* command successful, RC=*bb* RSN=*cc*.

Explanation: This is an audit trail message. The command completed, but a condition was detected that caused a non-zero return code to be generated. For example, if the Force option was used and a host vetoed the command, the command would still be processed.

The variable *aa* shows the name of the command.

The variable *bb* shows the return code.

The variable *cc* shows the reason code.

User response: None.

For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVR030I *aa* command failed, RC=*bb* RSN=*cc*.

Explanation: This is an audit trail message. The command either failed, or completed only partially. Partial completion will occur where a command applies to all switches, but a port name specified in the command does not exist in all of them.

The variable *aa* shows the name of the command.

The variable *bb* shows the return code.

The variable *cc* shows the reason code.

User response: None.

IHVR040I *aa*

Explanation: This message is the array header of the audit trail multiline input record. This message has a maximum length of 70 characters because it is used as part of a multiline WTO to record array data on multisystem commands that require such data to be recorded. This message contains the number and type of array elements that are used in the succeeding audit records. For example, the message could be one of the following:

```
IHVR040I 3 SCOPE HOSTS :
IHVR040I 15 DEVICES :
IHVR040I 5 CHPIDS:
```

User response: None.

IHVS001I *aa* on *bb* is processing an I/O-OPS command.

Explanation: Another user is processing an I/O operations command.

The variable *aa* shows the user ID that the command was entered from.

The variable *bb* shows the VTAM name of the I/O operations.

User ID ACB_SYNC is used when attempting to open the VTAM ACB at a timed interval.

User response: Either wait until the current owner has issued an UNLOCK command or, if you have the authority, issue a GETLOCK command to seize the lock. (If the user ID is HW_SYNCH, this I/O operations ID is updating its database and will release the lock when done.)

IHVS002I *aa* on *bb* has the I/O-OPS lock.

Explanation: User ID *aa* caused the processing lock to be assigned to the I/O operations whose VTAM name is *bb*.

The variable *aa* shows the user ID.

The variable *bb* shows the VTAM name of the I/O operations.

User response: See the response to IHVS001I.

IHVS003I *aa* on *bb* has taken control from you via GETLOCK.

Explanation: User id *aa* has entered a GETLOCK command to the I/O operations whose VTAM name is *bb*.

The variable *aa* shows the user ID.

The variable *bb* shows the VTAM name of the I/O operations.

User response: Wait until the other user enters the UNLOCK command. Then use I/O operations DISPLAY commands to determine what changed since the other user took control from you.

IHVS004I I/O-OPS internal error *aa*.

Explanation: I/O operations detected an internal processing error.

The variable *aa* shows the command code.

System programmer response: Record the sequence of events leading up to this message, this message number, and the command code. Report the problem to your IBM Support Center.

IHVT001E PC routine encountered error condition
aa, RSN=bb.

Explanation: The PC routine that writes trace records encountered the indicated error condition. The current trace entry has not been recorded. For details of the error condition see *z/OS MVS System Codes*.

The variable *aa* shows the system completion code.

The variable *bb* shows the value of the reason code.

User response: Determine the cause of the problem and take appropriate corrective action.

IHVT002I Component trace has not been registered.

Explanation: Component IHVAS has not been registered to the MVS component trace. No tracing is performed.

User response: Search the syslog for *IHVTnnn* messages to determine the reason why the trace could not be registered. If tracing is required you need to recycle the I/O operations address space to register the component trace.

IHVT003I Data space for tracing is not available.

Explanation: The data space that holds intermediate trace data could not be created. No tracing is performed.

User response: Search the syslog for *IHVTnnn* messages to determine the reason why I/O operations failed to create the data space. If tracing is required you need to recycle the I/O operations address space to create the data space.

IHVT004I CT registration encountered a problem,
RC=aa RSN=bb.

Explanation: The component trace registration encountered a problem related to the CTIIHVZZ parmlib member. The registration is retried without a parmlib member.

The variable *rc* shows the value of the return code.

The variable *rs* shows the value of the reason code.

User response: Analyze the return code and reason code (see *CTRACE* in *z/OS MVS Programming: Authorized Assembler Services Reference (ALESERV-DYNALLOC)*, SA22-7609).

IHVT005I Data space for tracing is not available,
RC=aa.

Explanation: The data space that holds intermediate trace data could not be made available to the I/O operations address space. No tracing is performed.

The variable *rc* shows the value of the return code of the *ALESERV* macro.

User response: Search the syslog for *IHVTnnn* messages to determine the reason why I/O operations failed to create the data space. If tracing is required you need to recycle the I/O operations address space to create the data space.

IHVT006I Unable to process trace record.

Explanation: An unrecoverable error occurred when I/O operations tried to process a trace record.

User response: Stop and restart I/O operations.

IHVT105I De-/Registration from/to CT failed,
RC=aa RSN=bb.

Explanation: The registration to or deregistration from the MVS component trace failed. Tracing is unavailable until the next startup of I/O operations.

The variable *rc* shows the value of the return code.

The variable *rs* shows the value of the reason code.

User response: Analyze the return code and reason code (see *CTRACE* in *z/OS MVS Programming: Authorized Assembler Services Reference (ALESERV-DYNALLOC)*, SA22-7609).

IHVT107I CT start/stop routine abended with completion code *aa*.

Explanation: The component trace start or stop routine abended with the system completion code *aa*. The *TRACE* command is not processed and the trace options remain unchanged.

The variable *aa* shows the system completion code of the abend.

User response: Contact your IBM Support Center.

IHVT110I CT display routine abended with completion code *aa*.

Explanation: The component trace display routine abended with the system completion code *aa*. The *DISPLAY* command is not processed.

The variable *aa* shows the system completion code of the abend.

User response: Contact your IBM Support Center.

IHVT170I Unknown trace option found: *aa*.

Explanation: The variable *aa* shows the trace option specified with the *OPTIONS* parameter of the *IPCS CTRACE* command. The following are valid values:

- MAX** Show all trace entries
- MORE** Show MIN entries plus processing relevant entries
- MIN** Show entry and exit trace entries only
- ALL** Includes all the following options

AM	The access method routines
API	The API routines running in the user's address space
AUDIT	The message service and audit routines
BASE	The routines that are common to all services
CMD	The routines that handle the command input
COMMUNICATION	The communication framework
CONFIG	The routines performing hardware configuration changes
DB	The database routines
QUERY	The service routines returning database information
RECOVERY	The recovery routines
SCHEDULE	The schedule framework
XM	The cross-memory services

User response: Correct or delete the option.

IHVT171I Unknown keyword found: *aa*.

Explanation: The parser found a keyword that is not valid. The keyword is ignored. Valid keywords are

- JOBNAME
- ASID
- TASKID
- MODULE
- POS

User response: Specify a correct keyword.

IHVT172I Too many value(s) specified for keyword *aa*.

Explanation: All keyword options that are allowed for the OPTIONS parameter support up to five different values. However, the parser found at least one additional value for the indicated keyword. The surplus values are ignored.

The variable *aa* shows the affected filter option.

User response: Correct the keyword option.

IHVT173I Value *aa* of keyword *bb* is missing.

Explanation: The parser expects a string of at least one character for the keyword option *aa* with the OPTIONS parameter. However, no string or a string representing the NULL string has been found.

The variable *aa* shows the value that is missing.

The variable *bb* shows the affected filter option.

User response: Correct the value.

IHVT175I Parsing the OPTION parameter stopped at column *aa*.

Explanation: The parser found an error condition when parsing the OPTION parameter. Processing is terminated.

The variable *aa* shows the column that caused the parser to stop

A column value of 0 can be caused by one of the following:

- A parameter consisting of blanks and commas only
- A parameter length greater than 32K
- An allocation error of temporary storage

A column value greater than 0 indicates one of the following:

- A keyword is not spelled correctly
- A keyword is incorrectly delimited
- A trace option is incorrectly delimited

User response: Correct the OPTION parameter, or increase the region size.

IHVT176I No SCOPE option specified.

User response: Correct the OPTIONS parameter. Refer to "Using Trace Services for I/O Operations" in *IBM Tivoli System Automation for z/OS User's Guide* for valid trace options.

IHVT184I The external writer has been stopped automatically.

Explanation: The component trace has been stopped either by a TRACE command or during I/O operations termination. The start/stop routine detected that the external writer was active, but has not connected any other component trace. As a result, the start/stop routine issued the TRACE CT,WTRSTOP command.

User response: None.

IHVT185I The external writer has been started automatically.

Explanation: The TRACE command specified the parameter for the external writer. The start/stop routine detected that the external writer has not been started yet. As a result, the trace routine issued the TRACE CT,WTRSTART command.

User response: None.

IHVT188I Parse service failed, RC=*aa*.

Explanation: The start/stop routine encountered a storage problem when parsing the OPTION parameter. The TRACE command is discarded.

The variable *aa* shows the return code of the service.

User response: Try the command later. If the problem

persists, contact your IBM Support Center.

IHVT191I Parameter WTR=DISCONNECT is not supported on a start trace command.

Explanation: The start/stop routine expects the name of the startup procedure of the external writer to be used. However, it found the reserved word DISCONNECT. The TRACE command is discarded.

User response: Correct the WTR parameter and reissue the TRACE command.

IHVT192I Invalid trace option specified.

Explanation: A non-positional trace option was found that is not supported. The TRACE command is discarded.

User response: Correct the OPTION parameter and reissue the TRACE command. For valid options see message IHVT170I.

IHVT193I No trace option specified.

Explanation: Either the OPTION parameter was omitted, or no value was specified for it. The trace command is rejected.

User response: Correct the OPTION parameter and reissue the TRACE command. For valid options see message IHVT170I.

IHVT194I Unknown trace option found: aa.

Explanation: An unsupported trace option has been detected. The trace command is rejected.

The variable *aa* shows the improper option.

User response: Correct the OPTION parameter and reissue the TRACE command. For valid options see message IHVT170I.

IHVT198I Data space could not be created, RC=aa RSN=bb.

Explanation: The data space that holds the intermediate trace buffers could not be created for the indicated reason. Tracing is unavailable until the next startup of I/O operations.

The variable *aa* shows the value of the return code.

The variable *bb* shows the value of the reason code.

User response: Analyze the return code and the reason code (see DSPSERV in *z/OS MVS Programming: Authorized Assembler Services Reference (ALESERV-DYNALLOC)*, SA22-7609).

IHVT199I Last trace buffer could not be written, RC=aa RSN=bb.

Explanation: The start/stop routine encountered an error when passing the last trace buffer to the external writer synchronously. The buffer is not available in the output data set.

The variable *aa* shows the value of the return code.

The variable *bb* shows the value of the reason code.

User response: Analyze the return code and the reason code (see CTRACEWR in *z/OS MVS Programming: Authorized Assembler Services Reference (ALESERV-DYNALLOC)*, SA22-7609).

IHVU002I HNDIUCV SET macro failed, I/O-OPS RC=aa, CMS RC=bb.

Explanation: A request from I/O operations in VM to identify itself as an IUCV program to CMS failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the CMS return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the CMS return code, refer to *z/VM CP Programming Services (SC24-6084)*.

IHVU003I I/O-OPS not logged on. CONNECT macro RC=aa RSN=bb.

Explanation: A request from a user in VM to establish an IUCV communication path to I/O operations failed because I/O operations is not logged on.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Log on I/O operations and reenter the command. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVU004I CONNECT macro failed, RC=aa RSN=bb.

Explanation: A request from a user to establish an IUCV communication path to I/O operations failed because the I/O operations virtual machine has not declared an external interrupt buffer.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Report the problem to your IBM Support Center. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVU005I Not IUCV authorized. CONNECT macro RC=*aa* RSN=*bb*.

Explanation: A request from a user in VM to establish an IUCV communication path to I/O operations failed because the user was not authorized to establish an IUCV path to I/O operations.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Contact your system programmer to obtain proper IUCV authorization. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVU006I CONNECT macro failed. I/O-OPS RC=*aa*, CMS RC=*bb*.

Explanation: A request from a user in VM to establish an IUCV communication path to I/O operations failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the CMS return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the CMS return code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVU007I SEND macro failed. I/O-OPS RC=*aa*, IUCV RC=*bb*.

Explanation: A request from a user in VM to send I/O operations command data to I/O operations failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV return code.

User response: Try the command again. If the error occurs again, contact your system programmer.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVU008I RECEIVE failed. *aa* on *bb* I/O-OPS RC=*cc*, IUCV RC=*dd*.

Explanation: A request from I/O operations in VM to receive command data through IUCV communications failed, and an attempt to send the failing return code and reason code back to the user through a SEVER also failed. This message is sent only to the system operator.

The variable *aa* shows the user ID.

The variable *bb* shows the system id.

The variable *cc* shows the I/O operations return code.

The variable *dd* shows IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVU009I Command results unknown. I/O-OPS RC=*aa*, IUCV IPAUDIT=*bb*.

Explanation: A request from I/O operations in VM to return data to a user through IUCV communications failed. The command was processed, but the results are not available.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV IPAUDIT reply (hexadecimal).

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the meaning of the IPAUDIT field, refer to the IUCV section of *z/VM CP Programming Services* (SC24-6084).

IHVU011I System macro *aa* failed. I/O-OPS RC=*bb* RSN=*cc*.

Explanation: A MVS system macro failed.

The variable *aa* shows the macro name.

The variable *bb* shows the I/O operations return code.

The variable *cc* shows the I/O operations reason code.

User response: Contact your MVS system programmer. Enter the DISPLAY RESULTS command at the I/O operations operator console to determine the status of the switch.

System programmer response: For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For information about MVS macros, refer to *z/OS MVS Programming: Authorized Assembler Services Reference* (SA22-7609 through , SA22-7612).

IHVU012I TSO Service Facility failed. I/O-OPS RC=*aa* RSN=*bb*.

Explanation: A request from I/O operations in MVS to IKJEFTSR failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the I/O operations reason code.

The I/O operations return codes and reason codes are defined as follows:

- When return code = 8 and reason code = 30B0 *xxxx*:
The TSO service routine returned a non-zero return code, which is the last four characters of the I/O operations reason code.
- When return code = 8 and reason code = 30B1 *xxxx*:
The TSO service routine abnormally ended. The abend code is the last 4 characters of the I/O operations reason code.

User response: Contact your MVS system programmer.

System programmer response: For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. Refer to *TSO Programming Facility II Description/Operations*. for an explanation of the TSO return codes and abend codes.

IHVU015I Unrecoverable error. The I/O-OPS program has terminated.

Explanation: User-to-I/O operations communication cannot occur because the I/O operations program stopped.

System programmer response: Report the problem to your IBM Support Center.

IHVU018I HNDIUCV CLR macro failed. I/O-OPS RC=*aa*, CMS RC=*bb*.

Explanation: A request from I/O operations in VM to remove itself from the IUCV environment failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVU019I *aa* on *bb* is processing an I/O-OPS command.

Explanation: Another user is processing a I/O operations command.

The variable *aa* shows the user ID.

The variable *bb* shows the I/O operations VTAM name.

User response: Contact the other user to determine when command processing is done and the UNLOCK command has been entered.

IHVU020I IUCVINI SET macro failed. I/O-OPS RC=*aa*, IUCVINI RC=*bb*.

Explanation: A request from I/O operations in VM to identify itself as an IUCV program to the Group Control System failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084). For information about the Group Control System, refer to *z/VM Group Control System* (SC24-6098).

IHVU021I RECEIVE macro failed. I/O-OPS RC=*aa*, IUCV RECEIVE RC=*bb*.

Explanation: A request from I/O operations in VM to receive command data from a user through IUCV failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVU022I REPLY failed. *aa* on *bb* I/O-OPS RC=*cc*, IUCV RC=*dd*.

Explanation: A request from I/O operations in VM to reply to the command data through IUCV communications failed. An attempt to send the failing return code and reason code back to the user also failed. This message is sent only to the system operator.

The variable *aa* shows the user ID.

The variable *bb* shows the system id.

The variable *cc* shows the I/O operations return code.

The variable *dd* shows the IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVU023I REPLY macro failed. I/O-OPS RC=*aa*, IUCV REPLY RC=*bb*.

Explanation: A request from I/O operations in VM to reply or send back command data to the user through IUCV failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVU024I ACCEPT failed. *aa* on *bb* I/O-OPS RC=*cc*, IUCVCOM RC=*dd*.

Explanation: A request from I/O operations in VM to accept a connection request from the user failed. An attempt to send the failing return code and reason code back to the user also failed. This message is sent only to the system operator.

The variable *aa* shows the user ID.

The variable *bb* shows the system id.

The variable *cc* shows the I/O operations return code.

The variable *dd* shows the IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084). For information about the Group Control System, refer to *z/VM Group Control System* (SC24-6098).

IHVU025I ACCEPT macro failed. I/O-OPS RC=*aa*, IUCVCOM RC=*bb*.

Explanation: A request from I/O operations in VM to accept a connection request from a user through IUCV failed.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084). For information about the Group Control System, refer to *z/VM Group Control System* (SC24-6098).

IHVU026I SEVER failed. *aa* on *bb* I/O-OPS RC=*cc*, IUCVCOM RC=*dd*.

Explanation: A request from I/O operations in VM to sever an IUCV communication path failed. An attempt to send the failing return code and reason code back to the user also failed. This message is only sent to the system operator.

The variable *aa* shows the user ID.

The variable *bb* shows the system id.

The variable *cc* shows the I/O operations return code.

The variable *dd* shows the IUCV return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVU027I STORAGE OBTAIN failed.

Explanation: There was a STORAGE OBTAIN failure in I/O operations while trying to access either subpool 230, 241, 245, or user private subpool 0.

System programmer response: Determine why there was not enough storage and correct the condition. Reenter the I/O operations command.

IHVU028I GETMAIN failed. *aa* on *bb* RC=*cc* RSN=*dd*.

Explanation: A request from I/O operations in VM to get main storage failed. An attempt to send the failing return code and reason code back to the user also failed. This message is sent only to the system operator.

The variable *aa* shows the user ID.

The variable *bb* shows the system id.

The variable *cc* shows the return code.

The variable *dd* shows the reason code.

System programmer response: Determine why there was not enough storage and correct the condition. Reenter the command. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVU029I I/O-OPS is not operational.

Explanation: The command failed because I/O operations is not running.

User response: Start I/O operations.

IHVU030I Another user is busy processing an I/O-OPS command

Explanation: Another user is processing an I/O operations command.

User response: Enter your command again. If it fails, either enter the GETLOCK command or wait for the other user to enter the UNLOCK command.

IHVV100I Device *aa*.

Explanation: The IEEVARY return code is 0. The reason code is 0.

The variable *aa* is either 'ONLINE' or 'OFFLINE'.

IHVV110I • IHVV143I

User response: See MVS message IEE302I or IEF281I.

IHVV110I Device already *aa*.

Explanation: The IEEVARY return code is 4. The reason code is 0.

The variable *aa* is either 'ONLINE' or 'OFFLINE'.

User response: See MVS message IEE302I or IEE303I.

IHVV120I Device pending offline.

Explanation: The IEEVARY return code is 8. The reason code is 0.

User response: See MVS message IEE794I.

IHVV130I Device came online with restrictions.

Explanation: The IEEVARY return code is 12. The reason code is 0.

User response: See MVS message IEE476I.

IHVV131I Device has no UCB.

Explanation: The IEEVARY return code is xx. The reason code is x.

User response: See MVS message IEE313I.

IHVV132I Device has no *aa* paths.

Explanation: The IEEVARY return code is 12 or 16, the reason code is 6 or 7. See explanation for MVS message IEE025I.

The variable *aa* shows either LOGICAL or PHYSICAL.

User response: None.

IHVV133I Device in use by system function.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 8.

User response: See MVS message IEE329I.

IHVV134I Device vary processing aborted.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 9.

User response: See MVS message IEE328I.

IHVV135I Dynamic pathing not operational on device.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 10.

User response: See MVS message IEE791I.

IHVV136I Device assigned to another system.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 11.

User response: See MVS message IEE791I.

IHVV137I Assign conflicts with current device status.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 12.

User response: See MVS message IEE791I.

IHVV138I Assign failed.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 13.

User response: See MVS message IEE791I.

IHVV139I Device pending offline and boxed.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 14.

User response: See MVS message IEE793I.

IHVV140I Device is managed by JES3.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 21.

User response: See MVS message IEF772I.

IHVV141I Console device is currently changing status.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 22.

User response: See MVS message IEE339I.

IHVV142I Console device does not support action.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 23.

User response: See MVS message IEE313I.

IHVV143I Vary of master console is rejected, force is required.

Explanation: The IEEVARY return code is 12 or 16. The reason code is 24.

User response: See MVS message IEE276I.

IHSV144I **Vary of master console is rejected, other active consoles exist.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 25.

User response: See MVS message IEE276I.

IHSV145I **Vary of master console failed.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 26.

User response: See MVS message IEE370I.

IHSV146I **Vary of master console candidate is rejected.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 27.

User response: See MVS message IEE367I.

IHSV147I **Device is a hardcopy console.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 28.

User response: See MVS message IEE229I.

IHSV150I **Device did not come *aa*.**

Explanation: The IEEVARY return code is 16. The reason code is 0.

The variable *aa* is either 'ONLINE' or 'OFFLINE'.

User response: See MVS message IEE845I or IEE846I.

IHSV151I **Device kept offline by a configuration manager.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 15.

User response: See MVS message IEE424I.

IHSV152I **Device is offline due to operator.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 16.

User response: See MVS message IEE113I.

IHSV153I **Device is in a system-managed tape library.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 17.

User response: See MVS message IEE108I.

IHSV154I **Device is in use by CUIR.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 18 (CUIR = control unit initiated reconfiguration). I/O operations does not remove or restore this device.

User response: See MVS message IEE529I.

IHSV155I **Device is not eligible for a vary operation.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 19.

User response: See MVS message IEE313I.

IHSV156I **Volume could not be read.**

Explanation: The IEEVARY return code is 12 or 16. The reason code is 20.

User response: See MVS message IEF502I, IEF503I, or IEF510I.

IHSV160I **Device is offline and pending boxed.**

Explanation: The IEEVARY return code is 20.

User response: See MVS message IEE845I.

IHSV170I **Device is pending offline and pending boxed.**

Explanation: The IEEVARY return code is 24.

User response: See MVS message IEE846I.

IHSV180I **Unexpected error occurred.**

Explanation: The IEEVARY return code is 32.

User response: An unexpected error occurred in MVS while processing the IEEVARY request. Follow your local procedures for reporting MVS errors.

IHSV200I **AUTOSWITCH for device *aa* (now *bb*).**

Explanation: The AUTOSWITCH attribute for the device was successfully changed.

The variable *aa* shows either REMOVED or RESTORED.

The variable *bb* shows either OFF or ON.

User response: None.

IHSV210I **AUTOSWITCH for device already *aa* (was *bb*).**

Explanation: Device is already in requested AUTOSWITCH mode.

The variable *aa* shows either REMOVED or RESTORED.

IHV230I • IHV2300I

The variable *bb* shows either OFF or ON.

User response: None.

IHV230I Coupling Facility is not available.

Explanation: Shared tape support relies on data sharing across the sysplex through a coupling facility.

User response: None.

IHV236I Device type not valid for use of AUTOSWITCH.

Explanation: The device specified for the AUTOSWITCH option was of an invalid type.

User response: None.

IHV240I Device is not offline.

Explanation: The unit must be OFFLINE before its AUTOSWITCH attribute can be changed.

User response: Vary the device offline.

IHV2300I LINK failed.

Explanation: There has been a link failure. Required module IEEVPTHI in SYS1.LINKLIB cannot be found.

System programmer response: Verify that the required operating system and all required PTFs have been installed.

IHVX000E System error detected, RC=*aa* RSN=*bb*.

Explanation: I/O operations has detected damage to storage or code and reports the return code and reason code, which are listed in Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Stop and start I/O operations to recover from the error, and report the problem to your IBM Support Center.

IHVX001I System service *aa* failed, RC=*bb* RSN=*cc* Mod=*dd*.

Explanation: The indicated system service called by I/O operations failed. I/O operations storage or code may be corrupted.

The variable *aa* shows the service that failed.

The variable *bb* shows the return code.

The variable *cc* shows the reason code.

The variable *dd* shows the name of the module.

User response: Contact your system programmer.

System programmer response: If the system service is either HCDSETUP or HCD OPEN, there might be a

problem with the IODF. Check the system log for related HCD messages.

For other system services, refer to the description of the system services in *z/OS MVS Programming: Authorized Assembler Services Reference* (SA22-7609 through SA22-7612), or to *z/OS MVS Programming: Sysplex Services Reference* (SA22-7618) for further information on the return code and the reason code.

Stop and start I/O operations.

If this does not solve the problem, report it to your IBM Support Center including the return code *bb*, the reason code *cc*, and the module name *dd*.

IHVX002I VARY SWITCH failed in module *aa*.

Explanation: I/O operations issued the VARY SWITCH system service. This message occurs when the VARY SWITCH service does not get the required amount of storage (STORAGE OBTAIN failed).

The variable *aa* shows the name of the module where the error occurred.

User response: Restart I/O operations if the hardware configuration was changed. If this does not solve the problem, report the problem to your IBM Support Center including the module name *aa*.

System programmer response: Determine why there was not enough storage and correct the condition.

IHVZ100I Unrecognized parameter, RC=*aa* RSN=*bb*.

Explanation: An incorrect parameter was found.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Report the problem to your IBM Support Center.

IHVZ200I I/O-OPS database update started.

Explanation: The I/O operations database is being updated due to hardware reconfiguration changes.

User response: None.

IHVZ210I I/O-OPS database update complete.

Explanation: The I/O operations database has been updated due to hardware reconfiguration changes.

User response: None.

IHVZ300I O/S communication established.

Explanation: All required operating system communication has been started for hardware reconfiguration changes.

User response: None.

IHVZ400I Communication error, RC=*aa* RSN=*bb*.

Explanation: An operating system communication error has occurred. Restart I/O operations if any hardware reconfiguration changes are made. If communication is established, message IHVZ300I is issued.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Restart I/O operations if any hardware reconfiguration changes are made. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

**IHVZ510I O/S communication error, RC=*aa*
RSN=*bb*.**

Explanation: An operating system communication error has occurred.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Report the problem to your IBM Support Center. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVZ600I ESTAE macro load error, RC=*aa* RSN=*bb*.

Explanation: I/O operations encountered an error loading the ESTAE macro.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Report the problem to your IBM Support Center. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVZ610I ESTAE macro end error, RC=*aa* RSN=*bb*.

Explanation: I/O operations encountered an error when ESTAE 0 was issued.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Report the problem to your IBM Support Center. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

**IHVZ620I ESTAE module delete error, RC=*aa*
RSN=*bb*.**

Explanation: I/O operations encountered an error when attempting to delete IHVSTAS from virtual storage.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Report the problem to your IBM Support Center. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVZ700I I/O-OPS RC=*aa*, IUCVINI SET RC=*bb*.

Explanation: A request from I/O operations in VM to identify itself as an IUCV program to the Group Control System failed. I/O operations cannot automatically process hardware reconfiguration changes.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCVINI SET return code.

User response: Restart I/O operations if any hardware reconfiguration changes are made.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084). For information about the Group Control System, refer to *z/VM Group Control System* (SC24-6098).

**IHVZ701I I/O-OPS RC=*aa*, IUCVCOM CONNECT
RC=*bb*.**

Explanation: A request from I/O operations in VM to establish communication with the *CONFIG CP system service failed. I/O operations cannot automatically process hardware reconfiguration changes.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCVCOM CONNECT return code.

User response: Restart I/O operations if any hardware reconfiguration changes are made.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084). For information about the Group Control System, refer to *z/VM Group Control System* (SC24-6098).

IHVZ710I I/O-OPS RC=*aa*, IUCVINI CLR RC=*bb*.

Explanation: A request from I/O operations to the Group Control System in VM to remove itself from the IUCV environment failed.

The variable *aa* shows the I/O operations return code.

IHVZ720I • IHVZ760I

The variable *bb* shows the IUCVCOM CLR return code.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084). For information about the Group Control System, refer to *z/VM Group Control System* (SC24-6098).

IHVZ720I I/O-OPS RC=*aa* RSN=*bb*.

Explanation: A connection between I/O operations and the *CONFIG CP system service has failed. Restart I/O operations if any hardware reconfiguration changes are made. Any subsequent I/O operations command attempts to reestablish the connection. If the attempt is successful, message IHVZ300I is sent.

The variable *aa* shows the return code.

The variable *bb* shows the reason code.

User response: Restart I/O operations if any hardware reconfiguration changes are made. For more information about return codes and reason codes, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385.

IHVZ730I I/O-OPS RC=*aa*, IUCV RECEIVE RC=*bb*.

Explanation: A request from I/O operations in VM to receive a reconfiguration record from the operating system failed. I/O operations cannot automatically process hardware reconfiguration changes.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV RECEIVE return code.

User response: Restart I/O operations if any hardware reconfiguration changes are made.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV return code, refer to *z/VM CP Programming Services* (SC24-6084). For information about the Group Control System, refer to *z/VM Group Control System* (SC24-6098).

IHVZ740I I/O-OPS RC=*aa*, IUCV RECEIVE CC=*bb*.

Explanation: A request from I/O operations in VM to receive a reconfiguration record from the operating system failed. I/O operations cannot automatically process hardware reconfiguration changes.

The variable *aa* shows the I/O operations return code.

The variable *bb* shows the IUCV RECEIVE return code.

User response: Restart I/O operations if any hardware reconfiguration changes are made.

System programmer response: For an explanation of the I/O operations return code, refer to Appendix A, "Return Codes, Status Codes, and Reason Codes," on page 385. For an explanation of the IUCV condition code, refer to *z/VM CP Programming Services* (SC24-6084).

IHVZ750I Unable to process hardware change.

Explanation: An unrecoverable error occurred when I/O operations tried to process hardware reconfiguration changes.

User response: Stop and restart I/O operations.

IHVZ760I ETR status changed.

Explanation: MVS has reported a change in status for the Sysplex Timer, or a port or link connected to a Sysplex Timer.

User response: Review the ETR configuration for the system issuing the message. This can be done via the MVS DISPLAY ETR command.

Chapter 9. Messages ING001I to INGY1203I

ING001I *aaaa* SERVICE FAILED, RC=*bb*,
 REASON=*cccccccc*

Explanation: SA z/OS detected an error.

The variable *aaaa* shows the service/command that failed.

The variable *bb* shows the value of the return code.

The variable *cccccccc* shows the value of the reason code.

User response: If an INGOMX error occurred, refer to *IBM Tivoli System Automation for z/OS Programmer's Reference*.

For a NetView command error, refer to the appropriate NetView manual.

In all other cases, contact your IBM Support Center.

System programmer response: None.

ING002I *name* IS NOT A SECONDARY
 AUTOMATION MANAGER

Explanation: The specified command or action is only applicable for a secondary automation manager.

The variable *name* shows the name of the automation manager that is not defined as secondary automation manager.

System action: Processing terminates.

Operator response: Specify the correct automation manager name and reissue the command.

ING003I *name* IS NOT A PRIMARY
 AUTOMATION MANAGER

Explanation: The specified command or action is only applicable for the primary automation manager.

The variable *name* shows that name of the automation manager that is not defined as primary automation manager.

System action: Processing terminates.

Operator response: Specify the correct automation manager name and reissue the command.

ING004I REQUEST *request* SUBMITTED

Explanation: The request to update the operation mode of the designated automation manager has been submitted, or the request was submitted to another work operator.

The variable *request* shows the submitted request.

System action: Processing continues.

ING005I RESOURCE *name* IS AMBIGUOUS

Explanation: There is more than one resource with the specified name within the domain of the automation manager in charge. Because the command is running in line mode, SA z/OS is unable to determine which of the resources the command should be applied to.

The variable *name* shows the name of the resource that is not unique.

System action: Processing terminates.

Operator response: Specify the fully qualified resource name and reissue the command.

ING006I THERE IS NO RESOURCE *name*

Explanation: None of the resources defined in the sysplex matches the specified name pattern.

The variable *name* shows the name of the resource that could not be found.

System action: None.

Operator response: Correct the name of the resource and reissue the command.

System programmer response: None.

ING007I *name* IS NOT AN APPLICATION
 GROUP

Explanation: The display "group members" action was requested for a resource that is not an application group.

The variable *name* shows the name of the resource that is not an application group.

System action: None.

Operator response: None.

ING008I *name* SERVICE FAILED, RC=*return_code*,
 REASON=*reason*.

Explanation: A service routine was called that ended with a non-zero return code.

The variable *name* shows the name of the function that encountered the error.

The variable *return_code* shows the return code that the failing service routine returned.

The variable *reason* shows descriptive text about the cause of the error.

System action: Processing terminates.

Operator response: Notify your system programmer.

ING009I • ING017I

System programmer response: Determine why the service routine failed and correct the problem. Examine the netlog for additional information. If necessary, contact your IBM Support Center for further assistance.

ING009I **UPDATE OF *parameter* COMPLETED;
SET TO *value***

Explanation: The requested function has been successfully completed.

The variable *parameter* shows the name of the parameter that was updated.

The variable *value* shows the new value of the parameter.

System action: None.

Operator response: None.

ING010I *warning/error information*

Explanation:

The variable *warning/error information* shows descriptive text from the automation manager about the cause of the error.

System action: None.

Operator response: None.

ING011I **CONFLICT DETECTED BETWEEN UP
AND DOWN TIMESLOTS**

Explanation: There is an overlap between an UP time slot and a DOWN time slot.

System action: None.

Operator response: Correct the time specification.

ING012I *name* **IS NOT AN AUTOMATION
MANAGER**

Explanation: The specified command or action is only valid for an automation manager.

The variable *name* shows the name of the system where the command could not be processed.

System action: Processing is terminated.

Operator response: Specify the correct automation manager name and reissue the command.

ING013I **PARAMETER *parameter_name* IS NOT
APPLICABLE FOR RESOURCES OF
TYPE *type***

Explanation: The specified parameter is not valid for resources of the specified type.

The variable *parameter_name* shows the name of the parameter that is not valid.

The variable *type* shows the type of the resource.

System action: Processing terminates.

Operator response: Correct the format and reissue the command.

ING014I **CONFLICTING PARAMETERS
SPECIFIED.**

Explanation: Self-explanatory.

System action: Processing terminates.

Operator response: Correct the format and reissue the command.

ING015I **NO MATCHING REQUEST FOUND
FOR *resource* - SOURCE=*source*
REQUEST=*request***

Explanation: An INGSET CANCEL was issued but the request identified by its type and source does not exist.

The variable *resource* shows the name of the resource.

The variable *source* shows the name of the source.

The variable *request* shows the name of the request that could not be found.

System action: Processing terminates.

Operator response: Correct the format and reissue the command.

ING016I **SCOPE=*scope* IS NOT VALID FOR *type*
REQUEST**

Explanation: The specified scope is not valid for the type of request.

The variable *scope* shows the scope that was specified for the request.

The variable *type* shows the type of the request.

System action: Processing terminates.

Operator response: Correct the format and reissue the command.

ING017I **EXPIRATION TIME *timestamp* HAS
PASSED ON *system***

Explanation: The specified expiration date or time is in the past.

The variable *timestamp* shows the specified timestamp.

The variable *system* shows the name of the system.

System action: Processing terminates.

Operator response: Specify an expiration time or date that is in the future and reissue the command.

ING018I ONLY ONE GROUP ALLOWED FOR ACTION *action*

Explanation: You specified more than one group name for an INGGROUP action of ADJUST or MEMBERS.

The variable *action* shows the action that could not be performed.

System action: Processing terminates.

Operator response: Specify only one group name and reissue the command.

ING019I REQUESTED ACTION IGNORED FOR RESOURCE *resource*

Explanation: An INGGROUP ACTION EXCLUDE, AVOID, DEFAULT, or INCLUDE was requested for a group of nature basic.

The variable *resource* shows the name of the resource.

System action: Processing terminates.

Operator response: Correct the command syntax and reissue the command.

ING020I DATASET *datasetname* **NOT FOUND OR NOT ACCESSIBLE**

Explanation: Either the specified data set does not exist, or it is not accessible, or, if it is a configuration data set it is not a PDS.

The variable *datasetname* shows the name of the data set that is not available.

System action: None.

Operator response: Correct the name of the data set and reissue the command.

ING021I STATISTICS COUNTERS RESET ON SYSTEM *sysname*

Explanation: The statistics counts have been reset on the specified system.

The variable *sysname* shows the name of the system whose statistics counts were reset.

System action: Processing continues.

Operator response: None.

Module: INGRVRPX

ING022I THRESHOLD SETTINGS FOR *resource* **CANNOT BE DELETED**

Explanation: The threshold settings for DEFAULTS, MVSESA and SUBSYSTEM cannot be deleted.

The variable *resource* shows the name of the resource that the threshold settings cannot be deleted for.

System action: None.

Operator response: None.

Module: INGRYTH0

ING023I *command* **COMMAND NOT ALLOWED TO RUN UNDER** *task* **TASK.**

Explanation: The specified command cannot run under the specified task.

The variable *command* shows the name of the command that cannot be run.

The variable *task* shows the name of the task that the command cannot be run under.

System action: Processing stops.

Operator response: Reissue the command under another task.

Module: INGRYRPX

ING024I DELETION OF INTERNAL REQUEST IS NOT ALLOWED.

Explanation: The selected request is created by the automation manager due to:

1. Vote propagation for an application group. This request cannot be canceled. It will be automatically removed when the vote that initiated the request is removed.
2. A service period UP or DOWN window. This request cannot be canceled. It will be automatically removed when the UP or DOWN window ends.

System action: None.

Operator response: None.

ING025I OUTBOUND CONNECTION WITH DOMAIN *domain* **RUNNING** *product* **ESTABLISHED.**

Explanation: The local system has established a communication path with the named system.

The variable *domain* identifies the remote system. It is the domain ID of the remote system.

The variable *product* is the SA z/OS release running on the remote system.

System action: None.

Operator response: None.

ING026I *clistname* **ROUTINE SENT TO DOMAIN** *domain* **LEVEL** *clist_level*.

Explanation: The local system has sent the current version of the INGRVXMT slave routine to the remote system. This enables the local system to communicate with the remote system via the "remote procedure call" technique rather than NNT sessions.

The variable *clistname* is the name of the slave routine.

The variable *domain* identifies the remote system by its domain ID.

The variable *clist_level* shows the last APAR number, if present, and compile date.

System action: None.

Operator response: None.

ING027I SLAVE ROUTINE *clistname* RECEIVED FROM DOMAIN *domain* RUNNING product LEVEL *clist_level*.

Explanation: Another system, identified by its domain ID, has sent a slave routine to communicate with the local system via the "remote procedure call" technique rather than NNT sessions.

The variable *clistname* is the name of the slave routine.

The variable *domain* is the domain ID of the remote system.

The variable *product* is the SA z/OS release level running on the remote system.

The variable *clist_level* shows the last APAR number and compile data of the slave routine.

System action: None.

Operator response: None.

ING028I *command*: PARAMETER *parameter* MISSING OR INVALID *value*

Explanation: The specified parameter is either wrong or missing.

The variable *command* is the name of the command being processed.

The variable *parameter* is the parameter that is invalid or missing.

The variable *value* is the value of the parameter that is wrong.

System action: None.

Operator response: None.

ING029I SLAVE ROUTINE *clistname* IS COMPILED. CANNOT BE SENT TO DOMAIN *domain*

Explanation: Because the clist that is named in the message is compiled, it cannot be sent to the remote system. This is done for two reasons:

1. The REXX runtime library might not be installed on the remote system.
2. The compiled REXX routine does not contain any indication of its software level (release number, last APAR).

The variable *clistname* is the name of the slave routine.

The variable *domain* identifies the remote system by its domain ID.

System action: None.

Operator response: None.

ING030I Group *group_name* cannot be moved to another system - *reason*

Explanation: An attempt was made to move the group away from the system that it currently runs on, but no other system is available.

The variable *group_name* is the move group concerned.

The variable *reason* explains in more detail why the group cannot be moved.

System action: None.

Operator response: None.

System programmer response: None.

ING031I EVENT SUBSCRIPTION CONSUMER *name* IS UNKNOWN.

Explanation: The consumer specified in the INGEVOBS call is not defined, which means that the consumer is not registered.

The variable *name* is the name of the consumer as specified in the INGEVOBS command.

System action: The command is rejected.

Operator response: Reenter the correct command, if applicable or register the consumer first.

Classes: None.

ING032I AVAILABILITY TARGET *avtgt* CANNOT BE LESS THAN THE SATISFACTORY TARGET *sattgt*

Explanation: You have specified either a satisfactory target value that is larger than the availability target, or an availability target that is less than the satisfactory target assigned to the server group.

The variable *avtgt* is the availability target assigned to the group.

The variable *sattgt* is the satisfactory target assigned to the group.

System action: The command is rejected.

Operator response: Reenter the correct command.

Classes: None.

ING033I **INBOUND SESSION STATUS
CORRECTED FOR** *domain*. **SET** *status*

Explanation: The gateway monitoring routine found an inbound session status that does not match the actual status. The session status is corrected.

The variable *domain* is the name of the domain that the local system has an inbound session with.

The variable *status* is the new inbound session status.

System action: Processing continues.

Operator response: None.

Classes: None.

ING034I **STATUS** *stat* **OF SUBSYSTEM** *subsys* **IS
NOT VALID FOR FUNCTION** *function*

Explanation: AOFrysAB was called to perform a function for a subsystem. The status of the subsystem does not allow this function.

The variable *stat* is the current status of the subsystem.

The variable *subsys* is the subsystem that the function is to be performed on.

The variable *func* is the function that is to be performed

System action: The function is not performed.

Operator response: None.

Classes: None.

ING035I *system_name* **IS NOT A VALID SYSTEM
FOR GROUP** *group_name*

Explanation: The specified system name is either unknown or not a valid choice for the group.

The variable *system_name* is the invalid system name.

The variable *group_name* is the move group concerned.

System action: None.

Operator response: Specify a valid system that the group can be moved to.

Module: INGRYMV0

Classes: None.

ING036I **GROUP** *group_name* **CANNOT BE
MOVED TO SYSTEM** *system_name* -
reason

Explanation: You attempted to move the group to the specified system either now or at the next recycle, but one of the following conditions is true:

- The system is down.

- The member on that system is not startable.
- The group is already running on that system.

System action: None.

Operator response: Specify another system that the group should be moved to.

Module: INGRYMV0

Classes: None.

ING037I **GROUP** *group_name* **CANNOT BE
BOXED - NOT ACTIVE**

Explanation: You attempted to box the group but the application group is not active.

System action: None.

Operator response: None.

Module: INGRYMV0

Classes: None.

ING038I **CONFLICTING PARAMETERS OR
VALUES SPECIFIED =>** *p1* - *p2*

Explanation: The specified parameters (or their values) conflict.

The variable *p1* is the first invalid parameter.

The variable *p2* is the second invalid parameter.

System action: None.

Operator response: Reenter the command with valid parameters.

Module: INGRYST0

Classes: None.

ING039I **SESSION** *name* **IS UNKNOWN**

Explanation: The specified session is not defined in the configuration file (ACF).

The variable *name* is the session name.

System action: None.

Operator response: Reenter the command with a valid session name.

Module: INGRVSS0

Classes: None.

ING040I **THE FOLLOWING IS MISSING:**
keywords

Explanation: A command was entered but one or more required parameters were not specified.

The variable *keywords* shows a list of all the missing parameters.

System action: The command is rejected.

Operator response: Enter the command again and specify the required parameters.

Classes: None.

ING041I ONE OF THE FOLLOWING IS MISSING: *keywords*

Explanation: A command was entered but a required parameter was not specified.

The variable *keywords* shows a list of possible parameters to be specified.

System action: The command is rejected.

Operator response: Enter the command again and specify one of the suggested parameters.

Classes: None.

ING042I *keyword1* MUST BE SPECIFIED WITH *keyword2*

Explanation: A command was entered but one of its parameters requires another one to be also specified.

The variable *keyword1* shows the parameter that requires *keyword2* to be specified.

The variable *keyword2* shows the additional parameter that is required.

System action: The command is rejected.

Operator response: Enter the command again and specify the missing parameter.

Classes: None.

ING043I *keyword1* MUST NOT BE SPECIFIED WITH *keyword2*

Explanation: A command was entered but one of its parameters disallows another one to be also specified.

The variable *keyword1* shows the parameter that excludes the use of *keyword2*.

The variable *keyword2* shows the parameter that is excluded by the use of *keyword1*.

System action: The command is rejected.

Operator response: Enter the command again and specify either *keyword1* or *keyword2*.

Classes: None.

ING044I *keyword1* MUST BE GREATER THAN OR EQUAL TO *keyword2*

Explanation: A command was entered with two parameters. The two values define a range. The second value must be greater than or equal to the first value.

The variable *keyword1* shows the parameter that defines the end of the range.

The variable *keyword2* shows the parameter that defines the start of the range.

System action: The command is rejected.

Operator response: Enter the command again and specify a value for *keyword1* that is greater than or equal to the value given for *keyword2*.

Classes: None.

ING045I *keyword* MUST BE IN THE FORMAT *format*

Explanation: A command was entered with a parameter that requires a certain data format.

The variable *keyword* shows the parameter that requires the format.

The variable *format* shows the format that is required as one of the following, or any other self-explanatory text:

MM/DD/YYYY	A valid date in the form month/day/year
HH:MM	A valid time in the form hour:minute
Cn	1–n characters
Dn	1–n numeric digits

Note that there might be additional lines that explain the problem in further detail.

System action: The command is rejected.

Operator response: Enter the command again and specify a value that conforms to the required format.

Classes: None.

ING046I *keyword* MUST BE ONE OF THE FOLLOWING: *list*

Explanation: A command was entered with a parameter that requires a certain value.

The variable *keyword* shows the parameter that requires the value.

The variable *list* shows a list of valid values that the parameter can take.

System action: The command is rejected.

Operator response: Enter the command again and specify a value from the list.

Classes: None.

ING047I *keyword* MUST BE IN THE RANGE FROM *from* TO *to*

Explanation: A command was entered with a parameter that must be in a certain range.

The variable *keyword* shows the name of the parameter that must be in the range.

The variable *from* shows the lowest possible value.

The variable *to* shows the highest possible value.

System action: The command is rejected.

Operator response: Enter the command again and specify a value in the given range.

Classes: None.

ING048I *subsystem* NOT DEFINED AS *type*
SUBSYSTEM

Explanation: The specified subsystem is not of the required type.

The variable *subsystem* identifies the subsystem.

The variable *type* defines the type (CICS, IMS, etc.).

System action: None.

Operator response: Specify a valid subsystem of the required type.

Classes: None.

ING049I No *xxxx* data to display for *subsystem*

Explanation: There are no commands specified in the policy for the specified subsystem.

The variable *xxxx* identifies the keyword that no commands have been defined for. It can be DISPINFO, IMSINFO, or CICSINFO.

The variable *subsystem* identifies the subsystem.

System action: None.

Operator response: None.

Classes: None.

ING050I HBEAT BEAT *parameter*

Explanation: A heartbeat was issued.

The variable *parameter* shows the parameters that were specified with the heartbeat.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING051I HBEAT RECV *parameter*

Explanation: A heartbeat was received.

The variable *parameter* shows the parameters that were specified.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING052I HBEAT FAIL *parameter*

Explanation: A heartbeat with the specified parameters failed.

The variable *parameter* shows the parameters of the failed heartbeat.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING053I HBEAT SDWN *parameter*

Explanation: A heartbeat with the specified parameters is shut down.

The variable *parameter* shows the parameters of the shutdown heartbeat.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING054I MAJOR RESOURCE *object name*
CANNOT BE FOUND

Explanation: An invalid major resource was specified for the creation of a minor resource.

The variable *object name* shows the name of the resource that could not be found.

System action: Processing continues unless no replacement anchor can be found.

Operator response: Inform your system programmer.

System programmer response: If the originator is a customized INGPOST call, correct the call. If not, contact your IBM Support Center.

ING055I INVALID 'STATE=*state*' CONVERTED
TO 'STATE=ALERT'

Explanation: An invalid state was specified in the INGPOST call. The state is converted to 'ALERT'.

The variable *state* shows the incorrect state that was converted to 'ALERT'.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: If the originator is a customized INGPOST call, correct the call. If not, contact your IBM Support Center.

ING057I *command* FAILED, GENALERT TEXT
PARAMETER LONGER THAN 244
CHARS

Explanation: The resulting GENALERT parameter string exceeds the maximum length for GENALERTs.

The variable *command* shows the command that could not be executed.

System action: The command is canceled.

Operator response: Inform your system programmer.

System programmer response: Contact your IBM Support Center.

ING058I *module function* UNEXPECTED
 RESPONSE FROM *service function* :
 return code

Explanation: The service called for the specified function returned an unexpected return code.

The variable *module* shows the name of the module where the error occurred.

The variable *function* shows the name of function that called the service.

The variable *service* shows the service that was called.

The variable *function* shows the function of the service that was called.

The variable *return code* shows the value of the return code that was issued.

System action: The command is canceled.

Operator response: Inform your system programmer.

System programmer response: Contact your IBM Support Center.

ING059I USERSTATUS FOR OBJECT *object*
 UPDATED TO *status value (status name)*

Explanation: The status of the object has been set to the referred value.

The variable *object* shows the object whose status was updated.

The variable *status value* shows the status value of the object.

The variable *status name* shows the status name of the object.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING060I USERDATA FOR OBJECT *object*
 UPDATED: *value*

Explanation: The user data of the object has been updated to the specified value.

The variable *object* shows the object whose status was updated.

The variable *value* shows the new status of the object.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING061I INGPOST *command* NOT ALLOWED
 FOR OBJECT *object*

Explanation: The specified INPOST command is not allowed for the object.

The variable *command* shows the name of the INGPOST command that is not allowed for the object.

The variable *object* shows the name of the resource that the command is not valid for.

System action: The command is canceled.

Operator response: Inform your system programmer.

System programmer response: If the originator is a customized INGPOST call, correct the call. If not, contact your IBM Support Center.

ING062I *command mode* PROCESSING STARTED

Explanation: The command with the specified mode has started.

The variable *command* shows the name of the command that processing started for.

The variable *mode* shows the mode of the command.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING063I *command mode* PROCESSING
 COMPLETED RC *return code*

Explanation: The command with the specified mode completed.

The variable *command* shows the name of the command that processing completed for.

The variable *mode* shows the mode of the command. The variable *return code* shows the value of the return code that was issued. A non-zero return code indicates that the command was in error.

System action: Processing continues.

Operator response: In case of a non-zero return code inform your system programmer.

System programmer response: In case of a non-zero return code investigate for proceeding.

ING064I ISSUING: *command*

Explanation: The command is issued by the command router.

The variable *command* shows the name of the command that is being issued.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING065I **INVOKED FROM NMC BY USER:** *user ID*

Explanation: The command was invoked by a user.
The variable *user ID* shows the user ID that the command was invoked from.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING066I **COMMAND NOT ISSUED:** *reason*

Explanation: The command could not be executed.
The variable *reason* shows the reason why the command could not be executed.

System action: The command is canceled.

Operator response: Make sure the specified command was valid within the context. If it was valid, inform your system programmer.

System programmer response: Investigate the cause of the failure and take the appropriate steps.

ING067I **REQUEST REJECTED; INGTOPO NOT INITIALIZED**

Explanation: The request was rejected because INGTOPO was not initialized.

System action: The command is canceled.

Operator response: Start INGTOPO with "init" and the appropriate NetView operator ID as parameter.

System programmer response: None.

ING068I **CHANGES WILL BE LOST - PRESS**
function_key **TO ACTIVATE**

Explanation: At least one group policy attribute has been modified but not yet committed.

The variable *function_key* is the function key that activates the group policy change.

System action: None.

Operator response: Press the appropriate function key to commit the changes.

System programmer response: None.

ING069E **ERROR LOADING** *topo file - reason*

Explanation: The interpretation of the topo file could not be processed successfully.

The variable *topo file* shows the name of the topo file that could not be loaded.

The variable *reason* shows the reason why the topo file could not be loaded.

System action: Processing stops.

Operator response: Inform your system programmer.

System programmer response: Check the topo file for the specified failure.

ING070E **UNEXPECTED RESPONSE FROM**
system RC return code

Explanation: The processing of a remote command on the specified system returned an unexpected return code.

The variable *system* shows the name of the system the returned the return code.

The variable *return code* shows the value of the return code.

System action: Processing continues, unless it was the last target system in the sysplex to make contact with.

Operator response: Check the final message ING063I and act according to its state.

System programmer response: None.

ING071I *command mode* **SUBMITTED**

Explanation: The command with the specified mode was submitted to the Topology Manager NetView operator ID.

The variable *command* shows the name of the submitted command.

The variable *mode* shows the mode of the submitted command.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING072E **INITIAL COMMUNICATION WITH**
system **FAILED**

Explanation: The attempt to communicate with the specified system was unsuccessful.

The variable *system* shows the name of the system.

System action: Processing continues.

Operator response: Inform your system programmer.

System programmer response: Investigate the cause of the failure and take the appropriate steps.

ING073E **"INIT" FAILED: MSM HAS NOT BEEN STARTED YET**

Explanation: The Tivoli NetView MultiSystem Manager (MSM) has not been started yet.

System action: Initialization of SA z/OS Topology Manager canceled.

ING080I • ING084I

Operator response: Inform your system programmer.

System programmer response: Start Tivoli NetView MultiSystem Manager (MSM). The MSM is a prerequisite for the SA z/OS Topology Manager.

ING080I *Monitor_name Session_name Logon_type
Exception_text*

Explanation: ING080I is issued on behalf of a monitor resource that uses the monitor command INGMTRAP to define a trap for one or more OMEGAMON[®] exceptions. It is issued for each exception that trips.

Monitor_name is the fully qualified monitor resource name in the format *monitor_name/MTR/system_name*, which indicates the monitor resource that this exception message was issued for. Note that *monitor_name* is known to INGMTRAP through the task global &SUBSAPPL.

Session_name is the name of the OMEGAMON session as it was specified by the NAME keyword on the INGMTRAP command.

Logon_type is the type attribute used by the OMEGAMON session named *Session_name* as defined in the automation policy.

Exception_text is the exception text as returned by OMEGAMON without the leading '+' sign.

ING081I *Monitor_name Session_name Logon_type
NO EXCEPTION FOUND*

Explanation: Message ING081I is never issued to an operator console, but is written to a monitor resource's history. This happens when the monitor resource uses the monitor command INGMTRAP to define a trap for one or more OMEGAMON exceptions and no exception has tripped.

Monitor_name is the fully qualified monitor name in the format *monitor_name/MTR/system_name*, which indicates the monitor resource that this exception message was issued for. Note that *monitor_name* is known to INGMTRAP through the task global &SUBSAPPL.

Session_name is the name of the OMEGAMON session as it was specified by the NAME keyword on the INGMTRAP command.

Logon_type is the type attribute used by the OMEGAMON session named *Session_name* as defined in the automation policy.

ING082I **NO SESSION OPERATORS HAVE
BEEN DEFINED – OMEGAMON
SESSION INITIALIZATION
TERMINATED.**

Explanation: The installation failed to define session operator entries in the automation policy.

System action: Automation initialization continues.

Operator response: Notify your system programmer.

System programmer response: Make sure you have defined at least one session operator AOFSESxx in the automation policy and assigned it to an automation operator, for example, AUTSESxx.

ING083I **COMMAND *command* ISSUED BY
operator IN SESSION *session_name*.**

Explanation: Message ING083 is issued each time a list of one or more commands is sent to the OMEGAMON session that is denoted by *session_name*.

command is the first command in the list of commands that was issued with the INGOMX API. *operator* is the operator or automation operator that issued the command.

session_name is the name of the OMEGAMON session that is to receive the command.

System action: None.

Operator response: None.

System programmer response: None.

ING084I **REQUEST FOR OMEGAMON
SESSION *session_name* FAILED.
REASON CODE *rsn***

Explanation: SA z/OS failed to communicate with the OMEGAMON session denoted by *session_name*. The variable *rsn* contains more specific error information.

System action: The request is not executed.

Operator response: None.

System programmer response: The reason code, *rsn*, provides details about the cause of the failure. It is in hexadecimal encoding and it can be any of the following:

00000001 (Authentication unsuccessful)

The userid defined for this session is not permitted to log on to OMEGAMON. Possible reasons are:

- The user ID is invalid.
- The password is invalid.
- SAFPW was specified as the password, but no password data set was specified in the NetView startup procedure.
- SAFPW was specified as the password, but the password found in the password data set is either invalid, does not exist, or was fetched using an invalid owner. See *IBM Tivoli System Automation for z/OS Planning and Installation* for information about how to set up your security environment to use this function.

00000002 (Not an OMEGAMON session)

The session that was established between SA z/OS and the VTAM APPLID defined for

this session is not an OMEGAMON session. Validate and, if necessary, correct your session definitions in the customization dialog.

00000003 (VTAM not available)

VTAM was not active at the time the INGOMX API was invoked.

00000004 (Timeout occurred)

The request was interrupted due to a timeout that has been specified for this session. Either the timeout is too aggressive or too few session operators are defined to handle all assigned sessions.

00000005 (Password update failed)

The password of the user defined for this session is controlled by the system using NetView. GETPW failed to update the password in the NetView password data set after OMEGAMON authentication with the new password.

Reset the session user's OMEGAMON password and re-initialize the system-controlled password in the NetView password data set.

00000006 (TAF fullscreen session not established)

NetView could not create a Terminal Access Facility (TAF) fullscreen session using the BGNSESS command. Refer to the netlog for additional messages issued by NetView.

0001nnnn (ATTACH of VOST failed with RC=nnnn)

SA z/OS failed to attach a virtual terminal. Refer to the NetView ATTACH command description for details of the return code represented by *nnnn* in bytes 2 and 3 of the reason code.

0002nnnn (DETACH of VOST failed with RC=nnnn)

SA z/OS failed to detach a virtual terminal. Refer to the NetView DETACH command description for details of the return code represented by *nnnn* in bytes 2 and 3 of the reason code.

0003nnnn (Internal error detected)

An internal error *nnnn* occurred within SA z/OS. Report this problem to your IBM Support Center.

ING085I OMEGAMON SESSION *session_name* {STARTED | STOPPED}. NEW STATE IS *status*.

Explanation: Message ING085I is issued each time SA z/OS established or destroyed an OMEGAMON session, denoted by *session_name*.

The action can be either of the following:

STARTED

The session was started.

STOPPED

The session was stopped.

The current status is indicated in variable *status*.

System action: None.

Operator response: None.

System programmer response: None.

ING087I *task_name*: PPI RECEIVER *receiver_name* READY

Explanation: The task *task_name* has initialized completely and the PPI receiver named *receiver_name* is ready for processing requests.

System action: Processing continues

Operator response: None

System programmer response: None

ING088I *task_name*: PPI RECEIVER *receiver_name* TERMINATED

Explanation: The task *task_name* has deleted the PPI receiver named *receiver_name*. No more requests can be processed.

System action: Processing terminates

Operator response: None

System programmer response: None

ING089I *task_name*: INVALID BUFFER RECEIVED, DATA DROPPED. RELATED INFO: *request_data*

Explanation: The PPI receiver owned by the task *task_name* received an invalid request buffer. The first 64 bytes are dumped with the message in the variable *request_data*.

System action: Processing continues

Operator response: None

System programmer response: Ensure that only the SA z/OS monitoring agent is sending requests to the PPI receiver that is owned by the task *task_name*.

ING090I TASK *task_name* IS NOW ACTIVE

Explanation: The task *task_name* has been activated.

System action: Processing continues.

Operator response: None.

System programmer response: None.

ING091I *task_name* ALREADY ACTIVE OR IN
PROCESS OF BECOMING ACTIVE

Explanation: You started the task *task_name* but the task is already active or in the process of becoming active.

System action: Processing terminates.

Operator response: None.

System programmer response: None.

ING092I *task_name: service_name* SERVICE
FAILED, RC=*rc*. ERROR
INFORMATION: *error_info*

Explanation: During the execution of the service *service_name*, an error occurred. The variable *error_info* indicates in what context the service was invoked by the task *task_name*.

For detailed error information, see *NetView: Customization - Using Assembler*. In the case of NetView Program-to-Program Interface (PPI) errors see *NetView: Application Programming Guide* for more information related to the service return code *rc*.

System action: Processing terminates.

Operator response: None.

System programmer response: Determine why the service routine failed and correct the problem. Examine the netlog for additional information. If necessary, contact your local IBM Support Center for further assistance.

ING093I *task_name: INVALID KEYWORD OR SYNTAX ERROR. RELATED INFORMATION: error_info*

Explanation: An invalid keyword or value was specified in the initialization member for the task *task_name*. Related error information is provided in the variable *error_info*.

System action: Processing terminates.

Operator response: None.

System programmer response: See "Step 2. Configure SA z/OS and NetView" in the chapter "Beginning the installation and configuration" in *IBM Tivoli System Automation for z/OS Monitoring Agent Configuration and User's Guide* for a list of valid keywords and values and correct the problem.

ING094I TASK *task_name* HAS TERMINATED

Explanation: The task *task_name* has terminated.

System action: Processing terminates.

Operator response: None.

System programmer response: None.

ING095I *task_name: NO ACTIVE OPERATOR FOUND*

Explanation: The PPI receiver owned by task *task_name* received a request for monitoring data but could not find an active automation operator to process the request.

System action: The message is sent back to the requestor and processing continues.

Operator response: None

System programmer response: Refer to *IBM Tivoli System Automation for z/OS Monitoring Agent Configuration and User's Guide* for instructions how to set up automation operators for this function and correct the problem.

ING096I function : *text*

Explanation: During processing the function request for monitoring data any of the following errors as indicated by *text* occurred. Examples for *text* are:

- NetView RC *rc* from: *command*
- Timeout occurred
- ING008I ...

System action: The message is sent back to the requestor and processing continues.

Operator response: None

System programmer response: If *text* contains an existing SA z/OS message ID, refer to the message help for problem resolution.

In case of a timeout, increase the time allotted for the longest request.

For return codes returned by SA z/OS or NetView services refer to related documentation or contact your local IBM Support Center for further assistance.

ING101I *subsystem* CONNECTION TO *db2id*
(*connection_id*) IS DOWN.

Explanation: When monitoring a connection between a subsystem and DB2, the connection was determined to be in the status not connected.

The variable *subsystem* shows the subsystem name as defined in SA z/OS.

The variable *db2id* shows the DB2 subsystem ID.

The variable *connection_id* identifies the affected connection.

System action: None.

Operator response: Check whether the subsystem is not connected to DB2 as a result of normal operation. If it is not, investigate why the connection is not up.

System programmer response: None.

ING102I *subsystem* CONNECTION TO *db2id*
(*connection_id*) IS UP.

Explanation: When monitoring a connection between a subsystem and DB2, the connection was determined to be in the status connected.

The variable *subsystem* shows the subsystem name as defined in SA z/OS.

The variable *db2id* shows the DB2 subsystem ID.

The variable *connection_id* identifies the affected connection.

System action: None.

Operator response: None.

System programmer response: None.

ING103I THREADS COULD NOT BE
TERMINATED FROM *subsystem*

Explanation: This message introduces a list of ING104I messages that show threads that remain with a DB2 master after a thread termination process has completed.

The variable *subsystem* shows the name of the DB2 master that threads could not be terminated for.

System action: The requested function could not be performed.

Operator response: Investigate why the threads that are listed in the following ING104I messages could not be terminated and retry the termination command.

System programmer response: None.

Classes: 40, 44.

ING104I *connection_ID* THREAD *correlation_ID*
ddf_ID

Explanation: This message shows a thread that remains with a DB2 master after a thread termination process has completed. It is part of a multiline message that begins with message ING103I.

The variable *connection_ID* shows the connection ID of a thread that could not be terminated.

The variable *correlation_ID* shows the correlation ID of a thread that could not be terminated.

The variable *ddf_ID* shows the token associated to a distributed database access thread that could not be terminated.

System action: The requested function could not be performed.

Operator response: Investigate why the indicated thread could not be terminated and retry the termination command.

System programmer response: None.

Classes: None.

ING106I INDOUBT THREADS EXIST FOR *name*

Explanation: There are indoubt threads existing with the DB2 subsystem named in the message. The message is given when invoking the INGB2 command checking for indoubt threads.

Variable *name* identifies the DB2 subsystem.

Operator response: None.

System action: Processing continues.

ING108I NO THREADS LEFT IN *subsystem*.

Explanation: There are no threads left with this DB2 subsystem for the thread termination process to cancel.

The variable *subsystem* identifies the DB2 subsystem.

System action: None.

Operator response: None.

System programmer response: None.

Classes: 40, 44.

ING109E *thread_number* THREADS COULD NOT
BE TERMINATED FROM *subsystem*.

Explanation: Threads remain with this DB2 subsystem after the thread termination process has completed.

The variable *thread_number* shows the number and type of threads.

The variable *subsystem* identifies the DB2 subsystem.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Investigate why the number of indicated threads are not terminated and retry the termination command.

Classes: 40, 44.

ING112I YOUR TSO DB2 (*subsystem*) THREAD
IS ABOUT TO BE TERMINATED BY
AUTOMATION.

Explanation: This message is broadcast to effected TSO users due to a DB2 thread termination request.

The variable *subsystem* identifies the DB2 subsystem.

System action: None.

Operator response: None.

System programmer response:

Classes: 40, 43.

ING113I YOUR TSO DB2 (*subsystem*) THREAD HAS BEEN TERMINATED BY AUTOMATION.

Explanation: This message is broadcast to effected TSO users as a result of a DB2 thread termination request.

The variable *subsystem* identifies the DB2 subsystem.

System action: Cancels the TSO users of the indicated DB2 subsystem.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

ING114E *jobname* CANCELLED BY AUTOMATION DUE TO *subsystem* THREAD TERMINATION.

Explanation: This message is issued for batch job cancellations due to a DB2 thread termination or DB2 shutdown process.

The variable *jobname* identifies the DB2 batch job.

The variable *subsystem* identifies the DB2 subsystem.

System action: Cancels the batch job for the indicated DB2 subsystem.

Operator response: None.

System programmer response: None.

Classes: 40, 44.

ING127A THREADS FOUND AFTER LAST CYCLE OF DB2 (*subsystem*), FORCE SHUTDOWN.

Explanation: After the final cycle (as defined by the DBITTH timeout variable), threads were still found to exist for this DB2 subsystem.

The variable *subsystem* identifies the DB2 subsystem.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Investigate why this process has not been able to terminate existing threads. If this DB2 needs to be shut down use the FORCE option.

Classes: 40, 44.

ING129E *jobname* CANCELLED. TABLESPACE *dbname.tsname* (*subsystem*) NEEDED TO BE STOPPED

Explanation: The job was canceled due to a tablespace that needed to be stopped.

The variable *jobname* shows the job name for the job canceled.

The variable *dbname.tsname* identifies the tablespace.

The variable *subsystem* identifies the DB2 subsystem.

System action: The identified tablespace is stopped and the job is canceled.

Operator response: None.

System programmer response: None.

Classes: 40, 44.

ING130I TABLESPACE *dbname.tsname* (*subsystem*) IS TO BE STOPPED. PLEASE STOP USING IT.

Explanation: The tablespace indicated is being stopped.

The variable *dbname.tsname* identifies the tablespace.

The variable *subsystem* identifies the DB2 subsystem.

System action: The identified tablespace will be stopped.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

ING131I YOU WERE CANCELLED, TABLESPACE *dbname.tsname* (*subsystem*) IS TO BE STOPPED.

Explanation: The indicated tablespace is being stopped and operator sessions using it are canceled.

The variable *dbname.tsname* identifies the tablespace.

The variable *subsystem* identifies the DB2 subsystem.

System action: The operator session is canceled.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

ING132I *thread_number* CANCELLED DUE TO STOP OF TABLESPACE *dbname.tsname* (*subsystem*).

Explanation: The tablespace indicated is being stopped and all threads are canceled.

The variable *thread_number* shows the number and type of threads canceled.

The variable *dbname.tsname* identifies the tablespace.

The variable *subsystem* identifies the DB2 subsystem.

System action: The indicated threads are canceled.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

ING133I YOUR USER ID WAS CANCELED IN ORDER TO ENABLE DSN TERMINATION FOR *subsystem*

Explanation: All threads are being stopped by operator request. TSO users with a DSN session are canceled in order to enforce the termination of their DSN session.

The variable *subsystem* identifies the name of the DB2 subsystem.

System action: The operator session is canceled.

Operator response: None.

System programmer response: None.

Classes: 40, 43.

ING140I ALERT '*alert*' FOR '*resource*' ON '*system*' AT *timestamp*

Explanation: This message is produced by SA z/OS and sent to targets such as System Automation for Integrated Operations Management (SA IOM), Netcool/OMNIBus, or the Tivoli Enterprise Console (TEC) for event notification.

The variable *alert* shows the alert identifier that was specified when INGALERT was invoked, or the default that was used.

The variable *resource* shows the resource name that was specified when INGALERT was invoked, or the default that was used.

The variable *system* shows the system name that INGALERT was invoked on.

The variable *timestamp* shows the time when the event was originally produced. It is in the format hh:mm:ss mm/dd/yyyy.

System action: The event is passed to the intended event notification target.

Operator response: None.

System programmer response: None.

Classes: None.

ING141I SENDING OF ALERT '*alertid*' FOR '*resource*' WAS SUCCESSFUL

Explanation: An event was successfully sent to targets such as System Automation for Integrated Operations Management (SA IOM), Netcool/OMNIBus, or the Tivoli Enterprise Console (TEC) for event notification. In the case of SA IOM, it has acknowledged receipt of the alert.

The variable *alertid* shows the alert identifier that was specified when INGALERT was invoked or the default that was used.

The variable *resource* shows the resource name that was specified when INGALERT was invoked or the default that was used.

System action: The event was successfully sent to the intended event notification targets

Operator response: None.

System programmer response: None.

Classes: None.

ING142I SENDING OF ALERT '*alertid*' FOR '*resource*' IGNORED

Explanation: An event was not sent to targets such as System Automation for Integrated Operations Management (SA IOM), Netcool/OMNIBus, or the Tivoli Enterprise Console (TEC) for event notification because the configuration does not allow this. Events are ignored in the following cases:

1. Event notification is not enabled on this agent for any of the intended event notification targets.
2. The Inform List of the resource does not contain the intended event notification target.
3. Code match checking with the message ID INGALERT of the subject resource returned either no data or the value IGNORE.

The variable *alertid* shows the alert identifier that was specified when INGALERT was invoked or the default that was used.

The variable *resource* shows the resource name that was specified when INGALERT was invoked or the default that was used.

System action: The event was not sent to the intended event notification targets.

Operator response: None.

System programmer response: If the event should not be ignored, determine the cause of the problem by using the INGCNTL command to check the alert mode on this agent and checking the Inform List and the code definitions for the subject resource in the automation policy.

Classes: None.

ING143I SENDING OF ALERT '*alertid*' FOR '*resource*' FAILED

Explanation: An event was not sent to targets such as System Automation for Integrated Operations Management (SA IOM), Netcool/OMNIBus, or the Tivoli Enterprise Console (TEC) for event notification because of an error. The error is detected by the command slave. Messages that are generated by the slave (or slaves) are appended to the above message text.

The variable *alertid* shows the alert identifier that was specified when INGALERT was invoked or the default that was used.

ING144I • ING147I

The variable *resource* shows the resource name that was specified when INGALERT was invoked or the default that was used.

System action: The event was not sent to the intended event notification targets.

Operator response: None.

System programmer response: Check the additional message lines and determine the cause of the problem.

Classes: None.

ING144I CDEMATCH FOR *entry type* WITH CODES (*code1,code2,code3*) RETURNED INVALID VALUE: *value*

Explanation: During code match checking a matching code definition was found. However, the value returned by the matching code definition is invalid.

The variable *entry* shows the entry name of the definition in the MESSAGES/USER DATA policy item of the automation policy where the matching code definitions were searched for.

The variable *type* shows the message ID of the definition in the MESSAGES/USER DATA policy item of the automation policy where the matching code definitions were searched for.

The variables *code1*, *code2*, and *code3* show the values that were used to search for a matching code definition in the MESSAGES/USER DATA policy item of the specified entry and type.

The variable *value* shows the value that is returned from the matching code definition.

System action: Further processing depends on the context. Processing usually stops.

Operator response: Check for more information.

System programmer response: Check the configuration and correct the problem.

Classes: None.

ING145I *type* INFORMATION IS INCOMPLETE

Explanation: An event is to be sent however the information about the event notification target is incomplete.

The variable *type* shows the event notification target that could not be notified. It is one of the following:

IOM System Automation for Integrated Operations Management (SA IOM) could not be reached. The information specified with the ALERTHOST parameter of the INGCNTL command is incomplete.

EIF An EIF event could not be generated. The EAS PPI receiver name to be specified with the EIFPPI parameter of the INGCNTL command is missing.

TTT A trouble ticket could not be generated because the address data for the TDI server or the data set characteristics for the trouble ticket detail data are missing or incomplete. This data should be specified with the TTTHOST and TTDDATA parameters of the INGCNTL command.

USR A user event could not be generated. The name of the automation procedure that is used to handle user events is missing. The procedure name should be specified with the USRHANDLER parameter of the INGCNTL command.

System action: The event is not sent to the indicated event notification target. The request is rerouted to the next available agent in the sysplex (if any).

Operator response: None.

System programmer response: Check the configuration and correct the problem.

Classes: None.

ING146I ALERTING IS DISABLED

Explanation: An alert is to be sent to targets such as System Automation for Integrated Operations Management (SA IOM), Netcool/OMNibus, or the Tivoli Enterprise Console (TEC) for event notification. However event notification is not enabled on this agent for any supported event notification target.

System action: The event is not sent to the indicated event notification target.

Operator response: Turn on event notification for the intended target with the INGCNTL command if desired.

System programmer response: None.

Classes: None.

ING147I ALERTING IS NOT SUPPORTED FOR RESOURCE TYPE '*type*'

Explanation: An event is to be sent to an event notification target such as IOM, EIF, TTT or USR. However, the resource type specified is not supported for event notification.

The variable *type* shows the type of the resource that was specified, which is not valid for event notification.

System action: The event is not sent to the intended event notification target.

Operator response: Correct the problem and reissue the command.

System programmer response: None.

Classes: None.

ING148I *function : request* **REQUEST TIMED OUT**

Explanation: A request timed out.

The variable *function* shows the name of the function that detected the time out.

The variable *request* shows the request that timed out.

System action: None.

Operator response: Check other messages for more information.

System programmer response: None.

Classes: None.

ING149I **LIST CONTROL SETTINGS**

Explanation: This message provides header information for the INGCNTL LIST command.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

ING150I *timestamp : object severity jobname DATA= data*

Explanation: ING150I represents an event that was injected into the system by an external or internal monitor. Typically INGMON is called from the NetView automation table as an action for this event. INGMON will find the corresponding monitor resources and trigger automation there if applicable.

The variable *timestamp* is the time that the event was originally produced or when the message was created. It is in the format hh:mm:ss mm/dd/yyyy.

The variable *object* specifies the monitored object that this event is related to.

The variable *severity* specifies the severity of the status of the monitored object that led to this event.

The variable *jobname* specifies the job name that the monitored object is related to or 'N/A' if there is none.

The variable *data* can be any data the monitor provides that might be useful for automation or recovery actions.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

ING151I **ATTACH OF** *name* **SUCCESSFUL**

Explanation: The ATTACH of a virtual operator station task (VOST) was successful.

The variable *name* specifies the attach name of the VOST. Note that this name is the content of TGLOBAL SUBSJOB, that is, the job name of the VOST management APL.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

ING152I **ATTACH OF** *name* **FAILED WITH RC=rc**

Explanation: The ATTACH of a virtual operator station task (VOST) failed with a return code greater than 0.

The variable *name* specifies the attach name of the VOST. Note that this name is the content of TGLOBAL SUBSJOB, that is, the job name of the VOST management APL.

The variable *rc* shows the return code given by the ATTACH command.

System action: The VOST is not started.

Operator response: Contact your system programmer.

System programmer response: Review the netlog for additional messages issued by the ATTACH command and correct the problem.

Classes: None.

ING153I *command* **OF** *name* **SUCCESSFUL**

Explanation: The stopping of a virtual operator station task (VOST) was initiated successfully. Note that the VOST may still be active for a certain time after this message is issued.

The variable *command* shows the command used to stop the VOST.

The variable *name* specifies the attach name of the VOST. Note that this name is the content of TGLOBAL SUBSJOB, that is, the job name of the VOST management APL.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Classes: None.

ING154I • ING160I

ING154I *command* OF *name* FAILED WITH RC=*rc*

Explanation: The stopping of a virtual operator station task (VOST) failed with a return code greater than 0.

The variable *command* shows the command used to stop the VOST.

The variable *name* specifies the attach name of the VOST. Note that this name is the content of TGLOBAL SUBSJOB, that is, the job name of the VOST management APL.

The variable *rc* shows the return code given by the command.

System action: The VOST is not stopped.

Operator response: Contact your system programmer.

System programmer response: Review the netlog for additional messages issued by the specified command and correct the problem.

Classes: None.

ING155I ENVIRONMENT CHECK FAILED FOR *command*. REASON=*rs*

Explanation: A command was issued in an unsupported environment.

The variable *command* shows the command that was executed.

The variable *rs* shows the reason code that can have the following values:

- 1 Required task globals not set
- 2 Running on an unsupported task
- 3 Configuration not fully loaded
- 4 Main include member could not be read
- 5 ACFLEVEL could not be determined
- 6 ACFLEVEL mismatch

System action: The command stops processing.

Operator response: Contact your system programmer.

System programmer response: See the documentation of the command that issued the message and correct the problem.

Classes: None.

ING156I *name* ENDED WITH RC=*rc*

Explanation: The command executed in a virtual operator station task (VOST) ended.

The variable *name* specifies the attach name of the VOST.

The variable *rc* shows the return code of the command that ran in the VOST.

System action: The VOST is stopped.

Operator response: None.

System programmer response: None.

Classes: None.

ING157I INVALID VALUE FOR *name*: *value*

Explanation: Data was found that is not valid.

The variable *name* specifies the context.

The variable *value* shows the invalid data.

System action: Processing stops.

Operator response: Contact your system programmer.

System programmer response: See the documentation of the command that issued the message and correct the problem.

Classes: None.

ING158I SENDING OF ALERT *alertid* FOR *resource* PARTIALLY FAILED

Explanation: An event was only partially sent to the designated targets.

Variable *alertid* identifies the alert identifier that was specified when INGALERT was invoked or the default was used.

Variable *resource* identifies the resource.

Operator response: None.

System action: Processing continues.

ING159I EXIT *exitname* RETURNED UNEXPECTED RESULTS. REASON *reason*

Explanation: An exit was called but returned either an invalid return code or invalid data.

The variable *exitname* shows the name of the exit that was called.

The variable *reason* shows why the exit is considered to be in error:

- 01 Unexpected return code
- 02 Invalid number of data lines returned
- 03 Invalid data found in the returned lines

System action: Further processing depends on the context. Processing usually stops.

Operator response: Check for more information.

System programmer response: Check the configuration and correct the problem.

Classes: None.

ING160I RESPONSE FROM SOAP SERVER:
server_name
attribute_name_1 sep attribute_name_2 sep
... attribute_name_n sep
attribute_value_1 sep attribute_value_2 sep
... attribute_value_n sep

Explanation: This message displays the names and

values of the attributes that have been requested by an ITM SOAP request. The attribute names and values are separated by the separator character, *sep*, of value X'FF'.

The first line of this multiline message denotes the SOAP server, *server_name*, that provided this response.

The second line contains the attribute names separated by the separator character. All the following lines contain the attribute values separated by the separator character.

This message is written to the console only.

System action: None.

Operator response: None.

System programmer response: None.

**ING161I SENDING SOAP REQUEST TO
SERVER: *server_name***

Explanation: A SOAP request is being sent to the SOAP server denoted by *server_name*. This message is written to the netlog only.

System action: None.

Operator response: None.

System programmer response: None.

**ING162I SOAP FAULT CODE *code*. REASON:
*reason***

Explanation: The SOAP request failed with a fault code of *code*. Possible codes are Client, VersionMismatch, MustUnderstand, and Server. The Client fault code indicates that the SOAP message was incorrectly formed.

The *reason* text indicates the particular fault.

This message is written to the console only.

System action: None.

Operator response: Contact your system programmer.

System programmer response: For a Client fault code, correct the SOAP message and try again. For any other fault code, contact your IBM support representative.

ING163I FAULT DETAIL: *text*

Explanation: This message provides further detail in variable *text* about the fault reported in the preceding ING162I message. This message spans multiple lines if further fault details are reported in the SOAP response.

This message is written to the console only.

System action: None.

Operator response: Contact your system programmer.

System programmer response: For a Client fault code

(see ING162I), correct the SOAP message and try again. For any other fault code, contact your IBM support representative.

**ING164I SOAP REQUEST FAILED, REASON:
reason SERVER: *server_name***

Explanation: An error occurred while attempting to process a SOAP request on the SOAP server denoted by *server_name*. The actual error reason is encoded in the separate *reason* variable.

This message is written to the console only.

The following reasons are defined:

- 00000004** A timeout occurred and the connection was terminated.
- 0001nnnn** INGOMX failed to connect to, send to, or receive from a socket created to communicate with the SOAP server denoted by *server_name*. The socket error is encoded in bytes 2 and 3 represented by variable *nnnn*.
- 0002nnnn** The output returned from the IBM Tivoli Monitoring SOAP server could not be handled by the XML parser. The XML parser reason code is encoded in bytes 2 and 3 represented by variable *nnnn*.
- 0003nnnn** An internal error occurred. The variable *nnnn* represents the internal error code.
- 0004xyyy** An HTTP error occurred while processing the request. If *x=0*, *yyy* denotes the 3-digit HTTP response code, for example, 404. If *x=1*, the variable *yyy* represents the internal error code.
- 0005nnnn** An error occurred while allocating or reading the SOAP request data set. The pipe QSAM return code is encoded in bytes 2 and 3 represented by variable *nnnn*.
- 0006nnnn** An error occurred while processing a CORR CMD pipe stage. The return code is encoded in bytes 2 and 3 represented by variable *nnnn*.

System action: None.

Operator response: Contact your system programmer.

System programmer response: For reason code 00000004, increase the timeout value. The parameter default variable INGOMX_WAIT can be set to increase the timeout.

For reason codes 0001nnnn and 0004xyyy, ensure you have connectivity to the SOAP server and verify the port and path of the SOAP service.

For reason code 0005nnnn, specify the correct data set name or fix the I/O error.

In any other case, contact your IBM support representative.

ING165I *protocol status-code reason-phrase*

Explanation: This message reflects the status line of a preceding HTTP request in case of a non-zero status code.

The variable *protocol* reports the HTTP protocol used by the server.

The variable *status-code* reports the status of the request.

The variable *reason-phrase* contains human readable information explaining the *status-code*.

This message is written to the netlog only.

System action: None.

Operator response: Contact your system programmer.

System programmer response: For status codes 1xx and 2xx, no action is required.

For status codes 3xx, check out message ING167I for the new location and try again.

For status codes 4xx, check out the details reported by message ING166I and fix the problem.

For status codes 5xx, collect the details reported by message ING166I and contact your IBM support representative.

ING166I **HTTP-RESPONSE INFORMATION:** *text*

Explanation: This message reports additional information about the HTTP response, where the variable *text* contains general header fields, response header fields, and entity header fields.

This message is written to the netlog only.

System action: None.

Operator response: Contact your system programmer.

System programmer response: See message ING165I about how to use this information depending on the status code returned by the server.

ING167I **NEW LOCATION:** *location*

Explanation: This message is issued as the result of a 3xx status code received as the response of the HTTP SOAP request. The request URI was either temporarily or permanently moved to a new location or is only accessible through the proxy server at the reported location. The request must be redirected to the new location reported by the variable *location*.

This message is written to the console only.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Correct the server information in the customization dialog to use the reported location and then try the request again.

| **ING170I** **THE FOLLOWING ENTRY-TYPES**
| **HAVE BEEN DELETED:** *entry-type*

| **Explanation:** This message is issued by the
| INGCLEAN utility when entry-types have been
| deleted. The deleted entry-types are appended to the
| above message text. Note that the entry-types for which
| a default has been restored are not included in this list.

| The variable *entry-type* displays any entry-types that
| have been deleted.

| **System action:** Processing was successful.

| **Operator response:** None.

| **System programmer response:** None.

| **Classes:** None.

ING200W **SA event with NoStatusChange reason**
in domain *domain*

Explanation: This message is created by the SA z/OS internal notification function to send current status information to the end-to-end automation adapter.

System action: None.

Operator response: None.

System programmer response: None.

ING201W **SA detected a status change of resource**
resource in domain *domain*. **Common**
observed state: *observed*, **Common**
desired state: *desired*, **Common**
operational state: *operational*, **Common**
compound state: *compound*

Explanation: This message is created by the SA z/OS internal event notification function when the status of a resource changes. It shows the resource's new status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING202I SA turned the resource *resource* request as requested in domain *domain*. **Common observed state:** *observed*, **Common desired state:** *desired*, **Common operational state:** *operational*, **Common compound state:** *compound*

Explanation: This message is created by the SA z/OS internal event notification function when a request has been issued to start or stop a resource. It shows the resource's current status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *request* indicates if an Online or Offline request has been issued.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING203I SA detected that the system *system* joined the domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the observed status of a system resource has changed from Sysgone.

The variable *system* shows the name of the system.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING204W SA detected that the system *system* left the domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the observed status of a system resource has become Sysgone.

The variable *system* shows the name of the system.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING205I SA turned off the automation for resource *resource* in domain *domain*. **Common observed state:** *observed*, **Common desired state:** *desired*, **Common operational state:** *operational*, **Common compound state:** *compound*

Explanation: This message is created by the SA z/OS internal event notification function when a request has been issued to switch on or off the automation flag of a resource. It shows the resource's current status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING206W SA detected a warning in domain *domain*. The resource *resource* has been degraded. **Common observed state:** *observed*, **Common desired state:** *desired*, **Common operational state:** *operational*, **Common compound state:** *compound*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the observed status of the resource has become degraded. It shows the resource's current status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING207W SA detected a warning in domain *domain*. The resource *resource* has a performance problem. Common observed state: *observed*, Common desired state: *desired*, Common operational state: *operational*, Common compound state: *compound*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the health status of the resource has become degraded. It shows the resource's current status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING208W SA detected a warning in domain *domain*. The resource *resource* has problems caused by a dependency to another resource. Common observed state: *observed*, Common desired state: *desired*, Common operational state: *operational*, Common compound state: *compound*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the startable status of the resource has become inhibited. It shows the resource's current status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING209W SA detected a problem in domain *domain*. The resource *resource* can't be turned into the requested state. Common observed state: *observed*, Common desired state: *desired*, Common operational state: *operational*, Common compound state: *compound*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports a problem in the automation state or the observed state. It shows the resource's current status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING210W SA detected a problem in domain *domain*. The resource *resource* needs a manual reset by the operator. Common observed state: *observed*, Common desired state: *desired*, Common operational state: *operational*, Common compound state: *compound*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the observed status of

the resource has changed to harddown. It shows the resource's current status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING211W SA detected a problem in domain *domain*. The resource *resource* has problems caused by an offline system. Common observed state: *observed*, Common desired state: *desired*, Common operational state: *operational*, Common compound state: *compound*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the observed status of the resource has changed to sysgone. It shows the resource's current status in the end-to-end automation resource data model.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *observed* shows the observed state of the resource.

The variable *desired* shows the desired state of the resource.

The variable *operational* shows the operational state of the resource.

The variable *compound* shows the compound state of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING212W SA event with unknown reason in domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager sends an event with an unknown reason.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING213I SA detected that system automation is ready in domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that the status of the automation domain is Online.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING214W SA detected that system automation is broken in domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that the status of the automation domain is Offline.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING215I SA detected a configuration policy refresh in domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that the automation policy has been refreshed in the automation domain.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING216I SA detected a new configuration in domain *domain*. New resource *resource* created.

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that a new resource has been created in the automation domain.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING217I SA detected a deleted configuration in domain *domain*. Resource *resource* deleted.

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that a resource has been deleted in the automation domain.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING218I SA detected a new configuration in domain *domain*. Request action *request* issued for resource *resource*

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that a new request has been issued for the resource in the automation domain.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *resource* shows the name of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING219I SA detected a deleted configuration in domain *domain*. Request action *request* deleted for resource *resource*

Explanation: This message is created by the SA z/OS internal event notification function to inform the

end-to-end automation adapter that a request for the resource in the automation domain has been deleted.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *resource* shows the name of the resource.

System action: None.

Operator response: None.

System programmer response: None.

ING220I SA detected a new configuration in domain *domain*. New relationship *relationship* with source resource *source* and target resource *target* created.

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that a new relationship has been created between the source resource and the target resource in the automation domain.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *relationship* shows the name of the relationship.

The variable *source* shows the name of the dependent resource.

The variable *target* shows the name of the supporting resource.

System action: None.

Operator response: None.

System programmer response: None.

ING221I SA detected a deleted configuration in domain *domain*. Relationship *relationship* with source resource *source* and target resource *target* deleted.

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that a relationship between the source resource and the target resource in the automation domain has been deleted.

The variable *domain* shows the name of the end-to-end automation domain.

The variable *relationship* shows the name of the relationship.

The variable *source* shows the name of the dependent resource.

The variable *target* shows the name of the supporting resource.

System action: None.

Operator response: None.

System programmer response: None.

ING222I SA detected an automation status change in domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function to inform the end-to-end automation adapter that properties have been changed in the automation domain.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING223W SA detected that the system *system* switched into maintenance mode by operator command in domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the desired status of a system resource has changed to Available.

The variable *system* shows the name of the system.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING224I SA detected that the system *system* switched back to automation mode by an operator command in domain *domain*

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the desired status of a system resource has changed to Unavailable.

The variable *system* shows the name of the system.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING225W SA suspended automation for resource *resource* in domain *domain* as requested.

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the automation flag of the resource has been switched off.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING226I SA resumed automation for resource *resource* in domain *domain* as requested.

Explanation: This message is created by the SA z/OS internal event notification function when the automation manager reports that the automation flag of the resource has been switched on.

The variable *resource* shows the name of the resource.

The variable *domain* shows the name of the end-to-end automation domain.

System action: None.

Operator response: None.

System programmer response: None.

ING230E Addressed consumer *consumer* not registered. Event not forwarded.

Explanation: An event was received with a consumer that was not registered as the target address. The event therefore cannot be forwarded to the addressed consumer.

The variable *consumer* shows the name of the event consumer that was specified as the target address of the event, and that the event should have been forwarded to.

System action: Processing of the event is terminated.

Operator response: None.

System programmer response: Check whether event notification has been correctly initialized and end-to-end automation has been registered as an event consumer.

ING231E Event notification service returns *rc*. Event not forwarded to *receiver_id*.

Explanation: The event could not be forwarded to the addressed consumer via the PIPE stage TECROUTE.

The variable *rc* shows the return code to PIPE stage TECROUTE.

The variable *receiver_id* shows the name of the PPI receiver ID.

System action: Processing of the event is stopped.

Operator response: None.

System programmer response: Check whether the status of the PPI receiver ID is active. The status of PPI receivers can be listed with the command DISPPI.

ING232I PPI receiver ID for E2E adapter is set to *receiver_id*.

Explanation: The PPI receiver ID for the message adapter of the event/automation service, which is used to forward events to the automation adapter, has been set to the specified receiver ID.

System action: None.

Operator response: None.

System programmer response: None.

ING233E Event forwarding task EVTOPER for E2E not defined.

Explanation: The primary agent cannot be started because the automated function EVTOPER, which is used to forward the events to the automation adapter, has not been defined in the automation policy.

System action: Processing stops.

Operator response: None.

System programmer response: Define the automated function EVTOPER.

ING234E Request '*request*' failed: RC=*rc* command='*command*' reason='*reason*'

Explanation: The specified command abended.
 The variable *request* shows the request that was received from the automation adapter for execution.
 The variable *rc* shows the return code of the command.
 The variable *command* shows the command that implements the request.
 The variable *reason* shows text that describes the error.

System action: Execution of the request stops.

Operator response: None.

System programmer response: Analyze the return code and reason code for the abending command.

ING235W No information from automation manager. *event* event not created for *consumer*.

Explanation: An error occurred when requesting information from the automation manager that needed to be included in a domain event. The event has not been created.

The variable *event* specifies the event to be created. Possible values are: READY, REFRESH, CHANGE.
 A READY event is created at initialization time of the automation domain and after having recovered from miscellaneous error situations.

A REFRESH event is created after having refreshed the configuration policy.

A CHANGE event is created after a takeover of the automation manager if the outage has not exceeded a given time interval.

The variable *consumer* shows the name of the event consumer that the event should have been forwarded to.

System action: Processing of the event is stopped.

Operator response: None.

System programmer response: Analyze the NetView log for further SA z/OS messages concerning the communication between the primary agent and the automation manager.

ING236W Task execution request *request* processed with warnings: COMMAND='*command*' REASON='*reason*'

Explanation: A warning was issued when executing a task execution request. The message variables are:

request The name of the task execution request
command The command that failed.
reason Text that describes the error.

System action: The request is executed with warnings.

Operator response: None.

System programmer response: Analyze the reason code of the warning message.

ING237W Request '*request*' processed with warnings: COMMAND='*command*' REASON='*reason*'

Explanation: A warning was issued when executing an end-to-end request.

The variable *request* shows the name of the end-to-end request.
 The variable *command* shows the issued command that implements the request.
 The variable *reason* specifies the warning message.

System action: The end-to-end request executes.

Operator response: Analyze the reason code of the warning message.

System programmer response: None.

ING238E Task execution request *tex_request* failed: ERROR=*error_code* SERVICE='*service*' RC=*return_code* REASON='*reason*'

Explanation: The REXX function INGRXTX0 and common REXX macros issue this message if there is an error. The message variables are:

tex_request

The name of the task execution request

error_code

The REXX function INGRXTX0 and the common REXX macros issue the following error codes:

- 1 RPC service failed: either INGPXSND, INGPXRCV or INGPXDEL.
- 2 Read from or write to the task execution data stream failed.
- 3 Unknown task execution operation name.
- 4 Execution of a task execution request started but failed. The REXX macro that implemented the execution of the request did not issue an error message. This is the default return code and the standard ING238E message is written with SERVICE=OperationName and RC=OperationReturnCode.
- 5 Authorization error: the user that made the request is not authorized to make it.
- 6 At least one request in the list of requests failed.
- 7 The execution of a task execution request failed for any other reason.
- 8 The execution of a task execution request failed because service INGPYAMC failed.
- 9 Execution of a task execution request was rejected because of expired expiration time.
- 24 Task execution request contains no meta data.
- 28 The task execution command handler aborts because there is not enough time to send back the response data. No response is returned.
- 32 The task execution command handler aborts because of a REXX signaling error.

service The service that failed.

return_code

The return code of the service that failed.

reason Text that describes the error.

System action: Execution of the request failed for the reason given in the message.

Operator response: None.

System programmer response: Check the netlog and correct the error.

ING239W Value in advanced automation option *name* not valid. Default *value* used.

Explanation: The value provided in the advanced automation option is not accepted as a valid value.

The variable *name* shows the name of the advanced automation option.

The variable *value* shows the value of the advanced automation option.

System action: Processing continues with the default value for the specified advanced automation option.

Operator response: None.

System programmer response: Correct the value provided in the advanced automation option.

ING240E Unexpected task execution request name *tex_request*.

Explanation: The REXX macro implemented to execute the task execution request was called with an unexpected task execution request name.

The variable *tex_request* shows the name of the task execution request.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING241E Unexpected task execution element type *type*.

Explanation: The REXX macro implemented to execute the task execution request was called with an unexpected task execution request type.

The variable *type* shows the type of task execution element.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING242E Cannot read task execution element from data stream.

Explanation: The REXX macro implemented to execute the task execution request cannot read the task execution element from the data stream.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING243E Cannot write task execution element to data stream.

Explanation: The REXX macro implemented to execute the task execution request cannot write to the data stream to return data to the calling routine.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING244E Invalid filter element name *filter*.

Explanation: The REXX macro implemented to execute the task execution request was passed an invalid filter element name.

The variable *filter* shows the name of the filter element.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING245E Duplicate filter element name *filter*.

Explanation: The REXX macro implemented to execute the task execution request was passed duplicate filter element names.

The variable *filter* shows the name of the filter element.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING246E Invalid combination of filter element names *filter1* and *filter2*.

Explanation: The REXX macro implemented to execute the task execution request was passed an invalid combination of filter element names.

The variables *filter1* and *filter2* show the names of the two filter elements.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING247E Invalid filter element *filter*. Filter value unknown: *value*

Explanation: The REXX macro implemented to execute the task execution request was passed an unknown filter value.

The variable *filter* shows the name of the filter.

The variable *value* shows the value of the filter.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING248E Invalid filter element *filter*. Filter operator unknown: *operator*

Explanation: The REXX macro implemented to execute the task execution request was passed a filter element with an unknown operator comparing the filter name and filter value.

The variable *filter* shows the name of the filter.

The variable *operator* shows the filter operator.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING249E Execution of a task execution request timed out. The expiration time '*time*' was too short – *missing_time*.

Explanation: The expiration time that was provided for the task execution request was too short.

The variable *time* shows the expiration time that was provided for the task execution request.

The variable *missing_time* specifies the missing time interval.

System action: Processing of the task execution request fails.

Operator response: Contact your system programmer.

System programmer response: Check the defined timeout interval for task execution requests. If necessary, increase it as described in *System Automation for z/OS: End-to-end automation adapter*.

ING250E Execution of a service timed out. The expiration time *time* was reduced to *number* seconds.

Explanation: A service routine timed out without exploiting the expiration time that was provided for the execution of the task execution request. The execution time was reduced to the maximum value for the WAIT parameter of the executed service.

The variable *time* shows the expiration time that was provided for the task execution request.

The variable *number* shows the value for the WAIT parameter of the executed service.

System action: Processing of the task execution request fails.

Operator response: None.

System programmer response: None.

ING251E Invalid combination of filter elements
filter1 and *filter2*. Values *value1* and *value2* are mutually exclusive.

Explanation: The REXX macro implemented to execute the task execution request was passed an invalid combination of values for the filter elements.

The variables *filter1* and *filter2* show the names of the two filter elements.

The variables *value1* and *value2* show the values of the two filter elements.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING252E Domain initialization failed.
Automation environment not initialized.

Explanation: Initialization of the automation domain failed because initialization of the system automation environment of the primary agent has not yet been completed.

System action: The start of the automation adapter fails.

Operator response: Restart the automation adapter after the initialization of the automation environment of the primary agent has completed.

System programmer response: None.

ING253E Domain initialization failed. Primary agent already started on system *system*.

Explanation: Initialization of the automation domain failed because the primary agent is already running on another system within the subplex.

The variable *system* shows the system name.

System action: The start of the automation adapter fails.

Operator response: Check whether the automation adapter should be stopped on the other system.

System programmer response: None.

ING254E Domain initialization failed. PPI receiver *receiver* **of EAS not ACTIVE.**

Explanation: Initialization of the automation domain failed because the PPI receiver ID of the NetView event/automation service is not active.

The variable *receiver* shows the name of the PPI receiver.

System action: The start of the automation adapter fails.

Operator response: Analyze the status of the PPI

receivers by using command DISPPI. Check whether the event/automation service has been started and has successfully initialized its receiver ID at the PPI.

System programmer response: None.

ING255E Requesting information from Automation Manager of domain *domain* **failed.**

Explanation: Requesting information from automation manager of an automation domain failed. Requesting information from automation is necessary during the initialization of the automation domain or when requesting information about the automation domain.

The variable *domain* shows the name of the automation domain.

System action: The task execution request fails.

Operator response: None.

System programmer response: Analyze the problem. Examine the NetView log for further information. If the problem persists contact your IBM Support Center.

ING256E Health check failed. Event forwarding switched off.

Explanation: Events were lost when trying to forward them from the primary agent to the automation adapter. Event forwarding has therefore been switched off, which results in failing the health check of the automation domain.

System action: The automation adapter will stop. A subsequent restart enforces the re-initialization of the automation adapter and a refresh of the status information about first-level automation resources on the end-to-end server.

Operator response: Check the communication path from the primary automation agent to the NetView event/automation service. As soon as the communication path is available again, the automation adapter can be restarted.

System programmer response: None.

ING257E Health check failed. PPI receiver *receiver* **of EAS not ACTIVE.**

Explanation: The PPI receiver identifier of the event/automation service is not in the status ACTIVE. This results in failing the health check of the automation domain.

The variable *receiver* shows the PPI receiver ID of the NetView event/automation service.

System action: The automation adapter does not send a heartbeat event. Thus the communication status of the domain on the end-to-end automation server changes.

Operator response: Analyze the status of the PPI receivers by using command DISPPI. Check whether the event/automation service has been started and has successfully initialized its receiver ID at the PPI.

System programmer response: None.

ING258E Execution of service *service* failed. Unexpected data received.

Explanation: The requested service did not return the expected data.

The variable *service* shows the name of the requested service.

System action: Processing of the task execution request fails.

Operator response: Notify your system programmer.

System programmer response: Determine why the service routine fails. Examine the NetView log for additional information.

ING259E Invalid request element. Request name unknown: *name*

Explanation: An end-to-end request with an unknown name was received.

The variable *name* shows the request name.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING260E Invalid request element. Request type unknown: *type*

Explanation: An end-to-end request with an unknown type was received.

The variable *type* shows the request type.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING261E Invalid request element. Multiple solicited requests are not supported.

Explanation: Multiple solicited end-to-end requests were received to be executed by the primary agent. Only one solicited end-to-end request can be accepted.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING262W An attempt was made to overwrite at least one restricted parameter. Input parameters have been ignored: *parameters*

Explanation: An end-to-end request was received with parameters that attempted to overwrite restricted parameters. The parameters are ignored.

The variable *parameters* shows a list of the ignored parameters.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING263E Request rejected because mandatory parameter SOURCE is missing.

Explanation: An end-to-end request was received without the specified originator of the request via parameter SOURCE.

System action: Processing of the task execution request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING264E Reset_from_NRE rejected because agent status of resource *resource* is CTLDOWN.

Explanation: A reset-from-non-recoverable-error request was received for a resource that is in agent status CTLDOWN. Such a request is not supported for resources in agent status CTLDOWN.

The variable *resource* shows the name of the resource that the request was issued for.

System action: Processing of the task execution request fails.

Operator response: Check the agent status of the resource that the request was issued for.

System programmer response: None.

ING265E Reset_from_NRE rejected because observed status of resource *resource* is not HARDDOWN.

Explanation: A reset-from-non-recoverable-error request was received for a resource that is not in status HARDDOWN. Such a request is only supported for resources in status HARDDOWN.

The variable *resource* shows the name of the resource that the request was issued for.

System action: Processing of the task execution request fails.

Operator response: Check the status of the resource.

System programmer response: None.

ING266E **Reset_from_NRE failed because no data to resource *resource* received from Automation Manager.**

Explanation: A reset-from-non-recoverable-error request was received for a resource. An error occurred when requesting information from the automation manager about the resource that the request was issued against.

The variable *resource* shows the name of the resource that the request was issued for.

System action: Processing of the task execution request fails.

Operator response: Check whether the resource that the request was issued for is known to the automation manager.

System programmer response: None.

ING267E **Reset_from_NRE rejected because resource *resource* is not of class APL.**

Explanation: A reset-from-non-recoverable-error request was received for a resource that is not of class APL. This request is only supported for resources of type APL.

The variable *resource* shows the name of the resource that the request was issued for.

System action: Processing of the task execution request fails.

Operator response: Check the type of the resource that the request was issued for.

System programmer response: None.

ING268E *request* **rejected because system name is missing.**

Explanation: An end-to-end request concerning a system was issued, but the system name was not specified.

The variable *request* shows the name of the end-to-end request.

System action: Processing of the task execution request fails.

Operator response: Check that the request was issued against a resource of class SYS.

System programmer response: None.

ING269E **Native command contains an invalid value for parameter \$WAIT: *wait***

Explanation: An invalid value was specified as the WAIT parameter when issuing a native command.

The variable *wait* shows the value for the WAIT parameter that was received.

System action: Processing of the task execution request fails

Operator response: Contact your IBM Support Center.

System programmer response: None.

ING270E **User *user* is not authorized to execute command: *command***

Explanation: The execution of the specified command is rejected by the authorization user exit AOFEXE2E.

The variable *user* shows the user ID that is associated with the command.

The variable *command* shows the rejected command resulting from the related end-to-end request.

System action: Processing of the task execution request fails.

Operator response: Contact your system programmer to obtain the necessary authorization to execute the command.

System programmer response: If it is appropriate, authorize the user to issue the specified command.

ING271E **Command execution failed. Waiting for asynchronous output timed out.**

Explanation: A timeout occurred when waiting for asynchronous output generated by a command that was issued from within a PIPE.

System action: Processing of the task execution request fails

Operator response: Retry the operation. If the problem persists, contact your system programmer.

System programmer response: Determine why the command does not return its output within the expected time. Examine the NetView log for additional information.

ING272E **More than one request in a list failed.**

Explanation: More than one request in a list of requests that were included in a task execution request failed.

System action: Processing of the related requests fails.

Operator response: Examine the NetView log for additional information about the single requests that were processed with errors.

System programmer response: None.

ING273W More than one request in a list processed with warnings.

Explanation: More than one request in a list of requests that were included in a task execution request were processed with warnings.

System action: Processing of the task execution request continues.

Operator response: Examine the NetView log for additional information about the single requests that were processed with warnings.

System programmer response: None.

ING274E A query to the automation manager failed due to unsuccessful execution of service INGPYAMC.

Explanation: An error occurred when requesting information from the automation manager via the automation manager API INGPYAMC.

System action: Processing of the task execution request fails.

Operator response: Check the communication path from the primary agent to the automation manager and examine the NetView log for additional information. If the problem persists contact your IBM Support Center.

System programmer response: None.

ING275E Processing failed for command: *command*

Explanation: A command was called that ended with a non-zero return code.

The variable *command* shows the failed command.

System action: Processing of the command stops.

Operator response: Determine why the command failed and correct the problem.

System programmer response: None.

ING276E Execution of a service timed out.

Explanation: The return code of the requested service indicates a timeout problem.

System action: The requested service does not complete successfully.

Operator response: Retry the operation. If the problem persists, contact your system programmer.

System programmer response: Determine why the service routine timed out. Examine the NetView log for additional information.

ING277I Command Handler INGRXCMJ FAILED: RC=*nn* ReasonText

Explanation: The command handler of the Automation Agent Java Framework detected an error.

The variable *RC* is the return code from the command handler INGRXCMJ as described below.

The variable *ReasonText* is the description of the error, as described below.

Table 4. Reason Text for Return Codes

Return Code	Reason Text
54	CMD_RC= <i>cmd_rc</i> CMD= <i>cmdshort</i> ERRMSG= <i>errormsg</i>
58	PIPE ERROR PIPE_RC= <i>pipe_rc</i>
59	PIPE ERROR STAGE= <i>stage</i> RC= <i>src</i> CMD= <i>cmdshort</i>
60	PIPE ERROR STAGE CORRWAIT ASYNCH TIMEOUT
<i>nn</i>	CMD= <i>short command</i> that failed

Note: For *rc*=1..32, message AOF356 was issued (command handler failed).

Table 5. Return Codes from INGRXCMJ

Return Code	Explanation
0	Success
4	Invalid RPC Type
8	REXX function failed
12	NetView pipe failed
16	buffer.0 is wrong
20	Route cmd failed
24	JVM stuff failed
28	INGPXRCV <i>rc</i> <>0
32	Remote_Command failed (for remote target only)
52	Invalid data stream
54	CMD_RC= <i>cmd_rc</i> CMD= <i>cmdshort</i> ERRMSG= <i>errormsg</i>
58	PIPE ERROR PIPE_RC= <i>pipe_rc</i>
59	PIPE ERROR STAGE= <i>stage</i> RC= <i>src</i> CMD= <i>cmdshort</i>

Table 5. Return Codes from INGRXCMJ (continued)

Return Code	Explanation
60	PIPE ERROR STAGE CORRWAIT ASYNCH TIMEOUT
74	INGEXEC error
75	PIPE error
76	At least one command failed on the remote system
77	Target missing

Note: For return codes 74 to 77, see AOF_JVM_Route_Command (AOF0210).

System action: The *command* has not been executed successfully.

Operator response: None.

System programmer response: Analyze the NetViewlog and search for further error messages which may provide more details about the reason of the error. If rc=54, look for the description of the associated command. If necessary, contact your IBM Contact Support Center.

ING300I Command *command* finished successfully

Explanation: A command has finished processing and achieved what it has been asked to do, for example, starting a resource.

The variable *command* is the command being executed.

System action: None.

Operator response: None.

System programmer response: None.

ING301I Command *command* failed - unexpected results after waiting for *hh:mm:ss*

Explanation: A command has finished processing but has not achieved what it has been asked to do. For example, when starting a resource, the resource is not in the expected state after the allowed time period.

The variable *command* is the command being executed.

The variable *hh:mm:ss* is the timer period.

System action: None.

Operator response: None.

System programmer response: None.

ING302I Nothing to cancel

Explanation: No request exists for the specified resource that originated from the same source.

System action: None.

Operator response: None.

System programmer response: None.

ING304I Waiting for termination of *name* aborted - *reason*

Explanation: System Automation for z/OS was waiting for the termination of the name component, but the termination command was aborted.

Variable *name* is the name of component.

Variable *reason* explains why the termination command was aborted.

Operator response: None.

System programmer response: If the component is OMVS, consider adding command F BPXOINIT,SHUTDOWN=FORKINIT command as SHUTINIT command for JES in the policy.

System action: Processing continues.

ING305I System shutdown started for *name*

Explanation: Indicates that the termination of the system has been started. The shutdown was either requested by the local operator, GDPS® or the Active/Active controller.

Variable *name* identifies the name of the system.

System programmer response: None required.

System action: Processing continues.

ING306I System shutdown nearly complete for *name*

Explanation: Indicates that the shutdown of the names system is nearly finished. Only NetView is still active.

Variable *name* identifies the name of the system.

System programmer response: None required.

System action: Processing continues.

ING307I System shutdown for *name* already in progress

Explanation: Indicates that System Automation for z/OS is already working on shutting down the named system.

System programmer response: None required.

System action: Processing continues.

ING308A **Waiting for termination of *name***

Explanation: System automation is in the middle of shutting down the system and is waiting for the termination of the named component / address space. The termination process is on halt till the named component becomes inactive.

Variable *name* identifies the name of the system.

Operator response: Stop the named component / address space.

System programmer response: None required.

System action: Processing continues.

ING309I **System shutdown for *name* redirected due to GDPS Tower, STOPAPPL=*resource***

Explanation: A system shutdown was requested by means of the INGREQ ALL command. Since the system is under control of GDPS, only the toplevel resource identified by means of the GDPS STOPAPPL parameter in the GEOPLEX OPTIONS member is terminated.

Variable *name* identifies the name of the system.

Variable *resource* identifies the GDPS STOPAPPL resource.

Operator response: None.

System programmer response: None required.

System action: Processing continues.

ING310I **data**

Explanation: This message is used to transport the completion block from the system being terminated back to the Active/Active Controller.

System programmer response: None.

System action: Processing continues.

ING311I **Workload *name* not defined in the SA policy**

Explanation: The Active/Active controller has requested either to Start, Stop a workload or to query the status of the workload, but the workload is not defined in the System Automation for z/OS policy. Workload must be defined in the GDPS_WORKLOAD policy under the MVS component.

Variable *name* identifies the workload.

System programmer response: Define the workload in the System Automation for z/OS policy, if necessary.

System action: Processing continues.

ING312I ***action* of Workload *name* failed.**

Explanation: The Active/Active controller has requested either to Start or Stop a workload, but this failed. See the NetView log for more details about the failure when the workload is managed by System Automation for z/OS or the system log otherwise.

Variable *action* identifies the action being performed (START or STOP).

Variable *name* identifies the workload.

System programmer response: None.

System action: Processing continues.

ING321I **Unsupported function *command* found in emulator *command*, Caller: *caller***

Explanation: The IMFEXEC command was invoked passing a function code that is not supported by System Automation for z/OS.

Variable *emulator* is the name of the command, for example IMFEXEC or OPSCMD.

Variable *command* identifies the function that is not supported by System Automation for z/OS.

Variable *caller* identifies the REXX script that called the IMFEXEC command.

System programmer response: Modify the script invoking the IMFEXEC command.

System action: Processing continues.

ING322I ***emulator function commands* are no longer allowed, Caller: *caller***

Explanation: The IMFEXEC CMD command was called, but no longer supported in the NetView/System Automation for z/OS environment.

Variable *emulator* is the name of the command, for example IMFEXEC or OPSCMD.

Variable *function* identifies the function.

Variable *caller* identifies the REXX script that called the IMFEXEC command.

System programmer response: Modify the script invoking the IMFEXEC command.

System action: Processing continues.

ING323I ***emulator function* does not support parameter *parameter*, Caller: *caller***

Explanation: The IMFEXEC command was called for a supported function, but the specified parameter is not supported by the System Automation for z/OS emulator.

Variable *emulator* is the name of the command, for example IMFEXEC or OPSCMD.

Variable *function* identifies the function.

| Variable *parameter* identifies the parameter that is not supported.
 | Variable *caller* identifies the REXX script that called the IMFEXEC command.

| **System programmer response:** Modify the script invoking the IMFEXEC command.

| **System action:** Processing continues.

| **ING325I alert-message**

| **Explanation:** This message is used to carry the alert text from an IMFEXEC ALERT function call

| **System programmer response:** None.

| **System action:** Processing continues.

| **ING326I No command specified**

| **Explanation:** The OPSCMD command was invoked but no command was specified.

| **System programmer response:** Modify the script invoking the OPSCMD command.

| **System action:** Processing continues.

| **ING327I parameter parameter value is not allowed**

| **Explanation:** The OPSCMD command was invoked specifying a value for a parameter that is not supported by the System Automation for z/OS provided emulator.

| Variable *parameter* names the parameter.

| Variable *value* is the value.

| **System programmer response:** Modify the script invoking the OPSCMD command.

| **System action:** Processing continues.

| **ING328I parameter parameter value is unknown**

| **Explanation:** The OPSCMD was invoked specifying a system that is not known in System Automation for z/OS.

| Variable *parameter* is SYSID/SYSTEM.

| Variable *value* is the name of the system that is unknown.

| **System programmer response:** Modify the script invoking the OPSCMD command.

| **System action:** Processing continues.

| **ING329I Job=jobname User=userid issued: command**

| **Explanation:** The OPSCMD was invoked to execute the specified command.

| Variable *jobname* is the name of the job who originated the OPSCMD call.

| Variable *userid* is the user id associated with the job.

| Variable *command* is the command being used.

| **System programmer response:** None.

| **System action:** The command is processed.

| **ING330I Command execution was successful**

| **Explanation:** A remote command was executed successfully.

| **System action:** The program exits.

| **Operator response:** None.

| **System programmer response:** None.

| **ING331I Command execution failed. Waiting for asynchronous timed out for output.**

| **Explanation:** A remote command execution failed due to timeout on the remote system.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **ING332I Command execution failed. RC=rc
 SYSTEM=sys Command=cmd
 REASON:xxxx**

| **Explanation:** A remote command executed on the System Automation for z/OS/NetView Agent failed due to return code *rc* on the remote system *sys*. Additional error description might be available and describes the reason for the error. See example:

```
ING332I Command execution failed, RC=20 SYSTEM=KEYA
COMMAND=INGRCVAC->INGRCVAL (GLOBAL.,E) REASON:
ING338I FUNCTION OR COMMAND INGRCVAL FAILED, RC=204
REASON=COMPOUND VARIABLE EXPECTED, BUT GOT A STEM
INGRCVAX: Function OPSVALUE completed lrc=404
retvalue=0
      80 +++      return
IRX0045I Error running OPSVALUE, line 80:
No data specified on function RETURN
      23 +++      var = OPSVALUE(vname,action,newval,oldval)
IRX0040I Error running T1, line 23:
Incorrect call to routine
```

| Command was executed on system KEYA.

| Command failed and error message ING338I describes the reason.

| Function OPSVALUE terminates and causes error
 | Incorrect call to routine on purpose.

| **System action:** The function or command terminated.

| **Operator response:** None.

| **System programmer response:** Analyse the netlog and search for further error messages. See explanation of error message ING338I.

ING333I • ING338I

| **Note:** Ignore the error IRX0045I, since No data specified on function RETURN was done on purpose. The real reason was detected earlier in the failing program which caused message ING346I.

ING333I Receiver is starting for PPI=xxx

| **Explanation:** The RPC receiver with PPI receiver id xxx is starting.
 | **System action:** The receiver is ready to receive remote command.
 | **Operator response:** None.
 | **System programmer response:** None.

ING334I Receiver is already running for PPI=xxx

| **Explanation:** The RPC receiver with PPI receiver id xxx is already running.
 | **System action:** The receiver terminates.
 | **Operator response:** Specify a different PPI receiver id if you want to start a new receiver.
 | **System programmer response:** None.

ING335I Execute remote command on behalf of JOBID=xxx CLIST=yyy COMMAND: cmd

| **Explanation:** A remote command is executed on behalf of a RPC issued by job id xxx and clist yyy. The cmd command should be executed on the System Automation for z/OS/NetView Agent.
 | **System action:** None.
 | **Operator response:** None.
 | **System programmer response:** None.

ING336I Receiver PPI=xxx will be stopped due to operator request

| **Explanation:** The RPC receiver with PPI receiver id xxx is stopping due to operator request. A stop-command has been received and issued by

| *Table 6. Return Codes of ING338I*

Module Name	ReturnCode and Description
INGRCVAF	204 Invalid name mask
	205 Invalid name mask, stem contains either + or *
	206 Invalid parmlist
	207 Invalid name mask, length of stem>8
	208 Invalid action
	212 Unsupported action
	216 Error in helper function, for example, INGPCGLB or INGPCREX

| 'INGRCRCV STOP PPI=xxx'

| **System action:** The receiver terminates.
 | **Operator response:** None.
 | **System programmer response:** None.

ING337I PPI RECEIVER STATUS LIST

| **Explanation:** These messages are output of the command 'INGRCRCV QUERY' and display all receiver PPI ids started via command 'INGRCRCV START'. The following example displays a list of two PPI receivers. One is running in a VOST.

```
ING337I PPI RECEIVER STATUS LIST
      STATUS  PPI      TASK      VOST
      -----  -
      ACTIVE  INGRRCV  DSI#0020  EMURCV2
      ACTIVE  PPIABC   BDOW
```

| **System action:** None.
 | **Operator response:** None.
 | **System programmer response:** None.

ING338I Function or command xxxx failed, RC=nn REASON=yyyy

| **Explanation:** A function or a command with name xxxx which is a part of the emulator failed with return code nn. Additional error description is given in yyyy.

| When an error occurred during initialization of the Emulator Environment then the reason might be 'action injected INIT STATUS=FAILED'. In this case any call to function INGRCVAL (OPSVALUE) is rejected.

| For explanation of return code nn see the table below.

| **System action:** Processing terminates.
 | **Operator response:** Notify your system programmer.
 | **System programmer response:** Determine why the function or command failed. Examine the netlog. If necessary, contact your IBM Support Center.

| Use Table 6 to determine the return codes used in message ING338I:

Table 6. Return Codes of ING338I (continued)

Module Name	ReturnCode and Description
INGRCVAL	200 Invalid call to routine 204 Invalid input name 205 Invalid old value for action=A 208 Invalid action 212 Unsupported action 216 Error in helper function, for example, INGPCGLB 218 Error when global variable not exist for action=O 220 Function aborted 224 Syntax error in subroutine(procedure) 236 EMULATOR INIT STATUS is not OK
INGRCVAX	0 OK 2 RPC timeout 4..24 RPC error 60 Invalid SOURCE name 104 Decode data items error 108 Expected at least 1 data item 112 Expected at least 1 data item 114 service routine failed, for example, INGPCREX 116 Host not supported 200 Execution of remote command failed Return code from remote command is unknown or invalid 200 Execution of remote command failed +nnn nnn is the remote return code from INGRCVAC or from INGRCVAL, for example, 404 means invalid input INGRCVAX

ING338I

Table 6. Return Codes of ING338I (continued)

Module Name	ReturnCode and Description
INGRCVAC	<p>0 OK 2 Mandatory parms missing 4 Error in AOF_PPI_PARSE_DATA 8 Error parsing DataItems 12 Error parsing parmlist 15 Error sending data, PIPE PPI error 16 Error sending data, PIPE error 20 INGRVAL failed 21 INGPCGLB failed 22 INGRVAF failed 23 INGRVAL failed for POOLIST 24 Unexpected error 28 Standard Safe not supported for action 34 Triggering message missing 36 ARCHIVE failed because RESTORE not OK 1nn ARCHIVE failed 2nn INGRVAL failed (remote command only) 3nn RESTORE failed</p> <p>where the return codes nn have the following meaning</p> <p>4 PUT init-rec failed 8 2nd OPEN failed 10 DEL failed 12 GET failed 14 GET KEYS failed 15 RESTORE VSAM failed 16 PUT failed 18 INQUIRE failed 19 Cannot retrieve output value 20 VSAM DEL for LOOKUP failed 21 PIPE EDIT for LOOKUP failed 22 PIPE LOOKUP failed 31 ARCHIVE_VSAM failed, mdfy.0 not equal mdfyval.0 32 ARCHIVE_VSAM failed, ref.0 not equal refval.0 33 ref. or refval. are not initialized by INGPCGLB 34 INGPCGLB(QUERY) failed 35 INGPCGLB(UPD_OR_APP) failed</p> <p>See also message ING344I for possible VSAM IO error description.</p>
INGRCRPC	<p>0 OK 2 TIMEOUT 4 Missing Input 8 INGPCREX call failed 9 INGPCSTK call failed 10 Decoding response data items failed (stem) 16 PPI error 20 REXX error such as Syntax, Novalue, and so on 24 Invalid data buffer 28 AOCT0000.REXMAC initialization failed 32 Error while preparing input stem</p>

Table 6. Return Codes of ING338I (continued)

Module Name	ReturnCode and Description
INGRCRCV	0 OK 4 Invalid action 8 RECEIVER already active 12 PPI error 16 PIPE error 20 Invalid parameter list 24 Service failed 28 Auto OPS environment checking failed 40 Initialization failed
INGRCMDH	0 OK 4 Not a RPC 8 Invalid RPC input data 16 Command Execution failed
INRCEXC	0 OK 1 Routing the command failed 2 Executing the command failed on THIS task 4 Bad input arguments 8 AOF_PPI_PARSE_DATA failed
INGRCCSA	201 Memory object name exist already 202 Init of lookup table failed 203 Invalid object name too long (max 12 char) 204 Invalid object name 205 Invalid size, not a number 206 Invalid parmlist 207 Invalid trace parameters 208 Invalid action 212 Unsupported action 216 Error in function INGPCCSA
INGRCRDY	2 INGRCRPC timeout 4..24 INGRCRPC error 104 Decode data items error 108 Expected at least 1 data item 112 Unexpected remote message 116 Host not supported 117 RDS working dataset not configured in CSA 118 Must be called as command 200 Execution of remote command failed Return code from remote command is unknown or invalid +nnn nnn is the remote return code from INGRCVAC or from INGRCVL, for example 404 means invalid input INGRCVL

ING338I

| *Table 6. Return Codes of ING338I (continued)*

Module Name	ReturnCode and Description
INGRCRDS	nnn<200 Error due to underlying services 201 VALUE missing or invalid 202 Mandatory parms missing 203 Table name missing or invalid 204 Error in AOF_PPI_PARSE_DATA 205 COLUMNS missing or invalid 206 Invalid parameter value 208 Error parsing DataItems 212 Error parsing parmlist 215 Error sending data, PIPE PPI error 216 Error sending data, PIPE error 220 No match, nothing updated or deleted 221 INGPCGLB failed 222 INGRCVAL syntax error 223 Invalid action 224 DEQ failed 225 Export to Dataset failed 226 Import to Dataset failed

Table 6. Return Codes of ING338I (continued)

Module Name	ReturnCode and Description
INGRCRDS underlying service	CreateTableControl: 1 Two words required or comma missing, for example, 'NAME Char(30)' 2 NOT numeric, for example, 'NAME Char(3x)' 3 Unsupported data type, for example, 'YEAR Integer' 4 Zero columns, at least one column is required 5 Table name too long (>30 characters) 6 Max row length exceeds implementation limit 32000 7 Invalid column name CreateTableRow: 11 Truncated at least one column 12 Unequal number of column names/values 13 Max length of a row is exceeded UpdateTableRow: 21 Invalid column name ReadTextTableData/Header 31 bad column delimiter 35 bad import header CreateTable: 100 Table exist already UpdateTableControl: 101 Table does not exist ModifyTableControl: 102 Table LOCK does not match IterateTable: InsertTable: 104 Table does not exist 105 Bad validation of TCR(corrupted row) 106 Row does not exist (dir.i) IterateTable: 110 Error, no COLUMN match 115 Error WHERE/SET clause parser 116 Error match where clause (Syntax error !?) 117 Error parse or update 118 Error copy table data CreateTable: InsertTable: GlobalDatastore: 120 INGRVAL failed, see GlobalDatastore(vn,'U') ListOfTables: ListOfPools: 130 INGPCGLB(QUERY_ACL) failed 131 INGPCGLB(QUERY_ACL) returns no hexdata Parser WHERE/SET: 41,42 parsing error, unmatched brackets (TokenizeWhereClause) 43 unsupported operator (TokenizeWhereClause) 44 parsing error, tokenizer failed (ParseValueClause) 46 parsing error, unmatched brackets (Tokenizer) 48,49 parsing error, column name not found (ResolveColumnsinWhereClause) 52 parsing error, invalid operator (ParseSetClause) 56 bad expression in match clause (SYNTAX, NOVALUE)

ING339I Receiver PPI=xxx recovers bad command execution, continue...

Explanation: The RPC receiver with PPI receiver id xxx executed a remote command which failed.

However, the receiver continues processing and waits for the next command.

System action: Execution of the current command terminates. Receiver continues.

Operator response: Notify your system programmer.

ING340I • ING346I

| **System programmer response:** Determine why the remote command failed. Examine the netlog.

| **ING340I Deleted *n1* of *n2* pattern matching global symbols**

| **Explanation:** The function INGRCVAF has filtered *n1* number of global variables. *n2* global variables have been deleted. If *n2* is smaller than *n1* then either the global variable cannot be found or an error occurred.

| **System action:** As many global variables have been deleted as possible.

| **Operator response:** None.

| **System programmer response:** None.

| **ING341I Receiver PPI=xxx has been stopped**

| **Explanation:** The RPC receiver with PPI receiver id xxx has been stopped.

| **System action:** The receiver terminates.

| **Operator response:** None.

| **System programmer response:** None.

| **ING342I No receivers are started**

| **Explanation:** Command 'INGRCRCV QUERY' does not find any receivers that were started.

| **System action:** None.

| **Operator response:** None.

| **System action:** None.

| **ING343I Parameter *xxxx* is not supported by function or command *yyyy***

| **Explanation:** The function or command *yyyy* was invoked passing a parameter that is not supported by System Automation for z/OS.

| Variable *xxxx* identifies the parameter that is not supported by System Automation for z/OS.

| Variable *yyyy* is the name of the command that detected the error.

| **System programmer response:** Modify the script invoking the command.

| **System action:** Processing continues.

| **ING344I SUBSJOB *job_name* FAILED: *error_text***

| **Explanation:** This message is used by some emulator commands to report an error and this might be trapped by the message table. With the 3rd token you can associate the error with a System Automation for z/OS application. The 3rd token *job_name* is the string specified in the field *job_name* in the System Automation for z/OS application information panel.

| You can use this message to set the resource, for example, to status abending.

| Command INGVAlUE ARCHIVE or RESTORE issues this message if a processing error occurred or a VSAM IO error. For example,

| INGVAlUE SUBSJOB EMUARCH FAILED: ARCHIVE VSAM_IO_ERROR..
| INGVAlUE SUBSJOB EMUARCH FAILED: RESTORE ERROR.....

| Token 3 is the subsystem job name, token 5 the subcommand that failed and token 6 the kind of error. For example, VSAM_IO_ERROR.

| **System action:** The originating command terminated.

| **Operator response:** None.

| **System programmer response:** Determine why the function or command failed. Examine the netlog.

| **ING345I Status of long global variables: *xxx***

| **Explanation:** Command 'INGVAlUE STATUS' reports the status of the long global variable framework. This message is also written if the command 'INGVAlUE RESTORE' was successful.

| *xxx* might have the following values:

- RESTORE SUCCESSFUL Emulator restored successfully the persistent global variables.
- INITIAL RESTORE NOT DONE Emulator initialization during System Automation for z/OS initialization was not done.
- INITIAL RESTORE FAILED Emulator initialization during System Automation for z/OS initialization failed.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Determine why the function or command failed. Examine the netlog if the status is not successful.

| **ING346I Abort function *xxx* on purpose at LINE *nnn***

| **Explanation:** The REXX function *xxx* detected an error at line *nnn*. The purpose is that the error causes invalid call to routine. This can only be achieved by returning nothing on purpose. See example:

| INGVAlUE FUNCTION or command INGRCVAl failed,
| RC=204 REASON=invalid character found
| INGVAlUE Abort function INGRCVAl on purpose at LINE 83
| 270 +++ return
| IRX0045I Error running INGRCVAl, line 270:
| No data specified on function RETURN

| **System action:** The function or command terminated.

| **Operator response:** None.

| **System programmer response:** Analyse the netlog and

| search for further error messages. See explanation of
| error message ING338I.

| **Note:** Ignore the error the IRX0045I, since No data
| specified on function RETURN was done on
| purpose. The real reason was detected earlier in
| the failing program which caused message
| ING346I.

| **ING347I Receiver PPI=*id* does not start due to an
| unsupported environment for *xxx***

| **Explanation:** The receiver task was started by
| command 'INGRCRCV START OPF=*xxx*'. The receiver
| searches for the existence of the automation operator
| task (for example, if *xxx* is AOFMDOPER then the
| tasks are AOFMDO nn , $nn=00,01,02,..$) but no active task
| was found.

| **System action:** Receiver terminates.

| **Operator response:** Notify your system programmer.

| **System programmer response:** Check if the operator
| tasks are defined or why they are inactive.

| **ING348I Receiver PPI=*id* starts with a degraded
| operator function array *mm* of *nn* for
| *xxx*.**

| **Explanation:** The receiver task was started by
| command 'INGRCRCV START OPF=*xxx*'. The receiver
| searches for the existence of the automation operator
| task (for example, AOFEMU nn , $nn=00,01,02,..$) but
| found at least one task that was inactive or not defined.

| The receiver task continues running but executing
| incoming commands might be degraded because the
| inactive tasks are skipped, and commands might be
| queued.

- | • The number *mm* is the number of active tasks
- | • The number *nn* is the number of defined tasks
- | • The variable *xxx* is the name of the operator function
| array.

| **System action:** Receiver continues.

| **Operator response:** Notify your system programmer.

| **System programmer response:** Check why not all
| operator tasks are inactive.

| **ING350I Improper Job Log Monitoring
| definitions detected for *name*, RSN=*rsn***

| **Explanation:** During initialization SA z/OS detected
| an improper definition for the Job Log Monitoring task.
| The variable *name* shows the subsystem name that is
| affected.

| The variable *rsn* shows the reason:

| **10** Job Log Monitoring interval defined but no
| message id.

| **11** Offset specification is invalid.

| **12** Token specification is invalid.

| **13** Message id defined even if JOBLOGALL
| has been specified.

| **14** Message offset is 0 but no token has been
| defined.

| **15** Token has been defined but no message
| offset.

| **16** Monitoring interval is invalid.

| **17** DD name specification is invalid.

| **18** Message id is invalid.

| **System action:** The definition is ignored.

| **Operator response:** Correct the definition and refresh
| the configuration if desired.

| **System programmer response:** None.

| **ING400I BCPM Work order *work_order* is created
| on BCPM server *server*, system *system*.**

| **Explanation:** This message displays the identifier of a
| work order created in the Business Continuity Process
| Manager (BCPM) as the result of an outage event on
| the system.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **ING401E Unable to create BCPM Work order.
| Return code *return_code* from BCPM
| server *server*, system *system*.**

| **Explanation:** An outage event from the system cannot
| be delivered or processed by the Business Continuity
| Process Manager (BCPM) server.

| **System action:** None.

| **Operator response:** Inform the IT Service Continuity
| administrator about the outage.

| **System programmer response:** Refer to the netlog for
| additional messages issued by the SA z/OS INGOMX
| command.

| **ING402I BCPM server *server* is notified about
| completion of task *task* with code *code*,
| system *system*.**

| **Explanation:** The task triggered by the Business
| Continuity Process Manager (BCPM) has completed.
| BCPM has received and processed the task completion
| information.

| **System action:** None.

| **Operator response:** None.

System programmer response: None.

ING403E Unable to notify BCPM server *server* about completion of task *task* with code *code*, system *system*. INGOMX command has returned *error_code*.

Explanation: The task triggered by the Business Continuity Process Manager (BCPM) has completed, but the BCPM cannot receive the task completion information. The INGOMX command ended with an error.

System action: None.

Operator response: Inform the IT Service Continuity administrator about the problem.

System programmer response: Refer to the netlog for additional messages issued by the INGOMX command. Check for the reason for the INGOMX error.

ING404E BCPM server *server* cannot process notification from task *task*. BCPM exception is *exception_code*.

Explanation: The task triggered by the Business Continuity Process Manager (BCPM) has completed, but the BCPM cannot process the task completion information.

System action: None.

Operator response: Inform the IT Service Continuity administrator about the problem.

System programmer response: None.

ING405I BCPM replied to the WTOR *wtor* with ID *reply_id* with *reply* on system *system*.

Explanation: The outstanding WTOR was reprocessed by the Business Continuity Process Manager (BCPM).

System action: A process requested WTOR is continued.

Operator response: None.

System programmer response: None.

ING406E BCPM unable to reply to the WTOR *wtor* with *reply* on system *system*. RC=*rc*.

Explanation: The outstanding WTOR is either no longer active and has already been reprocessed by an operator or automation product, or it cannot be reprocessed by the Business Continuity Process Manager (BCPM).

System action: None.

Operator response: Check for outstanding WTORs on the system. Inform the Service Continuity Team about the problem.

System programmer response: None.

ING407E BCPM unable to execute GDPS batch script *batch_script*. Return code *rc*, reason *reason*.

Explanation: The GDPS batch script was triggered by the Business Continuity Process Manager (BCPM), but cannot be executed.

System action: The script is not executed.

Operator response: Inform the IT Service Continuity administrator about the problem.

System programmer response: Refer to the netlog for additional messages issued by GDPS.

ING408I BCPM command *command* ended with rc=*rc*.

Explanation: The command triggered by the Business Continuity Process Manager (BCPM) has completed.

System action: None.

Operator response: None.

System programmer response: None.

ING500I *error_text*

Explanation: A program that has failed has produced this additional information regarding the error. See preceding messages.

System action: Execution probably halts. See subsequent messages.

Operator response: Contact your system programmer.

System programmer response: Assist the operator. Contact your IBM Support Center.

ING501I *debug_text*

Explanation: This is a generic informational message that shows the flow within the function.

System action: Execution continues.

Operator response: None.

System programmer response: None.

ING502I SA z/OS Discovery Library Adapter started *date time*

Explanation: This indicates the time that the SA z/OS DLA started execution.

System action: Execution continues.

Operator response: None.

System programmer response: None.

ING503I SA z/OS Discovery Library Adapter completed *date time*, RC=*nn*

Explanation: This indicates the time that the SA z/OS DLA completed execution.

System action: Execution finishes.

Operator response: The output file is now ready to be downloaded.

System programmer response: None.

ING504I Checksum data updated

Explanation: Following a successful run, the checksum data (member @CHCKSUM) stored in the output data set has been updated.

System action: Execution continues.

Operator response: None.

System programmer response: None.

ING505I Checksum data cleared; bad run

Explanation: The SA z/OS DLA has failed during its execution and, because it has not been able to generate a valid output book, it has cleared the checksum data.

System action: Execution continues.

Operator response: Analyze the problem and, when fixed, rerun the SA z/OS DLA to generate a valid book.

System programmer response: Assist the operator.

ING507I Data in *dataset_name* truncated; Increase output LRECL

Explanation: This indicates that the SA z/OS DLA tried to write a line to the output data set that was longer than the data set's LRECL.

The variable *dataset_name* identifies the fully qualified name of the data set.

System action: The book is generated with the truncated line. Depending on the nature of the truncated data, the iDML book that is generated may or may not be valid.

Operator response: Reallocate the data set with a longer LRECL and rerun the SA z/OS DLA.

System programmer response: Assist the operator.

ING508I Output of book *book* skipped; checksum matches

Explanation: The SA z/OS DLA has successfully executed, however the contents of the generated book are the same as the previously generated book in the same output data set. Generation of a book from the newly generated data is therefore suppressed in order

to preserve discovery date and time information in the previously generated book.

The variable *book* identifies the book.

System action: Execution continues.

Operator response: None.

System programmer response: None.

ING509I Data set *name* is not a partitioned data set

Explanation: The specified output data set is not a partitioned data set.

The variable *name* identifies the data set.

System action: None.

Operator response: Correct the name of the data set and reissue the command.

System programmer response: None.

ING510I *program* Directory OPEN failed for DDName *ddname*, RC *rc*

Explanation: The program was unable to open the directory for the data set that is pointed to by the *ddname*. The OPEN macro gave the specified return code.

System action: Execution probably halts. See subsequent messages.

Operator response: The output data set must be a PDS—you cannot use a sequential file. Contact your system programmer.

System programmer response: Assist the operator. Contact your IBM Support Center.

ING511I *program* Directory READ failed for DDName *ddname*, RC *rc* RS *rsn*

Explanation: The program tried to read the directory for the data set that is pointed to by the *ddname*, but received the unexpected return and reason codes indicated.

System action: Execution probably halts. See subsequent messages.

Operator response: The output data set must be a PDS—you cannot use a sequential file. Contact your system programmer.

System programmer response: Assist the operator. Contact your IBM Support Center.

ING512I *program* STORAGE macro call failed for type *type*, RC *rc*

Explanation: The program received an unexpected return code from a call to the STORAGE macro for the type of storage that was requested.

ING513I • ING806E

System action: Execution probably halts. See subsequent messages.

Operator response: Contact your system programmer.

System programmer response: There may be other messages in the netlog or syslog at the time of the incident that will give more details. If NetView is not running REGION=0M, try increasing its region size. Contact your IBM Support Center.

ING513I *program* **Codepage Translation Routine CUNLCNV returned RC rc, RS rsn**

Explanation: The program called the CUNLCNV routine and received an unexpected return and reason code.

System action: Execution probably halts. See subsequent messages.

Operator response: Contact your system programmer.

System programmer response: Contact your IBM Support Center.

ING800I **VERIFYING *type-model* HARDWARE AUTOMATION INTERFACE**

Explanation: A check of the type and model of the hardware automation interface is performed.

The variable *type-model* shows the type and model of the hardware being verified.

System action: The verification process for the specified processor is starting.

Operator response: None.

System programmer response: None.

ING801I **AUTOMATION INTERFACE NOT SUPPORTING *type-model* HARDWARE**

Explanation: Verification determined that the processor hardware is not supported

The variable *type-model* shows the type and model of the hardware.

System action: None.

Operator response: None.

System programmer response: None.

ING802I **HARDWARE INTERFACE COMPONENT HSAET32 CANNOT BE ACTIVATED**

Explanation: An error was reported by the HSAET32 internal command. The command that establishes and activates the hardware interface failed.

System action: None.

Operator response: None.

System programmer response: Check the netlog or the system log for HSAxxxx messages about the problem.

ING803I **HARDWARE INTERFACE AUTO OPERATOR *autotask* CANNOT BE STARTED**

Explanation: The named autotask failed to start. The hardware interface cannot be used.

The variable *autotask* shows the autotask that could not be started.

System action: None.

Operator response: None.

System programmer response: Check the netlog for additional messages. Correct the problem and activate the autotask again. This will initialize the hardware interface.

ING804I **HARDWARE AUTOMATION INTERFACE OF *type-model* PROCESSOR VERIFIED**

Explanation: The verification for the specified processor has ended.

The variable *type-model* shows the type and model of the hardware.

System action: None.

Operator response: None.

System programmer response: None.

ING805I *requestor* **REQUESTS TO CONNECT TO CPC *cpcaddr***

Explanation: The hardware interface tries to establish a connection to the indicated processor hardware.

The variable *requestor* shows the originator of the request.

The variable *cpcaddr* shows the address of the CPC.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGRVX80

Classes: 40.

ING806E **COMMUNICATION WITH CPC *cpcaddr* CANNOT BE ESTABLISHED**

Explanation: A failure occurred while the processor hardware was contacted through the hardware interface.

The variable *cpcaddr* shows the address of the CPC that a session could not be established with.

System action: None.

Operator response: None.

System programmer response: Check the netlog for AOFAXxxx messages to obtain more information, for example, the available condition and sense codes. Correct the problem and restart the session to the CPC.

Module: INGRVX80

ING807I OPERATOR *opid* IS WAITING FOR SYSOPS INITIALIZATION TO COMPLETE

Explanation: The operator task is currently executing the clist INGRX810 or INGRX805 and halts processing until the NetView common global variable AOFCOMPL has a value of YES, which indicates that SA z/OS SysOps initialization is complete. This halt is a temporary condition. Processing continues once SA z/OS SysOps is initialized. Message ING808I is issued to indicate that the continuation of the clist execution for the operator task.

The variable *opid* shows the name of the affected NetView operator task.

System action: None.

Operator response: None.

System programmer response: In situations when SA z/OS SysOps cannot initialize, you can use the STOP FORCE command to end the current command of such an operator. For more information about the SA z/OS common global variable AOFCOMPL, see *IBM Tivoli System Automation for z/OS Customizing and Programming*.

Module: INGRX805, INGRX810

ING808I OPERATOR *opid* ENDED TO WAIT FOR SYSOPS INITIALIZATION

Explanation: The operator task continues to execute the clist INGRX810 or INGRX805 after processing was halted. Message ING807I was issued to indicate the halt condition for the operator task.

The variable *opid* shows the name of the affected NetView operator task.

System action: None.

Operator response: None.

System programmer response: None.

Module: INGRX805, INGRX810

ING810I *requestor* REQUESTS TO DISCONNECT FROM CPC *cpcaddr*

Explanation: The hardware interface terminates the communication with the indicated processor hardware.

The variable *requestor* shows the originator of the request.

The variable *cpcaddr* shows the address of the CPC.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGRVX80

Classes: 40.

ING811I BCPII CONNECTION HEARTBEAT BETWEEN CPC *cpcaddr* AND *domainoperator* FAILED. LASTS(*stime*)

Explanation: For a period of 4 minutes, the Support Element of the addressed CPC has not responded to the BCPii heartbeat request of the operator task running in the NetView domain. The BCPii event connection is no longer working.

The variable *cpcaddr* is the Processor name defined as SA-PDB / Connection Flag / CPC Name of the processor in SNA netid.nau format. The Connection Flag indicates:

- L BCPii local connection. SA z/OS targets its own CPC.
- R BCPii connection to a remote CPC using a HMC for routing.
- . The Connection Flag information is not available.

The variable *domain* is the NetView domain name of the BCPii connection.

The variable *operator* is the operator task name of the event session of the affected BCPii connection.

The variable *stime* is the Date/Time stamp of the LAST successful connection start in the format yymmddhhmmss.

System action: The BCPii connection is internally restarted. If the restart is successful, message ING812I is issued. If the restart fails, message ING813E is issued with additional information.

Operator response: None.

System programmer response: None.

Module: INGCHW0E

Classes: 40.

ING812I BCPII LINK BETWEEN CPC *cpcaddr* AND *domainoperator* RE-ESTABLISHED. NEW LASTS(*stime*)

Explanation: After a BCPii event session heartbeat failure, the BCPii connection has been successfully restarted.

The variable *cpcaddr* is the Processor name defined as SA-PDB / Connection Flag / CPC Name of the processor in SNA netid.nau format. The Connection Flag indicates:

- L** BCPii local connection. SA z/OS targets its own CPC.
- R** BCPii connection to a remote CPC using a HMC for routing.
- The Connection Flag information is not available.

The variable *domain* is the NetView domain name of the BCPii connection.

The variable *operator* is the operator task name of the event session for the affected BCPii connection.

The variable *stime* is the Date/Time stamp of the LAST successful connection start in the format yymmddhhmmss. The connection start initializes the LASTS value and is therefore prefixed with NEW.

System action: Processing continues.

Operator response: None.

System programmer response: For additional information, refer to the event handler asynchronous response report AOFA0998, documented in the appendix section "Internal Transport Services "0Bx00xxx'" on page 442 of *IBM Tivoli System Automation for z/OS Messages and Codes*.

Module: INGCHW0E

Classes: 40.

ING813E BCPII CONNECTION RECOVERY FROM *domain/operator* TO CPC *cpcaddr* FAILED. RSN(*rsncode*) LASTS(*stime*)

Explanation: After a BCPii event session heartbeat failure between the targeted CPC and the operator task of the NetView domain, the internal BCPii connection restart failed.

The variable *cpcaddr* is the Processor name defined as SA-PDB / Connection Flag / CPC Name of the processor in SNA netid.nau format. The Connection Flag indicates:

- L** BCPii local connection. System Automation targets its own CPC.
- R** BCPii connection to a remote CPC using a HMC for routing.
- The Connection Flag information is not available.

The variable *domain* is the NetView domain name of the BCPii connection.

The variable *operator* is the operator task name of the event session for the affected BCPii connection.

The variable *rsncode* is the reason code. For full details, refer to the appendix section "Internal Transport Services "0Bx00xxx'" on page 442 of *IBM Tivoli System Automation for z/OS Messages and Codes*.

The variable *stime* is the Date/Time stamp of the LAST successful connection start in the format yymmddhhmmss.

System action: The BCPii connection to the addressed CPC is closed.

Operator response: Since the BCPii connections are monitored by SA z/OS or GDPS, a start of a failing connection is retried automatically at each defined monitor cycle time. No further action is required. If the heartbeat failure message ING811I was preceded by a ING814E message, additional error cause and reason information is available.

System programmer response: None.

Module: INGCHW0E

Classes: 40.

ING814E BCPII CONNECTION TO *domain/operator* ABORTED BY THE CPC *cpcaddr* SUPPORT ELEMENT. CAUSE(*cause_type*) REASON(*rsn_str*) LASTS(*stime*)

Explanation: The CPC Support Element has sent an 'Application_Ended' event over an active BCPii connection to indicate the communication with the NetView *domain/operator* is about to end. Note that this event message is provided only for BCPii connections to System z10™ and newer processors.

The variable *cpcaddr* is the Processor name defined as SA-PDB / Connection Flag / CPC Name of the processor in SNA netid.nau format. The Connection Flag indicates:

- L** BCPii local connection. SA z/OS targets its own CPC.
- R** BCPii connection to a remote CPC using a HMC for routing.
- The Connection Flag information is not available.

The variable *domain* is the NetView domain name of the BCPii connection.

The variable *operator* is the operator task name of the event session for the affected BCPii connection.

The variable *cause_type* is the action that caused the console application to end. Depending on the type given, a possible console application outage time can be predicted.

SHUTDOWN_CONSOLE

Long outage, manual intervention is required to recover.

RESTART_APPLICATION
 Medium outage of 10 - 20 minutes can be expected until the console application is fully operational.

RESTART_CONSOLE
 Medium outage of 10 - 20 minutes can be expected until the console device and console application are fully operational again.

UNKNOWN
 The type code in the event data is unknown. Long outage, manual intervention may be required to recover.
 The variable *rsn_str* refers to the issuer that caused the console application to end.

USER
 A logged-on SE user.

AUTOMATION
 An automation application.

OTHER
 Remote action such as SE reboot from an HMC.

UNKNOWN
 The reason code in the event data is unknown. The variable *stime* refers to the Date/Time stamp of the LAST successful connection start in the format yymmddhhmmss.

System action: Processing continues.

Operator response: Use the *cause_type* information of the message to determine the BCPii connection outage time that can be expected. In case of a long outage time, consider the SUSPEND command for the BCPii connection.

System programmer response: None.

Module: INGCHW0E

Classes: 40.

ING815I BCPII CONNECT FROM *domainoperator* TO CPC *cpcaddr* RETURNED: *resp*

Explanation: Request to contact the CPC Support Element of *cpcaddr* over the BCPii from *domain/operator* returned a response of *resp*.

The variable *cpcaddr* is the Processor name defined as SA-PDB / Connection Flag / CPC Name of the processor in SNA netid.nau format. The Connection Flag indicates:

- L** BCPii local connection. SA z/OS targets its own CPC.
- R** BCPii connection to a remote CPC using a HMC for routing.
- .** The Connection Flag information is not available.

The variable *domain* is the NetView domain name of the BCPii connection.

The variable *operator* is the operator task name of the event session for the affected BCPii connection.

The variable *resp* is:

SUCCESS
 SE connection established.

ERROR(*rsncode*)
 Failure to contact the SE.

rsncode
 For information about the reason code, please refer to the appendix section "Condition Codes" on page 434 of *IBM Tivoli System Automation for z/OS Messages and Codes*. Select the table that lists the reason code together with the error description.

System action: If the connect was successful, session initialization continues with CPC configuration discovery. Message ING817I will be issued when this is complete. If the connect processing failed, no further session initialization is done and processing stops.

When a BCPii connection monitoring interval is defined to SA z/OS or GDPS HMC monitoring is used for *cpcaddr*, the connect requests will be retried automatically.

Operator response: None.

System programmer response: None.

Module: INGCHW0A

Classes: 40.

ING816I BCPII AWAIT CMD SESSION TERMINATION TO CPC *cpcaddr* IN *delay* SECONDS: DOMAIN(*domain*) OPID(*operator*) CMD(*cmd*) LASTS(*stime*)

Explanation: The HW event handler for CPC (*cpcaddr*) session returned an error and the asynchronous session terminates. In order to perform a save memory cleanup, an ongoing HW command for the CPC running in task (*operator*) needs to be terminated first. Wait for (*delay*) seconds for normal command termination.

The variable *cpcaddr* is the Processor name defined as SA-PDB / Connection Flag / CPC Name of the processor in SNA netid.nau format. The Connection Flag indicates:

- L** BCPii local connection. SA z/OS targets its own CPC.
- R** BCPii connection to a remote CPC using a HMC for routing.
- .** The Connection Flag information is not available.

The variable *cmd* is the command token of the currently executed HW command.

The variable *domain* is the NetView domain name of the BCPii connection.

The variable *delay* is the time in seconds the command session is given to end.

The variable *operator* is the operator task name of the command session using the affected BCPii connection.

The variable *stime* is the Date/Time stamp of the LAST successful connection start in the format yymmddhhmmss.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGCHW0M

Classes: 40.

ING817I BCPII DISCOVERY OF *cpcaddr* FOR *domain/operator* RETURNED: *resp*

Explanation: The initial discovery request from *domain/operator* for configuration information of *cpcaddr* ended with a response of: *resp*

The variable *cpcaddr* is the Processor name defined as SA-PDB / Connection Flag / CPC Name of the processor in SNA netid.nau format. The Connection Flag indicates:

- L** BCPii local connection. SA z/OS targets its own CPC.
- R** BCPii connection to a remote CPC using a HMC for routing.
- .** The Connection Flag information is not available.

The variable *domain* is the NetView domain name of the BCPii connection.

The variable *operator* is the operator task name of the event session for the affected BCPii connection.

The variable *resp* is:

SUCCESS NEW LASTS(*stime*)
All required CPC configuration information has been retrieved. The BCPii connection is now fully operational.

stime
Date/Time stamp of the LAST successful connection start in the format yymmddhhmmss. The connection start initializes the LASTS value and is therefore prefixed with NEW.

FFDC(*ffdc_id*)
Failure to retrieve the required CPC configuration information. The BCPii session cannot be used.

ffdc_id
The four character id string represents the exit point from the discovery processing and can be used by IBM to locate the failing retrieval.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGCHW0A

Classes: 40.

ING818I BCPII LINK BETWEEN *domain/operator* AND CPC *cpcaddr* IS NOW TERMINATED. NEW STATUS(*connstat*) OLD LASTS(*stime*)

Explanation: The connection to the *cpcaddr* has been successfully closed from the *domain/operator* side. Before applications can use the connection again, they must restart it. If the connection was closed, because of a session suspension, before restarting it, the session must be resumed.

The variable *cpcaddr* is the Processor name defined as SA-PDB / Connection Flag / CPC Name of the processor in SNA netid.nau format. The Connection Flag indicates:

- L** BCPii local connection, SA z/OS targets its own CPC.
- R** BCPii connection to a remote CPC using a HMC for routing.
- .** The Connection Flag information is not available.

The variable *domain* is the NetView domain name of the BCPii connection.

The variable *operator* is the operator task name of the event session for the affected BCPii connection.

The variable *connstat* is the new connection-session status:

NOT_CONNECTED
Connection-session can be restarted again.

SUSPENDED
Connection is suspended from normal usage. It is required to resume it first, before restarting a BCPii session over this connection.

The variable *stime* is the Date/Time stamp of the LAST successful connection start in the format yymmddhhmmss. Since the connection is no longer active the LASTS is prefixed with OLD.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGCHW0C, INGCHW0D

| **Classes:** 40.

| **ING819I** **BCPII ROUTE** *domain/operator - cpcaddr*
| *cstat LASTS(stime)*

| **Explanation:** The BCPii connection monitor,
| configured for CPC *cpcaddr* has detected a connection
| status of *cstat*. Note that connection monitoring can be
| suppressed by specifying an interval value of 'NONE'
| in the PDB processor entry for a CPC. Processors
| specified in the GEOPLEX definitions of GDPS are
| automatically excluded from monitoring since GDPS
| provides the HMC Monitor to supervise BCPii
| connections.

| For the eligible CPCs, the BCPii connection monitor
| starts running after the SA z/OS agent initially
| activates the configured BCPii connections. The same
| applies if changes in the processor definitions, affecting
| the current or new BCPii connections are activated with
| an ACF REFRESH or ACF COLD at SA z/OS agent
| run-time.

| The variable *domain* is the NetView domain name of
| the BCPii connection.

| The variable *operator* is the operator task name of
| the event session for the affected BCPii connection.

| The variable *cpcaddr* is the Processor name defined
| as SA-PDB / Connection Flag / CPC Name of the
| processor in SNA netid.nau format. The Connection
| Flag indicates:

| **L** BCPii local connection. SA z/OS targets its
| own CPC.

| **R** BCPii connection to a remote CPC using a
| HMC for routing.

| • The Connection Flag information is not
| available.

| The variable *cstat* is:

| **CONNECTED**

| The BCPii connection is established.

| **NOT_CONNECTED**

| No BCPii connection is active.

| The variable *operator* is the operator task name of
| the event session for the monitored BCPii
| connection. If *cstat* is NOT_CONNECTED the
| LASTS field value is a dot.

| The variable *stime* is the Date/Time stamp of the
| LAST successful connection Start in the format
| yymmddhhmmss. If *cstat* is NOT_CONNECTED the
| LASTS field value is a dot.

| **System action:** If the connection status is
| NOT_CONNECTED, the monitor will try to start it. If
| the connection status is CONNECTED, no extra
| processing is done. In any case, the next monitor run is
| scheduled, using the configured monitor interval to set
| up a NetView timer for this monitor run.

| **Operator response:** When needed, the BCPii

| connection monitoring can be stopped by using the
| SA z/OS SETTIMER command dialog. Look for the
| 'AOFRASCD HWOPER01 INGRX830 1' cmd string and
| perform the delete action of the associated TIMER-ID in
| the dialog.

| **System programmer response:** None.

| **Module:** INGRX830

| **Classes:** 40.

| **ING920I** **COMMAND** *cmd* **ISSUED BY** *jobname*
| **INHIBITS OTHER COMMANDS/JOBS**
| **FROM EXECUTING**

| **Explanation:** SA z/OS found a command or job that
| inhibits at least one other command from completing
| its execution. The policy defined during customization
| does not allow for abending the command.

| The variable *cmd* shows the command name.

| The variable *jobname* shows the name of the job.

| **System action:** Processing continues.

| **Operator response:** Inform your system programmer.

| **System programmer response:** Check the execution
| time of the command. If the execution time is not
| reasonable, add the command to the policy along with
| a reasonable execution time. This allows the cancelation
| of the job if necessary.

| **Module:** INGRX744

| **Classes:** 40, 43.

| **TEC:** YES

| **ING921I** **THE JOB** *jobname* **IS BEING ABENDED**
| **DUE TO A HUNG COMMAND**
| **DETECTION ON RESOURCE** *resource*

| **Explanation:** SA z/OS detected a job that owns a
| resource that inhibits commands from executing
| successfully. The job is abended with U2047
| RSNC9D5C740 to avoid lockouts of other commands.

| The variable *resource* shows the major and minor
| name of the resource.

| The variable *jobname* shows the name of the job.

| **System action:** Automation abends the indicated job.

| **Operator response:** None.

| **System programmer response:** None.

| **Module:** INGRX744

| **Classes:** 40, 43.

| **TEC:** YES

**ING922I DUE TO COMMAND FLOOD
DETECTION THE JOB *job* IN *asid* ON
system IS BEING ABENDED**

Explanation: SA z/OS detected a job that is flooding the master or console address space with commands. To avoid lockouts of other commands the job is canceled and all of its commands are abended with U2047 RSNC9D5C740.

The variable *job* shows the name of the job.

The variable *asid* shows the address space ID running the job.

The variable *system* shows the name of the system running the job.

System action: SA z/OS cancels the indicated job and abends all its commands in the command class where the automation detected the command flooding.

Operator response: Inform your system programmer.

System programmer response: Determine the reason why the job was flooding the master or console address space with commands. Make the necessary changes to prevent future occurrences of this situation before you restart the job.

Module: INGRX746

Classes: 40, 43.

TEC: NO

**ING923I LONG-RUNNING ENQ
AUTOMATION IS ACTIVE**

Explanation: Automation has started monitoring of long-running ENQs sysplexwide.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Module: INGRX745

Classes: None.

TEC: NO

**ING924E AUTOMATION DETECTED THE JOB
job IN *asid* ON *system* FLOODING THE
COMMAND CLASS *cc***

Explanation: SA z/OS detected a job that floods the indicated command class. However, the policy defines the job as *non-cancelable*. For this reason the automation leaves the job and the commands that have been issued by the job untouched.

The variable *job* shows the name of the job issuing the command.

The variable *asid* shows the address space ID running the job.

The variable *system* shows the name of the system running the job.

The variable *cc* shows the name of the command class that is affected.

This message is issued approximately every 60 seconds until there are no more commands waiting for execution.

System action: Processing continues.

Operator response: Observe the command class to determine whether the job still floods the class. If this is the case, cancel the job because this is a serious situation that inhibits other commands from being executed in the indicated class.

After you have canceled the job clean up the command class. Use the command CMDS SHOW to determine the IDs of the commands that have been issued by the indicated job. Then use the command CMDS REMOVE to remove any commands that are waiting, and the command CMDS ABEND to terminate any commands that are executing.

System programmer response: None.

Module: INGRX746

Classes: 40, 43.

TEC: NO

**ING925E CF *cfname* IS IN NORMAL_OFFLINE
STATE, UNABLE TO *function***

Explanation: SA z/OS is unable to enable or populate the coupling facility. SA z/OS has issued V PATH(*cfname*,*chp*), ONLINE commands on all systems in the sysplex, but at least one system is unable to connect to the coupling facility. This condition has persisted for 30 seconds.

The variable *cfname* is the coupling facility name.

The variable *function* is either ENABLE or POPULATE.

System action: Processing continues.

Operator response: Contact your system programmer.

System programmer response: Ensure that every system in the sysplex has at least one channel path that is online to the coupling facility.

Classes: 40, 43.

TEC: YES

**ING926I JOB *jobname* INHIBITS OTHER
COMMANDS FROM EXECUTING.
RESOURCE=*qname rname***

Explanation: SA z/OS found a job that inhibits at least one other command from completing its execution. The policy defined during customization

does not allow for terminating the task that holds the resource lock.

The variable *jobname* shows the name of the job.

The variable *qname* shows the major resource name that is enqueued by the job and required by the command.

The variable *rname* shows the minor resource name that is enqueued by the job and required by the command.

System action: Processing continues.

Operator response: Contact your system programmer.

System programmer response: Check the execution time of the command. If the execution time is not reasonable, add the command to the policy along with a reasonable execution time. This allows the cancelation of the job if necessary.

Module: INGRX744.

Classes: 40, 43.

TEC: YES

ING927E **HSAPRGRS ENQ REQUEST FAILED.**
REQUEST *request* **RETURN CODE** *return code* **REASON CODE** *reason code*

Explanation: HSAPRGRS was called for ENQ processing but failed. This may be an ENQ/GRS issue or a programming error in HSAPRGRS. For ENQ failures the requested resource may already be in use and not available or the GRS environment may not be valid.

The variable *request* shows the request type for HSAPRGRS/GQSCAN.

The variable *return code* shows the return code from HSAPRGRS or the GQSCAN macro.

The variable *reason code* shows the reason code from HSAPRGRS or the GQSCAN macro.

System action: Processing is halted.

Operator response: Contact your system programmer.

System programmer response: Check the return codes listed below:

Code	Meaning
0	Request successfully completed.
4	
	CHNG Status cannot be changed. Other tasks share the resource.
	EXCL/SHR/TEST Resource is not immediately available.
	REL Resource control requested but not yet assigned.
	SCAN No resources matched the request.

8

CHNG Status cannot be changed. No ENQ has been requested.

EXCL/MOVE/SHR/TEST

A previous request was made. The task already has control of the resource.

REL Resource is already released or the task does not have control of the resource.

SCAN The area is too small.

For return codes 10 and 12 a reason code from macro GQSCAN is also returned (see *z/OS MVS Assembler Reference*).

10

SCAN GQSCAN failed. Check the reason code.

12

SCAN GQSCAN failed. Check the reason code.

20

CHNG/EXCL/MOVE/SHR/TEST

A previous request was made. The task does not have control of the resource.

24

EXCL/MOVE/SHR

ENQ limit reached.

32

No storage available for GQSCAN.

36

Resource not owned by the task with the TCB address.

-1

IRXEXCOM error.

-2

IRXEXCOM error.

*n*01

*n*th parameter not specified.

*n*02

Length of *n*th parameter is greater than 240.

*n*03

*n*th parameter doesn't represent a stem variable.

*n*04

Value of *n*th parameter is invalid.

*n*05

Invalid parameter.

1*nnn*

IRXEXCOM error '*nnn*'.

Classes: 40, 43.

INGC1010I Exit *exitname* has been disabled.

Explanation: The exit named in *exitname* has been disabled. *exitname* can have the values of XMEOUT for the CICS message exit, or XDTOOUT for the Transient Data Queue exit.

This message will be issued if an existing policy has enabled the specified exits and a refresh of the policy is in progress. The exits are disabled and then optionally re-enabled based on the new policy.

System action: Processing continues and the specified exit is not invoked to process messages.

Operator response: This is not an error message. The product will function correctly with one or both the exits disabled. However, this message should be followed by INGC1011I if the user has specified a message exit policy according to the *IBM Tivoli System Automation for z/OS Product Automation Programmer's Reference and Operator's Guide* manual.

INGC1011I Exit *exitname* has been enabled.

Explanation: The exit named in *exitname* has been enabled. *exitname* can have the values of XMEOUT for the CICS message exit, or XTDOUT for the Transient Data Queue exit.

This message will be issued if a refresh of the policy is in progress. The exits are enabled if the user has specified policy information in the MESSAGES/USER DATA policy. The XMEOUT exit will be enabled for any message in the policy that starts with DFH and has an OFFSET= value specified in the USER DATA section. The XTDOUT exit will be enabled for any message that has the OFFSET= and TDQUEUE= values specified in the USER DATA section.

System action: Processing continues and the specified exit will be invoked to process messages.

Operator response: None.

INGC1012I No message policy definitions found. Exits will not be enabled.

Explanation: There are no USER DATA policy definitions for the XMEOUT or XTDOUT exits and so the exits will not be enabled.

System action: Processing continues and the exits will not be invoked to process messages.

Operator response: If messages should be processed, check that the OFFSET= parameter has been specified in the USER DATA section of the message policy. Also check that the TDQUEUE= parameter is specified for Transient Data Messages.

INGC1013I Processing *function*. CICS *service* Service failed with EIBRCODE=*code*

Explanation: In the process of refreshing the message exit policy in the CICS subsystem, a CICS service has returned an unexpected return code.

The variable *function* can be one of:

XMEOUT

Policy was being processed for the XMEOUT exit.

XTDOUT

Policy was being processed for the XTDOUT exit.

BOTH

Policy was being processed for both the XMEOUT and XTDOUT exits.

The variable *service* can be one of:

DISABLE

Exit was being disabled via EXEC CICS DISABLE.

ENABLE

Exit was being enabled via EXEC CICS ENABLE.

EXTRACT

Exit information was being extracted via EXEC CICS EXTRACT.

FREEMAIN

Storage was being released via EXEC CICS FREEMAIN.

GETMAIN

Storage was being obtained via EXEC CICS GETMAIN.

MSGBUILD

A message was being built. EIBRCODE refers to the code returned from this routine and not an EXEC CICS function.

The variable *code* is the EIBRCODE of the EXEC CICS function or the return code as specified above.

System action: Processing is aborted and exits may or may not be enabled or disabled.

Operator response: For MSGBUILD make sure that the INGAMCE module is in the DFHRPL concatenation. If it is and for all other circumstances, report this error to service.

INGI1010I Automated Operator Exit Initialized for IMS *ims* Level *level*

Explanation: The SA z/OS exit has been initialized and the IMS control region is running at the specified level.

The variable *ims* is the IMS subsystem id.

The variable *level* is the Version Modification Release of the IMS.

System action: Processing continues.

Operator response: None.

INGI1011I No message policy definitions found. Exits will not be enabled for IMS *ims*.

Explanation: There are no USER DATA policy definitions for the IMS subsystem and so the exits will not be enabled

The variable *ims* is the IMS subsystem id.

System action: Processing continues and the exits will not be invoked to process messages.

Operator response: If messages should be processed, check that the OFFSET= parameter has been specified

in the USER DATA section of the message policy.

INGI1012I Exit DFSAOE00 has been disabled for IMS *ims*.

Explanation: The DFSAOE00 exit has been disabled.

This message will be issued if an existing policy has enabled the specified exits and a refresh of the policy is in progress. The exits are disabled and then optionally re-enabled based on the new policy.

The variable *ims* is the IMS subsystem id.

System action: Processing continues and the specified exit is still invoked, however as no message policy is present, it will not take any actions.

Operator response: This is not an error message. The product will function correctly with the exit disabled. However, this message should be followed by INGI1013I if you have specified message exit policy according to the *IBM Tivoli System Automation for z/OS Product Automation Programmer's Reference and Operator's Guide* manual.

INGI1013I Exit DFSAOE00 has been enabled for IMS *ims*.

Explanation: The DFSAOE00 exit has been enabled.

This message will be issued if a refresh of the policy is in progress. The exits are enabled if the user has specified policy information in the MESSAGES/USER DATA policy. The exit will be enabled for any message in the policy that has the OFFSET= value specified in the USER DATA section.

The variable *ims* is the IMS subsystem id.

System action: Processing continues and the specified exit will be invoked to process messages.

Operator response: None.

INGLCR01E Dynamic allocation failed for DSN=*file name*.

Explanation: Life Cycle Recording (LCR) cannot allocate the data set specified in *file name*.

System action: Either LCR cannot be enabled or the error occurred while offloading the data space to a data set (using the SAVE command).

Operator response: Check that you have specified a correct data set name via the command INGRXQRY LCR, or in HSAPRMxx or INGXINIT.

System programmer response: None.

INGLCR02I Close life cycle recording.

Explanation: SA z/OS starts closing Life Cycle Recording (LCR) for either the primary automation manager (PAM) or an automation agent.

System action: The data space is automatically offloaded to the data set.

Operator response: None.

System programmer response: None.

INGLCR03I Begin blocking access to LCR.

Explanation: When Life Cycle Recording (LCR) is about to close or offload the data space, activities by other tasks are blocked until the data space has been written to disk.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR04I End blocking access to LCR.

Explanation: The Life Cycle Recording (LCR) close or offload process has finished.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR05E ABEND *xxxx-rcrcrcrc* occurred for output media *media_type*

Explanation: An abend occurred during the I/O operation for a data set or data space for Life Cycle Recording (LCR).

The variable *xxxx* is the system completion code.

The variable *rcrcrcrc* is the corresponding reason code.

System action: Life Cycle Recording is disabled.

Operator response: Collect additional information about the reason for the abend, for example, the data set was too small. If you cannot fix the problem, report the error to your IBM Support Center.

System programmer response: None.

INGLCR06E Cannot open/write output media.

Explanation: Life Cycle Recording (LCR) cannot complete one of the following:

- Opening a data set or a data space
- Writing a record to the data space
- Offloading the data space to data set

System action: Life Cycle Recording is disabled.

Messages ING001I to INGY1203I

Operator response: Look for additional error messages, such as INGLCR05.

System programmer response: None.

INGLCR07I Life cycle recording has been enabled, NAME=dataset_name

Explanation: Life Cycle Recording (LCR) has been enabled with the *dataset_name* that will later be used as the offload data set.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR08I Life cycle recording is currently status, NAME=dataset_name

Explanation: Status message indicating the current *status* of Life Cycle Recording (LCR), which is either enabled or disabled.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR09I Life cycle recording will be disabled.

Explanation: An abend occurred and Life Cycle Recording (LCR) is about to be disabled.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR10I Life cycle recording has been successfully initialized, NAME=dataset_name

Explanation: Life Cycle Recording (LCR) has been initialized. A data set was allocated and an initial record was written (any old data is lost). The data space has been allocated.

System action: Life Cycle Recording is ready for the command to enable it.

Operator response: None.

System programmer response: None.

INGLCR11I Successfully flushed records, total size #rec(KB)=size1(size2).

Explanation: *size1* is the total number of checkpoint records ever written by Life Cycle Recording (LCR) to the data space. This is an accumulated number and is not reset due to possible wraparound.

size2 is the number of bytes actually allocated by the

data space. Later when the data space is saved to disk, *size2* is written.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR12I Switch output media for Life cycle recording.

Explanation: Life Cycle Recording (LCR) has switched the output media. The old data space is offloaded to the old data set. A new, empty data space is allocated and initialized with an initial record. A new data space is created.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR13E Switching output media for life cycle recording failed.

Explanation: Life Cycle Recording (LCR) cannot switch the output media.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR15E Creation of data space failed for DSP=file name

Explanation: Life Cycle Recording (LCR) cannot create a data space. This is caused either by the command INGRXQRY LCR, or while initializing the SA z/OS INGPXDST agent task or the SA z/OS primary automation manager (PAM).

System action: LCR cannot be enabled.

Operator response: Look for additional error messages, such as INGLCR23E.

System programmer response: None.

INGLCR21I Start offloading data space.

Explanation: Life Cycle Recording (LCR) starts offloading the data space to the data set.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR22I End offloading data space.

Explanation: Life Cycle Recording (LCR) has finished offloading the data space to the data set.

System action: None.

Operator response: None.

System programmer response: None.

INGLCR23E *function* failed, rc=*rc* frc=*frc* rsn=*rsn*

Explanation: Life Cycle Recording (LCR) detected an error while processing the function *function*. Diagnostic information is provided as the return code (*rc*), function return code (*frc*), and reason code (*rsn*).

The variable *function* shows the function that produced the error. It can be OPEN, CLOSE, OFFLOAD, or FLUSH.

The variable *rc* shows the return code:

- 32 REXX V-pool error
- 40 Communication environment not available
- 81 Initialization of lifecycle recording failed
- 99 Invalid input arguments
- <100 Internal error by INGPXILC
- 108 Internal. Invalid parameter list
- 109 Internal. LCR control block has not been initialized
- 112 Internal. Invalid function
- 116 Dynamic allocation failed
- 120 File open failed
- 121 File put failed

Note that within NetView an additional * may be appended to the end of the message text.

| **INGPC010I *pgm* is unable to perform function**
| **service, PPI_id=xxxxxxx**

| **Explanation:** The program *pgm* failed to perform the
| PPI function service with PPI using the PPI id
| xxxxxxx. Function could be:

- | • open DSIPHONE(OPENRECV)
- | • close DSIPHONE(CLOSE)
- | • send DSIPHONE(SEND)
- | • receive DSIPHONE(RECEIVER)

| **System action:** The program exits.

| **Operator response:** Check if a correct PPI receiver id
| was specified. Message INGPC011I might also be
| issued and provides additional PPI and DSIPHONE
| error information.

| **System programmer response:** None.

| **INGPC011I xxxxx, PPI Reason: yyyy**

| **Explanation:** The message provides DSIPHONE error
| information xxxxx and optional PPI error info yyyy. It
| will be issued in addition to INGPC010I.

| **System action:** None.

122, 123

Writing a control record failed while
closing the file

128 Get record failed

132 Function rejected because a sever error
occurred previously

136 ESTAE cannot be established

140 An abend occurred

144 DSP Create failed

145 DSP Delete failed

146 DSP Put failed

147 DSP Get failed

148 DSN is missing

149 Semicolon missing in INGRXQRY
command

150 Invalid size

>200 ENQ failed

>300 DEQ failed

>400 NetView ALLOC error

489 Invalid parameter list

The variable *frc* shows the function return code.
This is additional diagnostic information that is
intended for the IBM Support Center.

The variable *rsn* shows the reason code. This is
additional diagnostic information that is intended
for the IBM Support Center.

System action: LCR is disabled.

Operator response: Collect additional error messages,
if available.

System programmer response: None.

| **Operator response:** None.

| **System programmer response:** None.

| **INGPC012I The program *pgm* detected missing or**
| **invalid input for argument *n***

| **Explanation:** The program *pgm* failed because the
| input argument *n* is missing or invalid.

| **System action:** The program exits.

| **Operator response:** None.

| **System programmer response:** Check input
| arguments.

| **INGPC015I *cond* condition trapped in script *pgm***
| **Line: *sigl* Code: *rc* Description: *desc***

| **Explanation:** The program *pgm* failed due to a REXX
| script error.

| **System action:** The program exits.

| **Operator response:** Report problem to your system
| programmer unless the condition is triggered by the
| operator, such as a HALT condition.

| **System programmer response:** Report the problem to your IBM Support Center.

| **INGPC016I** *INGPCxxx service failed, RC=nn*

| **Explanation:** The program *INGPCxxx* failed with return code *nn*.

| **System action:** The program exits.

| **Operator response:** Report the problem to your system programmer.

| **System programmer response:** Report the problem to your IBM Support Center.

| **INGPC017I** *pgm failed. Reason*

| **Explanation:** The program *pgm* failed. Additional error info is provided in *Reason*.

| **System action:** The program exits.

| **Operator response:** Report the problem to your system programmer.

| **System programmer response:** Analyze the return code and/or additional error message. Resolve the problem and rerun the command.

| **INGPC018I** *pgm terminates with error, rc=nnn*

| **Explanation:** The program *pgm* terminates due to an error. Return code *nnn* is provided.

| **System action:** The program exits.

| **Operator response:** Report the problem to your system programmer.

| **System programmer response:** Analyze the return code and/or additional error message. Resolve the problem and rerun the command.

| **INGPC019I** *pgm does not support host: HOST ASP*

| **Explanation:** The program *pgm* does not support the REXX host environment. *HOST* and *ASP* is extracted from REXX 'parse source'.

| **System action:** The program exits.

| **Operator response:** Check why the program was called in the wrong environment.

| **System programmer response:** None.

| **INGPC020I** *pgm detected bad response: reason*

| **Explanation:** The program *pgm* failed since an unexpected response was received. *Reason* describes the error.

| **System action:** The program exits.

| **Operator response:** Report problem to your system programmer.

| **System programmer response:** Report the problem to your IBM Support Center.

| **INGPC021I** *pgm is called with invalid name cpyname*

| **Explanation:** The program *pgm* failed since the REXX program was copied with an unsupported name or alias *cpyname*.

| **System action:** The program exits.

| **Operator response:** Report problem to your system programmer.

| **System programmer response:** Use supported OPSxxxx names when copying or defining an alias of *pgm*.

| **INGPC022I** *pgm is called as cpyname for an invalid mode: mode*

| **Explanation:** The program *pgm* failed since it was called as command but only function call is supported or vice versa. The invalid mode is either COMMAND or FUNCTION.

| **System action:** The program exits.

| **Operator response:** Invoke the function only as the supported mode.

| **System programmer response:** None.

| **INGPC023I** *RACF check failed: reason. For info.*

| **Explanation:** On TSO the emulator commands check the TSO user id against a profile. This message will be issued if access is not granted or if a RACF error message occurred. *reason* might be one of the following:

- | • Access Allowed
- | • Profile not Defined
- | • User not Authorized
- | • HSAPRRAC rc=nnn

| *Info* is given in the following format, for example:
 | UserID=BDOW Clist=INGRCCMX Class=FACILITY
 | PROFILE=ING.EMULATOR.CMD

| The high level qualifier ING was customized via command "INGCSA SET NAME=EMU.RACF.HLQ VALUE=ING". The qualifier EMULATOR is a fixed string and CMD depends on the calling clist being one of the following: {CMD,VAL,SQL,DFLT}. If EMU.RACF.HLQ was not customized, RACF checking is ignored.

| **System action:** The calling program terminates.

| **Operator response:** Report the problem to your system programmer.

| **System programmer response:** Give the TSO user the correct RACF access rights or correct the RACF HLQ.

INGPC024I SA REXX function aborted due to no data specified on RETURN

Explanation: An SA REXX function running under TSO detected an error. A typical scenario might be: Due to the RACF checking of the TSO user id, the emulator for function OPSVALUE fails. Since a value must be returned for this error, the emulator must force an incorrect call to routine by returning nothing.

```
INGPC023I RACF check: Profile not defined.
For UserID=BDOW Clist=INGRCVAX
Class=FACILITY Profile=ING.EMULATOR.VAL
INGPC024I SA REXX function aborted due to no data
specified on RETURN.
      92 +++      return
IRX0045I Error running OPSVALUE line 92: No data
specified on function RETURN
      5 +++ VAL=OPSVALUE(GLOBAL.A,'U',AAA)
      5 +++ INTERPRET X
IRX0040I Error running OPSTEST, line 5: Incorrect call
to routine.
```

System action: The REXX function exits.

Operator response: Report the problem to the system programmer.

System programmer response: Analyze additional message accompanied with INGPC024I and fix the problem in the REXX user program.

INGPC025I Incorrect call to *name*

Explanation: The named program was called directly rather than from a REXX script.

Note: Because these INGX $nnnn$ t messages are written to the syslog, the message text must be in English.

INGX1001E PLX STACK OVERFLOW: *module*
compile_date info

Explanation: SA z/OS detected and internal error due to a PLX stack overflow.

The variable *module* shows the name of the module that detected the stack overflow.

The variable *compile_date* shows the date that the module was compiled.

The variable *info* gives additional information about the overflow.

System action: None. The function terminates with abend 3999.

Operator response: None.

System programmer response: Check the syslog for the associated messages and dump information, and contact your IBM Support Center.

INGX1002I SHORT ON XCF BUFFERS - RETRY

Explanation: The SA z/OS communication framework cannot send data segments due to the condition "Short On XCF Buffers". SA z/OS will try to resend the data

System action: The program exits.

Operator response: None.

INGPC026I *error information*

Explanation: An error occurred as reported by the previously issued message for example, INGPC017I. This message gives additional information.

System action: The program exits.

Operator response: Report problem to your system programmer.

System programmer response: If necessary, report the problem to your IBM Support Center.

INGPC027I *command. Parameter parm missing or invalid value*

Explanation: The specified parameter is either wrong or missing.

The variable *command* is the name of the command being processed.

The variable *parm* is the parameter that is invalid or missing.

The variable *value* is the value of the parameter that is wrong. The variable is only present when applicable.

System action: None.

Operator response: None.

segment after a timeout delay. If this is not successful a timeout error message ING008I is issued.

System action: None.

Operator response: None.

System programmer response: Check the syslog for the associated message HSAM1050E that contains further debugging information. If the problem persists consider increasing the number of XCF buffers. The maximum XCF buffer size for SA z/OS is 4028.

INGX1011I INBOUND DIAGNOSTICS: DIF=*nn*
ABS=*nn* QUE=*n* EXP=*n* SEC=*nn*
CPU=*nnn* OUT=*nn* RSP=*nn* FAF=*nn*

Explanation: This message is issued by the automation manager work item queue monitoring routine when it detects that more work items are arriving than can be processed by the automation manager within the monitoring interval.

DIF The number of incoming work items that have not yet been processed during the monitoring interval. It is the number of work items in the queue at the current monitoring time minus the number of work items in the queue at the

Messages ING001I to INGY1203I

last monitoring. If the number is positive and grows it is an indication that the automation manager is being put under stress.

ABS The total number of work items in the input and output queues. This value includes the number of work items that have been processed but that have not been through the cleanup process. The number is typically higher than the QUE number. If the number differs greatly from the QUE value, it might be an indication that the automation manager is unable to deliver command responses to the automation agents. A value that is slightly higher than QUE is acceptable.

QUE The current size of the work item queue.

EXP The number of work items in the input queue that have already expired. Expired work items are not processed by the automation manager but cause an ING008 timeout message.

SEC The monitoring interval. It is customizable and runs from 10 to 999 seconds.

CPU The total accumulated CPU time used by the automation manager

OUT The number of outbound orders that are waiting to be sent to the agent or agents.

RSP The query work items in the input queue.

FAF The number of work items in the input queue that do not require a response. Typically, these are status updates.

System action: Execution continues.

Operator response: None.

System programmer response: None.

INGX9701I ingadapter.sh
[-](start|stop|GenerateSampleKeys|
IBMSupport|traceON|traceOFF) [suffix]

Explanation: Describes the syntax of the automation adapter start/stop command. It is also shown if there are invalid arguments or no arguments specified.

System action: None.

Operator response: Specify command syntax as described.

INGX9702I Script ingadapter.sh completed successfully.

Explanation: The end-to-end automation adapter start script ended successfully.

System action: None.

Operator response: None.

INGX9703E Script ingadapter.sh completed unsuccessfully.

Explanation: An error occurred during script execution.

System action: Script execution is interrupted.

Operator response: Check previous error messages for more details.

INGX9704I Preparing the environment to start the automation adapter.

Explanation: The script ingadapter.sh performs a validation and a cleanup step before starting the automation adapter. The script determines whether an automation adapter on the same port is already running, or if there are any open connections left by a failed automation adapter shutdown. In the former case the script aborts and in the latter case the connections are dropped. Moreover, any existing pid-file of a failed automation adapter shutdown is deleted.

System action: None.

Operator response: If the automation adapter cannot be started or the connections cannot be dropped, check whether the user ID that was used to run the automation adapter has the correct access rights. For details see the chapter "Operating the end-to-end automation adapter" in *System Automation for z/OS End-to-end automation adapter*.

INGX9705E Cannot start automation adapter with the same configuration: hostname:port

Explanation: The automation adapter cannot be started twice with the same port number and hostname as specified in the automation adapter master configuration file. This is determined by examining an existing pid-file whose filename is constructed from the hostname and port number. The containing process-id is used to determine whether another automation adapter is running or if the pid-file has been left over by a failed automation adapter shutdown.

System action: Script execution is interrupted.

Operator response: Start the automation adapter with a different configuration.

INGX9706I Trying to drop possible open, unused connections. Processing port:
port_number

Explanation: The script tries to close possible open and unused connections that have been left by a failed automation adapter shutdown. These connections are detected with the netstat command. The connection will be dropped using the command netstat -d conid, if specific access rights have been granted.

System action: Connections are dropped.

Operator response: If the `ingadapter.sh` script cannot drop connections check whether the user ID that was used to run the automation adapter has the correct access rights. For details see the chapter “Operating the end-to-end automation adapter” in *System Automation for z/OS End-to-end automation adapter*.

INGX9707I Status of the automation adapter: adapter is running.

Explanation: Information about the status of the automation adapter.

System action: None.

Operator response: None.

INGX9708I Status of the automation adapter: adapter is not running.

Explanation: Information about the status of the automation adapter.

System action: None.

Operator response: None.

INGX9709E Rights to access process list have not been granted. Change permissions.

Explanation: The necessary rights to show all processes have not been granted to the current user. The script cannot correctly determine the automation adapter status.

System action: Script execution is interrupted.

Operator response: Change access rights to show all processes. For information about this topic see adapter manual.

INGX9710I Could not retrieve installed codepage. Trying with default codepage:

Explanation: The script was not able to determine the installed codepage because environment variable `E2A` has not been set correctly. It attempts to use the default codepage.

System action: The system uses the default codepage.

Operator response: Set the environment variable `E2A` to the desired codepage.

INGX9711E The automation adapter configuration file is missing.

Explanation: The automation adapter master configuration file, `ing.adapter{suffix}.properties`, was not found. The suffix is optional and can be specified as the second argument of `ingadapter.sh`.

System action: Script execution is interrupted.

Operator response: If you use a suffix make sure you

have specified it as the second argument for `ingadapter.sh`. The automation adapter master configuration file is expected to be located in the configuration directory. Also check the customization section of the script `ingadapter.sh`. Check manual for corresponding information.

INGX9712I Generating Sample Keys for test purposes.

Explanation: The script generates sample keystore and truststore files for SSL. These keys are for testing purpose only.

System action: None.

Operator response: None.

INGX9713E Keytool was not found. Please install a java-sdk and customize ingadapter.sh accordingly.

Explanation: The automation adapter script, `ingadapter.sh`, attempted to use the `keytool` from the Java SDK but the tool was not found. The `keytool` is used to generate sample keys for SSL.

System action: Cannot create sample keys.

Operator response: Install a Java SDK and adapt the path to the `keytool` using the variable `JAVA_KEYTOOL` in the customization section of the script `ingadapter.sh`.

INGX9714I Trace was set to: level.

Explanation: The script has successfully set the specified level. *level* is either `OFF` or `DEBUG_MAX`.

System action: None.

Operator response: None.

INGX9715E An error occurred while trying to set log level.

Explanation: The script failed in at least one case to set the specified log level. Use the counterpart of the command to reset the logger to its previous state. Ignore failures that are reported by the script in this undo step. Those commands that succeeded the first time will succeed in the undo step and those that did not will not succeed the second time.

System action: None.

Operator response: Undo changes by invoking the opposite trace command: for example, `traceOFF` if `traceON` has failed.

Messages ING001I to INGY1203I

INGX9730I Starting Automation Agent Java Plugin
Adapter Script: *start_parameters*

Explanation: The USS script of the System Automation Agent Java Adapter starts. The *start_parameters* are a list of name-value pairs which might be useful for diagnostic purposes.

System action: Processing continues.

Operator response: None.

System programmer response: None.

INGX9731I Finished Automation Agent Java
Adapter Script: *RC=retcode*

Explanation: The USS script of the System Automation Agent Java Adapter has been terminated. The final return code is provided in *retcode*.

System action: Processing finished.

Operator response: None.

System programmer response: If the *retcode* is not zero, look for additional messages and analyze the reason for the problem.

0 Success

2 USS script failed

>2 java mainline failed, see also message INGX9877I.

INGX9732I Cannot create logger output directory:
logdir

Explanation: The USS script of the System Automation Agent Java Adapter cannot create the logger output directory as specified by *logdir*.

System action: Processing terminates.

Operator response: None.

System programmer response: Check that the USS script has correct write access permissions.

INGX9733I BAD *java version foundVersion*, required *requiredVersion*

Explanation: The USS script compares the required minimum java version with the actual version found by *java version*.

System action: Processing terminates.

Operator response: None.

System programmer response: Install and set up at least the required java version.

INGX9800E Cannot find message file or message ID
file_or_msgid

Explanation: The message file cannot be loaded or the message ID cannot be found in the message file.

The variable *file_or_msgid* shows the name of the associated message file or ID.

System action: None.

Operator response: Check the class path.

System programmer response: None.

INGX9801E INGXLogger cannot setup
com.ibm.log.PDLogger

Explanation: Initialization of the message and trace log service failed.

System action: The message or trace data cannot be written to the log file.

Operator response: Check the classpath or the logger configuration file, *eezjlog.properties*.

System programmer response: None.

INGX9802I INGXLogger has successfully been
initialized using configuration file
config_file from path *path*.

Explanation: Initialization of the message and trace log service was successful. The logger configuration data were read from the specified configuration file.

The variable *config_file* shows the name of the configuration file that the logger configuration data was read from.

The variable *path* shows the path of the configuration file.

System action: Message and trace data will be written to the log file.

Operator response: None.

System programmer response: None.

INGX9803E INGXLogger is not available : *message*

Explanation: An attempt was made to write a message to the logger but the logger was not initialized.

The variable *message* shows the message that could not be written to the logger.

System action: The message is written to the *stderr.txt* file.

Operator response: None

System programmer response: None.

INGX9810I Timeout after *interval* seconds.

Explanation: A connection from the JVM to the SA z/OS communication manager (INGPXDST) timed out.

The variable *interval* shows the duration of the timeout.

System action: The response from the associated request is not received.

Operator response: Increase the timeout.

System programmer response: None.

INGX9820E JNI function *function* failed with return code *return_code*.

Explanation: A function call to the specified JNI DLL failed.

The variable *function* shows the function call that failed.

The variable *return_code* shows the return code that specifies the reason for the failure:

- 10 Internal—invalid field ID (attribute not available in Java class)
- 20 Internal—handle cannot be saved in Java long since long too small
- 21 Internal—handle is invalid, for example, null pointer
- 22 Handle cannot be created since no memory
- 23 Internal—handle iterator already used (number of slots too small)
- 26 Internal—invalid array size (Java array size is zero or too small)
- 27 Internal—invalid object (null pointer or an unexpected class)
- 30 Internal—invalid parameter (unexpected input argument)
- 31 Invalid buffer length (input data buffer has unexpected length)
- 32 A Java class that was accessed via JNI has thrown an exception
- 33 Creation of a new Java array failed
- 34 Creation of a new Java string failed
- 35 Out of memory
- 36 A Java class cannot be found
- 40 Catastrophic error—an unexpected exception occurred
- 1000 PPI receive timeout
- 11nn PPI subsystem not available
- 12nn PPI initialization of receiver queue failed
- 13nn The PPI receiver queue exists already
- 14nn PPI NetView posted PPI ECB due to subsystem error
- 15nn PPI cannot receive data
- 16nn PPI cannot delete receiver queue
- 17nn PPI cannot send data

where *nn* denotes the PPI-specific return code as described in the *NetView Application Programmer's Guide*, for example:

- 04 The PPI receiver is not active. For example, the SA z/OS agent is not available or is not properly customized.
- 24 The PPI is not active.
- 26 The receiver program is not defined, for example, PPI=YES has not been added to INGXINIT.
- 28 An active subsystem interface address space was found, but an active PPI address space was not found.
- 32 No NetView storage is available.
- 35 The receiver buffer queue is full.

System action: The corresponding request failed.

Operator response: Analyze the return code. If necessary, contact your system programmer.

System programmer response: Analyze the return code. For example:

- RC=1300 indicates that another automation adapter is running on the system. Verify with D OMVS,A=nnnn whether there is another automation adapter running, where *nnnn* is the ASID of INGEVE2E that you obtained with the NetView DISPPI command.
- RC=1726 indicates that there might be a mismatch of the XCF group ID in `ing.adapter.plugin.properties`. Verify whether it matches the XCF group ID that was defined in INGXINIT for the automation agent and HSAPRMxx for the automation manager. Alternatively, the statement PPI=YES might be missing in the INGXINIT member.

INGX9821E Object of class *class* has already been destroyed.

Explanation: Internal error.

System action: The corresponding request failed.

Operator response: Contact your IBM Support Center.

System programmer response: None.

INGX9822E Environment error. Check handle failed with return code *return_code*.

Explanation: Internal error.

System action: The corresponding request failed.

Operator response: Contact your IBM Support Center.

System programmer response: None.

INGX9823E Class *class* cannot load DLL *dll*.

Explanation: The JNI DLL cannot be loaded from the specified Java class.

System action: All communication services will fail.

Messages ING001I to INGY1203I

Operator response: Check the libpath.

System programmer response: None.

INGX9830E INGXStream failed reading from the data stream. Attempted to read *expected* bytes and got *actual* bytes.

Explanation: The data stream decoder expected to read *expected* number of bytes but got only *actual* bytes.

System action: The corresponding request fails.

Operator response: Check why connection was broken.

System programmer response: None.

INGX9831E INGXStream failed reading from the data stream. Found unexpected end-of-data.

Explanation: The data stream decoder did not find the end-of-data marker. Data might be corrupted. The connection might have broken or there is an internal data stream error.

System action: The corresponding request fails.

Operator response: Check why connection was broken.

System programmer response: None.

INGX9833E INGXStream failed due to invalid header.

Explanation: The data stream decoder found an invalid header in the data stream.

System action: The corresponding request fails.

Operator response: Contact your IBM Support Center.

System programmer response: None.

INGX9840E Execution of a remote REXX command handler aborted.

Explanation: The corresponding request cannot be executed within the SA z/OS NetView backend.

System action: The corresponding request fails.

Operator response: Analyze the NetView log for more SA z/OS messages or REXX syntax errors, or both.

System programmer response: None.

INGX9850E FileIO function *function* failed, *rc=rc* *reclen=rl* *fname=dsn* *fmode=mode*

Explanation: An I/O error occurred while accessing a z/OS data set from a Java program. For example, opening or reading the SMF report.

The variable *function* is the name of the function that detected the IO error.

The variable *rc* is the return code from the I/O service routine libingjrjio.so.

The variable *rl* is the local record length used to access the data set.

The variable *dsn* is the name of the data set.

The variable *mode* is the file mode used to access the data set.

System action: Processing terminates.

Operator response: Perform problem determination and correct the problem.

INGX9854E At least one input parameter is missing.

Explanation: This is a generic message that is issued when at least one input parameter was not specified or its value is null.

System action: Processing terminates.

Operator response: Perform problem determination, for example, check the input data, and correct the problem.

INGX9855E *function* has been stopped due to a failure, *rc=rc*

Explanation: An error occurred and the specified function terminates.

The variable *function* is the name of the function that terminates due to an error.

The variable *rc* is the return code:

16 The logger object cannot be instantiated.

32 Processing failed due to an exception.

40 Exception occurred during cleanup.

System action: Processing terminates.

Operator response: Perform problem determination, analyze associated exceptions and correct the problem.

INGX9856E Illegal syntax or format, *data*

Explanation: This is a generic message that is issued when a conversion error occurred in a Java program. For example, the data and time string from an SMF report are converted to an integrated timestamp of a specific format.

The variable *data* is diagnostic data that might describe the reason for the problem in more detail.

System action: Processing terminates.

Operator response: Perform problem determination, for example, check the input data, and correct the problem or contact your IBM Support Center.

INGX9860I *function* starting to feed the SA z/OS SMF report into DB2 table.

Explanation: The program is being started. It reads the SMF report, converts it to EEZ format and puts it into the DB2 table. This message is written to syslog.

The variable *function* is the name of the function that issued the message.

System action: None.

Operator response: None.

INGX9861I *function* is going to connect to DB2.

Explanation: The program tries to connect to the remote or local DB2 via JDBC driver.

The variable *function* is the name of the function that issued the message.

System action: None.

Operator response: None.

INGX9862I *function* is going to open the SMF report *dsn*

Explanation: The program tries to open the SMF report. The SMF report must have been generated before this program runs.

The variable *function* is the name of the function that issued the message.

The variable *dsn* is the name of the data set that contains the SMF report to be processed.

System action: None.

Operator response: None.

INGX9863I *function* is going to feed DB2 with the SMF report.

Explanation: The program attempts to read the SMF report, and converts each record from the SMF report to a row in the corresponding EEZ tables in DB2.

The variable *function* is the name of the function that issued the message.

System action: None.

Operator response: None.

INGX9864I *function* fed *num* events successfully into the DB2 table within *time* seconds.

Explanation: The program has successfully put the specified number of events from the SMF report into the DB2 table.

The variable *function* is the name of the function that issued the message.

The variable *num* is the number of records converted.

The variable *time* is the number of seconds taken by the program to feed DB2 with the SMF records.

System action: None.

Operator response: None.

INGX9865I Contents of EEZFeeder properties file: *config_parms*

Explanation: The input data from STDENV is converted to an internal properties file for the *eezfeeder.jar*.

The variable *config_parms* is the list of configuration parameters that is passed to *eezfeeder.jar*.

System action: None.

Operator response: None.

INGX9866I *function* completed successfully.

Explanation: The program has successfully put the SMF report into the EEZ tables in DB2.

The variable *function* is the name of the function that issued the message.

System action: None.

Operator response: None.

| **INGX9870I** System Automation Agent Java Plugin
| *'nickname'* failed: *'reason'*

| **Explanation:** The Automation Agent Java Framework
| invoked a plugin with name *nickname*. The plugin
| failed and a *reason* is provided which describes the
| error.

| **System action:** The Java plugin cannot execute the
| request. Processing of the Automation Agent Java
| Adapter continues.

| **Operator response:** None.

| **System programmer response:** Usually, the *reason*
| starts with a message id. Look for that message
| description in this case. Otherwise *reason* describes the
| error. Analyze the log file of the associated Automation
| Agent Java Adapter. Search for further error messages
| which may provide more details about the reason of
| the error.

| **INGX9871I** System Automation Agent RPC from
| Java Plugin to Netview failed:*'reason'*

| **Explanation:** The Automation Agent Java Framework
| attempted to call a NetView command but the attempt
| failed. The reason of the failure is described in the
| *reason* text.

| **System action:** The Java plugin cannot execute a
| command on the local System Automation NetView
| Agent. Processing of the Automation Agent Java
| Adapter continues.

Messages ING001I to INGY1203I

| **Operator response:** None.

| **System programmer response:** The *reason* starts with a message id. Look for that message description. Analyze the log file of the associated Automation Agent Java Adapter space. Search for further error messages which may provide more details about the reason of the error.

| **INGX9872I System Automation Agent Java Plugin with nickname 'nickname' not found.**

| **Explanation:** The Automation Agent Java Framework tries to invoke a plugin with name '*nickname*'. However a plugin with matching '*nickname*' cannot be found.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Look in the NetView log and search for correlated error messages in order to find the System Automation command that tried to call the Java plugin. Check if this command requires a Java plugin installation.

| **INGX9873I System Automation Java Plugins are not available or not initialized**

| **Explanation:** During the startup of the Automation Agent Java Adapter, it will be checked that at least one plugin is available and successfully initialized. If this is not the case, this message is issued and the address space stops.

| **System action:** Address space stops.

| **Operator response:** None.

| **System programmer response:** Search for additional messages that explain the reason why the plugin cannot be initialized. For example because the plugin class cannot be found in class path.

| **INGX9874I System Automation Agent Java Adapter 'subsjob' is active.**

| **Explanation:** Message used for automation of the Automation Agent Java Adapter via System Automation policy.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **INGX9875I System Automation Agent Java Adapter 'subsjob' is ready.**

| **Explanation:** Message used for automation of the Automation Agent Java Adapter via System Automation policy.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **INGX9876I System Automation Agent Java Adapter 'subsjob' has been stopped.**

| **Explanation:** Message used for automation of the Automation Agent Java Adapter via System Automation policy.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **INGX9877I System Automation Agent Java Adapter 'subsjob' has been stopped due to failure, rc=return code**

| **Explanation:** This message is used for automation of the Automation Agent Java Adapter via System Automation policy. SA Automation Status: ABENDED. The *return code* can be:

| 0 Success
| 4 abend posted by plugin
| 8 abend mainline

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **INGX9879I System Automation Agent Java Plugin with nickname 'nickname' cannot be invoked: *errortext*.**

| **Explanation:** The Automation Agent Java Framework tries to invoke a plugin with name '*nickname*'. However a plugin with matching nickname cannot be invoked because an error occurred. The *errortext* described the reason of the error.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Try to fix the problem if *errortext* gives you an indication. If necessary, contact IBM support Center.

| **INGX9880I System Automation Java Plugin 'nickname' cannot be loaded**

| **Explanation:** The Automation Agent Java Framework tries to reload the Java class representing the plugin with name '*nickname*'. However the class cannot be found.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** Adapt the classpath in the script ingjava accordingly.

INGX988II System Automation Java Plugin
'nickname' terminates adapter due to:
'description'

Explanation: The plugin encountered a problem that requires the adapter to terminate. The *'description'* describes the reason given by the plugin.

System action: Adapter terminates.

Operator response: None.

System programmer response: Try to solve the problem described by *'description'*.

INGX9901E INGXPluginLogger cannot initialize com.ibm.log.EEZLoggerJLog.

Explanation: An attempt to create a logger for the SA z/OS plug-in component was unsuccessful.

System action: No message or trace data is written to the logger output destination.

Operator response: None.

INGX9902I INGXPluginLogger has successfully been initialized using configuration file file from path path.

Explanation: The message and trace data of the SA z/OS plug-in will be written to the logger output destination.

System action: None.

Operator response: None.

INGX9904E Adapter plug-in initialization was unsuccessful.

Explanation: The initialization of the SA z/OS plug-in was not successful.

System action: Start up of plug-in is stopped.

Operator response: Installation or customization was erroneous, for example, the logger or the plug-in configuration file. Also check whether the automation adapter customization for NetView is correct.

INGX9905E Adapter function *function* failed due to exception from underlying component: *exception text*

Explanation: The specified adapter plug-in function was unsuccessful. An exception was caught from an underlying component.

System action: None.

Operator response: Read the original exception and follow its explanation.

INGX9906E Error message from SA z/OS: SA z/OS message

Explanation: Execution of a command using SA z/OS was unsuccessful. This message presents the extracted error message.

System action: None.

Operator response: For details about the extracted error message see *IBM Tivoli System Automation for z/OS Messages and Codes* or *Tivoli NetView for z/OS Messages and Codes*.

INGX9910I The command *cmd* will be performed for resource *res* on behalf of user *usr* with comment: *cmt*

Explanation: A user has issued a command that relates to a resource. This command has been received by the adapter and will be performed by automation on the indicated resource.

The variable *cmd* shows the end-to-end command that was issued against the adapter.

The variable *res* shows the name of the SA z/OS resources.

The variable *usr* shows the end-to-end operator name that issued the command.

The variable *cmt* shows the comment that was entered by the operator.

System action: The command has been received and will be executed for the resource.

Operator response: None.

INGX9911I The command *cmd* will be performed for system *sys* on behalf of user *usr*

Explanation: A user has issued a command that relates to a system. This command has been received by the adapter and will be performed by automation on the specified system.

The variable *cmd* shows the end-to-end command that was issued against the adapter.

The variable *sys* shows the name of the system that is involved with the command.

The variable *usr* shows the end-to-end operator name that issued the command.

System action: The command has been received and will be executed.

Operator response: None.

INGX9912I The command to move resource group *grp* away from system *sys* will be performed on behalf of user *usr* with comment: *cmt*

Explanation: A user has issued a request to move a resource group away from specified system. This

INGY1000I

command has been received by the adapter and will be performed by automation.

The variable *grp* shows the name of the group that is involved with the command.

The variable *sys* shows the name of the system that is involved with the command.

The variable *usr* shows the end-to-end operator name that issued the command.

The variable *cmt* shows the comment that was entered by the operator.

System action: The command has been received and will be executed.

Operator response: None.

INGX9913I The command to move resource group *grp* to the home system will be performed on behalf of user *usr* with comment: *cmt*

Explanation: A user has issued a command that relates to a resource. This command has been received by the adapter and will be performed by automation on the indicated resource.

The variable *grp* shows the name of the group that is involved with the command.

The variable *usr* shows the end-to-end operator name that issued the command.

The variable *cmt* shows the comment that was entered by the operator.

System action: The command has been received and will be executed.

Operator response: None.

INGX9914I The command to move resource group *grp* to system *sys* will be performed on behalf of user *usr* with comment: *cmt*

Explanation: A user has issued a command that relates to a resource. This command has been received by the adapter and will be performed by automation on the indicated resource.

The variable *grp* shows the name of the group that is involved with the command.

The variable *sys* shows the name of the system that is involved with the command.

The variable *usr* shows the end-to-end operator name that issued the command.

The variable *cmt* shows the comment that was entered by the operator.

System action: The command has been received and will be executed.

Operator response: None.

INGX9915I Command reset from non-recoverable error will be performed on resource *res* on behalf of user *usr*

Explanation: A user found a non-recoverable resource, which means the resource is not automated until a user issues the reset command. The reset command has been received by the adapter and will be performed by automation on the indicated resource.

The variable *res* shows the name of the SA z/OS resources.

The variable *usr* shows the end-to-end operator name that issued the command.

System action: The command has been received and will be executed for the resource.

Operator response: None.

INGY1000I *service* FAILED IN MODULE *module*, RC=*return_code*, REASON=*reasoncode* *error_info*

Explanation: An unexpected return code was retrieved from the service call mentioned in the message.

The variable *service* shows the name of the service that failed.

The variable *module* shows the name of the module where the error occurred.

The variable *return_code* shows the value of the return code that was issued.

The variable *reason_error_info* shows the value of the reason code that was returned and indicates the cause of the error.

Use Table 7 on page 293 to determine the cause of the error:

Table 7. Return and Reason Codes of INGY1000I

Module	Return Code	Reason Code	Explanation
INGPYENC	8	1	Incorrect parameter passed to the encoding routine.
INGPYENC	8	2	The generated instruction stream is on a higher level than allowed.
INGPYDEC	8	1	CRC value mismatch detected.
INGPYDEC	8	2	Incorrect table identifier found in the instruction stream.
INGPYDEC	8	3	Incorrect agent role/command combination found in the instruction stream.
INGPYDEC	8	4	Bad work item received: total length of work item is less than 0.
INGPXRPC INGPYRCV	0	20	REXX variable pool problem.
INGPXRPC INGPYRCV	0	24	A severe error occurred. See the netlog for further details.
INGPXRPC INGPYRCV	0	28	The sysplex communication manger is not available.
INGPXRPC INGPYRCV	0	2C	An abend occurred.
INGPOPPI	any	any	See the appropriate NetView manual where the service call that is mentioned in the message is described.

System action: Processing stops.

User response: Examine the return and reason codes from the service call and contact your system programmer.

**INGY1001I SEVERE ERROR DETECTED IN
MODULE *module*, REASON=*reason***

Explanation: An internal error has occurred.

The variable *module* shows the name of the module where the error occurred.

The variable *reason* shows the value of the reason code that was returned and indicates the cause of the error. It can have the following values:

- 1 An incorrect function code was specified for invoking the instruction builder.
- 2 An incorrect request code was specified for invoking the instruction builder.
- 3 The data contained in the passed data stem does not fit in the work buffer.
- 4 Incorrect parameters detected when the automation manager interface routine was called. The number of specified parameters exceeds the maximum number allowed.

System action: The request is canceled. A return code is sent to the program that issued the request.

User response: Examine the reason code for the possible cause of the error and contact your system programmer.

INGY1002I NOT ENOUGH STORAGE TO

HANDLE COMMAND/REQUEST

Explanation: A GETMAIN failure occurred.

System action: The command or process in progress when the error occurred terminates.

User response: None.

**INGY1003E ABNORMAL TERMINATION
abend_code, REASON=*abend_reason*,
MODULE=*module*,
FOOTPRINT=*footprint***

Explanation: A system service or instruction has failed.

The variable *abend_code* shows the value of the abend code.

The variable *abend_reason* shows the value of the abend reason.

The variable *module* shows the name of the module where the error occurred.

The variable *footprint* shows the information about the caller of the routine.

System action: The command or process in progress when the error occurred is terminated.

User response: None.

**INGY1004I AUTOMATION MANAGER CALL
FAILED. RC=*returncode*, REASON=*reason***

Explanation: An error occurred while communicating with the automation manager.

The variable *returncode* shows the value of the return code that was issued. It can have the following values:

- 4 The request was processed. However, the automation manager found minor problems with the work item composed for the request.
- 8 An instruction that is part of the work item is in error. The reason code provides more details about the cause of the error.
- 12 There is a severe problem with the work item. It did not pass the instruction interpretation phase in the automation manager.

The variable *reason* shows the value of the reason code that was issued. The reason code is for internal diagnosis only.

Reason Meaning

10E0 The configuration data set does not contain an entry for the system that the automation manager is running on.

See Appendix E, "Automation Manager Reason Codes," on page 459 for details.

System action: The instance terminates.

User response: Examine the netlog for additional messages with more details about the cause of the error. In case that return codes 8 or 12 were issued report the error to your IBM Support Center if necessary.

INGY1005I TOO MANY PARAMETERS SPECIFIED FOR FUNCTION *function*

Explanation: Too many parameters are specified for the function.

The variable *function* shows the name of the function that failed.

System action: The command or process in progress when the error occurred is terminated.

User response: Check whether WebSphere MQ is set up properly as a z/OS subsystem. See *WebSphere MQ for z/OS System Administration Guide* for further information. Also check whether your specifications in SA z/OS are correct. See HSAPRMxx for information about the automation manager part and INGXINIT for information about the SA z/OS NetView agent.

INGY1006I NO COMMUNICATION PATH AVAILABLE TO *destination*

Explanation: The communication with the designated automation manager could not be established.

The variable *destination* shows the name of the automation manager that the command was routed to.

System action: None.

User response: Wait until the designated automation

manager becomes available and reissue the command.

INGY1010E LICENSE MANAGER CALL *service*
FAILED, RC=*return_code*,
STATUS=*status_code*

Explanation: When SA z/OS was initializing, a license check for this instance of SA z/OS was performed. An error occurred and a correct license certificate could not be found. A combination of return code and status code can be used to determine the cause of the error.

The variable *service* shows the name of the service that failed.

The variable *return_code* shows the value of the return code that was issued.

The variable *status_code* shows the value of the status code that was returned.

System action: The SA z/OS initialization stops and SA z/OS terminates.

User response: Notify your system programmer.

System programmer response: Make sure that a valid license certificate for SA z/OS exists in the IBM License Manager database. If a valid license certificate exists and the error persists, contact your IBM Support Center.

INGY1011E NO LICENSE OBTAINED FOR SA z/OS VERSION 2

Explanation: When SA z/OS was initializing, a license check for Version 2 of SA z/OS was performed. A valid license certificate for Version 2 could not be found, or the IBM License Manager reported an error condition.

System action: The SA z/OS initialization stops and SA z/OS terminates.

User response: Notify your system programmer.

System programmer response: Make sure that a valid license certificate for SA z/OS exists in the IBM License Manager database. If a valid license certificate exists and the error persists, contact your IBM Support Center.

INGY1015E CONVERSION ERROR DETECTED. SEE INGDUMP DATA SET FOR DETAILS

Explanation: Routine INGPYDEC has detected a conversion error while decoding the buffer received from the automation manager. The buffer contains data that the INGPYDEC routine does not understand.

The INGPYDEC routine provides debugging information in the INGDUMP data set.

System action: SA z/OS continues ignoring the data that cannot be converted.

User response: Notify your system programmer.

System programmer response: Contact your IBM Support Center and provide the information in the INGDUMP data set.

INGY1020I Parameter *value* is not a valid *parameter*

Explanation: The specified parameter is invalid.

The variable *value* shows the value of the parameter. The value is invalid.

The variable *parameter* shows the name of the parameter.

System action: None.

Operator response: Correct the invalid parameter value.

System programmer response: None.

INGY1021I Required Parameter *parameter* missing.

Explanation: The named parameter is not specified.

The variable *parameter* shows the name of the parameter that is missing.

System action: None.

Operator response: Specify the missing parameter.

System programmer response: None.

INGY1022I REXX module *name* - Invalid syntax

Explanation: An SA z/OS REXX function detected a syntax error in its invocation. This error can result from extra operands being specified or expected operands that are missing.

The variable *name* is the name of the REXX routine that contains the error.

System action: The REXX function ends.

Operator response: Notify your system programmer that there are errors in the named REXX module.

System programmer response: Correct the statement in the REXX module.

INGY1023I REXX module *name* - Unknown variable *var_name*

Explanation: An SA z/OS REXX function detected a syntax error in its invocation. The specified variable name was not recognized.

The variable *name* is the name of the REXX routine that contains the error.

The variable *var_name* is the name of the variable that is unknown.

System action: The REXX function ends.

Operator response: Notify your system programmer that there are errors in the named REXX module.

System programmer response: Correct the statement in the REXX module.

INGY1024I REXX module *name* - Error occurred. Reason=*reason*

Explanation: An SA z/OS REXX function detected an error during processing.

The variable *name* is the name of the REXX routine that contains the error.

The variable *reason* indicates the cause of the error. It can have the following value:

- 1 Retrieval of the NetView common global that is associated with the variable failed.
- 2 SA z/OS is not fully initialized.

System action: The REXX function ends.

Operator response: Examine the reason code for the possible cause of the error and contact your system programmer if necessary.

System programmer response: For reason code 2, examine the NetView log for additional messages that explain why accessing the common global failed.

INGY1025I REXX module *name* - Unknown subsystem/job name *res_name*

Explanation: An SA z/OS REXX function detected a syntax error in its invocation. The specified subsystem name or its job name is not defined to SA z/OS.

The variable *name* is the name of the REXX routine that contains the error.

The variable *res_name* is the name of the resource. It can be a subsystem name or its associated job name.

System action: The REXX function ends.

Operator response: Notify your system programmer that there are errors in the named REXX module.

System programmer response: Correct the statement in the REXX module.

| INGY1090I Program *name* is not APF authorized

| **Explanation:** The SA z/OS discovery engine was invoked but the module named in the message does not reside in an APF authorized library. The discovery engine must be authorized in order to pickup sensitive data such as program name or program parameters from the various address spaces.

| Variable *name* holds the name of the program.

| **System programmer response:** Place the named module into an authorized library.

| **System action:** The discovery engine terminates.

INGY1097I Job *jobname* - STC*name* - User ended.
RC=*rc*

Explanation: This message indicates that a started task (STC) that was initiated by SA z/OS has ended. The return code informs SA z/OS of the result and acts accordingly. Additional messages may be issued to the netlog.

The variable *jobname* indicates the job name that this STC was assigned.

The variable *STCname* indicates the STC that was invoked.

The variable *User* indicates the user that this job was run under.

The variable *rc* indicates the return code returned by the job:

- 0 Okay
 - 1 There was a parameter mismatch
 - 3 A timeout occurred
- 10nn**
nn is the return code from the STC

System action: None.

Operator response: None.

System programmer response: None.

INGY1100I AN SA AUTOMATION MANAGER
TAKEOVER HAS BEEN REQUESTED

Explanation: The SA z/OS automation manager found an WebSphere MQaccess problem. A takeover to another automation manager instance was forced. Message INGY1103I shows the reason for the takeover.

System action: An SA z/OS automation manager takeover is triggered. However, if the automation manager runs on the last system within the XCF group the takeover request is ignored. Instead, a switch from WebSphere MQ to XCF communication will be performed.

User response: None.

INGY1101I FAILED MQ FUNCTION WILL BE
RETRIED

Explanation: A WebSphere MQ access problem has been detected, the failed function will be retried. See message INGY1103I for further information about the WebSphere MQ problem.

System action: The failed SA z/OS function is retried.

User response: None.

System programmer response: Examine preceding message INGY1103 or INGY1104 to understand the WebSphere MQ specific reason for the failure.

INGY1102I AUTOMATION STATE QUEUE IS
CORRUPTED, PROCESSING
CONTINUES

Explanation: A problem with the StateQueue has been detected. The SA z/OS automation manager now frees the StateQueue access. The queue is marked as GET/PUT disabled.

System action: The SA z/OS automation manager tries to continue its work without the StateQueue. However, severe problems may result in a takeover when the failed UOW ends. If the SA z/OS manager can continue, it will monitor the queue and try to reconnect when the queue is GET/PUT enabled again.

User response: Analyze the queue problem described in message INGY1103I. Redefine the StateQueue and mark it as GET/PUT enabled.

INGY1103I MQ FUNCTION *function* FAILED.
REASON = MQ*reason*'

Explanation: A WebSphere MQ function failed.

The variable *function* shows the function that failed.

The variable *reason* shows the reason code that indicates the cause of the error. See *WebSphere MQ for z/OS Application Programming Reference* for a description of the reason code.

System action: Depending on the severity and the SA z/OS instance having that problem it can be one of the following:

- The WebSphere MQ function may be retried.
- A manager takeover might be triggered.
- The SA z/OS instance terminates.

User response: When the instance terminates, analyze the WebSphere MQ problem and restart the failed instance.

INGY1104I SEVERE MQ LINKAGE PROBLEM.
REASON = *reason*

Explanation: This message indicates that WebSphere MQ is not set up correctly.

The variable *reason* shows the reason code that indicates the cause of the error.

System action: The instance terminates.

User response: See *WebSphere MQ for z/OS System Administration Guide* for further information. If you cannot resolve the problem contact your IBM Support Center.

INGY1105I MQ SUBSYSTEM *mqssid* NOT
DEFINED OR INVALID.

Explanation: This message indicates that WebSphere MQ is not set up correctly.

The variable *mqssid* shows which WebSphere MQ definition is missing or invalid.

System action: The instance terminates.

User response: Refer to *WebSphere MQ for z/OS System Administration Guide* (SC34-6585-00) for further information. If you cannot resolve the problem contact your IBM Support Center.

INGY1106I INVALID MAXMSGL SPECIFICATION FOUND FOR THE *queue*, SPECIFIED VALUE = *yourValue*, REQUIRED VALUE = *requiredValue*. PROCESSING CONTINUES

Explanation: SA z/OS found inconsistent definitions in the queue. This can happen when samples that are provided with SA z/OS are used that do not reflect the latest WebSphere MQ versions.

The variable *queue* shows the name of the queue that contains incorrect definitions.

The variable *yourValue* shows the value that is not correct.

The variable *requiredValue* shows the value that should be specified.

System action: The instance continues processing, but the overall processing performance might be impacted when the required value is less than the one you specified.

User response: Correct the length and rerun the queue definition jobs when applicable.

INGY1107I PROBLEM DETECTED WITH THE AUTOMATION STATE QUEUE. QUEUE REMOVED FROM PROCESSING

Explanation: This message is issued after INGY1102, when the manager was able to complete processing. However, the State Queue remains corrupted and is now removed.

System action: The automation manager frees the State Queue from all accesses. After that you are able to redefine the queue.

User response: Analyze the reason for the corruption and redefine the queue.

INGY1108I AUTOMATION STATE QUEUE INSERTED AGAIN

Explanation: This message is related to message INGY1107. It contains notification that the manager tried to reinstate the queue again. This will happen when the manager detects that the queues are marked GET/PUT enabled again.

System action: The manager tries to reinstate the State Queue again.

User response: None.

INGY1109I RECOVERY FOR THE *queue* HAS BEEN STARTED. ACTIONCODE = *action*

Explanation: The manager detected a Queue Full condition and decided to recover from that situation.

The variable *queue* shows the name of the queue that is full.

The variable *action* shows the recovery action that is taken.

System action: The automation manager automatically tries to resolve queue full situations. The following action codes can occur:

- 1 Query requests from the work item queue are deleted without any processing.
- 2 The manager stops the update process to avoid an overflow of the agent queue.
- 3 All agents are asked to enter a slowdown mode. That means only important requests are sent to the manager.
- 4 Is a combination of actions 1 and 3.
- 5 The agent queue will be determined and 'zombie' messages are deleted. 'Zombie' messages are orders and responses and agents that are currently not active.
- 6 Is a combination of actions 1 and 5.

User response: None.

INGY1111I WAITING FOR MQ QUEUE MANAGER *mqssid* TO COME UP

Explanation: Either the automation manager or one of its agents is waiting for the WebSphere MQ manager to come up. As long as is not running as an WebSphere MQ application, the automation manager cannot be restarted without possible loss of data.

The variable *mqssid* shows the subsystem ID of the WebSphere MQ Manager that SA z/OS tries to connect to.

System action: Processing continues.

User response: None.

INGY1112I WAITING FOR MQ PEER/LOCAL RECOVERY TO COMPLETE

Explanation: Takeover is waiting for WebSphere MQ peer/local recovery to complete.

System action: Takeover processing will actively poll WebSphere MQ to recognize when the WebSphere MQ peer recovery has been completed.

User response: None.

INGY1113I SEVERE MQ QUEUE RECOVERY PROBLEM, CANNOT CONTINUE

Explanation: During a takeover SA z/OS detected a severe WebSphere MQ related processing problem.

System action: The takeover cannot be completed HOT, and will be reset to WARM. However, processing continues.

User response: Analyze the reason of this problem using WebSphere MQ diagnostic means.

INGY1114I FOUND DEFECT WORKITEMS, CANNOT CONTINUE

Explanation: During a takeover SA z/OS detected that an automation manager request in the work item queue has been rolled back twice. It looks like a HOT takeover would again result in a termination.

System action: The takeover cannot be completed HOT, and will be reset to WARM. However, processing continues.

User response: Analyze the reason of the repeated termination of the manager.

INGY1120I Starting *modname*

Explanation: This message is written to indicate that the program has started.

The variable *modname* is the name of the module being started.

System action: None.

User response: None.

Classes: None.

Module: INGPUSMF

INGY1121I Ending *modname*

Explanation: This message is written to indicate that the program has ended.

The variable *modname* is the name of the module that has ended.

System action: None.

User response: None.

Classes: None.

Module: INGPUSMF

INGY1122I Unable to open the *filename* file. RC=*retcode*

Explanation: SA z/OS attempted to open the named file. The OPEN service failed.

The variable *filename* is the ddname of the data set.

The variable *retcode* is the return code from the OPEN call that failed.

System action: The INGPUSMF utility program ends.

User response: Examine the return code from the service call.

Classes: None.

Module: INGPUSMF

INGY1123I *filename* has unexpected record length of *nnn*

Explanation: While opening the named file, the utility program detected that the record length does not match the expected length of 80.

The variable *filename* is the ddname of the data set.

The variable *nnn* is the record length.

System action: The file that is in error is ignored and considered to be not present.

User response: Allocate the data set with the correct record length.

Classes: None.

Module: INGPUSMF

INGY1124I Specified user parameters

Explanation: This message documents the options that the INGPUSMF utility program works with.

System action: None.

User response: None.

Classes: None.

Module: INGPUSMF

INGY1125I Invalid parameter detected: *option*

Explanation: The INGPUSMF utility program encountered an invalid parameter option.

The variable *option* identifies the option that is invalid.

System action: The option that is in error is ignored.

User response: None.

Classes: None.

Module: INGPUSMF

INGY1126I All SMF records will be selected for processing

Explanation: This message indicates that no filtering option was specified when invoking the INGPUSMF utility program. Filter options are SYSID, FROM, and TO.

System action: None.

User response: None.

Classes: None.

Module: INGPUSMF

INGY1127I *nnnnn* records {read in | processed}

Explanation: This message records the number of records either read or processed.

The variable *nnnnn* is the number of records either read in or processed.

System action: None.

User response: None.

Classes: None.

Module: INGPUSMF

INGY1130I *debug_text*

Explanation: This is a generic informational message that shows debug information.

System action: Execution continues.

Operator response: None.

System programmer response: None.

INGY1131I PPI Receiver *name* is active.
TASK=*taskname*

Explanation: The task identified by its *name* has been activated.

System action: Execution continues.

Operator response: None.

System programmer response: None.

INGY1132I PPI Receiver *name* has terminated.
TASK=*taskname*

Explanation: The task identified by its *name* has been terminated.

System action: Execution continues.

Operator response: None.

System programmer response: None.

INGY1133E Command *command* to receiver failed
with RC=*retcode* - *description*

Explanation: The SA z/OS USS process exit routine or its corresponding receiver attempted to execute a NetView PPI command that has failed.

The variable *command* indicates in the CNMNETV function that failed.

The variable *receiver* is the name of the PPI receiver that experienced the error.

The variable *retcode* is the return code from the CNMNETV call.

The variable *description* gives an explanation of the error.

For detailed error information, see *NetView for z/OS Programming: Assembler*. In case of NetView Program-to-Program Interface (PPI) errors see *NetView for z/OS Application Programmer's Guide* for more information related to the service return code *rc*.

System action: Processing terminates.

Operator response: Report the error condition to your system programmer.

System programmer response: Report the problem to the SA z/OS service team for further diagnosis.

INGY1201E Unable to load NetView module
module-name

Explanation: The TWS command exit module supplied by SA z/OS was unable to load the named module for execution.

System action: The command exit module supplied by SA z/OS returns a code of 4 to TWS. This forces the operation into an error state with error code 'OAUT'. Processing continues.

Operator response: Determine why the exit could not load the module. Ensure that the NetView load libraries are in either the link list or the STEPLIB concatenation for the TWS Controller address space.

System programmer response: If the module name is CNMNETV, ensure that the NETVIEW.CNMLINK data set is either in the system link list or in the STEPLIB concatenation for the TWS Controller address space.

INGY1202E PPI Command *command-name* to
receiver-id failed with *error-text*

Explanation: The TWS command exit module supplied by SA z/OS attempted to execute a NetView PPI command that has failed.

command-name has the following values:

Command Name	Meaning
QUERYPPPI	PPI request 1, Query if PPI services are available
QUERYRCV	PPI request 2, Query the receiver ID status
SEND	PPI request 14, Send a buffer to a receiver ID

receiver-id is the name of the PPI receiver that experienced the error. This can be ignored for the QUERYPPPI command.

error-text gives an explanation of the error, as follows:

INGY1203I

Text	Meaning
Receiver is Disabled	RC=4. The receiver is not active. The receiver ID is defined but not active, so PPI has accepted a copy of the data buffer and queued it to the receiver ID queue.
PPI is Available	RC=10. The PPI is available to process requests. This is the result of a QUERYPPPI command.
Receiver is Active	RC=14. The PPI receiver ID specified is active. This is the result of a QUERYRCV command.
Receiver is Inactive	RC=15. The PPI receiver ID specified is inactive. This is the result of a QUERYRCV command.
Receiver already Active	RC=16. The PPI receiver ID specified is already active.
Receiver ECB not zero	RC=18. The ECB specified in the request is not zero. This occurs if the program does not clear the ECB before executing a RECEIVE command.
Request code not defined	RC=20. The request code is not defined.
Not in Primary Mode	RC=22. The program that issued this request is not running in PRIMARY mode. PPI uses z/OS Cross-Memory services so the program must be running in PRIMARY mode for it to be successful.
Not Authorized	RC=23. The program that issued this request is not authorized. If the receiver ID has been defined to accept buffers only from authorized programs, this error occurs if the sender program is not authorized.
PPI is not Available	RC=24. The PPI is not available. This is the result of a QUERYPPPI command.
ASCB ID is Invalid	RC=25. The ASCB address is not correct. In a RECEIVE command, the ASCB address is specified. If this does not match the address specified in the OPEN command this error occurs.
Receiver is not Defined	RC=26. The receiver program is not defined. This is the result of a QUERYRCV command.

Text	Meaning
No PPI Address Space	RC=28. The NetView SSI address space is running, but no PPI function is defined to it. The NetView SSI address space should be enabled to process PPI requests.
No Data in Queue	RC=30. The data buffer queue for the receiver ID is empty. This is the result of issuing a RECEIVE command but the buffer queue is empty.
Receive buffer is too small	RC=31. The receive buffer specified in the request is too small to hold the queued buffer. This occurs when the RECEIVE command specifies a buffer area that is too small to hold the queued data buffer. Re-issue the request specifying a larger buffer area.
No NetView Storage Available	RC=32. The PPI has run out of NetView Storage.
Buffer length is invalid	RC=33. The buffer length supplied in the request is invalid.
Receive Queue is full	RC=35. The receiver ID queue is full. This occurs for a SEND command when the receiver's buffer queue has reached its defined limit.
ESTAE cannot be established	RC=36. A recovery environment could not be established.
SENDER, RECEIVER IDs are invalid	RC=40. Either the sender or receiver ID is not valid.
Processing error has occurred	RC=90. An internal processing error has occurred.

System action: The command exit module supplied by SA z/OS returns a code of 4 to TWS. This forces the operation into an error state with error code 'OAUT'. Processing continues.

Operator response: Report the error condition to your system programmer.

System programmer response: Report the problem to the SA z/OS service team for further diagnosis.

INGY1203I Request failed for *adname-opnum* (*jobname*) on *wsname*

Explanation: This is an information message to describe the TWS function that incurred an error during processing. This error message is always issued in addition to error messages INGY1201 and INGY1202.

adname is the TWS Application Description.

opnum is the TWS Operation Number in the Application Description.

jobname is optional and is the TWS jobname that is associated with the Operation Number.

wsname is the TWS Workstation name that was processing the Operation at the time of the error.

System action: The command exit module supplied by SA z/OS returns a code of 4 to TWS. This forces the operation into an error state with error code 'OAUT'. Processing continues.

Operator response: No action is required. This message provides information that can be used to help diagnose the problem.

System programmer response: None.

INGY1300I *aa bb cc dd ee*

Explanation: The Job Log Monitoring has read a message from a log data set and has sent this message to NetView for automation.

This message is a multi-line message. The first line is a label line showing the following tokens:

The variable *aa* is the owner of the job. The value is 8 characters long and filled with blanks if necessary.

The variable *bb* is the name of the job. The value is 8 characters long and filled with blanks if necessary.

The variable *cc* is the job identifier. The value is 8 characters long and filled with blanks if necessary.

The variable *dd* is the ddname of the corresponding spooled data set. The value is 8 characters long and filled with blanks if necessary.

The variable *ee* is the message id that has been defined in the ISPF dialogs as one of the filter criteria. The value is up to 32 characters long and left-justified. The variable shows the value 'N/A' for all non-SA controlled jobs.

Each subsequent line represents a line of the original message.

System programmer response: Processing continues.

Operator response: None.

System programmer response: None.

INGY1301I **Job Log Monitoring is ready for requests.**

Explanation: The Job Log Monitoring task has been started and is now ready for processing requests.

System action: Processing continues.

Operator response: None.

System programmer response: None.

INGY1302I **Job Log Monitoring is being terminated.**

Explanation: Job Log Monitoring task has been stopped by a NetView stop command or during NetView termination. All opened data sets are closed and deallocated. And all monitoring information is deleted.

System action: The task is terminated.

Operator response: None.

System programmer response: None.

INGY1303I **Job Log Monitoring has been suspended.**

Explanation: The Job Log Monitoring task has been stopped on request. All opened data sets are closed and deallocated. The monitoring information is kept for restarting the task later.

System action: The task is terminated.

Operator response: None.

System programmer response: None.

INGY1304I **JES2 is not the primary job entry subsystem.**

Explanation: SA z/OS only supports Job Log Monitoring for JES2. Since the task may be started before any job entry subsystem is started the job entry subsystem check is delayed until the first request is received.

System action: The task is terminated.

Operator response: None.

System programmer response: Remove all definitions from Job Log Monitoring from your PDB and rebuild the PDB. This prevents SA z/OS from starting the task the next time when the configuration is being refreshed.

INGY1305I **Job Log Monitoring has been started for *uid-jnm-jid-ddn*.**

Explanation: The Job Log Monitoring task has allocated the spool data set that corresponds to indicated ddname and monitors new incoming messages. Note that empty messages lines are suppressed.

The variable *uid* is the owner of the job.

The variable *jnm* is the name of the job.

The variable *jid* is the job identifier.

The variable *ddn* is the ddname of the monitored spool data set.

System action: Processing continues.

Operator response: None.

System programmer response: None.

INGY1306I Job Log Monitoring has been stopped for uid-jnm-jid-ddn.

Explanation: The Job Log Monitoring task has removed the corresponding spool data set from monitoring. The data set has been closed and deallocated.

The variable *uid* is the owner of the job.

The variable *jnm* is the name of the job.

The variable *jid* is the job identifier.

The variable *ddn* is the ddname of the monitored spool data set.

System action: Processing continues.

Operator response: None.

System programmer response: None.

INGY1307I Error condition detected when processing uid-jnm-jid-ddn, RC=rc RSN=rsn.

Explanation: The Job Log Monitoring detected an error condition when it processes the spool data set that corresponds to the ddname. The return and reason codes describe the particular reason.

The variable *uid* is the owner of the job.

The variable *jnm* is the name of the job.

The variable *jid* is the job identifier.

The variable *ddn* is the ddname of the monitored spool data set.

The variable *rc* shows the return code:

8 Message retrieval failed.

12 Stop monitoring failed.

16 Start monitoring failed.

The variable *rsn* shows the reason code in hexadecimal characters. The first byte of the reason code describes the function, the second byte is the return code of the failing macro, and the last two bytes represent the reason code of the failing macro:

Table 8. First byte (function code) of the reason code of messages INGY1307I and INGY1308I

Function	Failing Macro	Terminate
'01'x	ESTAE	Yes
'11'x	STORAGE OBTAIN	Yes
'12'x	STORAGE RELEASE	Yes
'21'x	DYNALLOC ALLOCATE	Yes

Table 8. First byte (function code) of the reason code of messages INGY1307I and INGY1308I (continued)

Function	Failing Macro	Terminate
'22'x	DYNALLOC FREE	Yes
'31'x	GENCB ACB	
'32'x	GENCB RPL	Yes
'33'x	OPEN ACB	
'34'x	MODCB RPL	
'35'x	CLOSE ACB	Yes
'36'x	POINT RPL	
'37'x	GET RPL	
'41'x	GET CGLOBAL	Yes
'42'x	SET CGLOBAL	Yes
'51'x	RACROUTE AUTH	
'61'x	CREATE Name/Token	Yes
'62'x	DELETE Name/Token	Yes
'63'x	READ Name/Token	Yes

System action: The task is terminated depending on the function that was performed (see Table 8).

Operator response: None.

System programmer response: Correct the error condition that relates to the function specific return and reason codes. Restart the task if necessary.

INGY1308I Internal error condition detected. RC=rc RSN=rsn.

Explanation: The Job Log Monitoring could not initialize successfully.

The variable *rc* shows the return code.

The variable *rsn* shows the reason code. For a description of the reason codes, refer to Table 8.

System action: The task is terminated when *rc* shows a value of 12 or higher.

Operator response: None.

System programmer response: If the task is terminated contact your IBM Support Center.

INGY1309I Job Log Monitoring encountered abend SCC=scc RSN=rsn.

Explanation: The Job Log Monitoring encountered the indicated abend. The system completion and reason codes describe the abend condition.

| The variable *scc* shows the system completion code.

| The variable *rsn* shows the abend reason code.

| **System action:** After all data sets have been closed and deallocated the task is terminated.

| **Operator response:** None.

| **System programmer response:** When the cause of the abend is eliminated you may restart the task using the NetView command START.

| **INGY1310I Syntax error on *parm*.**

| **Explanation:** The command processor for Job Log Monitoring detected a syntax error.

| The variable *parm* shows the parameter in error.

| **CMD** A required parameter is missing. Or, an improper combination of parameters or an unknown parameter has been detected.

| **DDN** The keyword value is not a valid ddname.

| **INTERVAL**

| The monitoring interval is not in the range 1 to 3600.

| **JOBID** The job identifier is incorrect or too long.

| **JOBNM**

| The job name is incorrect or too long.

| **POLL** The polling time is not in the range from 1 to 3600.

| **OWNER**

| The name of the job owner is incorrect or is too long.

| **REQ** The request is not one of the following: START, STOP, STATUS, SUSPEND

| **System action:** The command is rejected.

| **Operator response:** Correct the error condition and reissue the command.

| **System programmer response:** None.

| **INGY1311I Application *jobnm(subsysnm)* will not be monitored, RSN=*rsn***

| **Explanation:** Monitoring is disabled for the indicated application because no filter criteria have been defined or the application does not run under control of JES2.

| The variable *jobnm* shows the job name of the application.

| The variable *subsysnm* shows the subsystem name of the application.

| The variable *rsn* shows one of the following reasons:

| **1** No filter criteria have been defined.

| **2** Application does not run under JES2 control.

| **System action:** Processing continues.

| **Operator response:** None.

| **System programmer response:** Correct the policy data, rebuild the ACF, and refresh the configuration data.

| **Note:** You need to define at least one filter criteria even if all messages are processed.

| **INGY1312I Command processing failed, RC=*rc* RSN=*rsn*.**

| **Explanation:** The command processor for Job Log Monitoring detected an internal error.

| The variable *rc* shows the return code of the internal error condition.

| The variable *rsn* shows the reason code of the internal error condition:

| **12** No filter criteria found for the job.

| **16** Job has not been defined for monitoring.

| **20** No filter criteria could be found.

| **System action:** The command is rejected.

| **Operator response:** In case of reason code 20 verify that the filter criteria have been defined and that the list INGRJLMP ran successfully.

| **System programmer response:** None.

| **INGY1313I Job Log Monitoring task is not active**

| **Explanation:** The command processor found the Job Log Monitoring task inactive.

| **System action:** The command is not processed.

| **Operator response:** Start the task and reissue the command.

| **System programmer response:** None.

| **INGY1314I *uid-jnm-jid-ddn* is already being monitored**

| **Explanation:** Job Log Monitoring does already monitor the spool data set that corresponds to the indicated ddname.

| The variable *uid* is the owner of the job.

| The variable *jnm* is the name of the job.

| The variable *jid* is the job identifier.

| The variable *ddn* is the ddname of the monitored spool data set.

| **System action:** The command is ignored.

| **Operator response:** None.

| **System programmer response:** None.

INGY1315I *uid-jnm-jid-ddn* has not been activated for monitoring.

Explanation: Job Log Monitoring has not been instructed to monitor the spool data set of the indicated data set.

The variable *uid* is the owner of the job.

The variable *jnm* is the name of the job.

The variable *jid* is the job identifier.

The variable *ddn* is the ddname of the monitored spool data set.

System action: The command is ignored.

Operator response: None.

System programmer response: None.

INGY1316I Automation environment has not been initialized.

Explanation: The Job Log Monitoring task detected that the automation environment has not been initialized. If the task was already active all monitoring is delayed until the automation environment is initialized.

System action: The task is terminated if it was started by the command.

Operator response: Retry the command after the automation environment has been initialized.

System programmer response: None.

INGY1317I Job *uid-jnm-jid* not found.

Explanation: The Job Log Monitoring task did not find the indicated job active or on the output queue.

The variable *uid* is the owner of the job.

The variable *jnm* shows the job name.

The variable *jid* shows the job identifier.

System action: The command is rejected.

Operator response: In case a parameter was misspelled, correct the value and reissue the command.

System programmer response: None.

INGY1318I Job Log Monitoring has been terminated for *uid-jnm-jid-ddn*.

Explanation: The Job Log Monitoring task did not find the job active, but on the output queue. After all messages have been processed the monitoring is terminated for the spool data set that corresponds to the indicated ddname.

The variable *uid* is the owner of the job.

The variable *jnm* is the name of the job.

The variable *jid* is the job identifier.

The variable *ddn* is the ddname of the monitored spool data set.

System action: Processing continues.

Operator response: None.

System programmer response: None.

INGY1319I No Job Log Monitoring definitions found.

Explanation: The Job Log Monitoring task did not find any monitoring definitions. However, because of the ACF information SysOps assumes that the monitoring definitions are available.

System action: The ACF-based request is rejected. Only manual monitoring requests are honored.

Operator response: None.

System programmer response: Check the NetView log and the policy database definitions as to why the task could not find any monitoring definitions. The clist INGRJLMP prepares and stores the definitions used by the monitoring task.

INGY1320I DDname *ddn* of *jnm/jid* not found

Explanation: The Job Log Monitoring task did not find the indicated ddname in JCL image of the job.

Variable *ddn* shows the ddname.

Variable *jnm* shows the job name.

Variable *jid* shows the job identifier.

System action: The command is rejected.

Operator response: In case one of the keyword values was misspelled, correct the value and reissue the command.

System programmer response: None.

INGY1321I Multiple job ids (*jid/jid*) exist for *jnm*.

Explanation: An INGJLM START | STOP command was issued without the JOBID parameter. However, the Job Log Monitoring function found at least two job ids for the specified job name. The first two job ids are presented in the message.

The variable *jnm* shows the job name.

The variable *jid* shows a job identifier.

System action: The command is rejected.

Operator response: Reissue the command using a specific job id.

System programmer response: None.

-
- | **INGY1322I** Multiple jobs (*uid*/*uid*) exist for *jnm*.
- | **Explanation:** An INGJLM START | STOP command
| was issued without the OWNER parameter. However,
| the Job Log Monitoring function found at least two jobs
| for the specified job name but with different owners.
| The owners of the first two jobs are presented in the
| message.
- | The variable *uid* shows an owner of the job name.
- | The variable *jnm* shows the job name.
- | **System action:** The command is rejected.
- | **Operator response:** Reissue the command using a
| specific job id.
- | **System programmer response:** None.

Messages ING001I to INGY1203I

Chapter 10. Messages ISQ001I to ISQ901I

ISQ001I *cmd_entity* CANNOT FIND
PARAMETER ENTITY OR ENTITY
NEEDS MORE PARAMETERS.

Explanation: The parameter entity specified with the command was not found in the ProcOps tables, or if it is defined additional parameter(s) are required to process the command successfully.

cmd_entities can be target system names with console designators SC or OC, or path names together with a console designator of 'S'. A valid *cmd_entity* is also an ensemble name, which has to be specified without a console designator.

System action: The command does not complete.

Operator response: If this message does not show a command name, see the NetView log if additional error messages are shown at the time of the error and contact your system programmer.

System programmer response: If the used parameter entity is not defined make sure it is defined using the SA z/OS Customization Dialogs. Use the Dialog Build function to generate a new ACF to be used with ProcOps. Reload the ACF and recycle ProcOps, then repeat the command.

Modules: ISQXLOC, ISQXUNL, ISQOVRD

ISQ003I SEVERE INTERNAL ERROR FOUND
BY *routine-name*.

Explanation: An internal error has occurred. After this message appears, other messages that contain information about the error are written to the NetView log.

System action: The command or process in progress when the error occurred does not complete.

Operator response: Write down the routine name shown in this message. Also, save or print a copy of the NetView log, and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ004I INDEX TO *target-table* IN *source-table*
ENTRY *entry-location*.

Explanation: This message may be preceded by message ISQ003I. It contains internal error information and is written to the netlog. This message is for IBM service use only.

System action: SA z/OS processing continues. Other messages indicate whether the command or process associated with this message succeeded or failed.

Operator response: Write down the information shown in this message. Also, save or print a copy of the netlog, and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ005I UNEXPECTED RETURN CODE
return-code FROM *program*.

Explanation: This message may be preceded by message ISQ003I. It contains internal error information and is written to the netlog. This message is for IBM service use only.

System action: SA z/OS processing continues. Other messages indicate whether the command or process associated with this message succeeded or failed.

Operator response: Save or print a copy of the netlog and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ006I MSGID: *message-identifier* TASK:
task-name COMMAND: *command-name*
CSECT: *module-name*.

Explanation: This is a netlog entry. It appears only if the DEBUG parameter of the ISQSTART command was specified.

System action: None.

Operator response: Follow the directions for the message identified within this message.

System programmer response: None.

ISQ011I LOSS OF CONSOLE LOCK [TO
task-name].

Explanation: You no longer own the lock for the port that you issued the ISQSEND command to. Another task took the lock for the console using the ISQOVRD command while the ISQSEND command was running. Because of this, the results of issuing the ISQSEND command are unpredictable.

If another task currently owns the lock for the port, it is identified by the variable *task-name* in the message. If no task currently owns the port, no *task-name* variable appears in the message.

System action: SA z/OS processing continues. Other messages indicate whether the command or process associated with this message succeeded or failed.

Operator response: Determine which task should

ISQ012I • ISQ021I

properly own the lock for the console and establish this ownership.

System programmer response: None.

ISQ012I **INCORRECT SYNTAX FOR**
command-name **COMMAND.**

Explanation: The command identified in this message was not entered correctly.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command after checking to be sure its syntax is correct.

System programmer response: None.

ISQ013I *command-name*: **EITHER TARGET AND**
CONSOLE OR PATH AND
CONNECTION REQUIRED.

Explanation: The command you entered (*command-name*) requires that you specify either a path name and a connection character or a target name and a console identifier.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, including all needed parameters.

System programmer response: None.

ISQ014I *command-name*: **PARAMETER NUMBER**
argument-number **IS problem.**

Explanation: The command argument entered (identified by *argument-number*) is not valid, for one of several reasons. The variable *problem* identifies the reason why the argument is not valid. Possible reasons are:

- There are too many characters in the argument.
- The argument is not a valid option for the command.
- A value that is not acceptable was entered for the argument. (In this case, the *problem* variable lists the acceptable values.)

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command after checking to be sure its syntax is correct.

System programmer response: None.

ISQ015I *command-name*: *entity-name* **IS NOT A**
VALID ENTITY NAME.

Explanation: As part of the command *command-name*, the entity name you entered is not valid. Entity names can be target system, ensemble or target hardware

names. Refer to the command help to learn which entity types are valid for *command-name*.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, making sure that the entity name is correct and that it identifies a defined target system, ensemble or target hardware name.

System programmer response: None.

ISQ016I *command-name*: *console-ID* **IS NOT A**
VALID CONSOLE IDENTIFIER.

Explanation: As part of the command *command-name*, you entered a console identifier (*console-ID*) that is not valid. The only valid console identifiers are:

- SC
- OC
- SCA
- SCB
- OCA
- OCB

Not all of these console identifiers are valid for all commands.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, specifying a valid console identifier.

System programmer response: None.

ISQ017I *command-name* **COMPLETED**
SUCCESSFULLY.

Explanation: Processing of the command completed successfully.

System action: None.

Operator response: None.

System programmer response: None.

ISQ021I **TARGET PATH** *target-name* **port HAS**
NOT BEEN INITIALIZED

Explanation: This message appears in response to the ISQEXEC command. The specified target system *target-name* was not initialized.

System action: The command terminates before it is complete.

Operator response: Initialize the target system using command ISQXIII, then repeat the ISQEXEC command.

System programmer response: None.

ISQ022I *command-name* **COMMAND HAS BEEN QUEUED FOR PROCESSING BY TASK**
task-name.

Explanation: This message appears in response to the ISQEXEC command. A user automation command (*command-name*) was sent to a target control task (*task-name*) for processing.

System action: None.

Operator response: None.

System programmer response: None.

ISQ023I **No command entered in** *command-name*
request.

Explanation: This message appears in response to a command. The command was issued without any *Command* parameter specified.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, specifying a value for the *command* parameter.

System programmer response: None.

ISQ024I *command-name* **PROCESSING UNDER TASK**
task-name **WAS SCHEDULED BY**
initiating-task.

Explanation: This message is written to the NetView log as part of an audit trail when the DEBUG parameter of the ISQSTART command has been specified. This message is for IBM service use only.

System action: None.

Operator response: None.

System programmer response: None.

ISQ025I *task- name* **AND SUBSEQUENT TASKS ARE NOT STARTED.**

Explanation: An attempt was made to start a NetView task but no confirmation message was received within the required time.

The variable *task- name* shows the name of the task that did not start.

System action: SA z/OS processor operations initialization continues but function and/or performance may be limited. No attempt will be made to start other tasks in the same group. For example, if ISQBT003 is indicated as the task that failed to start no attempt will be made to start task ISQBT004 and higher.

Operator response: Take a note of the details and contact your system programmer.

System programmer response: Ensure that the

NetView operator and task definitions built from the SA z/OS dialogs have been included into DSIOPF and DSIDMN respectively. Further messages may be available in the NetView log if processor operations is started with the DEBUG option on ISQSTART.

ISQ026I *value-passed* **IS NOT A VALID**
type-of-value **VALUE FOR** *command-name.*

Explanation: This message is written to the NetView log and is for IBM service use only.

System action: SA z/OS processing continues. Other messages indicate whether the command or process associated with this message succeeded or failed.

Operator response: Save or print the NetView log, and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ027I **TEXT IS MISSING FOR ISQ**
message-number **MESSAGE.**

Explanation: No message text exists for a message entry. Message ISQ027I is issued in place of the message that text is missing for.

System action: None.

Operator response: Contact your system programmer. Find the message that text is missing for in this manual or the online help and follow the operator response instructions given for it.

System programmer response: Make sure the messages library (ISQ.ISQMLIB) is concatenated to the NetView data sets. Check the appropriate DSIIISQxx message member, where xx are the first two digits of the message number. If the text is missing, edit the member.

ISQ028I **ENTRY COUNT DISCREPANCY BETWEEN INTERESTED OPERATOR LIST AND INDEX.**

Explanation: This message is written to the NetView log and is for IBM service use only.

System action: SA z/OS processing continues. Other messages indicate whether the command or process associated with this message succeeded or failed.

Operator response: Save or print the NetView log, and contact your system programmer.

System programmer response: If this problem recurs after SA z/OS is restarted, contact an IBM support representative.

ISQ030I CANNOT RUN *command-name* UNTIL PROC-OPS HAS STARTED.

Explanation: Processor operations must be started before this command can be executed. If *command-name* variable is blank, this message was issued from internal ProcOps routines that requires Processor Operations to be started. If the *command-name* variable contains a command name, the ISQCHK command was executed with the *command-name* as parameter. See command ISQCHK for more information.

System action: The command is not processed.

Operator response: Start processor operations by issuing the ISQSTART command.

System programmer response: None.

ISQ031I TARGET HARDWARE *target-hardware-name* TYPE IS NOT VALID.

Explanation: This message is written to the netlog and is for IBM service use only.

System action: SA z/OS processing continues. Other messages indicate whether the command or process associated with this message succeeded or failed.

Operator response: Save or print the NetView log, and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ032I COMMAND *command_name* CANNOT BE EXECUTED. REASON CODE: *rsn*

Explanation: This message appears in response to an ISQCCMD or ISQECMD command. The associated reason code contains the error condition, as follows:

- 1 The command has been identified as using the ProcOps communication paths SNMP, HTTP, or TCP, but the required SA Subtower statement PROCOPS is not defined in the NetView style sheet.
- 2 The command has been identified as using the ProcOps communication paths SNMP, HTTP, or TCP, but ProcOps is not active.
- 3 The command has been identified as using the SA z/OS BCPii communication path, but the connection is either not defined correctly or has not started.

System action: The ISQCCMD or ISQECMD command is rejected.

Operator response: Initiate that the error is corrected and reenter the command.

System programmer response: For reason code 1, make sure there is an SA.TOWER statement for

PROCOPS available in your NetView style sheet definitions, before you start ProcOps.

For reason code 2, make sure that ISQSTART is processed before issuing an ISQCCMD or ISQECMD common command. If you expect this command to use the BCPii path, make sure that the processor and system definitions are correctly defined for an INTERNAL (BCPii) connection.

For reason code 3, make sure SA z/OS is fully initialized. If it is fully initialized, use the SA z/OS test command HSAET32 STATUS to validate that the BCPii services are enabled. If they are enabled, verify that the processor and system definitions are correctly defined for an INTERNAL (BCPii) connection in your SA z/OS policy database.

ISQ033I THE INTERESTED OPERATOR LIST FOR *target-system-name console-ID* IS FULL.

Explanation: This message appears in response to the ISQXMON and ISQXIII commands. A console can have a maximum of 12 entries in its interested operators list. The list for this console is currently full.

System action: If this message appears in response to the ISQXMON command, processing of the command ends before it is complete and no operator is added to the interested operators list. If this message appear in response to the ISQXIII command, processing continues.

Operator response: Determine whether or not all operators currently on the interested operators list need to be there. Reenter the command after someone has left the list.

System programmer response: None.

ISQ034I THE SYSTEM LIMIT OF *xxx* INTERESTED OPERATORS WAS REACHED.

Explanation: This message appears in response to the ISQXMON and ISQXIII commands. The maximum number of interested operators is currently defined. No operators can be added to any interested operators list.

The parameter *xxx* describes the number of reached operators.

System action: If this message appears in response to the ISQXMON command, processing of the command ends before it is complete and no operator is added to the interested operators list. If this message appears in response to the ISQXIII command, processing continues.

Operator response: Determine whether or not any operator IDs can be taken off the list. Reenter the command after modifying the list.

System programmer response: None.

ISQ035I *number-of-arguments* ARGUMENTS ARE
REQUIRED ON THE *command-name*
COMMAND.

Explanation: The specified command that does not have the correct number of arguments.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, making sure that it is correct.

System programmer response: None.

| **ISQ036I** *operator-name* NOW HAS INTERESTED
| OPERATOR LIST STATUS *status* FOR
| *entity-name*.

| **Explanation:** The operator has been added to or
| removed from the list of interested operators to receive
| messages from *entity-name*.

| The variable *operator-name* is the name of the
| operator issuing the ISQXMON command.

| The variable *status* can be one of the following:

- | – ON indicating operator has been added to the list
- | – OFF indicating operator has been taken from the list.

| The variable *entity-name* can be a valid:

- | – Ensemble Name
- | – Target System Name

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

ISQ037I *command-name* HAS COMPLETED
WITH A RETURN CODE OF *return-code*.

Explanation: This is a netlog trace message that appears when the DEBUG parameter of the ISQSTART command specified. This message is for IBM service use only.

System action: None.

Operator response: None.

System programmer response: None.

ISQ041I *ccmd* IS NOT SUPPORTED BY
ISQSEND, USE ISQCCMD.

Explanation: The common command *ccmd* is not supported by the ISQSEND command. It is supported by the ISQCCMD command.

System action: The ISQSEND command is rejected.

Operator response: Reenter the command using ISQCCMD.

System programmer response: None.

ISQ042I *command-name: console-type* console has
not been defined for target system
target-system-name.

Explanation: Processing of the command ends before it is completed because the specified console was not defined for the target system. The variable *console-type* shows the type of console that needs to be defined. Possible values for console-type are:

- Active operating system
- Active system
- Backup operating system
- Backup system
- An MVS system
- A VM

System action: Processing of the ISQXCLS command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Make sure the console for the target system is defined in the SA OS/390 configuration dialogs

ISQ044I DIAGNOSTIC TRACE MESSAGE:
trace-message.

Explanation: This is a NetView log trace message appears when debugging is active for SA z/OS. Debugging is activated when the ISQSTART command is issued with the DEBUG option specified.

System action: None.

Operator response: None.

System programmer response: None.

ISQ052I Command *command-name* failed.

Explanation: Processing of the command identified in this message ends before it is complete. Other messages provide additional information about the problem.

System action: Processing of the command ends before it is complete.

Operator response: Try to determine the cause of the failure, using information from the messages that appear. Reenter the command. If the problem persists, contact your system programmer.

System programmer response: Refer to *IBM Tivoli System Automation for z/OS User's Guide* and to *IBM Tivoli System Automation for z/OS Operator's Commands* for information about the command that failed. Use this information and message information to determine the cause of the failure.

ISQ059I ALL LPARS ARE ALREADY ACTIVE
ON SYSTEM *target-hardware-name*.

Explanation: This message appears in response to a command to initialize a target system running on a logically partitioned target hardware. All logical partitions on the target hardware are active. No additional target systems can be initialized for this target hardware.

System action: Processing of the command ends before it is complete.

Operator response: Close one of the target systems that are active on the target hardware and reenter the command.

Caution: If you are using the ISQXIPL command or the ISQCCMD ACTIVATE command with the FORCE option to initialize a target system, you may disrupt a target system that is already active on the target hardware.

System programmer response: None.

ISQ061I LOCK STACK {OVERFLOW |
UNDERFLOW} IN PROGRAM
program-name.

Explanation: An internal error has occurred. This message is for IBM service use only.

System action: Processing of the internal command ends before it is complete.

Operator response: Record the information in this message and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ062I TARGET *system-name* HAS BEEN
INITIALIZED.

Explanation: This message appears in response to the ISQXIII command to initialize a target system. Processing of the command completed successfully.

This does not mean necessarily that the connection to the SE/HMC of the processor for the target system itself was successfully established.

For target systems, a ProcOps status of INITIALIZED means that in case of communication problems during ISQXIII processing, ProcOps will automatically retry to establish a connection. This depends on the setting of a POLL FREQUENCY value in SA customization dialogs for the processor of this target system.

System action: None.

Operator response: None.

System programmer response: None.

ISQ063I TARGET *system-name* IS INITIALIZED
ALREADY.

Explanation: This message appears in response to the ISQXIII command to initialize a target system. The target system is already initialized.

System action: Processing of the command ends before it is complete. The target entity is not re-initialized.

Operator response: None.

System programmer response: None.

ISQ064I ENSEMBLE *ensemble_name*
COMMUNICATION INITIALIZED.

Explanation: This message appears in response to the ISQXIII command for the specified *ensemble_name*. Processing of the command completed successfully.

This does not necessarily mean that the connection to the ensemble HMC was successfully established.

For ensemble HMC connections, a status of CONNECTING means that in case of communication problems with the ensemble HMC, ProcOps will automatically retry to establish a connection. This depends on the setting of a POLL FREQUENCY value in the SA customization dialogs for the ensemble.

System action: None.

Operator response: None.

System programmer response: None.

ISQ065I ENSEMBLE *ensemble_name* IS ALREADY
INITIALIZED.

Explanation: This message appears in response to a ISQXIII command to initialize an ensemble. The ensemble connection is already active.

System action: Processing of the command ends before it is complete. The connection is not re-initialized.

Operator response: None.

System programmer response: None.

ISQ066I ENSEMBLE *ensemble_name*
COMMUNICATION IS NOW CLOSED.

Explanation: This message appears in response to the ISQXCLS command for the specified *ensemble_name*. Processing of the command completed successfully.

For ensemble HMC connections, a status of CLOSED is set. In case of any communication problems during ISQXCLS processing, ProcOps will always set this status and reset the communication on the ProcOps side.

System action: None.

- | **Operator response:** None.
 | **System programmer response:** None.

ISQ067I *PF-key* **IS NOT A VALID FUNCTION KEY. PLEASE SELECT ANOTHER.**

Explanation: This message appears when you press a function key that is not defined for the current status panel.

System action: Processing of the command continues.

Operator response: Select a function key from the list of valid keys at the bottom of the status panel.

System programmer response: None.

-
- | **ISQ068I** **ENSEMBLE** *ensemble_name*
 | **COMMUNICATION IS ALREADY**
 | **CLOSED**

| **Explanation:** This message appears in response to the ISQXCLS command for the specified *ensemble_name*. Processing of the command completed successfully.

| The ensemble HMC connection already has a status of CLOSED.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

ISQ069I **Cursor is not on a valid** *object*.

Explanation: You tried to select an option on a status panel, but the cursor was not positioned so that an option could be selected.

System action: Processing of the command continues.

Operator response: Move the cursor to the line that contains what you want to select and press the PF key again.

System programmer response: None.

ISQ070I **UNABLE TO LOCATE** *data-identifier*.

Explanation: This message appears in response to an attempt to display a status panel. The panel cannot be displayed, due to an internal error.

System action: Processing of the command continues.

Operator response: Save or print a copy of the NetView log and contact your system programmer.

System programmer response: Contact an IBM support representative.

| **ISQ071I** *fname* **zBX BASE OPERATION PANEL**
 | **FUNCTION IS COMPLETE**

| **Explanation:** This message appears in response to a completed zBX operation panel function. The variable *fname* contains the panel function operation name.

| **System action:** ISQXDST panel ISQEZBX processing continues.

| **Operator response:** None.

| **System programmer response:** None.

| **ISQ072I** *fname* **zBX BASE OPERATION PANEL**
 | **FUNCTION NOT EXECUTED**

| **Explanation:** This message appears in response to a rejected zBX operation panel function. The variable *fname* contains the panel function operation name. The zBX object at the panel cursor position is not valid for the function.

| **System action:** ISQXDST panel ISQEZBX processing continues.

| **Operator response:** Reposition the cursor and repeat the function.

| **System programmer response:** None.

| **ISQ073I** **TARGET HARDWARE** *thw_node*
 | **ENSEMBLE NODE INFORMATION**
 | **NOT VALID FOR zBX**

| **Explanation:** The ISQXDST ensemble view panel ISQEENS is displayed and the PF2 key was pressed to invoke the zBXCntl function for an ensemble THW node. The panel invocation failed for one of the following reasons:

- | • The THW node name could not be located in the ProcOps THW table and was also not found in the THW node list of the ensemble
- | • The address of the dynamically allocated buffer with the zBX information was zero.

| **System action:** The zBXCntl panel ISQEZBX is not displayed, ISQXDST processing continues.

| **Operator response:** Inform your system programmer.

| **System programmer response:** Make sure the SA policy definitions for the ensemble and processors are correct. Rebuild the ACF and recycle ProcOps. Look for diagnostic ISQ511I messages in the netlog happening during the ISQSTART processing. If the problem persists, contact IBM support.

| **ISQ074I** **CONNECTION TO ENSEMBLE**
 | *ens_name* **IS NOT ESTABLISHED**

| **Explanation:** The ISQXDST ensemble view panel ISQEENS is displayed and PF2 key was pressed to invoke the zBXCntl function for an ensemble THW

| node. The panel invocation failed for the following
| reason:

| The connection to the ensemble HMC of ensemble
| *ens_name* is not active.

| **System action:** The zBXCntl panel ISQEZBX is not
| displayed, ISQXDST processing continues.

| **Operator response:** From the ISQEENS panel press
| PF4 to establish the ensemble connection (ISQXIII).
| Wait until the connection status changes to ACTIVE,
| then press PF2 again. If the connection status does not
| change to ACTIVE, look for additional messages in the
| netlog and inform your system programmer.

| **System programmer response:** If additional messages
| are available from the netlog, look if they indicate a
| possible network or SA policy definition problem.
| Correct the problem and retry the ISQXDST request.

ISQ075I SCHEDULING OF COMMAND
command_name ENDED WITH RC *cmd_rc*

Explanation: The command *command_name* issued per
command character from the ISQXDST target system
summary panel for a selected target system, was
scheduled for execution on the target system's control
task. The scheduling ended with a return code of
cmd_rc. When *command_name* completes, the target
system status will be updated.

System action: None.

Operator response: None.

System programmer response: None.

ISQ076I LAST TARGET IS ALREADY
DISPLAYED; CANNOT SCROLL
DOWN.

Explanation: This message appears when you press
the PF8 key to scroll the screen down and there are no
more target systems to display.

System action: The request to scroll the screen is
ignored.

Operator response: None.

System programmer response: None.

ISQ077I ALREADY AT THE TOP; CANNOT
SCROLL UP.

Explanation: This message appears when you press
the PF7 key to scroll the screen up and there are no
previous target systems to display.

System action: The request to scroll the screen is
ignored.

Operator response: None.

System programmer response: None.

ISQ078I PATH *path-name* IS NOW CLOSED.

Explanation: This message appears in response to a
command to close a port. The command was processed
successfully and the port is now closed.

System action: None.

Operator response: None.

System programmer response: None.

ISQ079I COMMAND *command_name* ENDED
WITH RC *cmd_rc*

Explanation: The command *command_name* issued per
command character from the ISQXDST target system
summary panel for a selected target system, ended
with a return code of *cmd_rc*.

System action: None.

Operator response: None.

System programmer response: None.

ISQ080I *command-name* MAY TAKE SEVERAL
MINUTES.

Explanation: The command you entered normally
takes up to several minutes to be processed.

System action: Processing of the command continues.

Operator response: None.

System programmer response: None.

ISQ081I RESOURCE DISCOVERY OF *thw-name*
ZBX IS NOW STARTING

| **Explanation:** The ensemble connection to manage the
| zBX resources of *thw-name* has been established and the
| automatic discovery of the zEnterprise® Blade
| Extension resource begins. This process may take
| several minutes. The message is part of the ISQXIII
| processing of an ensemble and is issued for each
| *thw-name* ensemble node that has a zBX attached.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

ISQ082I RESOURCE DISCOVERY OF *thw-name*
ZBX IS NOW COMPLETE

| **Explanation:** The ensemble connection to manage the
| zBX resources of *thw-name* has been established and the
| automatic discovery of its zEnterprise Blade Extension
| resources is now complete. This message follows
| message ISQ081I. Several minutes may pass between
| the two messages.

| **System action:** None.

- | **Operator response:** None.
 | **System programmer response:** None.

| **ISQ087I** **ENSEMBLE** *ensemble_name*
 | **CONNECTION WAS NOT ACTIVE.**

- | **Explanation:** One of the following conditions exists:
- | • The command you issued did not complete because the connection to the ensemble named in the message is not active.
 - | • The command you issued to activate the connection to the ensemble identified in the message did not complete successfully.

| **System action:** Processing of the command ends before it is complete.

| **Operator response:** If the command you issued did not complete because the ensemble connection was not active, issue an ISQXIII command to activate the connection and then reenter the original command.

| If the command you issued to activate the ensemble connection failed, see the message that is displayed preceding or following this one to learn why the command failed.

| **System programmer response:** None.

ISQ088I **CONNECTION TO TARGET**
HARDWARE *target-hardware-name* **NOT**
ESTABLISHED.

Explanation: This message appears in response to an ISQCCMD common command request with a target hardware name as parameter. There is no connection established to this target hardware and the common command is not executed.

System action: None.

Operator response: In order to establish a connection to the target hardware, at least one target system running on the target hardware must have been initialized. To perform the necessary initialization use the ISQXIII command, followed by a valid target system name for this target hardware. This establishes a connection to the target hardware and then the ISQCCMD common command request can be retried.

System programmer response: None.

ISQ089I **TARGET** *target-system-name* **HAS NOT**
BEEN INITIALIZED.

- Explanation:** One of the following conditions exists:
- The command you issued did not complete because the target system identified in the message is not initialized.
 - The command you issued to initialize the target system identified in the message did not complete successfully.

System action: Processing of the command ends before it is complete.

Operator response: If the command you issued did not complete because the target system is not initialized, issue an ISQXIII command to initialize the target system and then reenter the original command.

If the command you issued to initialize the target system failed, see the message that is displayed preceding or following this one to learn why the command failed.

System programmer response: None.

ISQ090I **THE MESSAGE MONITOR TASKS**
COULD NOT BE STOPPED.

Explanation: This message appears when the ISQSTOP command is unable to stop the processor operations message monitor tasks.

System action: Processing of the ISQSTOP command continues.

Operator response: This is an informational message only. It may, however, indicate that additional errors may follow.

System programmer response: This message may indicate that the operator task or autotask that the operator ISQSTOP command was invoked on is not allowing processor operations OST tasks in the NetView address space sufficient CPU time to free their resources and to terminate. If possible, ensure the ISQSTOP command is issued from tasks that receive limited message traffic, DOM's, and automation. This message is more likely to be received on a focal-point system that is running on a uniprocessor or an LPAR on a multiprocessor that is defined to run with only one processor.

ISQ091I **PROC-OPS SHUTTING DOWN. DATA**
MAY NOT BE CURRENT. PLEASE
EXIT.

Explanation: This message appears on a SA z/OS status panel. SA z/OS is shutting down as a result of the ISQSTOP command. Data displayed on the panel may not be accurate.

System action: Processing of the ISQSTOP command continues.

Operator response: Press a function key to exit from the status panel.

System programmer response: None.

ISQ092I **PROC-OPS HAS STOPPED.**

Explanation: This message appears when you try to display a status panel after processor operations has been shut down. It is also displayed when you exit

from a status panel after processor operations has been shut down.

System action: None.

Operator response: None.

System programmer response: None.

ISQ095I *message-ID* **AUTOMATION FAILED.**
error-description **ERROR.**

Explanation: The automation routine that should normally be processed in response to the message *message-ID* was not processed. The variable *error-description* contains text describing the type of error that occurred.

System action: The automation routine is not processed.

Operator response: Save or print the NetView log and contact your system programmer.

System programmer response: If the *error-description* variable contains the words "message text error" check the text of the message that automation failed for in the NetView log. Make sure that the message table is correct and that none of the original entries have changed. If you still cannot locate the source of the problem, contact an IBM support representative.

If the *error-description* variable contains any other text, contact an IBM support representative.

ISQ100I **UNABLE TO OBTAIN STORAGE FOR INTERNAL TABLES.**

Explanation: The amount of storage required for processor operations internal tables cannot be allocated.

System action: Processing of the command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Increase the amount of storage available, by increasing NetView storage allocations. If this does not solve the problem, contact an IBM support representative.

ISQ101I **UNABLE TO ALLOCATE DATA SET**
data-set-name.

Explanation: The option file that was specified in the ISQSTART command cannot be allocated. See the preceding NetView messages for details. You should only specify an option file should when advised to do so by IBM support.

System action: Processing of the command ends before it is complete.

Operator response: Determine whether the specified data set name is valid. If it is, correct the problem with the option file data set.

System programmer response: None.

ISQ102I **PROC-OPS IS ALREADY STARTED.**

Explanation: The ISQSTART command was issued with an incorrect start parameter that was specified when processor operations was already running. If processor operations is already running, you can only use ISQSTART paramter DEBUG to initiate a warm start in debug mode.

System action: Processing of the command ends before it is complete.

Operator response: If you want to restart processor operations with new configuration data, issue the ISQSTOP command and then issue ISQSTART with the ACF parameter.

If you want to restart processor operations with the same configuration, issue the ISQSTART command without the ACF parameter to initiate a warm start.

System programmer response: None.

ISQ103I **WARM START ATTEMPTED WHEN PROC-OPS WAS NOT ACTIVE. COLD START REQUIRED.**

Explanation: The ISQSTART command was issued without a configuration parameter when processor operations was not active. If processor operations is not active, the ISQSTART command must include the ACF parameter.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the ISQSTART command, including the ACF parameter.

System programmer response: None.

ISQ104I **ERROR FREEING TABLE** *table-name.*
TABLE IS NOT FREED.

Explanation: This message is a response to the ISQSTOP command. Due to an error the command did not complete successfully.

System action: Processing of the ISQSTOP command ends before it is complete.

Operator response: Record the information in this message and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ106I **COLD START PROCESSING STARTED.**

Explanation: This message appears in response to an ISQSTART command that includes the ACF parameter. Processing of the command completes successfully and

processor operations continues to start.

System action: None.

Operator response: None.

System programmer response: None.

**ISQ107I WARM START PROCESSING
STARTED.**

Explanation: This message appears in response to an ISQSTART command that does not include the ACF parameter. Processing of the command completed successfully and the processor operations warm start continues.

System action: None.

Operator response: None.

System programmer response: None.

ISQ108I EMPTY DATA SET *data-set-name*.

Explanation: The option file that was specified in the ISQSTART command does not contain any data records.

System action: Processing of the command ends before it is complete.

Operator response: Determine whether the specified data set is valid and contains appropriate XML control records. You should only specify an option file when advised to do so by IBM support.

System programmer response: None.

**ISQ109I DATA SET *data-set-name* IS NOT A
VALID XML FILE.**

Explanation: The data set that was specified in the ISQSTART command does not contain XML style data.

System action: Processing of the command ends before it is complete.

Operator response: Determine whether the specified option file is valid and contains appropriate XML control records. The first character of the data set must be <.

You should only specify an option file when advised to do so by IBM support.

System programmer response: None.

**ISQ110I ISQSTOP ISSUED FROM A TARGET
CONTROL TASK.**

Explanation: The ISQSTOP command is running under a target control task.

System action: Processing of the ISQSTOP command continues, but the target control task is not ended.

Operator response: Issue a NetView STOP TASK

command to end the target control task after processing of the ISQSTOP command is complete.

System programmer response: None.

ISQ111I *task-name* WILL NOT BE CANCELLED.

Explanation: The ISQSTOP command is running under the target control task identified in this message. This task is not canceled by the command.

System action: Processing of the ISQSTOP command continues, but the target control task is not ended.

Operator response: Issue a NetView STOP TASK command to end the target control task after processing of the ISQSTOP command is complete.

System programmer response: None.

**ISQ112I UNEXPECTED END OF CONTROL
FILE DETECTED.**

Explanation: The end of the control file was reached before all of the required information was read.

System action: Processing of the command ends before it is complete.

Operator response: Check that the correct control file data set is specified.

System programmer response: None.

**ISQ113I CONTROL FILE CONTAINS A
RECORD OF AN INCORRECT
LENGTH.**

Explanation: The control file contains a record that is either too long or too short.

System action: Processing of the command ends before it is complete.

Operator response: Make sure that the correct control file data set is specified.

System programmer response: None.

**ISQ114I UNABLE TO OPEN THE CONTROL
FILE DATA SET.**

Explanation: The request to OPEN the control file data set failed.

System action: Processing of the command ends before it is complete.

Operator response: Check that the correct control file data set is specified on the ISQSTART command.

System programmer response: None.

ISQ115I I/O ERROR OCCURRED READING THE CONTROL FILE DATA SET.

Explanation: The specified control file data set could not be read.

System action: Processing of the command ends before it is complete.

Operator response: Make sure that the control file data set is properly allocated and that the correct data set was specified.

System programmer response: None.

ISQ116I AN ERROR CONDITION *condition-code* HAS BEEN RAISED.

Explanation: This message is written to the NetView log and is for IBM service use only.

System action: Processing of the command ends before it is complete.

Operator response: Save or print a copy of the NetView log and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ117I Target hardware *thw* is not referenced by any target systems.

Explanation: The ISQSTART processing detected that no target systems are associated with the target hardware identified in the message.

thw

message variable containing the name of the target hardware without a reference.

System action: The definition of the target hardware identified in the message is ignored.

Operator response: Contact your System Programmer.

System programmer response: If you want this target hardware to be defined, use the configuration dialogs to change at least one target system definition so that it references this target hardware. Then rebuild the processor/system entry in the Policy Database and refresh the ACF on the ProcOps FP system. In order to activate the changes in ProcOps, issue command ISQSTOP, followed by a ISQSTART ACF command.

ISQ118I No target hardware has been specified for target system *tgt*

Explanation: The ISQSTART processing detected that no target hardware has been specified for the target system identified in this message.

tgt

message variable containing the name of the referenced target system.

System action: The target system is ignored.

Operator response: Contact your System Programmer.

System programmer response: If you want the target system defined, specify a target hardware for this system using the PROCESSOR policy under the system entry type. Then rebuild the processor/system entry in the Policy Database and refresh the ACF on the ProcOps FP system. In order to activate the changes in ProcOps, issue command ISQSTOP, followed by a ISQSTART ACF command.

ISQ119I Target system *tgt* referenced in LPAR TABLE is not defined

Explanation: The ISQSTART processing detected that the target system name used in the LPAR TABLE of a target hardware (processor) is not defined.

tgt

message variable containing the name of the referenced target system.

System action: The target system is ignored.

Operator response: Contact your System Programmer.

System programmer response: Verify the PROCESSOR entries for ProcOps in your Policy Database. Make sure that the LPAR list entries contain only target system names of actually defined SYSTEM entries, or leave the system field blank. Then rebuild the processor/system entry in the Policy Database and refresh the ACF on the ProcOps FP system. In order to activate the changes in ProcOps, issue command ISQSTOP, followed by a ISQSTART ACF command.

ISQ121I SENDING ISQXIII *target-system-name* TO TARGET CONTROL TASK *task-name*.

Explanation: The command to initialize the target system *target-system-name* was sent to the target control task *task-name* for processing.

System action: Processing of the command continues.

Operator response: Wait until the target system is initialized before taking any action.

System programmer response: None.

ISQ122I FAILURE DURING INITIALIZATION OF ENTITY *entity-name*. CLOSING ENTITY.

Explanation: An entity can be either a target system or an ensemble name. This message appears in response to a ISQXIII command to initialize a connection to this entity. An error occurred that prevented the entity from being initialized, and the entity was closed.

System action: The entity is closed.

| **Operator response:** Look for other messages that describe the error that caused the failure. Follow the directions given in the descriptions of those messages in this book.

| **System programmer response:** None.

| **ISQ123I** *command-name: entity* NAME IS
| NEITHER A TARGET SYSTEM, PATH
| OR AN ENSEMBLE.

| **Explanation:** The processing of the command ended before it was complete because the entity name specified was not recognized as a target system, path or ensemble by ProcOps.

| Variable *command-name* shows the name of the command that this message is a response to.

| Variable *entity* identifies the target system, path or ensemble specified to be initialized.

| **System action:** The command is not processed.

| **Operator response:** Make sure to specify the name of a defined entity in the command.

| **System programmer response:** None.

| **ISQ124I** This ProcOps configuration is not
| usable

| **Explanation:** Processor Operations minimum requirement is to have at least one valid target hardware definition with at least one valid target system definition selecting this hardware, or, to have at least one valid target hardware definition with a valid ensemble definition referencing this hardware.

| **System action:** ISQSTART processing is aborted.

| **Operator response:** Save the NetView log and inform the system programmer.

| **System programmer response:** Look for the following messages in the NetView log to identify the reasons why the required ProcOps definitions/tables could not be built: ISQ117I, ISQ118I, ISQ126I, ISQ127I, ISQ129I. Correct the ProcOps definitions in SA policy DB and rebuild the ACF for your ProcOps FP system. Refresh the SA configuration on this system, then issue the ISQSTART command again.

| **ISQ125I** ENSEMBLE *ensemble-name* TARGET
| HARDWARE MEMBER LIMIT
| EXCEEDED.

| **Explanation:** The member definition of *ensemble-name* has more target hardware members defined than allowed. The current maximum of target hardware members supported with zEnterprise is eight (8).

| The ISQSTART processing continues, but all target hardware member names of ensemble *ensemble-name* following the eighth target hardware name are ignored as ensemble members.

| **System action:** ISQSTART processing continues.

| **Operator response:** Make sure that not more than the maximum number of target hardware names are selected as ensemble members. For ProcOps, target hardware and ensembles definitions are made using the SA z/OS Customization Dialogs.

| **System programmer response:** None.

| **ISQ126I** ENSEMBLE *ensemble-name* HAS NO
| TARGET HARDWARE MEMBERS.

| **Explanation:** The definition for *ensemble-name* does not have a target hardware name specified as member.

| The ISQSTART processing continues, but the specified *ensemble-name* is not included in the ProcOps runtime-list of ensembles.

| **System action:** ISQSTART processing continues.

| **Operator response:** If you want to manage the ensemble with ProcOps, make sure that at least one of the target hardware names defined as a node (member) for this ensemble is also defined to ProcOps. Then, select this target hardware as an ensemble member. Target hardware and ensembles definitions are made using the SA z/OS Customization Dialogs.

| **System programmer response:** None.

| **ISQ127I** Target hardware definition *thw* ignored
| by ProcOps.

| **Explanation:** The ISQSTART processing detected a target hardware definition with a processor connection protocol other than SNMP or TCP/IP. ProcOps cannot manage this processor. This informational message is displayed when BCPII/INTERNAL processor connections and ProcOps processor connections are configured to run on the same SA agent.

| *thw*

| message variable containing the name of the referenced target hardware.

| **System action:** ProcOps ignores the target hardware definition.

| **Operator response:** None.

| **System programmer response:** If you want this target hardware to be managed by ProcOps, specify a supported connection protocol for the processor. Then rebuild the processor entry in the Policy Database and refresh the ACF on the ProcOps FP system. In order to activate the changes in ProcOps, issue command ISQSTOP, followed by a ISQSTART ACF command.

ISQ128I THE CONTROL FILE CONTAINS AN INCONSISTENT VERSION IDENTIFIER.

Explanation: This message appears in response to the ISQSTART command. The control file specified in the command has a different version number than expected.

System action: Processing of the ISQSTART command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Generate the control file again. If doing this does not solve the problem, verify that the installation procedure you used is correct. Make sure that the version of the configuration dialogs and the generate function you are using is the same as the version of the ISQSTART command.

ISQ129I Table TARGET *table-name* has no entries

Explanation: The ISQSTART processing detected that the table identified in this message contains no information. The TARGET HARDWARE table is built from processor information and the TARGET SYSTEM table is built from the available system information of the loaded ACF.

table-name

message variable containing the name of the empty table.

System action: Processing stops.

Operator response: Contact your system programmer.

System programmer response: Review the configuration dialogs and enter all target hardware/processor information and all target system information that is needed. At least one target hardware with at least one target system is required to start ProcOps successfully. Then rebuild the processor/system entry in the Policy Database and refresh the ACF on the ProcOps FP system. In order to activate the changes, issue an ISQSTART ACF command.

ISQ130I THE STORED VALUE IS: *value*

Explanation: This message appears in response to the ISQVARS command. The MSG option was specified for a GET request. *value* is the value of the processor operations keyword requested.

System action: None.

Operator response: None.

System programmer response: None.

ISQ131I TOO FEW PARAMETERS HAVE BEEN ENTERED.

Explanation: This message appears in response to the ISQXOPT command or the ISQVARS command. Not all of the parameters that should have been specified for the command were specified.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, making sure that all necessary parameters are specified.

System programmer response: None.

ISQ132I SECOND PARAMETER, *value-entered*, IS NOT VALID.

Explanation: This message appears in response to the ISQVARS command. The value entered for the second parameter of the command is not valid. The valid values for this parameter are:

- TGT
- THW
- LIST
- GRP
- OCF
- LUN

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, making sure that the value of the second parameter in the command is valid.

System programmer response: None.

ISQ133I INCORRECT FORM OF COMMAND, *form-used*, HAS BEEN USED WITH THE "LIST" OPTION.

Explanation: This message appears in response to the ISQVARS command. The LIST option was specified together with the PUT option in the command. This is not a valid form of the command. The LIST option can be used only if the ISQVARS command is issued from an automation procedure and a form of the GET option is specified.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the command, specifying a form of the GET option with the LIST option.

System programmer response: None.

ISQ134I "MSG" AND "LIST" OPTIONS ARE MUTUALLY EXCLUSIVE.

Explanation: This message appears in response to the ISQVARS command. The command was issued with both the MSG and LIST options specified. This is not a

valid form of the command. Commands that contain the LIST option must be issued from an automation procedure. Commands that contain the MSG option originate from an operator.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Make sure that an ISQVARS command that has the LIST option specified originates from a NetView command list or a REXX automation procedure.

System programmer response: None.

ISQ135I UNKNOWN FORM OF COMMAND, form-used, HAS BEEN ENTERED.

Explanation: This message appears in a response to the ISQVARS command. The command was issued with neither the GET option nor the PUT option specified. For an ISQVARS command to be valid, either a form of the GET option or the PUT option must be specified.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, making sure that either a form of the GET option or the PUT option is specified.

System programmer response: None.

ISQ136I UNBALANCED QUOTES IN COMMAND.

Explanation: This message appears in response to the ISQVARS command. An incorrect number of quotes were used in the command.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, making sure that it contains the correct number of quotes.

System programmer response: None.

ISQ137I QUOTED STRING MUST BE THE LAST ARGUMENT.

Explanation: This message appears in response to the ISQVARS command. A string of characters with quotes around it was found in a parameter other than the last parameter in the command. Quotes can only be used in the last parameter of the ISQVARS command.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the command, making sure that quotes are not used in any parameter of the command other than the last parameter.

System programmer response: None.

ISQ138I UNABLE TO WRITE TO CALLER'S LOCAL VARIABLE, local-variable-name.

Explanation: This message appears in response to an ISQVARS command that includes a form of the GET option. The information requested in the command has been obtained. However, this information cannot be written to a NetView variable (if a variable name is specified in the command) or displayed on the console (if the MSG option is specified in the command).

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, making sure that a NetView variable is correctly specified, or that the MSG option is correctly specified. Make sure that the MSG option is not enclosed in quotes.

System programmer response: None.

ISQ139I UNABLE TO RETRIEVE CALLER'S LOCAL VARIABLE, local-variable-name.

Explanation: This message appears in response to an ISQVARS command that includes the PUT option. The local variable or literal value specified in the command cannot be read.

System action: Processing of the ISQVARS command ends before it is complete. The keyword specified in the command is not updated with the value of a local variable or literal value.

Operator response: Reenter the command. If you specify a local variable, make sure that its name is spelled correctly. If you specify a literal value, make sure that it is enclosed in quotes.

System programmer response: None.

ISQ140I ISQVARS PUT SUCCESSFUL.

Explanation: Processing of the ISQVARS command completed successfully.

System action: None.

Operator response: None.

System programmer response: None.

ISQ141I MSG IS A KEYWORD AND CANNOT BE USED AS A VARIABLE NAME.

Explanation: This message appears in response to an ISQVARS command. The term MSG was used as a variable name in the command. Because MSG is the name of an option of the ISQVARS command, it cannot be used as a variable name.

ISQ142I • ISQ151I

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command. If you use the MSG option, make sure that it appears in its proper position in the command. Also, make sure that you have specified the GET option for the command.

System programmer response: None.

ISQ142I **VARIABLE *variable-name* TRUNCATED TO MAXIMUM LENGTH, 255.**

Explanation: This message appears in response to an ISQVARS command. The data for the variable identified in this message has been collected, but it has been truncated to 255 bytes. This occurs when the amount of data collected exceeds 255 bytes and the ISQVARS command was issued from a NetView command list or a high-level language other than REXX. When the ISQVARS command is issued in this way, it uses NetView global variables. These variables have a limit of 255 bytes of data. This limit does not exist when the ISQVARS command is issued from a REXX automation procedure.

System action: Only the first 255 bytes of data collected for the variable are stored with it.

Operator response: None.

System programmer response: None.

ISQ143I **VARIABLE TRUNCATED TO MAXIMUM LENGTH, 32,767.**

Explanation: This message appears in response to an ISQVARS command. The data for the variable identified in this message was collected, but it was truncated to 32,767 bytes. This occurs when the amount of data collected exceeds 32,767 bytes and the ISQVARS command was issued from a REXX automation procedure.

System action: Only the first 32,767 bytes of data collected for the variable are stored with it.

Operator response: None.

System programmer response: None.

ISQ146I **KEYWORD *keyword* MAY NOT SET A VALUE TO BLANKS. INTERNAL DATA UNCHANGED.**

Explanation: This message appears in response to the ISQVARS command. The command specified that the internal variable referenced by ISQVARS keyword *keyword* should be set to blanks. This internal variable cannot be set to blanks.

System action: Processing of the ISQVARS command ends before it is complete. The value of the internal variable referenced in the command is not changed.

Operator response: Refer to *IBM Tivoli System Automation for z/OS Programmer's Reference* for the values that the internal variable referenced in the ISQVARS command can be set to. Reenter the ISQVARS command to set the internal variable to a valid value.

System programmer response: None.

ISQ149I **STRING ENTERED, *string*, CONTAINS TOO MANY CHARACTERS. INTERNAL DATA UNCHANGED.**

Explanation: This message appears in response to the ISQVARS command. An attempt was made to set the internal variable referenced by an ISQVARS keyword to a value containing more than the allowed number of characters.

System action: Processing of the ISQVARS command ends before it is complete. The value of the internal variable referenced by the keyword is not changed.

Operator response: Reenter the ISQVARS command, specifying a valid value for the internal variable referenced by the ISQVARS keyword.

System programmer response: None.

ISQ150I **THE MAXIMUM CP DESIGNATION SUPPORTED FOR THE *processor-type* IS *CP-number*.**

Explanation: This message appears in response to the ISQVARS command. An attempt was made to set the internal variable for CP*x* and *x* is a number greater than the maximum CP number supported for the processor type (*processor-type*). The variable *CP-number* shows the greatest CP number supported by the processor.

System action: Processing of the ISQVARS command ends before it is complete. No internal variable value is changed.

Operator response: Reenter the ISQVARS command, specifying the keyword for a CP that has a number supported by the processor type.

System programmer response: None.

ISQ151I **VECTORS ARE NOT SUPPORTED ON THIS MACHINE.**

Explanation: This message appears in response to the ISQVARS command. An attempt was made to set the value of an internal variable for a vector. (The ISQVARS keyword specified in the command was *vectorn*, where *n* is the number of the vector.) The processing of the command ends before it completes because the processor does not support vectors.

System action: Processing of the ISQVARS command ends before it is complete. No internal variable values are changed.

Operator response: None.

System programmer response: None.

ISQ152I ONLY THREE OR FOUR CHARACTER ADDRESSES SUPPORTED. ENTERED VALUE WAS *address*.

Explanation: This message appears in response to the ISQVARS command. An attempt was made to set the internal variable referenced by ISQVARS keyword *dvol* to a value consisting of more than four characters. The value specified for this internal variable can have a maximum of four characters.

System action: Processing of the ISQVARS command ends before it is complete. No internal variable value is changed.

Operator response: Reenter the ISQVARS command, specifying a valid value with the *dvol* keyword.

System programmer response: None.

ISQ153I PROCESSING WEIGHT OF 0 IS NOT PERMITTED.

Explanation: This message appears in response to the ISQVARS command. An attempt was made to set the internal variable referenced by ISQVARS keyword *lparpw* to 0. The value of this internal variable must be between 1 and 999 (inclusive) or it must be D (dedicated).

System action: Processing of the ISQVARS command ends before it is complete. No internal variable values are changed.

Operator response: Reenter the ISQVARS command, specifying a valid value with the *lparpw* keyword.

System programmer response: None.

ISQ161I *target-hardware-name* IS NOT A VALID TARGET HARDWARE NAME.

Explanation: This message appears in response to the ISQVARS or ISQIPSWT command. The target hardware specified in the command is not defined to SA z/OS.

System action: Processing of the ISQVARS or ISQIPSWT command ends before it is complete.

Operator response: Issue the ISQVARS or ISQIPSWT or command again, making sure that *target hardware name* is correct. If you receive this message again in response, contact your system programmer.

System programmer response: Make sure that the target hardware name used by the operator is the name of a target hardware defined in the SA z/OS configuration dialogs. If the target hardware is defined in the configuration dialogs and you still cannot solve the problem, contact an IBM support representative.

ISQ162I *keyword* IS NOT A VALID KEYWORD.

Explanation: This message appears in response to the ISQVARS command. The keyword specified in the command is not valid.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, making sure that the keyword is spelled correctly and that it is a valid ISQVARS keyword. For a list of valid ISQVARS keywords, refer to the *IBM Tivoli System Automation for z/OS Programmer's Reference*.

System programmer response: None.

ISQ163I *target-system-name* IS NOT A VALID TARGET NAME.

Explanation: This message appears in response to the ISQVARS command when the target system name specified in the command is not valid. It also appears when the CHSTATUS is not available to a zSeries, or 390-CMOS processor.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, making sure that the target system name is correct. If you receive this message again in response, contact your system programmer.

System programmer response: Make sure that the target system name used by the operator is the name of a target system defined in the SA z/OS configuration dialogs. If the target system is defined in the configuration dialogs and you still cannot solve the problem, contact an IBM support representative.

ISQ164I *path-name* IS NOT A VALID PATH NAME.

Explanation: This message appears in response to the ISQVARS command. The path specified in the command is not valid.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, making sure that the path is correct. If you receive this message again in response, contact your system programmer.

System programmer response: Make sure that the pathname used by the operator is the name of a processor defined in the SA z/OS customization dialogs. If the processor is defined in the customization dialogs and you still cannot solve the problem, contact an IBM support representative.

**ISQ166I NOT AN LPAR-MODE SYSTEM:
KEYWORD IS ONLY FOR LPAR-MODE
SYSTEMS.**

Explanation: This message appears in response to the ISQVARS command. The keyword specified in the command is not valid for the target system or port ID specified. It is valid only for a target system running on a logically partitioned target hardware.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Make sure that the target system name or the port ID and the keyword you specify in the ISQVARS command are correct. If both are correct and you receive this message when you issue the ISQVARS command, contact your system programmer.

System programmer response: Determine why there is a conflict between the keyword used and the target system name or port ID specified. Make sure that the definitions in the SA z/OS configuration dialogs were entered correctly, and make sure that the correct data set name was specified for the ISQSTART command used to start SA z/OS.

**ISQ167I CONSOLE CONNECTION *port-ID* IS
NOT AN OPERATOR CONSOLE.**

Explanation: This message appears in response to the ISQVARS command. The port specified in the command is not valid for use with the keyword specified. The keyword can only be used with a port that is connected to an operator console.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Make sure that the port letter and the keyword you specify in the ISQVARS command are correct. If both are correct and you receive this message when you issue the ISQVARS command, contact your system programmer.

System programmer response: Determine why there is a conflict between the keyword used and the port letter specified. Make sure that the definitions in the SA z/OS configuration dialogs were entered correctly, and make sure that the correct data set name was specified for the ISQSTART command used to start SA z/OS.

**ISQ168I CONSOLE CONNECTION *port-ID* IS
NOT A SYSTEM CONSOLE.**

Explanation: This message appears in response to the ISQVARS command. The port specified in the command is not valid for use with the keyword specified. The keyword can only be used with a port that is connected to a system console.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Make sure that the port letter and the keyword you specify in the ISQVARS command are correct. If both are correct and you receive this message when you issue the ISQVARS command, contact your system programmer.

System programmer response: Determine why there is a conflict between the keyword used and the port letter specified. Make sure that the definitions in the SA z/OS configuration dialogs were entered correctly, and make sure that the correct data set name was specified for the ISQSTART command used to start SA z/OS.

**ISQ169I THE TARGET HARDWARE IS IN
LPAR MODE BUT NO TARGETS ARE
INITIALIZED.**

Explanation: This message appears in response to the ISQVARS command. The target hardware specified in the command is indicated as being in LPAR mode, but no associated target systems were initialized.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Contact an IBM support representative.

**ISQ170I CANNOT LOCATE SYSTEM
CONSOLE.**

Explanation: This message appears in response to the ISQVARS command. The port definition and the target hardware definition in the SA z/OS configuration dialogs conflict with one another, making identification of the system console impossible.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Contact an IBM support representative.

**ISQ171I CANNOT LOCATE OPERATOR
CONSOLE.**

Explanation: This message appears in response to the ISQVARS command. The port definition and target system definition in the SA z/OS configuration dialogs conflict with one another, making identification of the operator console impossible.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Contact an IBM support representative.

**ISQ172I "PUT" NOT ALLOWED FOR
KEYWORD *keyword*.**

Explanation: This message appears in response to the ISQVARS command. A keyword that can only be used with a form of the GET option was specified with the PUT option.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, making sure that the correct keyword and PUT or GET option are specified.

System programmer response: None.

**ISQ174I INTERNAL LOCKING SEQUENCE
ERROR.**

Explanation: This message appears in response to the ISQVARS command. Because of an internal error, the requested information could not be accessed.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Save or print a copy of the NetView log, and contact your system programmer.

System programmer response: Contact an IBM support representative.

**ISQ175I INTERNAL ERROR. UNABLE TO
OBTAIN LOCK FOR "PUT".**

Explanation: This message appears in response to the ISQVARS command. A NetView lock could not be obtained for the PUT option of the command. This is an internal problem.

System action: Processing of the ISQVARS command ends before it is complete. Internal data is not changed.

Operator response: Save or print a copy of the NetView log, and contact your system programmer.

System programmer response: Contact an IBM support representative.

**ISQ176I NON-HEXADECIMAL CHARACTERS
ENTERED, INTERNAL DATA
UNCHANGED.**

Explanation: This message appears in response to the ISQVARS command. You tried to write (by specifying the PUT option) nonhexadecimal characters to a keyword field that can contain only hexadecimal characters.

System action: Processing of the ISQVARS command ends before it is complete. No internal data is changed.

Operator response: Reenter the ISQVARS command, making sure that only hexadecimal characters (0-9, A-F) are written to the keyword field.

System programmer response: None.

**ISQ177I NON-INTEGGER CHARACTERS
ENTERED, INTERNAL DATA
UNCHANGED.**

Explanation: This message appears in response to the ISQVARS command. You tried to write (by specifying the PUT option) noninteger characters to a keyword field that can contain only integer characters.

System action: Processing of the ISQVARS command ends before it is complete. No internal data is changed.

Operator response: Reenter the ISQVARS command, making sure that only integer characters (0-9) are written to the keyword field.

System programmer response: None.

**ISQ178I UNKNOWN OPERATING SYSTEM
TYPE FOR TARGET *target-system-name*.**

Explanation: This message appears in response to the ISQVARS command. The target system identified in this message is defined as having a type of operating system not supported by SA z/OS.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Make sure that the SA z/OS control file was generated properly and that the correct control file data set was specified for the ISQSTART command.

**ISQ179I TARGET *target-system-name* IS NOT
RUNNING {MVS | VM}.**

Explanation: This message appears in response to the ISQVARS or ISQXDRL command. The ISQVARS keyword specified in the command is not valid for the specified target system. This keyword is only valid for a target system that is running the MVS or VM operating systems.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, specifying a different keyword or target system.

System programmer response: None.

**ISQ182I INCORRECT CHARACTERS
ENTERED. ONLY *characters* ALLOWED.**

Explanation: This message appears in response to the ISQVARS command. You tried to write (by specifying the PUT option) values to a keyword field that are not valid values for that field. The variable *characters* in this message shows the valid values for the field.

ISQ183I • ISQ188I

System action: Processing of the ISQVARS command ends before it is complete. No internal data is changed.

Operator response: Reenter the ISQVARS command, making sure that only valid values are written to the keyword field.

System programmer response: None.

ISQ183I **NUMBER ENTERED, *value*, TOO LARGE. INTERNAL DATA UNCHANGED.**

Explanation: This message appears in response to the ISQVARS command. You tried to write (by specifying the PUT option) a value that is too large for the keyword field that it is being written to. The variable *value* shows the value you tried to write to the keyword field.

System action: Processing of the ISQVARS command ends before it is complete. No internal data is changed.

Operator response: Reenter the ISQVARS command, making sure that you specify a value that is not too large for the keyword field that it is being written to.

System programmer response: None.

ISQ184I **INCORRECT TIME, *time-submitted*, ENTERED. INTERNAL DATA UNCHANGED.**

Explanation: This message appears in response to the ISQVARS command. You tried to write (by specifying the PUT option) a time offset value that is not valid. Valid offsets are:

- Integers in the range from -23 to +23 (inclusive)

System action: Processing of the ISQVARS command ends before it is complete. No internal data is changed.

Operator response: Reenter the ISQVARS command, making sure that you specify a valid time offset value.

System programmer response: None.

ISQ185I *parameter-value* **FORM NOT ALLOWED FOR KEYWORD *keyword*.**

Explanation: This message appears in response to the ISQVARS command. You specified a parameter (*parameter-value*) that cannot be used with the keyword you specified (*keyword*).

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Reenter the ISQVARS command, specifying a different keyword or a different parameter value. Valid parameter values are:

- LIST
- TGT
- THW
- GRP

- OCF
- LUN

System programmer response: None.

ISQ186I **INCORRECT *keyword* STATUS, *desired-value* ENTERED. INTERNAL DATA UNCHANGED.**

Explanation: This message appears in response to the ISQVARS command. You cannot change the internal variable (the ISQVARS keyword) from its current value to the value you specified (*desired-value*).

System action: Processing of the ISQVARS command ends before it is complete. The value of the internal variable (ISQVARS keyword) is not changed.

Operator response: Refer to the *IBM Tivoli System Automation for z/OS Programmer's Reference* to learn which changes are valid for the internal variable (ISQVARS keyword) that you are trying to change.

System programmer response: None.

ISQ187I **TARGET STATUS CANNOT BE CHANGED MANUALLY FROM *value*.**

Explanation: This message appears in response to the ISQVARS command. You cannot use the ISQVARS command to change the current value (*value*) of the internal variable (keyword) you specified.

System action: Processing of the ISQVARS command ends before it is complete. The value of the internal variable (keyword) is not changed.

Operator response: Refer to *IBM Tivoli System Automation for z/OS Programmer's Reference* to learn which values can be changed and which values cannot be changed for the internal variable (keyword) you are trying to change.

System programmer response: None.

ISQ188I **INCONSISTENT INFORMATION, CANNOT DETERMINE {*target-system-name* | *partition-name*} STATUS.**

Explanation: This message appears in response to the ISQVARS command. The status of the target system or logical partition cannot be identified. This may be the result of internal variables having been changed through the use of the ISQVARS command.

System action: Processing of the ISQVARS command ends before it is complete. No internal variable is changed.

Operator response: Refer to the *IBM Tivoli System Automation for z/OS Programmer's Reference* to learn what the restrictions there are for the internal variable (keyword) that you are trying to change.

System programmer response: None.

ISQ189I *partition-name* IS NOT A VALID LOGICAL PARTITION NAME.

Explanation: This message appears in response to the ISQVARS command. The logical partition you specified is not defined.

System action: Processing of the ISQVARS command ends before it is complete.

Operator response: Make sure the name of the logical partition is spelled correctly and that the logical partition is defined to SA z/OS.

System programmer response: None.

ISQ191I QUALIFIED NAME ENTERED, *name*, IS NOT VALID.

Explanation: The qualified name was not entered correctly. The variable *name* shows the name that was entered.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, making sure the qualified name is entered correctly. The correct form is: *target-hardware-name.partition-name*

System programmer response: None.

ISQ192I LPAR NAME MISSING. ONLY SYSTEM NAME, *target-hardware-name*, GIVEN.

Explanation: The qualified name was not entered correctly. The target hardware name was entered, but it was not followed by the name of a logical partition.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, making sure the qualified name is entered correctly. The correct form is: *target-hardware-name.partition-name*

System programmer response: None.

ISQ194I *name* IS NOT VALID TARGET NAME, BUT IS VALID TARGET HARDWARE NAME.

Explanation: The *name* entered on the target form of an ISQVARS command is not a valid processor operations target system name, but it is a valid processor operations target hardware name.

System action: The ISQVARS command is rejected.

Operator response: Reenter the ISQVARS command, making sure the target system name is correct.

System programmer response: None.

ISQ195I ENSEMBLE NAME *ensname* COULD NOT BE FOUND.

Explanation: The name entered on the ensemble form of an ISQVARS command is not a valid processor operations ensemble name.

System action: The ISQVARS command is rejected.

Operator response: Reenter the ISQVARS command, making sure the ensemble is correct.

System programmer response: None.

ISQ201I *command-name*: ERROR SENDING *variable* TO *target-system-name target-console-type*. RESPONSE WAS: *message-text*.

Explanation: The SA z/OS program received an unexpected reply (shown by *message-text*) to the command identified in this message.

System action: Processing of the command identified in this message ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Look in the NetView log to determine what went wrong during command processing. Correct the problem and reenter the command.

ISQ202I UNABLE TO UPDATE TARGET *target-system-name* IPL STATUS.

Explanation: This message is generated by an SA z/OS automation routine. The IPL status for a target system could not be updated.

System action: None.

Operator response: Issue the ISQSEND command.

System programmer response: None.

ISQ203I *target-system-name* : DUPLICATE [SYSRES | VOLUME] *volser* FOUND ON DEVICE(S) *device-address-1* (AND *device-address-2*). OPERATOR INTERVENTION REQUIRED.

Explanation: This message is generated by an SA z/OS automation routine if the following three conditions are true:

- The default response selection in the policy database for this target system is set to NO.
- During the IPL process one of the following was detected:
 - A duplicate sysres condition. The operating system sends one of the following messages to the automation routine:
 - Message IEA213A, if you use MVS/ESA SP V5, or OS/390

- Message IEA212A, if you use MVS/ESA SP V4, or previous releases
- A duplicate volume condition. The operating system sends one of the following messages to the automation routine:
 - IEA214A if you use MVS/ESA SP V5, or OS/390
 - IEA212A if you use MVS/ESA SP V4, or previous releases
- The automation routine cannot identify (in conjunction with the reply list in the policy database for this target system) one device address as being required.

System action: The IPL process is suspended.

Operator response: If a duplicate sysres condition was detected, verify that you have booted the system from the correct device address. If the device address is correct, enter R 00,CONT at the operator console. The system sets the address offline, where the duplicate sysres was located, and allows the IPL to continue. If the device address was not correct, restart the system with the correct device address (see also message IEA213A).

If a duplicate volume condition was detected, determine which of the two addresses is correct and reply to the other address, which should be set offline (R 00,dev-address). Then the IPL process continues (see also message IEA214A).

System programmer response: Check the reply list and the default response selection in the policy database for this target system. Then specify the correct values.

ISQ204I UNEXPECTED ERROR OCCURRED IN IPL FOR TARGET *target-system-name*. IPL FAILED.

Explanation: This message is generated by a SA z/OS automation routine. The IPL process failed for the target system identified in this message. The SA z/OS program received one of the following unexpected MVS messages: IEA305A, IEA310A, IEA317A, IEA320A, IEA332A, IEA906A.

System action: None.

Operator response: Check the NetView log to identify the MVS message that the automation program responded to. The reply sent for this message by the automation program was not correct. Send the correct reply to the target system, using the ISQSEND command.

System programmer response: Check that the configuration dialogs have a correct reply for messages from the target system identified in this message.

ISQ210I ENTITY *entity_name* WAS NOT LOCKED.

Explanation: This message appears in response to the ISQOVRD and ISQXUNL commands. The port that the command was issued for was not locked at the time of the command. Processing of the command ends.

Entity names can be either ensemble, target-system or path names. Note that a console designator is shown in addition to a path or target-system name specification. Ensemble name entities do not have this extension.

System action: None.

Operator response: None.

System programmer response: None.

ISQ211I *task-name* OWNS THE LOCK FOR ENTITY *entity_name*.

Explanation: This message appears in response to the ISQOVRD, ISQXLOC, and ISQXUNL commands. For the ISQOVRD command, it indicates that the lock for the entity was removed and shows the last owner of the lock (*task-name*). For the ISQXLOC and ISQXUNL commands, it indicates that the command was not performed because another task (*task-name*) owns the lock for the *entity_name*.

Entity names can be either ensemble, target-system or path names. Note that a console designator is shown in addition to a path or target-system specification. Ensemble name entities do not have this extension.

System action: For the ISQOVRD command, processing completes and the lock is removed from the entity. For the ISQXLOC and ISQXUNL commands, processing of the command ends before it is complete.

Operator response: If you tried to issue the ISQXLOC command or the ISQXUNL command, wait to issue these commands until after the lock is removed from the *entity_name*.

System programmer response: None.

ISQ212I THE LOCK FOR ENTITY *entity_name* HELD BY *task-name* HAS BEEN REMOVED.

Explanation: This message appears in response to the ISQOVRD and ISQXUNL commands. Processing of the command completed successfully and the lock is removed from the *entity_name*.

Entity names can be either ensemble, target-system or path names. Note that a console designator is shown in addition to a path or target-system specification. Ensemble name entities do not have this extension.

System action: None.

Operator response: None.

| **System programmer response:** None.

| **ISQ213I** *task-name* **NOW OWNS THE LOCK FOR**
| **ENTITY** *entity_name*.

| **Explanation:** This message appears in response to the
| ISQXLOC command. Processing of the command
| completed successfully.

| Entity names can be either ensemble, target-system or
| path names. Note that a console designator is shown in
| addition to a path or target-system specification.
| Ensemble name entities do not have this extension.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **ISQ214I** *task-name* **ALREADY HAS THE LOCK**
| **FOR ENTITY** *entity_name*.

| **Explanation:** This message appears in response to the
| ISQXLOC command. The task that the command was
| issued from already owns the lock for the entity that
| the lock was requested for.

| Entity names can be either ensemble, target-system or
| path names. Note that a console designator is shown in
| addition to a path or target-system specification.
| Ensemble name entities do not have this extension.

| **System action:** Processing of the ISQXLOC command
| ends before it is complete.

| **Operator response:** None.

| **System programmer response:** None.

ISQ220I *command:* **INCORRECT NUMBER OF**
 PARAMETERS.

Explanation: An incorrect number of parameters was
entered.

System action: The command is rejected.

Operator response: Check message description and
enter command again.

System programmer response: None.

ISQ221I *command: target-hardware-name* **IS NOT A**
 VALID TARGET HARDWARE NAME.

Explanation: The target hardware *target-hardware-name*
is not defined in ProcOps.

System action: The command is rejected.

Operator response: Check message description and
enter command again.

System programmer response: None.

ISQ222I *command: keyword* **IS NOT A VALID**
 keyword. **VALID KEYWORDS ARE**
 list-of-keywords.

Explanation: The entered keyword *keyword* is not a
valid keyword. It must be one of the list of keywords
given.

System action: The command is rejected.

Operator response: None.

System programmer response: None.

ISQ223I *command: target-hardware-name* **IS NOT**
 DEFINED AS PSM.

Explanation: The target hardware,
target-hardware-name, that was entered is not of type
PSM

System action: The command is rejected.

Operator response: None.

System programmer response: None.

ISQ224I *command:* **THERE IS NO VM HOST**
 DEFINED FOR PSM *target-hardware-name*

Explanation: There is no VM host defined for the
ProcOps Service Machine (PSM) target hardware
target-hardware-name VM host and VM name are
mandatory to issue commands for the PSM.

System action: The command is rejected.

Operator response: None.

System programmer response: None.

ISQ225I *command:* **THERE IS NO VM NAME**
 DEFINED FOR PSM *target-hardware-name*

Explanation: There is no VM name defined for the
ProcOps Service Machine (PSM) target hardware
target-hardware-name. VM host and VM name are
mandatory to issue commands for the PSM.

System action: The command is rejected.

Operator response: None.

System programmer response: None.

ISQ226I *command:* **THE VM HOST**
 target-system-name **OF PSM**
 target-hardware-name **IS NOT ACTIVE.**

Explanation: The VM host *target-system-name* is not
active. The command cannot be sent.

System action: The command is rejected.

Operator response: None.

System programmer response: None.

ISQ227I *command*: COMMAND
 target-hardware-name keyword
 STATUS(*status*).

Explanation: This message appears in response to the command *command*. Processing of command *keyword* ended. The variable *status* indicates the ending status of processing the command. Possible values for status are:

- ACCEPTED
- FAILED

The variable *target-hardware-name* shows the name of the target hardware associated with the command.

System action: Processing of the command is complete.

Operator response: For all ending status, look for other messages for further information.

System programmer response: None.

ISQ228I *command*: PSM *target-hardware-name*
 STATUS(*status*).

Explanation: This message appears in response to the command *command*. The variable *status* indicates the status of ProcOps Service Machine (PSM) *target-hardware-name*. Possible values for status are:

- LOGGED ON
- LOGGED OFF
- NOT DEFINED

System action: Processing of the command *command* is complete.

Operator response: None.

System programmer response: None.

ISQ229I *command*: AUTHORIZATION ERROR.

Explanation: The user has an insufficient authorization level for the entered command.

System action: The command is rejected.

Operator response: None.

System programmer response: None.

ISQ232I *task-name* HAS STARTED.

Explanation: This message is generated by a SA z/OS automation routine. The message monitor task *task-name* has started.

System action: Processing of the command continues.

Operator response: None.

System programmer response: None.

ISQ236I *command_name*: OPERAND *operand* IS
 NOT VALID FOR *mode* TARGET
 SYSTEMS.

Explanation: The operand you entered, *operand*, is not valid for *mode* target systems.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command with an operand that is supported for the target system mode identified in this message.

System programmer response: None.

ISQ237I COMMAND *command* IS NOT VALID
 FOR *mode* TARGET SYSTEMS.

Explanation: The command you entered, *command*, is not valid for *mode* target systems.

System action: Processing of the command ends before it is complete.

Operator response: Issue a command that is supported for the target system mode identified in this message.

System programmer response: None.

ISQ238I *group name* IS NOT A VALID GROUP
 NAME.

Explanation: This message is issued when a group name specified on a command is not valid.

System action: Processing of the command ends before it is complete.

Operator response: Issue the command again making sure that the *group name* is correct. If you receive this message again contact your system programmer.

System programmer response: Make sure the group name used by the operator is the name of a group defined in configuration dialogs. If the group name is defined in the dialogs, verify it has some valid systems connected to it.

ISQ239I *subgroup name* IS NOT A VALID
 SUBGROUP NAME.

Explanation: This message is issued when a subgroup name specified on a command is not valid.

System action: Processing of the command ends before it is complete.

Operator response: Issue the command again making sure that the *subgroup name* is correct. If you receive this message again contact your system programmer.

System programmer response: Make sure the subgroup name used by the operator is the name of a subgroup defined in configuration dialogs. If the

subgroup name is defined in the dialogs, verify it has some valid systems connected to it.

ISQ301I *&1*: RETURN CODE FROM *&3* WAS
 &2.

Explanation: This message appears in response to the ISQCCMD command. It is issued whenever a nonzero return code is returned from a process or subprocess called by the command. The variable *&1* shows the common command processed through the ISQCCMD command.

System action: The ISQCCMD command is rejected.

Operator response: Examine other messages that may appear for clues about the problem.

System programmer response: None.

ISQ302I *target-system-name common-command-name*,
 KEYWORD *keyword*: **VALUE**
 '*character-string*' **IS NOT ALLOWED FOR**
 HMC CONNECTED TARGETS.

Explanation: The CTRLCONS common command was entered with TYPE(ALTERNATE), but the target system is connected to an HMC, not to an SE.

System action: The ISQCCMD command is rejected.

Operator response: None. As long as the target system is connected to an HMC, TYPE(ALTERNATE), cannot be used with the CTRLCONS common command.

System programmer response: None.

ISQ303I *target-system-name common-command-name*,
 KEYWORD *keyword*: **THE STRING**
 '*character-string*' **IS NOT A VALID**
 VALUE. VALID VALUES ARE: *value1* [,
 value2, *value3*, ...].

Explanation: An incorrect value was specified for a keyword (identified by the variable *keyword*) in a common command (identified by the variable *common-command*). Valid values for this keyword are listed at the end of the message. The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the ISQCCMD command, making sure that valid values are specified for all common command keywords.

System programmer response: None.

ISQ304I *target-system-name common-command-name*:
 THE STRING *character-string* **IS NOT A**
 VALID KEYWORD. VALID
 KEYWORDS ARE: *keyword-list*.

Explanation: This message appears in response to the command. A keyword that is not valid was found during processing of the command identified in this message. The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The command is rejected.

Operator response: Reenter the command, making sure that you specify a valid keyword. (The valid keywords are shown in the *keyword-list* provided in this message.)

System programmer response: None.

ISQ305I *target-system-name common-command-name*:
 REQUIRED PARAMETER, *keyword*,
 NOT FOUND.

Explanation: This message appears in response to the *common command*. A keyword that must be specified in the command was missing. The variable *keyword* identifies the missing keyword. The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The command is rejected.

Operator response: Reenter the command, making sure that you specify the required keyword.

System programmer response: None.

ISQ306I *target-system-name common-command-name*:
 THE UNRECOGNIZED STRING
 '*character-string*' **WAS FOUND. NO**
 PARAMETERS ARE EXPECTED.

Explanation: This message appears in response to the ISQCCMD command. A keyword was specified for a common command that has no parameters (and, therefore, cannot contain any keywords). The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The ISQCCMD command is rejected.

Operator response: Make sure that you entered the correct common command. If you entered the correct command, reenter it and do not specify keywords as part of the command.

System programmer response: None.

ISQ307I **LOAD: IPLPARM WAS SPECIFIED WITHOUT IPL ADDRESS.**

Explanation: This message appears in response to the ISQCCMD command. The LOAD common command was issued with the IPLPARM keyword specified but without the IPL address.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the LOAD common command, making sure that both the IPLPARM keyword and the IPL address are specified.

System programmer response: None.

ISQ308I *target-system-name common-command-name,*
KEYWORD *keyword:* **THE STRING**
'character-string' **IS NOT A VALID**
VALUE.

Explanation: An incorrect value was specified for a keyword (identified by the variable *keyword*) in a common command (identified by the variable *common-command*). The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the ISQCCMD command, making sure that valid values are specified for all common command keywords.

System programmer response: None.

ISQ309I *target-system-name common-command-name,*
KEYWORD *keyword:* **VALUE**
'character-string' **IS NOT ALLOWED FOR**
COMMAND *'character-string'*. **VALID**
VALUES ARE: *value1* [, *value2*, *value3*, ...].

Explanation: An incorrect value was specified for a keyword (identified by the variable *keyword*) in a common command (identified by the variable *common-command*). The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the ISQCCMD command, making sure that valid values are specified for all common command keywords.

System programmer response: None.

ISQ310I *target-system-name* **COMMAND**
'command' *variable-name1* **'YES'** **NOT**
ALLOWED WITH *variable-name2* **'YES'**

Explanation: This message appears in response to the ISQCCMD ICNTL or PROFILE Update command for either WLME or PWIC to value YES. Updating WLME

to YES is only allowed if PWIC has a value of NO and vice versa.

System action: The ISQCCMD command is rejected.

Operator response: Enter the PROFILE or ICNTL command to set the value of the variable that inhibits the update (either PWIC or WLME) to a value of NO. Then reenter the failed command.

System programmer response: None.

ISQ311I **ISQCCMD: TARGET NOT SPECIFIED.**

Explanation: This message appears in response to the ISQCCMD command. A target system name was not specified in the command. A target system name must be specified. If the target was specified in the form of *target_hardware_name.lpar_name*, a valid target system name has not been found for this combination of hardware and LPAR name.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the command, making sure to specifying a valid target system name or *target_hardware_name.lpar_name* combination.

System programmer response: None.

ISQ312I **ISQCCMD: COMMAND NOT SPECIFIED.**

Explanation: This message appears in response to the ISQCCMD command. A common command name was not specified in the command. A common command name must be specified.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the ISQCCMD command, specifying a common command name.

System programmer response: None.

ISQ313I *command-name:* **COMMAND**
character-string **NOT RECOGNIZED.**

Explanation: This message appears in response to the ISQCCMD or ISQECMD command. A common command name that is not valid was specified in the command. A valid common command name must be specified.

System action: The ISQCCMD/ISQECMD command is rejected.

Operator response: Reenter the ISQCCMD/ISQECMD command, specifying a valid common command name.

System programmer response: None.

ISQ314I *target-system-name common-command:*
**REQUIRED PARENTHESIS NOT
 FOUND FOLLOWING** *keyword*.

Explanation: This message appears in response to the ISQCCMD command. No parenthesis was found following a keyword, when the keyword must be followed by a value that is surrounded by parentheses. The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the command, specifying a value surrounded by parentheses following the keyword.

System programmer response: None.

ISQ315I *target-system-name common-command:*
**MATCHING PARENTHESIS NOT
 FOUND FOLLOWING** *keyword*.

Explanation: This message appears in response to the ISQCCMD command. Only one parenthesis was found following a keyword, when the keyword must be followed by a value that is surrounded by parentheses. The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the command, specifying a value surrounded by parentheses following the keyword.

System programmer response: None.

ISQ316I *target-system-name common-command-name,*
MULTIPLE ENTRIES OF KEYWORD
keyword **FOUND**.

Explanation: The same keyword was found more than once in the command string.

System action: The ISQCCMD command is rejected.

Operator response: Reenter the ISQCCMD command, making sure that each keyword only occurs once.

System programmer response: None.

ISQ317I *target-system-name* **COMMAND**
'command' **WLME UPDATE 'YES' NOT
 ALLOWED, VALUE OF PWI IS OUT
 OF RANGE**.

Explanation: This message appears in response to the ISQCCMD ICNTL or PROFILE Update command to set WLME to the value YES. The value of PWI (initial processing weight) is outside the range that is defined by the PWMN (minimum processing weight) and PWMX (maximum processing weight) fields.

System action: The ISQCCMD command is rejected.

Operator response: Update the values for PWMN, PWMX, and PWI such that PWI fits in the range that is defined by PWMN and PWMX. Reenter the failed command.

System programmer response: None.

ISQ318I *target-system-name* **COMMAND**
'command' variable-name **UPDATE VALUE
 LESS THAN MIN VALUE** *'min-value'*

Explanation: This message appears in response to the ISQCCMD 'ICNTL' or 'PROFILE' Update command for a particular *variable-name*. The update fails because the update value is less than the allowed or defined minimum value.

System action: The ISQCCMD command is rejected.

Operator response: Depending on the variable to be updated, either enter the PROFILE or ICNTL 'Update' command to set the variable that defines the minimum value (for example, PWMN defines the minimum value for PWI) to an appropriate value, or change the value of the variable to be updated to a value that is greater than or equal to the allowed minimum value. Then reenter the failed command.

System programmer response: None.

ISQ319I *target-system-name* **COMMAND**
'command' variable-name **UPDATE VALUE
 GREATER THAN MAX VALUE**
'max-value'

Explanation: This message appears in response to the ISQCCMD 'ICNTL' or 'PROFILE' Update command for a particular *variable-name*. The update fails because the update value is greater than the allowed or defined maximum value.

System action: The ISQCCMD command is rejected.

Operator response: Depending on the variable to be updated, either enter the PROFILE or ICNTL 'Update' command to set the variable that defines the maximum value (for example, PWMX defines the maximum value for PWI) to an appropriate value, or change the value of the variable to be updated to a value that is less than or equal to the allowed maximum value. Then reenter the failed command.

System programmer response: None.

ISQ320I **UNKNOWN STATUS CODE** *status-code*.

Explanation: An asynchronous status change message delivered a status code that is unknown to ProcOps.

System action: The asynchronous status change message is not processed.

Operator response: Save or print a copy of the NetView log, and contact your system programmer.

System programmer response: Contact an IBM support representative.

ISQ321I *target-system-name* CTRLCONS
TSTIME(*time-stamp*) STATUS(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the CTRLCONS common command ended.

The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ322I *target-system-name* OOCOD
TSTIME(*time-stamp*) STATUS(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the OOCOD common command ended.

The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ323I *target-system-name* TCDATA
TSTIME(*time-stamp*) STATUS(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the TCDATA common command ended.

The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ324I *target-system-name* TCM
TSTIME(*time-stamp*) STATUS(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the TCM common command ended.

The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ325I *target-system-name* too many occurrences
for parm '*parm-name*': allowed *value*,
passed *value*.

Explanation: This message appears in response to the ISQCCMD command. The command allows multiple occurrences of the parameter *parm-name*, but the maximum number of allowed occurrences has been exceeded.

The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: Processing of the ISQCCMD command is rejected.

Operator response: Correct the number of occurrences of the parameter *parm-name* and retry ISQCCMD.

System programmer response: None.

ISQ326I *target-hardware-name* **RESERVE**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the RESERVE common command ended.

The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-hardware-name* shows the name of the target hardware that is associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ327I **Image profile update for PWI not allowed if CPS value is set to 0.**

Explanation: The validation of a PWI update request failed. The initial processing weight value of the image profile can only be set or changed if it has shared logical processors defined. The corresponding CPS value found in the profile was 0.

The request is rejected. This message supplements message ISQ419I.

Note that this validation is performed on System z9[®] processors, or later. On CMOS /390 or zSeries processors, the PWI image profile update cannot check the defined logical processor count.

System action: Processing of the ISQCCMD PROFILE command ends before it is complete.

Operator response: None.

System programmer response: Make sure the image profile has the shared logical processors defined and repeat the request.

ISQ328I **ICNTL UPDATE for PWI not allowed if current PWI is 0.**

Explanation: During the validation of an update request of the processing weight for a logical partition, it was detected that its current processing weight is 0. This indicates that the LPAR does not have a shared processor assigned for it, which is required for a processing weight update.

The request is rejected. This message supplements message ISQ419I.

System action: Processing of the ISQCCMD ICNTL

command ends before it is complete.

Operator response: None.

System programmer response: If you are uncertain about the shared or dedicated logical processors that have been assigned for an LPAR, use its image activation profile on the HMC for verification.

ISQ329I *target-system-name* **POWERMOD**
TSTIME(*time-stamp*) **STATUS**(*status*)

Explanation: This message appears in response to the ISQCCMD command. Processing of the POWERMOD common command ended.

The variable *target-system-name* shows the name of the target system associated with the command.

The variable *status* indicates the ending status of processing of the commands. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ330I *entity-name task-name status protocol*
endpoint stack-flag IP-addr lock-holder

Explanation: This message appears in response to the ISQXCON host-based status command for one or multiple target hardware connections. It is part of the multi-line message ISQ350I. The meaning of the msg-fields (columns) is as follows:

The variable *entity-name* indicates the Processor name or a zEnterprise Ensemble name, as defined in the SA z/OS customization dialog.

The variable *task-name* indicates the name of the Processor operations message monitoring task associated with the *entity-name* for HW event processing. A value of '*' indicates that currently no task is associated.

The variable *status* indicates the current session status of the connection path to the entity. Possible settings are:

A- Active

indicates the session endpoint connection is active.

D- Dormant

indicates the connection is not activated since last Processor operations start or Resume operation for this connection.

C- Closed

indicates a previously active session has been closed or the session was set to an initial closed status at Processor operations start time or a previously suspended connection was resumed again.

O - cOnnecting (Ensemble connections only)

indicates that an ensemble session is not yet fully active. If polling is configured for this connection, Processor operations will retry to start a session in the specified time interval.

P- Problem

indicates the connection protocol detected a session problem.

S- Suspended

indicates the connection path is suspended. No sessions can be established to the connection endpoint. This status can only be changed using the RESUME function (ISQXCON RESUME). The status is valid for the Processor operations instance for which the ISQXCON SUSPEND command was entered. Processor operations stores the status using the DSISVRT NetView VSAM DB in order to keep it persistent.

?- (undetermined)

indicates the session status cannot be determined or an error occurred using Processor operations ISQVARS service for variable NVCS / ENSCS.

The variable *protocol* indicates the Network Protocol defined for this connection. Possible settings are:

SNMP

indicates the connection uses the TCP/IP SNMP based services of the System z® Application Programming Interfaces.

TCPIP

indicates the connection uses the socket interfaces to communicate with a Processor operations Service Machine for VM guest support.

HTTP

indicates the connection uses the HTTP based zEnterprise Web Services Application Programming Interfaces to manage Ensemble and Blade Extension resources

???

indicates the protocol of the connection cannot be determined.

The variable *endpoint* indicates the connection endpoint type. Possible settings are:

SE System z Support Element

HMC

Hardware Management Console

PSM

Processor operations Service Machine

* No endpoint, connection not active

The variable *stack-flag* indicates:

Y - Yes

the z/OS default IP stack is used for the connection,

N - No

another IP address other than the default IP stack is used. Depending on the connection protocol use ISQVARS variables IPSTACK or PSMIPST to determine the z/OS system-started task name of this IP stack.

The variable *IP-addr* indicates the IP address in ASN notation or hostname of the connection endpoint. For TCPIP connections, the configured port numbers of the PSM msg and cmd server applications are displayed. A value '???' is shown if the address information cannot be determined due to an internal error.

The variable *lock-holder* is only displayed if a NetView autotask or operator holds a lock for the connection to prevent others from using it or to serialize processing. Two lock types may be shown:

(lock holder)

indicates User initiated lock set with ISQXLOC command

< lock holder >

indicates Internal set lock INGHWCMD/ INGHWCMD

System action: None.

Operator response: None.

System programmer response: None.

ISQ332I *target-system-name common-command-name:*
KEYWORD *keyword1* **IS NOT**
COMPATIBLE WITH *keyword2*.

Explanation: This message appears in response to the common command. A keyword that is identified by variable *keyword1* cannot be used together with the keyword that is identified by variable *keyword2*. The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: The command is rejected.

Operator response: Reenter the command, making sure that you specify the correct keywords.

System programmer response: None.

ISQ333I *target-system-name common-command-name:*
OPTION *option* **IS REQUIRED FOR**
keyword.

Explanation: This message appears in response to the common command. An option that must be specified in the command was missing. The variable *target_system_name* shows the name of the target system that is associated with the command.

System action: The command is rejected.

Operator response: Reenter the command, making sure that you specify a required option. (The required option is shown in the variable *option* that is provided in this message.)

System programmer response: None.

| **ISQ334I** *target-object* **SECLOG**
| **TSTIME**(*time-stamp*) **STATUS**(*status*)

| **Explanation:** This message appears in response to the
| ISQCCMD command. Processing of the SECLOG
| common command ended.

| The variable *target-object* shows the name of the
| target system or target hardware that is associated
| with the command.

| The variable *status* indicates the ending status of
| processing of the command. Possible values for
| status are:

- | – ACCEPTED
- | – FAILED
- | – REJECTED

| **System action:** Processing of the ISQCCMD command
| is complete.

| **Operator response:** For all ending statuses, look for
| other messages that contain information about why
| processing of the command ended as it did.

| **System programmer response:** None.

ISQ335I *target-system-name* **STPDATA**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the STPDATA common command ended.

 The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

 The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ336I *target-system-name* **STP**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the STP common command ended.

 The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

 The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

| **ISQ337I** **The SECLOG status for** *thw_name* **is**
| *seclog_status*

| **Explanation:** This message appears in response to the
| SECLOG common command of ISQCCMD.

| The variable *thw_name* shows the name of the target
| hardware that is associated with the command.

| The variable *seclog_status* indicates if Security Log
| messages from the connection endpoint are
| forwarded to Processor operations. Depending on
| the configuration, the connection endpoint can be
| either a processor Support Element (SE) or a
| Hardware Management Console (HMC):

- | ON - Security Log message forwarding is active
- | OFF - Security Log message forwarding is
| inactive

| **System action:** Processing of the ISQCCMD command
| is complete.

| **Operator response:** None.

| **System programmer response:** None.

| **ISQ350I** **CONNECTION STATUS SUMMARY**

| **Explanation:** This is the header message of a
| multi-line message returned in response to the
| ISQXCON STATUS command. After this header line
| one or multiple ISQ330I messages are following.
| Message ISQ355I closes the multi-line message.

ISQ351I • ISQ356I

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

| **ISQ351I** **THE *req* FOR *ent* FAILED. HWcmd RC:**
| *hwrc*

| **Explanation:** The ISQXCON function request *req* for the specified entity *ent* failed. *hwrc* contains the hardware function return code. The *req* can be either SUSPEND or RESUME.

| **System action:** The function ends before it is complete.

| **Operator response:** Inform the System Programmer.

| **System programmer response:** Use command AOCTRACE ISQXCON,ON to enable the debug mode and repeat the request. Contact IBM Service and have the NetView log with the debug information available if this error still occurs.

ISQ352I **TOO MANY PARAMETERS HAVE BEEN ENTERED.**

Explanation: The specified command includes too many parameters.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, specifying the correct number of parameters.

System programmer response: None.

ISQ353I **TARGET NAME ENTERED,**
| *target-system-name*, **CONTAINS TOO**
| **MANY CHARACTERS.**

Explanation: The target system name specified in the command has too many characters.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command, specifying the correct target system name.

System programmer response: None.

ISQ354I **VIEW COULD NOT PRESENT PANEL**
| *panel-name*. **VIEW RETURN CODE =**
| *return-code*.

Explanation: The panel identified in this message could not be displayed using the NetView VIEW command. The NetView VIEW command is issued automatically during SA z/OS processing to display a panel.

System action: Processing of the command to display a panel ends before it is complete.

Operator response: Write down the name of the panel that could not be displayed and the return code. Contact your system programmer.

System programmer response: Try to identify the problem by examining the panel that could not be displayed. Refer to the description of the VIEW command in *NetView Customization Guide* for a description of the return code displayed with this message.

ISQ355I **END OF CONNECTION STATUS**
| **SUMMARY**

| **Explanation:** This is the closing message of multi-line message ISQ350I returned in response to the ISQXCON STATUS command.

| **System action:** None.

| **Operator response:** None.

| **System programmer response:** None.

ISQ356I **ISQXCON PARAMETER ERROR:**
| *parameter_error_information*

| **Explanation:** The ISQXCON parameter specification is incorrect. *parameter_error_information* can be one of the following:

| **Unknown command specified**

| The ISQXCON sub-commands can be: STATUS, SUSPEND, or RESUME.

| **Command allows no options**

| The ISQXCON sub-commands STATUS and RESUME have no options.

| **FORCE option is invalid**

| The ISQXCON sub-command SUSPEND FORCE specification error.

| **Not a valid target hardware name**

| The specified target hardware name or ensemble name is not defined to Processor operations in the SA z/OS policy data base.

| **No matching target hardware name found**

| The target hardware name was specified using wildcard option *, but no match was found. The ensemble name was specified using wildcard option *, but no match was found.

| **System action:** The function ends before it is complete.

| **Operator response:** Correct the parameter error and repeat the command.

| **System programmer response:** None.

ISQ357I **SESSION TO *thw* IS ACTIVE AND FORCE IS NOT SPECIFIED. REQUEST IGNORED.**

Explanation: The ISQXCON SUSPEND request to the specified target hardware requires the FORCE option to be specified because there are still one or more target system sessions active, using the connection.

System action: The request for the target hardware *thw* specified in the message ends before it is complete.

Operator response: Repeat the SUSPEND with the FORCE option to close all target systems of that target hardware using ISQXCLS first.

System programmer response: None.

ISQ358I **CONNECTION STATUS OF *thw* MUST BE SUSPENDED. REQUEST IGNORED.**

Explanation: The ISQXCON RESUME request can only be processed for the specified target hardware if its connection path status is SUSPENDED.

System action: The request for the target hardware specified in the message ends before it is complete.

Operator response: Repeat the RESUME request if the connection path to the target hardware has the correct status.

System programmer response: None.

ISQ359I **CONNECTION TYPE *ctype* TO *thw* NOT SUPPORTED. REQUEST IGNORED.**

Explanation: The connection protocol *ctype* is not supported by ISQXCON.

System action: The request for the target hardware *thw* specified in the message ends before it is complete.

Operator response: None.

System programmer response: None.

ISQ400I *target-system-name* **UNKNOWN
TSTIME(*time-stamp*)
STATUS(REJECTED).**

Explanation: This message appears in response to the ISQCCMD command. The common command specified within the ISQCCMD command could not be processed correctly. The variable *target-system-name* shows the name of the target system associated with the command.

System action: The ISQCCMD command is rejected.

Operator response: Look for earlier messages to indicate the reason for the problem.

System programmer response: None.

ISQ401I *target-system-name* **ACTIVATE
TSTIME(*time-stamp*) STATUS(*status*).**

Explanation: This message appears in response to the ISQCCMD command. Processing of the ACTIVATE common command ended.

The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system that is associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ402I *target-system-name* **DEACTIVATE
TSTIME(*time-stamp*) STATUS(*status*).**

Explanation: This message appears in response to the ISQCCMD command. Processing of the DEACTIVATE common command ended. The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ403I **ccmd common command not allowed for PSMs.**

Explanation: This message appears in response to the ISQCCMD command. The ccmd common command was issued for a ProcOps Service Machine (PSM). In the context of VM Second Level System support this command can only be used for a PSM.

System action: The ISQCCMD command is rejected.

Operator response: None

System programmer response: None

ISQ407I *target-system-name* **LOAD**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the LOAD common command has ended. The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ408I *target-system-name* **RESTART**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the RESTART common command ended. The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ409I *target-system-name* **CBU**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the CBU common command ended. The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system associated with the command. If the common command was entered with a

target-hardware-name, the first *target-system-name* found in the list of target systems associated with this hardware is shown.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ410I *target-system-name* **START**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the START common command ended. The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ411I *target-system-name* **STOP**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the STOP common command ended. The variable *status* indicates the ending status of processing of the command. Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ412I **CPC function** *function-name* **is not available with connection type** *type*.

Explanation: The hardware function *function-name* is not available with a SNA based NetView connection (NVC) to your CPC or a TCPIP based connection (TCPIP) to your ProcOps Service Machine (PSM).

System action: Processing of the ISQCCMD command ends.

Operator response: Refer to the ISQCCMD common command documentation to get the information about the supported connection types for this request.

System programmer response: None.

ISQ413I *target-system-name* **EXTINT**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the EXTINT common command ended. The variable *status* indicates the ending status of processing of the command.

Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system associated with the command.

System action: Processing of the ISQCCMD command is complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ414I *target-system-name* **SYSRESET**
TSTIME(*time-stamp*) **STATUS**(*status*).

Explanation: This message appears in response to the ISQCCMD command. Processing of the SYSRESET common command ended. The variable *status* indicates the ending status of processing of the command.

Possible values for *status* are:

- ACCEPTED
- FAILED
- REJECTED

The variable *target-system-name* shows the name of the target system associated with the command.

System action: Processing of the ISQCCMD command complete.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ415I *ccmd* **COMMAND CANCELLED.**

Explanation: The NetView operator canceled the common command *ccmd*

System action: The command is canceled.

Operator response: Reenter the command if desired.

System programmer response: None.

ISQ416I **FAILURE ISSUING** *ccmd* **COMMAND.**
RETCODE=*ccmd_rc*

Explanation: Your attempt to issue the common command *ccmd* failed. The NetView RUNCMD command received a nonzero return code *rc*.

System action: Processing of the command ends before it is complete.

Operator response: Reenter the command.

System programmer response: Determine the reason for the failure. Possible meanings of nonzero return codes are:

- | | |
|----|---|
| 4 | The RUNCMD command failed or no response was returned |
| 16 | The command was canceled by the CANCMD command |
| 24 | Some command list data was truncated |
| 28 | The service point application returned more than 132 responses for the RUNCMD |
| 32 | COS command timeout exceeded (specified in NetView Constants module) |
-

ISQ417I *ccmd* **STATUS**(*ccmd_completion_status*)
CONDITION(*cond_code*)
SDATA(*sense_data*)

Explanation: The specified status was received from the target in response to a common command *ccmd*. For an NVC connection refer to the section "Operations Management Condition Code Reference" in *S/390 Managing Your Processors* (GC38-0452) for information on operations management condition codes. For connection types other than NVC refer to Appendix C, "Response Messages, Error Strings, Condition Codes," on page 397.

System action: If *ccmd_completion_status* is SUCCESS, the common command *ccmd* was processed at the target. If *ccmd_completion_status* is FAILED, the status field will be followed by a CONDITION and possibly SDATA fields detailing the reason for the failure. It will also be followed by a text explanation describing the failure. Both of these messages will appear in SA z/OS status panels for the most recent events.

Operator response: None. This is an informational message only.

ISQ418I • ISQ424I

System programmer response: None. This is an informational message only.

ISQ418I PROFILE NOT KNOWN ON *thwname*

Explanation: A common command that was previously issued specified a profile name that was unknown at the destination target hardware *thwname*. This message supplements message ISQ417I or ISQ419I.

System action: Processing of the command ends before it is complete.

Operator response: Check the profile names associated with the target hardware *thwname*. Profile names that are used as common command parameters must exist on the target hardware Support Element.

System programmer response: Make sure that an existing activation profile name is used with the common command. Note that the specified profile name must be of type RESET or LOAD for the CCNTL common command.

ISQ419I ISQCCMD *ccmd* PROCESSING ON *thwname* IS COMPLETE.

Explanation: Processing of the common command *ccmd* is complete. The target hardware *thwname* is now ready to process new requests. This message supplements message ISQ417I.

System action: Processing of the ISQCCMD command is complete.

Operator response: None.

System programmer response: None.

ISQ420I FUNCTION CANNOT BE PERFORMED IN PRESENT STATE OF THE TARGET HARDWARE SYSTEM.

Explanation: The target is in an incompatible mode to process the command. This message supplements message ISQ417I.

System action: Processing of the command ends before it is complete.

Operator response: Reexamine the command in relation to the state of the target. Reenter the common command, specifying the correct target.

System programmer response: Reenter the common command, specifying the correct target.

ISQ421I OPERATIONS MANAGEMENT IS NOT ENABLED ON *thwname*

Explanation: The support element at the target hardware name *thwname* is not enabled to perform operations management commands. Ensure that the support element at the target is enabled for operations management from the hardware system console. This

message supplements message ISQ417I.

System action: Processing of the command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Enable operations management at the target hardware system console.

ISQ422I COMMAND NOT SUPPORTED.

Explanation: The target support element does not support the operations management command. This message supplements message ISQ417I.

System action: Processing of the command ends before it is complete.

Operator response: Ensure that the command is a supported operations management command. Reenter the command, specifying a supported common command.

System programmer response: Reenter the command, specifying a supported common command.

ISQ423I INCORRECT AUTHORIZATION CODE FOR *thwname*

Explanation: The authorization token supplied on the previous command sent to the target hardware name *thwname* was rejected by the target. This message supplements message ISQ417I.

System action: The associated command is rejected.

Operator response: Contact your system programmer.

System programmer response: Verify that the correct operations management authorization token was defined for the NetView Connection associated with this target hardware. Correct the operations management authorization token and reenter the command.

ISQ424I TARGET SYSTEM RESOURCES BUSY. *ccmd* COMMAND WILL BE RETRIED.

Explanation: System resources at the target are temporarily busy, and the target hardware cannot act on the request. This message supplements message ISQ417I.

System action: The command will be retried once.

Operator response: None. This is an informational message only.

System programmer response: None. This is an informational message only.

**ISQ425I TARGET SYSTEM RESOURCES BUSY.
 ccmd COMMAND FAILED**

Explanation: System resources at the target are busy; the receiver is unable to process the requested command after being retried. The command is not processed.

System action: The command is rejected.

Operator response: Reenter command when the target support element is available. If this response appears again, contact your system programmer.

System programmer response: If the problem persists, contact an IBM service representative.

**ISQ426I SYSTEM IS RUNNING. FORCE(YES)
 MAY BE NEEDED ON *target_name***

Explanation: A disruptive common command was issued to an operational target system or target hardware, *target_name*, without FORCE(YES) specified. The command failed. This message supplements message ISQ417I.

System action: The command is rejected.

Operator response: Determine whether the target really is meant to be disrupted. If so, reenter the command with the FORCE(YES) parameter.

System programmer response: Determine whether the target really is meant to be disrupted. If so, reenter the command with the FORCE(YES) parameter.

ISQ427I CPC *thwname* IS NOT POWERED ON

Explanation: A common command that requires a powered-on target hardware was issued to target hardware *thwname*, which was powered off.

System action: The command is rejected.

Operator response: Enter the ACTIVATE common command to the appropriate target system, then reenter the original command, if necessary.

System programmer response: None.

**ISQ428I POWER-ON RESET NOT COMPLETE
 ON *thwname***

Explanation: A common command that requires power-on reset to be complete was issued to target hardware *thwname*, which was in the "power-on reset required" state. This message supplements message ISQ417I.

System action: The command failed.

Operator response: Determine the cause of the failure on the target hardware *thwname* that caused the target to be power-on reset required. Enter an ACTIVATE command, then reenter the original common command.

System programmer response: None.

**ISQ429I SEND A STOP COMMAND, THEN
 RESEND THE ORIGINAL REQUEST**

Explanation: The target system support element is unable to perform the common command because the targeted CPU is not in the stopped state. This message supplements message ISQ417I.

System action: The processing of the command ends before it is complete.

Operator response: Enter the STOP common command, then reenter the original command, specifying the correct target system.

System programmer response: None.

**ISQ430I HARDWARE ERROR. *ccmd*
 COMMAND WILL BE RETRIED**

Explanation: The common command *ccmd* failed due to a hardware error at the target. The command will be retried once. This message supplements message ISQ417I.

System action: The command is retried.

Operator response: None. This is an informational message only.

System programmer response: None. This is an informational message only.

**ISQ431I HARDWARE ERROR. *ccmd*
 COMMAND FAILED.**

Explanation: The common command *ccmd* failed due to a hardware error at the target a second time. This message supplements message ISQ417I.

System action: The processing of the command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Check the NetView hardware monitor for generic alerts with this problem. Analyze the hardware system console information.

**ISQ432I LOCAL OPERATOR OR ANOTHER
 APPLICATION IS IN CONTROL OF
 *target_system_name***

Explanation: A local operator or another systems management application has obtained exclusive control of the target system. This prevents use of SA z/OS status data that may not be up-to-date and the acceptance by the hardware of SA z/OS common commands. This message supplements message ISQ417I.

System action: The processing of the command ends

before it is complete. The target has been unconditionally reserved.

Operator response: Commands may be reentered to the target when the target is released by the local operator.

Programmer response: None.

ISQ433I Insufficient CPs to complete ACTIVATE request

Explanation: An ACTIVATE common command failed due to insufficient CPs at the target. This message supplements message ISQ417I.

System action: The processing of the ACTIVATE command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Verify that the number of dedicated CPs specified in the profile used with the ACTIVATE command does not exceed the number of physical processors at the target hardware.

ISQ434I ONE OR MORE CHANNEL PATHS ARE IN ERROR <chpid-numbers>

Explanation: The CHPID numbers, *chpid-numbers*, were found to be in error during processing of the command. They were defined in the CHPMAP operand of the image profile. This message supplements message ISQ417I.

System action: The processing of the command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Determine the cause of the channel path error and correct the problem.

ISQ435I INSUFFICIENT AMOUNT OF MAIN STORAGE TO PROCESS ACTIVATE REQUEST

Explanation: An ACTIVATE Common Command failed due to insufficient main storage at the target. This message supplements message ISQ417I.

System action: The processing of the ACTIVATE command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Verify that the amount of main storage defined in the profile used with the ACTIVATE request does not exceed the amount of main storage available at the target hardware.

ISQ436I INSUFFICIENT AMOUNT OF EXPANDED STORAGE TO PROCESS ACTIVATE REQUEST

Explanation: An ACTIVATE common command failed due to insufficient expanded storage at the target. This message supplements message ISQ417I.

System action: The processing of the ACTIVATE command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Verify that the amount of expanded storage defined in the profile used with the ACTIVATE request does not exceed the amount of expanded storage available at the target hardware. Determine the cause of a storage error if the profile is correct.

ISQ437I NO S/390 CHANNELS ARE CONFIGURED TO THIS LOGICAL PARTITION

Explanation: A common command was issued to a logical partition that does not have any channels configured. This message supplements message ISQ417I.

System action: The processing of the command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Consult System/390 hardware documentation to properly configure the channels. After configuring the channels, reenter the common command

ISQ438I ERROR IN SPECIFICATION OF CENTRAL STORAGE ORIGIN

Explanation: Activation of a target was requested with an incorrect central storage origin specified.

System action: The processing of the ACTIVATE command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Determine the cause of the storage error. Reenter the common command after specifying the central storage origin.

ISQ439I REQUEST FOR CENTRAL STORAGE CANNOT BE SATISFIED

Explanation: The common command did not complete because the target hardware was unable to satisfy the command's implicit request for central storage. The activation profile identified in the CNAME parameter requested an amount of central storage that would have included the hardware system area (HSA). This message supplements message ISQ417I.

System action: The processing of the command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Determine the correct storage request. Reenter the common command after correcting the storage specification.

ISQ440I A RESOURCE REQUIRED BY A LOGICAL PARTITION WAS NOT DEFINED

Explanation: A required resource for the logical partition was not defined in the associated profile. This message supplements message ISQ417I.

System action: The processing of the command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Verify that the Image Profile for the logical partition agrees with the available resources.

ISQ441I A REQUESTED RESOURCE IS NOT AVAILABLE TO SATISFY THE ACTIVATE REQUEST

Explanation: A resource error exists that may indicate a configuration problem or insufficient resources to process the command. This message supplements message ISQ417I.

System action: The processing of the ACTIVATE command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Determine the required resource and correct the error. Check SDATA code, if any, to identify the specific resource error.

ISQ442I COMMAND *ccmd* NOT SUCCESSFUL ON *target_name*

Explanation: The common command *ccmd* was not successful on the target system or target hardware *target_name*. This message supplements message ISQ417I.

System action: The processing of the command ends before it is complete.

Operator response: Contact your system programmer.

System programmer response: Determine the reason for the failure from the condition report code and sense data (SDATA) information in the command response.

ISQ443I SYSRESET HAD QUALIFIED SUCCESS: *qualification_information*

Explanation: The common command SYSRESET was partially successful. Possible *qualification_information* is:

- “Check PU status”
- “Check I/O status”
- “Check both PU and I/O status. An IOP re-IML (power-on reset) is in progress.”
- “Sense data = ‘*nnnnnnnnnn*’”.

This message supplements message ISQ417I.

System action: Processing of the SYSRESET command is complete.

Operator response: Contact your system programmer.

System programmer response: The instructions given by the processor’s built-in problem analysis facilities may provide additional information. If not, refer to 390-CMOS, or zSeries *Managing Your Processors*.

ISQ444I ACTIVATE HAD QUALIFIED SUCCESS. SUCCESSFUL STEP CODE: *step_number*

Explanation: The target system support element partially completed the ACTIVATE request. *step_number* is a bit map returned to indicate how far the ACTIVATE process completed.

System action: Processing of the ACTIVATE command is complete.

Operator response: See your system programmer.

System programmer response: The instructions given by the processor’s built-in problem analysis facilities may provide additional information. If not, refer to 390-CMOS or zSeries *Managing Your Processors*.

ISQ445I ACTIVATE THE LOGICAL PARTITION AND RESEND THE REQUEST.

Explanation: The target system support element is unable to perform the command because the target system is in the deactivated state.

System action: Processing of the command ends before it is complete.

Operator response: Enter an ACTIVATE common command to the target system, then reenter the original command.

System programmer response: None.

ISQ446I COMMAND ISSUED: *ccmd* *ccmd_parameters*

Explanation: The common command *ccmd* was issued in the message with the parameter shown.

System action: None.

Operator response: This is an informational message only.

System programmer response: This is an informational message only.

**ISQ447I ALERT RECEIVED ON *thwname*. CPC
HARDWARE FAILURE.**

Explanation: An alert was received for target hardware *thwname* indicating that there was a CPC hardware failure at the target. More information about the type of failure can be seen in the second line of the "Last Significant Message" field of the "Target Hardware" panel. Use the NetView NPDA panels to get more information about the alert.

System action: The target hardware status is updated.

Operator response: Contact your system programmer.

System programmer response: The instructions given by the processor's built-in problem analysis facilities may provide additional information. If not, refer to 390-CMOS or zSeries *Managing Your Processors*.

**ISQ448I ALERT RECEIVED ON *thwname*:
STORAGE SUBSYSTEM FAILURE.**

Explanation: An alert was received for target hardware *thwname* indicating that there was a storage subsystem failure. More information about the type of failure can be seen in the second line of the "Last Significant Message" field of the "Target Hardware" panel. Use the NetView NPDA panels to get more information about the alert.

System action: A processor operations target hardware failure alert is generated.

Operator response: Contact your system programmer.

System programmer response: The instructions given by the processor's built-in problem analysis facilities may provide additional information. If not, refer to 390-CMOS or zSeries *Managing Your Processors*.

**ISQ449I ACTIVATE had qualified success. Sense
Data = *nnnnnnnn***

Explanation: An ACTIVATE common command completed at the target system, but there were some errors that are identified by the code in sense data. This message supplements message ISQ417I.

System action: The ACTIVATE command partially completes.

Operator response: Contact your system programmer.

System programmer response: Refer to 390-CMOS or zSeries *Managing Your Processors* for SDATA sense data information. Determine the reason for the errors that are indicated in the SDATA sense data information.

Examine the SDATA code to determine whether action is required.

**ISQ450I ALERT RECEIVED {on *thwname* | FOR
target_system_name}:PROCESSOR
ENTERED HARD WAIT**

Explanation: An alert was received for target hardware *thwname* indicating that one or more processors loaded a disabled wait.

System action: The target attention status is updated.

Operator response: See your system programmer.

System programmer response: For System z processors more information about the alert may be available in the NetView log if a target system, which is initialized to processor operations, is affected by the disabled wait. Look for ISQ900I messages which include a GETIPSW response report AOFA0025.

**ISQ451I ALERT RECEIVED ON *thwname*.
INVALID PASSWORD.**

Explanation: An alert was received for target hardware *thwname* indicating that an operator entered an incorrect password. More information about the type of failure can be seen in the second line of the "Last Significant Message" field of the "Target Hardware" panel. Use NetView NPDA panels to get more information about the alert.

System action: The System Console logon is disallowed.

Operator response: See your system programmer.

System programmer response: Use NetView NPDA panels to get more information about the alert. Follow the instructions given by the processor's built-in problem analysis facilities. Otherwise, use normal support channels for problem determination or contact an IBM support representative.

**ISQ452I UNRECOGNIZED PROFILE NAME
SPECIFIED IN THE AUTOACT
OPERAND OF THE RESET PROFILE**

Explanation: The profile name specified in the AUTOACT operand of the Reset profile that was specified on the ACTIVATE common command was not recognized by the receiving support element at the target hardware. This message supplements message ISQ417I.

System action: ACTIVATE common command is rejected.

Operator response: Contact your system programmer.

System programmer response: Examine the Reset profile that was specified by the ACTIVATE command. Correct the profile name specified in the AUTOACT

operand of the Reset profile, or define the profile if necessary.

ISQ453I **TIMED COMMAND REJECTED
BECAUSE TIMED OPERATIONS
QUEUE IS FULL**

Explanation: The OCF timed operations queue at the target support element is currently full, and the most recent timed command cannot be accepted for future processing. This message supplements message ISQ417I.

System action: The timed command is rejected.

Operator response: Contact your system programmer.

System programmer response: Refer to 390-CMOS or zSeries *Managing Your Processors* for information.

ISQ454I **TIMING WINDOW SPECIFIED IN
COMMAND REQUEST HAS EXPIRED.
REQUEST WILL NOT BE HONORED.**

Explanation: The current request specified a window of time for processing (using the XATIME, XBTIME operands) that has already expired. The request cannot be carried out. This message supplements message ISQ417I.

System action: The timed command rejected.

Operator response: Examine the XATIME and XBTIME operands specified on the common command. Reenter the command with a valid timing window.

System programmer response: None.

ISQ455I **INVALID XATIME OPERAND**

Explanation: The common command was issued with an XATIME operand that is incorrect.

System action: The timed command is rejected.

Operator response: Determine the proper form of the XATIME operand. Correct the XATIME operand and reenter the command.

System programmer response: None.

ISQ456I **INVALID XBTIME OPERAND**

Explanation: The common command was issued with an XBTIME operand that is incorrect.

System action: The timed command is rejected.

Operator response: Determine the proper form of the XBTIME operand. Correct the XBTIME operand and reenter the command.

System programmer response: None.

ISQ457I **INVALID INTERVAL OPERAND**

Explanation: The common command was issued with an INTERVAL operand that is incorrect.

System action: The command is rejected.

Operator response: Determine the proper form of the INTERVAL operand. Correct the INTERVAL operand and reenter the command.

System programmer response: None.

ISQ458I **INVALID COUNT OPERAND**

Explanation: The common command was issued with a COUNT operand that is incorrect.

System action: The command is rejected.

Operator response: Determine the proper form of the COUNT operand. Correct the COUNT operand and reenter the command.

System programmer response: None.

ISQ459I **COMPLETION REPORT FOR
COMMAND *ccmd* RECEIVED.
STATUS(*cmd_status*) COND(*cmd_cond*)
SDATA(*sdata*)**

Explanation: Completion of a previously issued timed request or completion of an ACTIVATE for a logical partition when a nontimed ACTIVATE request has been sent to a primary OCF. Refer to *S/390 Managing Your Processors* for information on operations management condition codes.

System action: If *ccmd_completion_status* is SUCCESS, common command *ccmd* was processed at the target. If *ccmd_completion_status* is FAILED, the status field will be followed by a CONDITION and possibly SDATA fields detailing the reason for the failure. It will also be followed by a text explanation describing the failure. Both of these messages will appear in processor operations status panels for the most recent events.

System action: The ACTIVATE command is complete.

Operator response: If *ccmd_completion_status* is FAILED, refer to the text explanation of the failure.

System programmer response: This is an informational message only.

ISQ460I **ALERT RECEIVED ON *thwname*.
SUPPORT ELEMENT FAILURE**

Explanation: An alert was received for target hardware *thwname* indicating that the support element failed. Remote operations to the affected targets may no longer be possible.

System action: None.

Operator response: Contact your system programmer.

System programmer response: Refer to 390-CMOS or *zSeries Managing Your Processors* for information.

ISQ461I **TARGET SYSTEM** *target_system_name*
EXISTS BUT IS NOT INITIALIZED.
INITIALIZATION IS BEING
REQUESTED.

Explanation: Activation of a target system has caused activation of another target system on the same target hardware. This can occur on OCF-based processors if the Reset profile specifies that multiple LPARs should be activated.

System action: Processor operations automatically initializes the additional target systems being activated by ISQXIII and updates their status.

Operator response: None. This is an informational message only.

System programmer response: None. This is an informational message only.

ISQ462I **TARGET SYSTEM** *target_system_name* **IS**
INITIALIZED BUT IS NO LONGER
ACTIVE. CLOSE PROCESSING IS
BEING REQUESTED.

Explanation: Processor operations has detected that a target system is initialized but is not the target system running on the target hardware. This can occur on OCF-based processors if a basic mode target system has been initialized, but the target hardware is now running in LPAR mode.

System action: Processor operations automatically closes the target system using ISQXCLS.

Operator response: None. This is an informational message only.

System programmer response: None. This is an informational message only.

ISQ463I **CBU REQUEST FOR**
target_hardware_name **CANNOT BE**
EXECUTED. CONDITION CODE:
cond_code

Explanation: The ISQCCMD CBU command for the named target hardware cannot be executed because of the condition shown in *cond_code*. Refer to *IBM Tivoli System Automation for z/OS Messages and Codes* for a list of the possible condition codes and their meaning.

System action: The CBU request is not executed. ISQCCMD processing terminates with a reject messages.

Operator response: Check whether the CBU request has been issued for the correct hardware. Usually this message is displayed when a ACTIVATE, TESTACT, or UNDO CBU command is requested, but the CBU

feature is not installed. Use the CBU STATUS command to determine the current CBU status for this target hardware.

System programmer response: If your hardware has CBU installed and enabled, contact IBM system support.

ISQ464I **CBU DATA OF** *target_hardware_name* **IS:**
cbu_information

Explanation: This message is the response to an ISQCCMD CBU CMD(STATUS) command. The *cbu_information* string contains the current status of the target hardware CBU feature. If the feature is installed, additional information, depending on the actual status is displayed. This information is: CBU activation and expiration dates, and number of test activations left for use.

System action: None. Informational message only.

Operator response: None.

System programmer response: None.

ISQ468I **CCNTL error on** *thwname*. **Cannot access**
activation profiles.

Explanation: A common command that was previously issued specified a profile name that could not be validated. Access to the Support Element profiles failed. This message supplements message ISQ419I.

System action: Processing of the command ends before it is complete.

Operator response: Inform your system programmer.

System programmer response: Use the PROFILE common command to test the access to the activation profiles of the target hardware *thwname*. If there are problems with the access, make a note of the condition codes that are displayed and contact an IBM support representative.

ISQ505I **ISQCMON COMMAND COMPLETE,**
CONDITION=*result-of-command*.

Explanation: Processing of the ISQCMON command is complete and it has resulted in one of the following *result-of-command* conditions:

add user

The user who issued the command is added to the SA z/OS control group.

del user

The user who issued the command is removed from the SA z/OS control group.

System action: Processing of the command continues.

Operator response: If necessary, reenter the command.

System programmer response: None.

ISQ511I *character-string*

Explanation: This message acts as the prefix for all DEBUG mode trace messages.

System action: Processing of the command continues.

Operator response: These messages can be suppressed by selecting the **Debug Off** option of the Monitor Task Control Options panel (which is displayed using the ISQXDST dialog), or by issuing the ISQSTART command without the DEBUG and *data_set_name* parameters.

System programmer response: None.

ISQ527I **ISQCMON PARAMETER NOT RECOGNIZED.**

Explanation: This message appears in response to the ISQCMON command. The command was issued with an option other than ON or OFF.

System action: Processing of the ISQCMON command ends before it is complete.

Operator response: Reenter the ISQCMON command, to make sure that you specify either ON or OFF.

System programmer response: None.

| ISQ531I *target-hardware-name command-name*
TSTIME(*time-stamp*) ENDED RETURN
CODE (*return-code*).

Explanation: This message appears at the end of processing a command.

The variable *target-hardware-name* shows the name of the target hardware that is associated with the command.

The variable *command-name* indicates the command executed.

The variable *return-code* indicates the command ending return code.

System action: Processing of the *command-name* command completes.

Operator response: None.

System programmer response: None.

| ISQ535I *path-name port-id* **STATUS IS** *lock-status*
BY *task-name*.

Explanation: The task that is identified in this message either obtained or released the lock for the port that is identified in the message. This message can be used as a key for non-SA z/OS automation routines.

System action: None.

Operator response: None.

System programmer response: None.

| ISQ536I **COMMAND:** *cmd-name* **LOCATION:**
loc-name **RECEIVED RC:** *cmd-rc* **FROM:**
called-cmd

Explanation: A command called from a processor operations routine has returned an unexpected return code.

The variable *cmd-name* shows the processor operations command name or module name that was running.

The variable *loc-name* shows the location within *cmd-name* where the problem occurred.

The variable *cmd-rc* shows the unexpected return code issued from *called-cmd*.

The variable *called-cmd* shows the name of the command and, optionally, the parameters of the command that returned the unexpected return code.

System action: Processing terminates.

Operator response: Write down the information shown in this message. Also, save or print a copy of the netlog and contact your system programmer.

System programmer response: Contact an IBM support representative.

| ISQ537I **COMMAND:** *cmd-name* **RECEIVED:**
excpn **FROM:** *source* **DATA:** *error-data*

Explanation: A processor operations command encountered an unexpected exception.

The variable *cmd-name* identifies the processor operations command name or module name that was running.

The variable *excpn* identifies which type of error occurred.

The variable *source* indicates where the error may have originated from.

The variable *error-data* contains data that may help with the problem resolution.

System action: Processing terminates.

Operator response: Save or print a copy of the netlog and contact your system programmer.

System programmer response: Try to determine the cause of the problem with the help of the information given in *excpn*. One possible exception is:

INVALID_DATA_0001: An incorrect message was received. Specifically, a line of data received from a Support Element (SE) was expected to contain either a CPC name or an LPAR name and had neither. The processor operations target hardware name for the SE is given in *source* and the line of data in error is repeated in *error-data*. This problem can occur in CPCs with more than ten CPUs if the Licensed

ISQ600I • ISQ607I

Internal Code (LIC) in the SE is not up-to-date.
Check this with your IBM customer engineer.

If the cause of the problem cannot be determined contact an IBM support representative.

ISQ600I **PROCESSOR OPERATIONS
COMMAND CAN NOT BE ISSUED TO
THE SPECIFIED GROUP OR
SUBGROUP.**

Explanation: This message is issued when a group or subgroup has been specified that is not valid for Processor Operations commands.

System action: Specified action will not be taken.

Operator response: Specify the correct group or subgroup for the Processor Operations command. If you want to issue a command to the specified group or subgroup, you must update the configuration dialogs.

System programmer response: None.

ISQ602I **PROC-OPS AUTOMATION TABLE
PROCESSING OPTION *tpo* IS
INVALID.**

Explanation: During ISQSTART or ISQSTOP processing, the processor operations supplied automation table member ISQMSG01 was processed using option value *tpo*, which is not valid. The valid options are START or STOP.

System action: The ISQSTART or ISQSTOP processing terminates.

Operator response: None.

System programmer response: Contact the IBM system support.

ISQ603I **PROC-OPS AUTOMATION TABLE IS
ALREADY *tblstat*.**

Explanation: During ISQSTART or ISQSTOP processing, the processor operations supplied automation table member ISQMSG01 was found already in the desired state. Automation table states *tblstat* values are 'activated' for ISQSTART and 'stopped' for ISQSTOP.

System action: The ISQSTART or ISQSTOP processing continues.

Operator response: None.

System programmer response: None.

ISQ604I **PROC-OPS AUTOMATION TABLE
tblaction failed.**

Explanation: During ISQSTART or ISQSTOP processing, the processor operations supplied automation table member ISQMSG01 processing

tblaction failed, because the NetView AUTOTBL command, issued internally, did not respond with a successful completion message BNH360I. The *tblaction* values can be 'removal' for ISQSTOP or 'activation' for ISQSTART.

System action: The ISQSTART or ISQSTOP processing terminates.

Operator response: None.

System programmer response: Verify if automation table member ISQMSG01 is available and can be processed successfully using the 'INSERT' and 'REMOVE' options of the AUTOTBL command. After the problem is corrected, rerun ISQSTART or ISQSTOP command.

ISQ605I **PROC-OPS AUTOMATION TABLE
tblaction was successful.**

Explanation: During ISQSTART or ISQSTOP processing, the processor operations supplied automation table member ISQMSG01 was processing successfully. The *tblaction* values can be 'removal' for ISQSTOP or 'activation' for ISQSTART.

System action: The ISQSTART or ISQSTOP processing continues.

Operator response: None.

System programmer response: None.

ISQ606I **PROC-OPS CONFIGURATION
VARIABLES *vpo* was successful.**

Explanation: During ISQSTART the BUILD or ISQSTOP the DESTROY of additional configuration variables was performed successfully.

System action: The ISQSTART or ISQSTOP continues.

Operator response: None.

System programmer response: None.

ISQ607I **PROC-OPS CONFIGURATION
VARIABLES OPTION *vpo* is not valid.**

Explanation: During ISQSTART or during ISQSTOP the invalid processing option *vpo* was used. Valid processing options are BUILD or DESTROY.

System action: The ISQSTART or ISQSTOP processing terminates.

Operator response: None.

System programmer response: Contact the IBM system support.

ISQ608I **PROC-OPS ISQVARS VARIABLE** *keyvar*
FOUND EMPTY, PROCESSING
TERMINATED.

Explanation: During ISQSTART a processor operations configuration was used that contains insufficient ISQVARS information to build additional required configuration variables. Additional variables need to be created for all processors with a connection type other than NVC. The keyword variable *keyvar* can be one of the following: THWLIST, NPNETID, NPLU, CTYPE, TAUTH, SEADDR or HMCADDR.

System action: The ISQSTART processing terminates.

Operator response: None.

System programmer response: Make sure your processor operations control file used for ISQSTART contains target hardware definition. For target hardware with a connection type other than NVC, verify that it has all required fields defined using the customization dialog. Add the processor information and rebuild your processor operations control file. Issue ISQSTART with the new build configuration file.

ISQ609I **Target Hardware Name must not be blank.**

Explanation: A blank value for the mandatory ISQIPSWT command parameter *target hardware name* was provided.

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: Provide a valid target hardware name with the ISQIPSWT command.

System programmer response: None.

ISQ610I **Invalid parameter name entered:**
parameter-name.

Explanation: The second parameter for the ISQIPSWT command was provided, but was not FORCE(YES) or FORCE(NO).

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: Provide a valid FORCE parameter with the ISQIPSWT command.

System programmer response: None.

ISQ611I **Invalid 'FORCE' option - must be 'FORCE(YES)' or 'FORCE(NO)'**.

Explanation: The FORCE parameter for the ISQIPSWT command was provided, but the value was not YES or NO.

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: Provide a valid FORCE parameter value with the ISQIPSWT command.

System programmer response: None.

ISQ612I **Target Hardware Connection Type is not SNMP, but** *target-hardware-connection-type.*

Explanation: The target hardware is not SNMP-connected.

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: None.

System programmer response: Make sure that the target hardware connection type is correctly defined in the SA z/OS customization dialog.

If the connection type is correctly defined (that is, the connection type really is not SNMP), you cannot use the ISQIPSWT command for this target hardware.

If the connection type is not correctly specified, change the value to SNMP, rebuild the processor operations control file and restart processor operations.

ISQ613I **Target Hardware Variable SEADDR or HMCADDR must not be blank.**

Explanation: One of the processor operations variables, SEADDR or HMCADDR (or both), is blank.

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: None.

System programmer response: Make sure that both values are correctly defined in the SA z/OS customization dialog. To avoid restarting processor operations, the SEADDR or HMCADDR values can be set using the ISQVARS command.

ISQ614I **IP connect to alternate SE address**
value-of-HMCADDR failed.

Explanation: The ISQIPSWT command tries to contact the IP address that is stored in HMCADDR, but fails.

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: None.

System programmer response: Check whether the SE's alternate network card works properly. Additionally, make sure that its value is correctly defined in the SA z/OS customization dialog (the TCP/IP Address or Hostname for Support Element second Adapter or a HMC entry field in the PROCESSOR INFO policy item of the PRO entry).

ISQ615I Primary SE connection *value-of-SEADDR* still active - use command with FORCE(YES).

Explanation: The ISQIPSWT command checks whether communication to the IP address that is stored in SEADDR is possible. In this case, FORCE(YES) must be passed as the second command parameter in order to successfully execute the command.

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: Repeat the ISQIPSWT command with FORCE(YES).

System programmer response: None.

ISQ616I Connection failed for system *system-name* - ISQXIII return code *ISQXIII-return-code*.

Explanation: After switching the SEADDR and HMCADDR values, the ISQIPSWT command tries to re-initialize the systems of the target hardware that was previously stopped by ISQIPSWT. The ISQXIII command that was used to perform this task encountered an error.

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: None.

System programmer response: Browse the netlog for additional error messages. With regard to ISQXIII, look for AOFA0002 responses with associated condition and sense data. If possible, correct errors and retry ISQIPSWT.

ISQ617I Command partially executed - IP addresses were not changed.

Explanation: This message is issued with message ISQ615I to indicate that only some logic of the ISQIPSWT command has been executed, namely, the closing of target systems. Before exiting, the ISQIPSWT command will undo the switching of values between variables SEADDR and HMCADDR.

System action: Processing of the ISQIPSWT command ends before it is complete.

Operator response: Make a note of the target systems that were closed by the (failed) ISQIPSWT command. Before rerunning ISQIPSWT, you can start these manually using the ISQXIII command.

System programmer response: None.

ISQ618I *target-hardware-name* ISQIPSWT TSTIME(*time-stamp*) STATUS(*status*).

Explanation: This message appears in response to the unsuccessful execution of an ISQIPSWT command. Processing of the ISQIPSWT command ended.

The variable *target-hardware-name* shows the name of the target hardware that is associated with the command.

The variable *status* indicates the command ending status. Possible values for status are FAILED or REJECTED.

System action: Processing of the ISQIPSWT command completes.

Operator response: For all ending statuses, look for other messages that contain information about why processing of the command ended as it did.

System programmer response: None.

ISQ619I Command inhibited by initialization pending for target hardware *target-hardware-name*.

Explanation: This message appears in response to the ISQIPSWT command. Processing of the ISQIPSWT common command detected that one or more target systems are currently being processed by the ISQXIII command (initialize target system connection). In order to avoid interference with the running initialization process, the ISQIPSWT command terminates with a 'rejected' message.

The variable *target-hardware-name* shows the name of the target hardware that is associated with the command.

System action: Processing of the ISQIPSWT command completes.

Operator response: Check that all target system connection commands have ended for the command target hardware and repeat the ISQIPSWT command.

System programmer response: None.

ISQ700I *pppppppp SC mmmmmm*

Explanation: A message has been received from the ProcOps Service Machine (PSM). The original message was received by the virtual machine that the PSM Control Program is running in.

The variable *pppppppp* gives the name of the PSM that the message has been received from.

The variable *mmmmm* gives the text of the message that was received by the PSM.

System action: None.

Operator response: None. This is an informational message only.

System programmer response: None

ISQ701I **PSM Receive Program started. Version**
nnn

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been started in its Message Monitoring Task (MMT). This is a long running program that receives messages issued by guest machines on the VM system that are controlled by the PSM.

The variable *nnn* gives a value that indicates the level of the program run. This is used for service purposes only.

System action: None. This is an informational message only. Normal processing continues.

Operator response: None.

System programmer response: None

ISQ703I **PSM Receive Program for PSM** *pppppppp*
starting on task *ttttttt*.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been started in its Message Monitoring Task (MMT).

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

The variable *ttttttt* gives the name of the autotask that this copy of the program is running in.

System action: Normal processing continues.

Operator response: None.

System programmer response: None

ISQ704I **PSM Receive Program for PSM** *pppppppp*
will connect to IP address *nn.nn.nn.nn*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been started in its Message Monitoring Task (MMT).

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

The variable *nn.nn.nn.nn* gives the dotted-decimal IP address of the remote VM system that this copy of the program will connect to in order to receive messages from the PSM running in that VM system.

System action: Normal processing continues.

Operator response: None.

System programmer response: None

ISQ705I **PSM** *pppppppp* **has port address** *nnnnn*
for message traffic.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been started in its Message Monitoring Task (MMT).

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

The variable *nnnnn* gives the port number on the remote VM system that this copy of the program will connect to in order to receive messages from the PSM running in that VM system.

System action: Normal processing continues.

Operator response: None.

System programmer response: None

ISQ706I **CONNECT done for socket** *sss*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has issued a TCPIP CONNECT to set up a connection to its associated PSM. The CONNECT has completed with a return code of zero.

The variable *sss* gives the socket number used for the CONNECT call.

System action: Normal processing continues.

Operator response: None.

System programmer response: None

ISQ707I **CONNECT done for socket** *sss*. **Awaits**
actual connection from PSM.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has issued a TCPIP CONNECT to set up a connection to its associated PSM.

The CONNECT has completed with a return code of zero, but the full two-way communication has not been set up. The PSM is not responding to the CONNECT. Because the socket operates in non-blocking mode, the program waits until the PSM issues a TCPIP ACCEPT, at which time a full two-way connection is in place.

The variable *sss* gives the socket number used for the CONNECT call.

System action: Normal processing continues.

Operator response: As soon as the Message Server on the PSM issues a LISTEN, this link should connect and an ISQ709I message should be seen. If it does not, ensure that the PSM on the remote VM system is active and that all parts of the IP connection between the PSM and the ProcOps NetView are functioning.

Once the IP link to the PSM is restored the program

should connect automatically and the ISQ709I message should be seen. This reconnection may not occur immediately after the link is established.

System programmer response: None

ISQ708E CONNECT for socket *sss* gave TCPIP error number *eeee*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been issued a TCPIP CONNECT to set up a connection to its associated PSM. The CONNECT has completed with an error that prevents further processing.

The variable *sss* gives the socket number used for the CONNECT call.

The variable *eeee* gives the non-zero TCPIP error number returned by the CONNECT call.

System action: the PSM Message Receiver program terminates.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned, if possible. Use this to correct the error, then reinitialize the PSM using ISQXIII for each of its target systems that is desired to be initialized.

ISQ709I Handshake done for socket *sss* to PSM *pppppppp*. Session fully established.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has established a TCPIP connection.

The PSM has accepted the connection and has sent back a valid handshaking sequence. The two ends of the connection are now communicating with each other. Up to this point a successful CONNECT may have been done, but it does not indicate two-way communication. From this point on, unsolicited messages may be received from the PSM.

The variable *sss* gives the socket number used.

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

System action: Normal processing continues.

Operator response: None

System programmer response: None

ISQ710I Socket *nnn* is in non-blocking mode.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been obtained a TCPIP socket and has set it to non-blocking mode.

The variable *nnnnnnnn* gives the socket number obtained. This socket number may be used in later messages.

System action: Normal processing continues.

Operator response: None.

System programmer response: None.

ISQ711E FCNTL request to set non-blocking mode for socket *sss* ended with error number *eeee*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been used the FCNTL request to set non-blocking mode for a socket. The request ended in error.

The variable *eeee* gives the non-zero TCPIP error number returned by the FCNTL request.

System action: the PSM Message Receiver program terminates. All initialized target systems on the PSM are closed.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, if possible, then reinitialize the PSM using ISQXIII by issuing the ISQXIII command for each of its target systems that are to be initialized.

Otherwise, report the problem to your IBM Service representative.

ISQ712I PSM Receive Program for PSM *pppppppp* on task *ttttttt* has been canceled.

Explanation: As part of the shutdown of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has detected that is canceled by a NetView RESET or CANCEL command has been issued, indicating that the program should stop execution.

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

The variable *ttttttt* gives the name of the autotask that this copy of the program is running in.

System action: The PSM Message Receiver program terminates with a zero return code. Such a cancel is normally done as part of the normal termination of a

PSM connection, when the last initialized target system on a PSM is closed. All target systems on the PSM are then closed.

Operator response: None

System programmer response: None

ISQ713I **Socket *sss* closed. RC =*rrrr* error number = *eeee***

Explanation: As part of the termination of a Message Receiver program, a CLOSE has been issued for a previously obtained TCPIP socket. The termination may have been requested or be the result of an error.

The variable *sss* gives the socket number of the socket that was closed.

The variable *rrr* gives the return code from the CLOSE call.

The variable *eeee* gives the TCPIP error number returned by the CLOSE call. A value of zero indicates no error was detected.

System action: Termination of the ProcOps Service Machine (PSM) Message Receiver program continues. All initialized target systems on the PSM are closed.

Operator response: None

System programmer response: None

ISQ714E **SELECT call for socket *sss* action *a* gave error number *eeee***

Explanation: The ProcOps Service Machine (PSM) Message Receiver program issued a TCPIP SELECT call to check whether it could proceed with a READ or WRITE request. The SELECT gave an error.

The variable *sss* gives the socket number of the socket that the SELECT was made for.

The variable *a* gives the desired action that the SELECT was made for:

- A value of R indicates that the program was ready to do a READ.
- A value of W indicates that the program was ready to do a WRITE, or that the program was ready to connect to the PSM.
- Any other value indicates that the program was waiting.

The variable *eeee* gives the TCPIP error number returned by the SELECT call.

System action: The PSM Message Receiver program terminates. All initialized target systems on the PSM are closed.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the `errno.h` header file for a complete description of the error number returned. Use this to correct the error, if possible. Then reinitialize the PSM by issuing the

ISQXIII command for each of its target systems that are to be initialized.

Otherwise report the problem to your IBM Service Center.

ISQ715E **Unrecoverable error for socket *sss* to PSM *pppppppp*. Program ending.**

Explanation: The ProcOps Service Machine (PSM) Message Receiver program has sustained an error that it cannot recover from. This message is preceded by one or more error messages detailing the cause of the terminating error.

The variable *sss* gives the socket number of the socket that the error occurred on.

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

System action: The PSM Message Receiver program terminates. All initialized target systems on the PSM are closed.

Operator response: Report the error to your system programmer.

System programmer response: Refer to earlier error messages.

ISQ717I **Session on socket *sss* to PSM *pppppppp* has been closed by PSM.**

Explanation: the ProcOps Service Machine (PSM) Message Receiver program has sustained a signal indicating that the connection has been closed as a result of program action by the PSM. This may be the result of an intended termination, or an unexpected error detected by the PSM.

The variable *sss* gives the socket number of the socket that the error occurred on.

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

System action: The PSM Message Receiver program terminates with a return code of zero.

Operator response: Check whether the PSM terminated the connection as a result of a normal termination request. If it did not, investigate why the PSM issued the termination. To reinitialize the PSM, issue the ISQXIII command for each of its target systems that are to be initialized.

System programmer response: None

ISQ718E **PSM Receive Program failed to obtained socket. Return code = *rrrrrr***

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been started in its Message

Monitoring Task (MMT). It issued a TCPIP SOCKET call, but this was not successful.

The variable *rrr* gives the non-zero error number returned by the SOCKET call.

System action: The PSM Message Receiver program terminates. All target systems on the PSM are closed.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to determine the cause of the error and then re-issue the original request.

Otherwise report the problem to your IBM Service Center.

ISQ719I PSM Receive Program has obtained socket number *nnnn*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been obtained a TCPIP socket.

The variable *nnnn* gives the socket number obtained. This socket number may be used in later messages.

System action: Normal processing continues.

Operator response: None.

System programmer response: None.

ISQ720E RECV request on socket *sss* gave TCPIP error number *eeee*

Explanation: The ProcOps Service Machine (PSM) Receive Program issued a TCPIP RECV request that ended in error.

The variable *sss* gives the socket number used in the RECV request.

The variable *eeee* gives the TCPIP error number returned by the RECV request.

System action: The PSM Message Receiver program terminates. All initialized target systems on the PSM are closed.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, if possible. Then use the ISQXIII command to initialize each of the target systems on the PSM that you want to initialize.

ISQ721E FCNTL call to set non-blocking mode for socket *sss* ended with TCPIP error number *eeee*

Explanation: As part of the processing to send a request to a ProcOps Service Machine (PSM) on VM, the PSM Send Program ISQVMSND has issued a FCNTL request to set non-blocking mode for its TCPIP socket. The request ended in error. No command has been sent to the PSM.

The variable *sss* gives the socket number used in the FCNTL request.

The variable *eeee* gives the TCPIP error number returned by the FCNTL request.

System action: Execution of the PSM Send Program terminates with an error return.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, if possible, then reissue the failing command, if appropriate.

ISQ722I PSM Send Program for PSM *pppppppp* on task *ttttttt* has been canceled.

Explanation: While sending a request to a ProcOps Service Machine (PSM), the PSM Send Program has detected that the command is to be canceled immediately. At the point of cancelation, the request may or may not have been sent to the PSM. Also, some, none, or all of the immediate responses to the request (if any) may have been returned as ISQ700I messages.

Any indirect messages issued as a result of the request executing on the PSM (for example, CP commands to a guest machine) are returned as ISQ900I messages using the PSM Message Receiver and are not affected.

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

The variable *ttttttt* gives the name of the autotask that this copy of the program is running in.

System action: The PSM Send Program ends. Commands that called the Send Program may end successfully or unsuccessfully.

Operator response: None.

System programmer response: None.

ISQ723E SELECT call for socket *sss* action *a* gave TCPIP error number *eeee*

Explanation: While sending a request to a ProcOps Service Machine (PSM), the PSM Send Program issued a SELECT request to check whether it could proceed

with a READ or WRITE request. The SELECT request gave an error.

The variable *sss* gives the socket number of the socket that the SELECT was made for.

The variable *a* gives the desired action that the SELECT was made for:

- A value of R indicates that the program was ready to do a READ.
- A value of W indicates that the program was ready to do a WRITE, or that the program was ready to connect to the PSM.
- Any other value indicates that the program was waiting.

The variable *eeee* gives the TCPIP error number returned by the SELECT request.

System action: Execution of the Send Program terminates with an error. Depending on the point at which the error occurred, the request may or may not have been sent to the PSM, and some, all, or none of the responses, if any, may have been received.

Any indirect messages issued as a result of the request executing on the PSM (for example, CP commands to a guest machine) are returned as ISQ900I messages using the PSM Message Receiver and are not affected.

Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, if possible, then reissue the failing command, if applicable.

ISQ724E PSM Send Program timed out waiting to connect for PSM *pppppppp*.

Explanation: While attempting to send a request to a ProcOps Service Machine (PSM), the PSM Send Program issued a TCPIP CONNECT request to the PSM. The request timed out.

The variable *pppppppp* gives the name of the PSM.

System action: The request is not sent to the PSM. An error return code is given to the caller of the PSM Send Program

If this occurred as a result of a command such as ISQXIII or ISQSTART, the PSM and its guest target systems enter error status.

If this occurred as a result of a ProcOps common command, the command is not executed.

Operator response: Ensure that the PSM is initialized and can be reached through the IP network. When resolved, and if appropriate, reissue the command that failed.

System programmer response: None

ISQ725E Caller not authorized to use command *cccccccc*

Explanation: While attempting to send a request to a ProcOps Service Machine (PSM), the PSM Send Program discovered that the caller did not have NetView authorization to run the command.

The variable *cccccccc* gives the name of the ProcOps command processor that was invoked.

System action: The request is not sent to the PSM. An error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Report the problem to your system programmer

System programmer response: Check that the security definitions in NetView are correct. (Command authorization is done using either Command Authorization Table definition, or through definition to an SAF product such as RACF.)

If necessary, use NetView or SAF product facilities, or both, to update and refresh the definitions used.

ISQ726E Caller not authorized to address PSM *ttttttt*

Explanation: While attempting to send a request to a ProcOps Service Machine (PSM), the PSM Send Program discovered that the caller did not have NetView authorization to send a request to the indicated PSM.

The variable *ttttttt* gives the name of the PSM that the request was directed to.

System action: The request is not sent to the PSM. An error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Report the problem to your system programmer.

System programmer response: This message is likely to be preceded by NetView messages BHN236E and BNH237E, giving details of the failed authorization. Use these to check that the security definitions in NetView are correct.

Access to specific PSMs is controlled by the authorization definition for the TARGET operand of the ISQVMSND command. (Command authorization is done using either Command Authorization Table definition, or through definition to a SAF product such as RACF.)

If necessary, use NetView or SAF product facilities, or both, to update and refresh the definitions used.

ISQ727E Authorization checking ended with return code *cccc*

Explanation: While attempting to send a request to a ProcOps Service Machine (PSM), the PSM Send Program issued a call to the CNMSCOP routine to check that the caller was authorized to issue the request. The call resulted in an unexpected return code.

The variable *cccc* is the return code.

System action: The request is not sent to the PSM. An error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Report the problem to your system programmer

System programmer response: Use the return code to investigate why the unexpected return code was given. The meaning of the return code for CNMSCOP is given in the section "HLL Service Routine Reference" in the NetView *Customization: Using PL/I and C* manual.

Check that the security definitions in NetView are correct. Access to specific PSMs is controlled by the authorization definition for the TARGET operand of the ISQVMSND command. (Command authorization is done using either Command Authorization Table definition, or through definition to a SAF product such as RACF.)

If necessary, use NetView or SAF product facilities, or both, to update and refresh the definitions used.

ISQ728E PSM Send Program failed to obtain socket. TCPIP error number: *eeee*

Explanation: While attempting to send a request to a ProcOps Service Machine (PSM), the PSM Send Program failed to obtain an IP socket.

The variable *eeee* gives the non-zero TCPIP error number returned by TCPIP.

System action: The request is not sent to the PSM. An error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, then reissue the request if appropriate.

ISQ729E Message Receiver handshake failed for socket *ss* to PSM *pppppppp*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has established a TCPIP connection. The PSM has accepted the connection and

has sent back a handshaking sequence. The sequence sent back does not match that sent.

The variable *ss* gives the socket number used.

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

System action: The connection is broken and the PSM Message Receiver program terminates. All target systems on the PSM are closed.

Operator response: Report the problem to your system programmer.

System programmer response: Check that the authorization tokens used at both ends of the connection match. If necessary make the necessary changes.

For NetView, the Customization Dialogs must be used, and a new ProcOps control file must be built. ProcOps is then refreshed with this updated control file.

For the PSM, stop the PSM, update the ISQPARM DATA file, and restart the PSM.

ISQ730E Unrecoverable error for socket *ssss* to PSM *pppppppp*. Send Program ending.

Explanation: While attempting to send a request to a ProcOps Service Machine (PSM), the PSM Send Program encountered an error that prevented further processing. This message is preceded by other messages that give details of the error.

The variable *ssss* gives the socket number of the socket being used. In some cases the error may be encountered before a socket has been obtained. In these cases the socket number displayed is -1.

The variable *pppppppp* gives the name of the PSM.

System action: An error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful. The request may or may not have been sent to the PSM.

Operator response: Messages preceding this indicate in more detail the nature of the problem. Use these to respond to the error, if appropriate. Otherwise, report the problem to your system programmer.

System programmer response: Messages preceding this indicate in more detail the nature of the problem. Use these to respond to the error.

ISQ731I CONNECT for socket *sss* gave TCPIP error number *eeee*.

Explanation: While attempting to send a request to a ProcOps Service Machine (PSM), the PSM Send Program issued a TCPIP CONNECT to set up a connection to its associated PSM. The CONNECT has completed with an error that prevents further processing.

The variable *sss* gives the socket number being used.

The variable *eeee* gives the non-zero TCPIP error number returned by the connect call.

System action: The request is not sent to the PSM. The Send Program terminates and an error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, then reissue the request if appropriate.

ISQ732E Session on socket *sss* to PSM *pppppppp* has been closed by PSM.

Explanation: While attempting to send a request to a ProcOps Service Machine (PSM), the PSM Send Program has received a signal from the PSM that the connection has been forcibly reset, possibly as a result of the termination of the PSM or a thread of the PSM.

The variable *sss* gives the socket number used.

The variable *pppppppp* gives the name of the PSM.

System action: The request may or may not have been sent to the PSM. Some, all, or none of the responses from the PSM may have been received. The Send Program terminates and an error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Check whether the PSM terminated the connection as a result of a normal termination request. If it did not, investigate why the PSM issued the termination. Otherwise, report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, then reissue the request if appropriate.

ISQ733I RECV request on socket *ssss* for PSM *pppppppp* gave TCPIP error number *eeee*

Explanation: While sending a request to a ProcOps Service Machine (PSM), the PSM Send Program has attempted to read a response. The TCPIP RECV that was issued ended in error.

The variable *sss* gives the socket number being used.

The variable *pppppppp* gives the name of the PSM.

The variable *eeee* gives the non-zero TCPIP error number returned by the RECV call.

System action: The request has been sent to the PSM. Some, all, or none of the responses from the PSM may

have been received. The Send Program terminates and an error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, then reissue the request if appropriate.

ISQ734E WRITE request on socket *ssss* gave TCPIP error *eeee*

Explanation: While sending a request to a ProcOps Service Machine (PSM), the PSM Send Program issued a TCPIP WRITE call. The call ended in error.

The variable *sss* gives the socket number being used.

The variable *eeee* gives the non-zero TCPIP error number returned by the RECV call.

System action: The request may or may not have been received and executed by the PSM. No responses from the PSM have been received. The Send Program terminates and an error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, then reissue the request if appropriate.

ISQ735I PSM *pppppppp* has authorization token *ttttttt*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM Message Receiver program ISQVMRCV has been started in its Message Monitoring Task (MMT).

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

The variable *ttttttt* gives the authorization token that is being used to validate the caller to the PSM.

System action: Normal processing continues.

Operator response: None.

System programmer response: None.

ISQ736E Message traffic (including heartbeats) from PSM *pppppppp* has stopped.

Explanation: Once a TCPIP session is established between a ProcOps Service Machine (PSM), and its Receive Program in NetView, the Receive Program expects to receive either normal message traffic or heartbeat messages at regular intervals. No such messages have been received for an extended period.

The variable *pppppppp* gives the name of the PSM that this copy of the program is communicating with.

System action: The Receive Program terminates. All target systems on the PSM are closed.

Operator response: Check that the PSM is still operating. If the PSM or one of its threads has failed, correct the error and then re-initialize the active target systems on the PSM using ISQXIII commands

System programmer response: None

ISQ738E PSM Send Program timed out waiting to read for PSM *pppppppp*.

Explanation: While processing a request to a ProcOps Service Machine (PSM), the PSM Send Program timed out when waiting for responses to the request it sent to the PSM.

The variable *pppppppp* gives the name of the PSM.

System action: The request may or may not have been sent to the PSM. Some, all, or none of any possible responses may have already been received (as ISQ700I messages). The Send Program terminates and an error return code is given to the caller.

If this occurred as a result of a command such as ISQXIII or ISQSTART, the PSM and its guest target systems enter error status.

If this occurred as a result of a ProcOps common command, the command may or may not have executed.

Operator response: Ensure that the PSM is initialized and can be reached through the IP network. When resolved, and if appropriate, check the status of the target systems on the PSM, and reissue the command that failed, if appropriate.

System programmer response: None.

ISQ739E Cannot contact PSM *pppppppp*. Likely PSM not active or bad IP address.

Explanation: While sending a request to a ProcOps Service Machine (PSM), the PSM Send Program issued a TCPIP WRITE. The WRITE ended with TCPIP error ECONNREFUSED. The attempt to connect to a socket was refused because there was not a process listening or because the queue of connection requests was full

and the underlying protocol does not support retransmissions.

The variable *pppppppp* gives the name of the PSM.

System action: The request has not been received and executed by the PSM. No responses from the PSM have been received. The Send Program terminates and an error return code is given to the caller. Higher level commands calling the PSM Send Program are likely to be unsuccessful.

Operator response: Check that the PSM is active. If necessary, start the PSM. Then reenter the request, if appropriate. If this is not successful, report the error to your system programmer.

System programmer response: Check that the IP address that is specified for the PSM in the customization dialogs is correct and that the build has created the correct ProcOps configuration in the automation control file (ACF), or ProcOps control file for systems earlier than SA z/OS 3.3. Make sure the updated ACF information gets activated on the focal point system. Recycle ProcOps with the new configuration information.

ISQ750E An attempt to set the TCPIP stack gave error number: *eeee*

Explanation: As part of the processing of a request to a ProcOps Service Machine (PSM), or as part of the initialization of the PSM Receive Program, an attempt was made to choose the active TCPIP image stack that the application will connect to. The call was unsuccessful.

The variable *eee* gives the non-zero TCPIP error number returned by the SETIBMOPT call.

System action: The program (either the PSM Send Program or the PSM Receive Program) terminates. In the case of the PSM Send Program, no request is sent.

Operator response: Likely values for the TCPIP error number include 1011 (EIBMBADTCPNAME), which indicates that the desired image stack is not active. If appropriate, start the job or system task that corresponds to this stack. Otherwise, report the error to your system programmer.

System programmer response: Refer to the *errno.h* header file for a complete description of the error number returned. Use this to correct the error, then reissue the request if appropriate.

If the stack name is incorrect, use the customization dialogs to set a new TCPIP Stack name for the PSM. Use the Build step to create the correct ProcOps configuration in the automation control file (ACF), or ProcOps control file for systems earlier than SA z/OS 3.3. Make sure the updated ACF information gets activated on the focal point system. Recycle ProcOps with the new configuration information.

ISQ751E **ISQGSMSG routine returned non-zero return code: *rr***

Explanation: A message has been received by the ProcOps Service Machine (PSM) Receive Program. The ISQGSMSG service routine has been called to route it to interested operators. The routine failed.

The variable *rr* gives the non-zero return code from the ISQGSMSG routine.

System action: The message is not routed, displayed or logged.

Operator response: Report the error to your system programmer.

System programmer response: Report the problem to your IBM Service Center.

ISQ752E **The program failed to resolve IP address for host *hhhh*. TCPIP error number: *rr***

Explanation: As part of the initialization of the ProcOps Service Machine (PSM) Send or Receive Program, an attempt was made to resolve the IP address for PSM host *hhhh*. The call to the LE getaddrinfo routine was unsuccessful.

The variable *rr* is the non-zero return code from the getaddrinfo routine.

System action: The program (either the PSM Send Program or the PSM Receive Program) terminates. In the case of the PSM Send Program, no request is sent.

Operator response: Report the error to your system programmer.

System programmer response: Check that the host name specified for the PSM in the customization dialogs is correct. Use the Build step to create the correct ProcOps configuration in the automation control file (ACF), or ProcOps control file for systems earlier than SA z/OS 3.3. Make sure the updated ACF information gets activated on the focal point system. Recycle ProcOps with the new configuration information.

ISQ753I **PSM *pppppppp* forces IPv4 protocol for message traffic. PSMIPv4 flag is *ff***

Explanation: The ProcOps Service Machine (PSM) Send Program and PSM Receive Program force will use IPv4 protocol for communication with PSM host.

The variable *pppppppp* gives the name of the PSM.

The variable *rr* gives the value of the **Force to use IPv4 for Hostname** flag in the ProcOps Service Machine Information panel of the customization dialog.

System action: Normal processing continues using the IPv4 protocol.

Operator response: None.

System programmer response: None.

ISQ800I *originator console_designator message.*

Explanation: This message appears whenever SA z/OS detects a message related to a target ensemble.

The variable *message* contains the message detected by SA z/OS, including both the message identifier and the message text. Messages that are detected include:

- Messages issued by SA z/OS that refer to a target ensemble
- New notification messages received from ensemble HMC regarding a set of predefined management events. These events include:
 - Addition and removal of managed objects to/from the inventory of resources that are managed by the HMC
 - Changes to specified properties of managed object instances
 - Changes to the operational status of managed objects
 - Completion of asynchronously processed jobs.

The variable *originator* contains the originator information of the message. It has the following format:

- target ensemble name

The variable *console_designator* contains the type of console the message comes from:

- SC designates hardware system console

ISQ800I messages are written to the NetView log. They are not displayed to an operator.

ISQ800I are used to cause the processing of SA z/OS-supplied automation routines. They also can be used to cause the processing of user-written automation routines.

System action: Automation processing is triggered by the ISQ800I prefix. The remainder of the ISQ800I message determines which (if any) automation routine is processed.

Operator response: None.

System programmer response: None.

ISQ801I *originator console_designator message.*

Explanation: This message appears whenever SA z/OS detects a message related to a target ensemble.

The variable *message* contains the message detected by SA z/OS, including both the message identifier and the message text. Messages that are detected include:

- Messages issued by SA z/OS that refer to a target ensemble

- New notification messages received from ensemble HMC regarding a set of predefined management events. These events include:
 - Addition and removal of managed objects to/from the inventory of resources that are managed by HMC.
 - Changes to specified properties of managed object instances
 - Changes to the operational status of managed objects
 - Completion of asynchronously processed jobs.

The variable *originator* contains the originator information of the message. It has the following format:

- target ensemble name

The variable *console_designator* contains the type of console the message comes from:

- SC designates hardware system console

Each ISQ801I message is displayed only to operators defined as interested operators for the target system that the message is associated with. Interested operators are defined using the configuration dialogs and the ISQXMON command.

System action: None.

Operator response: None.

System programmer response: None.

ISQ850I ISQECMD: TARGET NOT SPECIFIED OR INVALID TARGET SPECIFICATION

Explanation: This message appears in response to the ISQECMD command. A target system name was not specified in the command or is invalid. A valid target system must be specified.

System action: The ISQECMD command is rejected.

Operator response: Reenter the command, making sure to specify a valid target system name.

System programmer response: None.

ISQ851I ISQECMD *ccmd* PROCESSING ON *thwname* IS COMPLETE

Explanation: Processing of the common command *ccmd* is complete. The target hardware *thwname* is now ready to process new requests.

System action: Processing of the ISQECMD command is complete.

Operator response: None.

System programmer response: None.

ISQ852I ISQECMD: INVALID COMBINATION OF PARAMETERS *parm1* AND *parm2* - ONLY ONE PARAMETER ALLOWED.

Explanation: This message appears in response to the ISQECMD command. A command was issued with an invalid combination of command parameters.

System action: The ISQECMD command is rejected.

Operator response: Reenter the command, making sure to specify only one parameter.

System programmer response: None.

ISQ853I ISQECMD: THE STRING *string* IS AN UNRECOGNIZED PARAMETER FOR COMMAND *ccmd*

Explanation: This message appears in response to the ISQECMD command. A command was issued with a parameter not valid for the command.

System action: The ISQECMD command is rejected.

Operator response: Reenter the command, making sure to specify the parameter correctly.

System programmer response: None.

ISQ854I ISQECMD: 'CPC' PARAMETER VALUE *value* MUST MATCH TARGET HARDWARE NAME *thwname*

Explanation: This message appears in response to the ISQECMD command. A command was issued with a Target Hardware Name as command target, but a 'CPC(*cpc-name*)' parameter was provided as well with a different name.

System action: The ISQECMD command is rejected.

Operator response: Reenter the command, making sure that Target Hardware Name and '*cpc-name*' of the CPC parameter match.

System programmer response: None.

ISQ855I ISQECMD: TARGET IS NOT AN ENSEMBLE MEMBER.

Explanation: This message appears in response to the ISQECMD command. The Ensemble Name or Target Hardware Name provided with the command could not be resolved as a valid ensemble member.

System action: The ISQECMD command is rejected.

Operator response: Reenter the command, making sure that the Ensemble Name or Target Hardware belongs to one of the defined ensembles.

System programmer response: None.

ISQ856I ISQECMD: *ccmd* TSTIME(*time-stamp*)
STATUS(*ccmd-status*)

Explanation: This message appears in response to the ISQECMD command. It shows the status of the command executed:

SUCCESS
REJECTED
FAILED

System action: Processing of the ISQECMD command is complete.

Operator response: Analyse *ccmd-status*; if REJECTED or FAILED, look for other messages indicating the reason why the command was not processed successfully.

System programmer response: None.

ISQ857I ISQECMD: 'CPC' *cpc-name* IS NOT
PART OF ENSEMBLE *ensemble-name*

Explanation: This message appears in response to the ISQECMD command. The CPC parameter provided is not defined as being part of the ensemble the command is executed for.

System action: The ISQECMD is rejected.

Operator response: Reenter the command, making sure that *cpc-name* is part of the ensemble.

System programmer response: None.

ISQ858I ISQECMD: CONNECTION TO
ENSEMBLE *ensemble-name* NOT
ESTABLISHED

Explanation: This message appears in response to an ISQECMD command request with an ensemble name as parameter. There is no connection established to this ensemble and the command is not executed.

System action: None.

Operator response: To perform the necessary initialization, use the ISQXIII command, followed by a valid ensemble name. This establishes a connection to the ensemble and then the ISQECMD command request can be retried.

System programmer response: None.

ISQ859I ISQECMD: ENSEMBLE *ensemble-name* IS
LOCKED BY USER *user-id*

Explanation: This message appears in response to an ISQECMD command. The ensemble the command is executed for, is locked by another user and the command is rejected.

System action: The ISQECMD is rejected.

Operator response: Remove the lock using command

ISQXUNL or ISQOVRD and reenter the command.

System programmer response: None.

ISQ900I *originator console_designator message.*

Explanation: This message appears whenever SA z/OS detects a message related to a target system.

The variable *message* contains the message detected by SA z/OS, including both the message identifier and the message text. Messages that are detected include:

- Messages issued by SA z/OS that refer to a target system
- New message lines on a target system console, including multiline messages.

The variable *originator* contains the originator information of the message. It can have the following format:

- target system name
- target hardware name.partition name

The variable *console_designator* contains the type of console the message comes from:

- OC designates operating system console.
- SC designates hardware system console.

ISQ900I messages are written to the NetView log. They are not displayed to an operator.

ISQ900I messages are used to cause the processing of SA z/OS-supplied automation routines. They also can be used to cause the processing of user-written automation routines.

System action: Automation processing is triggered by the ISQ900I prefix. The remainder of the ISQ900I message determines which (if any) automation routine is processed.

Operator response: None.

System programmer response: None.

ISQ901I *originator console_designator message.*

Explanation: This message appears whenever SA z/OS detects a message related to a target system.

The variable *message* contains the message detected by SA z/OS, including both the message identifier and the message text. Messages that are detected include:

- Messages issued by SA z/OS that refer to a target system.
- New message lines on a target system console, including multiline messages.

The variable *originator* contains the originator information of the message. It can have the following format:

- target system name
- target hardware name.partition name

Messages ISQ001I to ISQ901I

The variable *console_designator* contains the type of console the message comes from:

- OC designates operating system console.
- SC designates hardware system console.

Each ISQ901I message is displayed only to operators defined as *interested operators* for the target system that the message is associated with. *Interested operators* are defined using the configuration dialogs and the ISQXMON command.

System action: None.

Operator response: None.

System programmer response: None.

Chapter 11. Messages ISQX998I and ISQX999I

ISQX998I *mmmmmmmmmm 00000000 F aaaaaaaa llllll*

Explanation: This is a trace message from the processor operations storage management. This message appears only if the debug flag (ISQ#MTRC) in the communication vector map is set. The debug flag can be set with the DEBUG parameter in the ISQSTART and ISQSTOP commands, or dynamically by using the ISQTCC command and switching the monitor on.

The variable *mmmmmmmmmm* shows the name of the Csect issuing the message.

The variable *00000000* shows the name of the NetView task (TVBOPID).

The variable *F* shows the Get /Free indicator where:

- E** A DSIFRE macro with MAINTSK=YES was issued
- F** A DSIFRE macro was issued
- G** A DSIGET macro was issued
- H** A DSIGET macro with MAINTSK=YES was issued

The variable *aaaaaaa* shows the address of allocated or freed storage.

The variable *llllll* shows the last six digits (decimal value) of the bytes allocated or freed from the DSIGET/DSIFRE macro.

System action: The interval timer device driver starts. Initialization of the PC continues.

Operator response: None.

System programmer response: None.

ISQX999I *mmmmmmmmmm 00000000 F lllllll rrrr*

Explanation: This is the continuation message of message ISQX998. It is issued in the following cases:

- A nonzero return code was set by the DSIGET / DSIFRE macro.
- The number of bytes of allocated or freed storage is greater than 999999 (decimal value). In this case the whole length will be displayed.

The variable *mmmmmmmmmm* shows the name of the Csect issuing the message.

The variable *00000000* shows the name of the NetView task (TVBOPID).

The variable *F* shows the Get Free indicator where:

- E** A DSIFRE macro with MAINTSK=YES was issued.
- F** A DSIFRE macro was issued.
- G** A DSIGET macro was issued.
- H** A DSIGET macro with MAINTSK=YES was issued.

The variable *llllllll* shows the number of bytes (decimal value) of allocated or freed storage.

The variable *rrrr* shows the return code from the DSIGET / DSIFRE macro.

System action: The interval timer device driver starts. Initialization of the PC continues.

Operator response: None.

System programmer response: None.

Messages ISQX998I and ISQX999I

Chapter 12. Messages ISQtt001I to ISQtt0723I

The following messages are issued by the PSM application. The message ID has the following form:

ISQttnnnns *text*

where *tt* is the component (thread) that issued the message. Possible values are:

MA	Main thread
CS	Command Server
MS	Message Server
MH	Message Handler
CN	Console Server
LG	Logger

In many cases a message is issued only by one thread and this value is shown in the description. In some cases it may be issued by two or more threads, and the message ID shows a generic format:

nnnn is the 4-digit message number

s is the severity of the message

I	Information
W	Warning
E	Error

The messages in this chapter are sorted according to 4-digit message number.

ISQMA0001I PSM Main Thread entered.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the main thread of its multitasking control program has started.

System action: Normal processing continues

Operator response: None. This is an informational message only.

System programmer response: None.

ISQMA0002I Waiting for Command Server thread to initialize.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the main thread of its multitasking control program has issued a request to start the Command Server thread. The main thread waits for the Command Server thread to initialize before starting the Logger and Console server threads.

System action: Normal processing continues.

Operator response: If this message is not followed by messages indicating that the Command Server has

started processing, report this to your system programmer.

System programmer response: If this message is not followed by messages indicating that the Command Server has started processing, restart the program with the Command Server trace activated.

ISQMA0005I Main Thread ending (others may still be running).

Explanation: The Main Thread of the PSM control program is ending. It has attempted earlier to stop all other threads, and these may, or may not, have terminated successfully.

System action: Normal processing continues

Operator response: If the CMS console then issues a "Ready" message, all other threads have ended. To terminate any running threads enter a HX command followed by a BEGIN command. Repeat this sequence of commands until the "Ready" message is issued. At this point the PSM control program can be restarted.

System programmer response: None.

ISQMA0101I Thread *ttt* started as *aaaaaaa*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), or as the result of a request to the PSM, the PSM control program has successfully started a thread.

The variable *ttt* gives the numerical ID of the thread that has started.

The variable *aaaaa* gives the description of the thread. It may be one of: CSERV (Command Server), MSERV (Message Server), MH (Message Handler), LOGGER (Logger) or CNSERVER (Console Server).

System action: Normal processing continues.

Operator response: Ensure that the thread indicated subsequently initializes correctly. If it does not, report the problem to your system programmer.

System programmer response: None. If a thread does not initialize correctly and there are no messages that indicate an obvious cause, set the trace for the failing component and restart the PSM Control program.

ISQMA0103I *aaaaa* thread cannot be started. Reason code *rrrr*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), or as the result of request to the PSM, the PSM control program has attempted to start a new thread. The request was unsuccessful.

The variable *aaaaa* gives the description of the thread. It may be one of: CSERV (Command Server), MSERV (Message Server), MH (Message Handler), LOGGER (Logger) or CNSERVER (Console Server).

The variable *rrrr* gives the reason code returned by the call to ThreadCreate.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual. Use this to resolve the problem, if possible. Otherwise report the problem to your IBM Service Center.

ISQMA0104E Command Server thread failed during its initialization.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the PSM control program started the Command Server thread. The thread then terminated unexpectedly.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Report the problem to your IBM Service Center.

ISQMA0106E EventMonitorCreate failed with reason code *rrrr*. Call was *ccccccc*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), or as the result of request to the PSM, the PSM control program has attempted to create a new multitasking event monitor. The request was unsuccessful.

The variable *rrrr* gives the reason code returned by the call to EventMonitorCreate.

The variable *ccccccc* gives the name of the event monitor. This is an internal name used by the PSM Control Program.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual. Use this to resolve the problem, if possible. Otherwise report the problem to your IBM Service Center.

ISQMA0107E SIGNAL routine failed with reason code *rrrr* for event *eeeeeee*

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), or as the result of request to the PSM, the PSM control program has attempted to signal (set) an event. The request was unsuccessful.

The variable *rrrr* gives the reason code returned by the call to EventSignal.

The variable *eeeeeee* gives the name of the event. This is an internal name used by the PSM Control Program.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual. Use this to resolve the problem, if possible. Otherwise report the problem to your IBM Service Center.

ISQMA0110E ADDEVENT routine failed adding event *eeeeee*. No space in table.

Explanation: As part of the initialization of a ProcOps Service Machine (PSM), the internal ADDEVENT service was attempted for an event. This is an internal failure.

The variable *eeeeee* gives the name of the event.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: This is an internal failure. Report the problem to your IBM Service Center.

**ISQMA0111E Thread aaaaa could not be deleted.
Reason code rrrr. Continuing...**

Explanation: As part of the shutdown of a ProcOps Service Machine (PSM), or as the result of a request to the PSM, the PSM control program has attempted to delete an existing thread. The request was unsuccessful.

The variable *aaaaa* gives the description of the thread. It may be one of: CSERV (Command Server), MSERV (Message Server), MH (Message Handler), LOGGER (Logger) or, CNSERVER (Console Server).

The variable *rrrr* gives the reason code returned by the call to ThreadDelete.

System action: The PSM control program continues.

Operator response: The thread may be left running. If so, use the CP command HX (halt execution) to terminate it. If several of these messages are given, several uses of the HX command may be needed. The PSM control program may not be restartable without a re-IPL of the PSM virtual machine or a re-logout of the PSM virtual machine. Report the problem to your system programmer.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual. Use this to resolve the problem, if possible. Otherwise report the problem to your IBM Service Center.

**ISQCS0200E Command Server cannot start.
TERMINATE failed with result: rrrr**

Explanation: As part of the initialization of the Command Server thread, a TCPIP TERMINATE was issued. The TERMINATE failed.

The variable *rrrr* is a string containing a nonzero return code, an error name, and an error message.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program.

**ISQCS0201E Command Server cannot start.
INITIALIZE failed with result: rrrr**

Explanation: As part of the initialization of the Command Server thread, a TCPIP INITIALIZE was issued. The INITIALIZE failed.

The variable *rrrr* is a string containing a nonzero return code, an error name, and an error message.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program.

ISQCS0204E No valid focal point addresses found.

Explanation: As part of the initialization of the Command Server thread, an attempt was made to access the ISQADDRS DATA file in order to read the list of valid IP addresses for ProcOps NetView focal points. No data set was found.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Create an ISQADDRS DATA file with at least one valid entry. Restart the PSM control program in its virtual machine.

**ISQCS0205I Command Server valid FP addresses are
nnn.nnn.nnn.nnn, nnn.nnn.nnn.nnn, ...**

Explanation: As part of the initialization of the Command Server thread, the ISQADDRS DATA file was read. The quoted list of valid IP addresses for ProcOps NetView focal points was found. Where an entry in the file was in node name form, the equivalent dotted decimal form of its address is given.

The variable *nnn.nnn.nnn.nnn* is one of the following:

- The dotted decimal IPv4 address as read from the file
- The hexadecimal IPv6 address as read from the file
- The dotted decimal IPv4 address as resolved from the node name

Node names in the IPv6 environment are not yet supported.

System action: Execution continues.

Operator response: None. This is an informational message.

System programmer response: None.

ISQCS0208I Session from *nnn.nnn.nnn.nnn* has invalid address. Connection refused.

Explanation:

The variable *nnn.nnn.nnn.nnn* is the client IP address that attempts to connect to the PSM.

System action: Execution continues.

Operator response: None. This is an informational message.

System programmer response: None.

ISQCS0209E In ISQADDRS file, *aaaaa* could not be resolved as a valid address.

Explanation: As part of the initialization of the Command Server thread, the ISQADDRS DATA file was read in order to create a list of valid IP addresses for ProcOps NetView focal points. The entry quoted is neither a valid dotted decimal address nor could it be resolved as a valid address by a GETHOSTBYNAME call.

The variable *aaaaaa* is the entry as read from the file.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Correct the entry in the ISQADDRS DATA file or ensure that the appropriate entry exists in the files used by TCPIP to resolve names to IP addresses. Restart the PSM control program in its virtual machine.

ISQCS0210E Errors found in ISQADDRS file.

Explanation: As part of the initialization of the Command Server thread, the ISQADDRS DATA file was read in order to create a list of valid IP addresses for ProcOps NetView focal points. Errors were found in one or more entry.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: See any associated ISQCS0209E and ISQCS0211E messages for details. Correct the entries in the ISQADDRS DATA file or ensure that the appropriate entry exists in the files used by TCPIP to resolve name to IP addresses. Restart the PSM control program in its virtual machine.

ISQCS0211E In ISQADDRS file, *nnn.nnn.nnn.nnn* is not a valid dotted decimal address.

Explanation: As part of the initialization of the Command Server thread, the ISQADDRS DATA file was read in order to create a list of valid IP addresses for ProcOps NetView focal points. The entry quoted

was entered in dotted decimal form, but is not a valid IP address.

The variable *nnn.nnn.nnn.nnn* is the entry as read from the file.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Correct the entry in the ISQADDRS DATA file. Restart the PSM control program in its virtual machine.

ISQCS0212E ACCEPT call for socket *ss* gave unexpected result: *rrrrr*

Explanation: In issuing an ACCEPT for an incoming CONNECT the Command Server received an unexpected result and was unable to complete the connection.

The variable *ss* is the socket number in use.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program.

ISQCS0213W *nn.nnn.nnn.nnn*

Explanation: The Command Server thread has received an internal request to start focal point operations to a NetView that is not its current focal point. That is, it already has a running connection between the Message Server and another NetView.

This message is sent on the existing connection immediately before it is terminated. (It is not displayed on the VM console.) It effectively informs the current ProcOps focal point NetView that a new ProcOps focal point has been established.

The variable *nn.nnn.nnn.nnn* is the dotted decimal IP address of the new ProcOps focal point NetView.

System action: Immediately following this message, the Message Server connection to the current focal point NetView is terminated. On the current focal point NetView, target systems on the PSM will then go to a target system status of CLOSED.

Operator response: None.

System programmer response: None.

ISQCS0214E Attempt to RECEIVE request for socket *ss* gave result: *rrrr*

Explanation: In issuing an RECV for a connection with a ProcOps NetView the Command Server received an unexpected result and was unable to continue with the connection.

The variable *ss* is the socket number in use.

The variable *rrrr* is the TCPIP reason that resulted (this is displayed as a number and text description).

System action: The request from the ProcOps NetView is not processed. The connection with the ProcOps NetView is closed.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, reenter the request or restart the PSM Control Program, whichever is appropriate.

ISQCS0215W TCPIP session on socket *ss* canceled by NetView node.

Explanation: In issuing an RECV for a session with a ProcOps NetView the Command Server received a record of length zero. This usually indicates that the connection has been closed by the peer node (ProcOps NetView).

The variable *ss* is the socket number in use.

System action: The request from the ProcOps NetView is not processed. The connection with the ProcOps NetView is closed.

Operator response: Report the problem to your system programmer.

System programmer response: Investigate whether the receipt of a zero length message resulted from a session being closed at the ProcOps NetView end. If so, this is normal. Otherwise determine the causes of the zero length message.

ISQCS0216I Command *cccc* executed. Return code = *nn*

Explanation: The Command Server has executed a request received from either a remote ProcOps NetView or the CMS console. The request was interpreted to be a CP or CMS command and was executed on the PSM's virtual machine.

The variable *cccc* is the full text of the command executed, including all blanks and special characters.

The variable *nn* is the return code that resulted from the execution of this command. Because the command is executed using a CMS PIPE command,

the return code resulting may be that of the PIPE command. For example, an invalid command gives a return code of -27.

System action: No special processing is done if the return code is not zero.

Operator response: None. This is an informational message.

System programmer response: None.

ISQCS0217I Main response is <*rrrrrrrr*>

Explanation: The Command Server has executed a request from a remote ProcOps NetView and has produced output (none of which is yet sent). There may be several lines of output to be sent back to the remote NetView.

The variable *rrrrrrrr* is the full text of the first (and main) response message to be sent, including all blanks and special characters.

System action: The Command Server then attempts to send the responses back to the NetView that entered the request, using TCPIP WRITE requests.

Operator response: None. This is an informational message.

System programmer response: None.

ISQCS0218I Request *rrrrrr* done.

Explanation: The Command Server has received a special ProcOps request from either a remote ProcOps NetView or the VM console. The request has been executed without any observed problems.

The variable *rrrrrr* is the full text of the request, including all blanks and special characters.

System action: Other messages, that result from the actual execution of the request, may also be returned to the caller.

Operator response: None. This is an informational message.

System programmer response: None.

ISQCS0219I *rrrrrr* cannot be done. No previous FP known.

Explanation: The Command Server has received a request from the VM console. The request cannot be done because it requires that a previous focal point be known. For example, an INITCOM request from the VM console that no FP is given in cannot be performed if there was no previous known focal point that the Message Server can be connected to.

The variable *rrrrrr* is the full text of the request, including all blanks and special characters.

System action: The request is not done.

Operator response: Reenter the request. Specify an IP address in the INITCOM request from the console; or, issue the INITCOM request from a ProcOps NetView. In the latter case, the IP address of the ProcOps NetView is its implicit operand.

System programmer response: None.

ISQCS0220I Send request for response on socket *ss* gave result: *rrrr*

Explanation: The Command Server has completed processing a request from a remote ProcOps NetView. In attempting to send one of the response messages created, a TCPIP SEND request has been issued, but has failed with the indicated result.

The variable *ss* is the socket number in use.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The request may or may not have executed completely. Some earlier responses to the request may have been sent successfully. Termination of the request ends. The Command Server closes the connection and makes itself available to receive further requests.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible, then reenter the request and restart the PSM.

ISQCS0221I No parameters in ISQPARM DATA file, or no file.

Explanation: As part of the initialization of the PSM, an attempt has been made to access and read the ISQPARM DATA file. The request has been unsuccessful. Either there is no file, or there are no valid data records in the file.

System action: The PSM terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Ensure that there is a ISQPARM DATA file and that it contains the necessary definitions. When done, restart the PSM.

ISQCS0222I No specification for *sssssss* in ISQPARM DATA file.

Explanation: As part of the initialization of the PSM, the ISQPARM DATA file has been read. A required definition for the keyword indicated has not been found.

The variable *sssssss* is the required keyword that was not found.

System action: The PSM terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Ensure that there is a definition in the ISQPARM DATA file for the required keyword. When done, restart the PSM.

ISQCS0223I Invalid component *cccccc*

Explanation: An ISQTRACE or ISQACT request has been sent to the PSM. The PSM component (thread) referenced in the request is not a valid value. No action is taken.

The variable *cccccc* is the component value entered.

System action: The request terminates.

Operator response: Check that the request syntax is correct and that one of the valid values was used. Reenter the request.

System programmer response: None.

ISQCS0224I Trace for *cccccc* set to *vvv*

Explanation: An ISQTRACE special request has been sent to the PSM. The PSM has changed the trace setting for the component.

The variable *cccccc* is the component (thread) value entered.

The variable *vvv* is the requested trace setting of the PSM component: ON or OFF.

System action: The request completes successfully.

Operator response: None.

System programmer response: None.

ISQCS0225W *tttttt* not known as a valid thread.

Explanation: As part of the execution of either a TERMCOM or INITCOM request to the PSM, an attempt was made to delete a thread for a PSM component. The thread appeared to be not running.

The variable *tttttt* is the name of the PSM component's thread.

System action: Execution continues.

Operator response: If there are subsequent problems, report this message to your system programmer.

System programmer response: If the ProcOps termination, or initialization does not complete correctly for this PSM, check that the indicated thread was really inactive.

ISQCS0226I Message Queue has *nnn* members.

Explanation: An ISQUERY Q request was entered to determine the number of messages in the PSM's Message Queue.

The variable *nnn* is the number of messages in the queue.

System action: The request completes.

Operator response: None.

System programmer response: None.

ISQCS0227I List of Events and Event Monitors Defined.

Explanation: An ISQCMD Q E request was entered to display the events and event monitors currently in use by the PSM. This is the header line for the responses received. This is followed by two lists. One list contains the names of all events known to the PSM Control program (including all its threads). The other list contains the numerical IDs of the event monitors known. In some cases information about the event monitor is also shown.

System action: The request is completed.

Operator response: None.

System programmer response: None.

ISQCS0228E Invalid value *vvvv* in request *rrrrrr*

Explanation: A special request was received by the PSM. The value of one of its operands was incorrect.

The variable *vvvv* is the invalid value entered in the request.

The variable *rrrrrr* is the full text of the request, including all blanks and special characters.

System action: The request is not executed.

Operator response: If this is the result of an operator request, check for correct syntax. Reenter the request. If this is not an operator request, report the problem to your system programmer.

System programmer response: Report the problem to your IBM Service Center.

ISQCS0229E Component cannot be started. Still active.

Explanation: An ISQACT request was received by the PSM to start a thread. The thread is already apparently running.

The variable *vvvv* is the invalid value entered in the request.

The variable *rrrrrr* is the full text of the request, including all blanks and special characters.

System action: The request is not executed.

Operator response: If this is the result of an operator request, check for correct syntax. Reenter the request. If this is not an operator request, report the problem to your system programmer.

System programmer response: Report the problem to your IBM Service Center.

ISQCS0230E *q* is invalid queue designator. Use M (messages) or C (cmds).

Explanation: An ISQCLEAR request was received by the PSM to clear a queue. The value entered for the queue designator is invalid.

The variable *q* is the invalid value entered for the queue designator.

System action: The request is not executed.

Operator response: Determine why the thread is still running.

System programmer response: None.

ISQCS0231E Queue cleared.

Explanation: An ISQCLEAR request was received by the PSM to clear a queue. All entries in the queue are now deleted.

System action: The request is complete.

Operator response: None.

System programmer response: None.

ISQCS0232E ISQCLEAR failed with reason: *rrrr*

Explanation: An ISQCLEAR request was received by the PSM to clear a queue. During the request a QueueReceiveImmed call was issued but did not complete successfully.

The variable *rrrr* is the reason code returned as a result of the QueueReceiveImmed request.

System action: The request does not complete. Some of the entries on the queue may have been deleted. Some entries may still remain on the queue. The request is terminated and the PSM awaits further requests. Future actions involving the queue may or may not execute correctly.

Operator response: Report the problem to your system programmer.

System programmer response: The appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual contains explanations for the reason code returned. Use this to determine the cause of the message. If necessary, contact IBM for service.

ISQCS0233E SELECT request ended in error. Result:
rrrr

Explanation: The PSM Command Server is waiting for a connection to a ProcOps NetView. As part of this process, a TCPIP SELECT call was issued. The call gave the error result indicated.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The Command Server thread of the PSM terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program. If this is not possible, contact IBM for service.

ISQCS0234E Message Server initialized, but not Message Handler.

Explanation: As part of the initialization of a connection to a PSM, an internal INITCOM request was sent to the PSM. The PSM Message Server thread was initialized but its required Message Handler thread appeared not to initialize.

System action: No further action is taken.

Operator response: Investigate other messages issued by the PSM, specifically any from the Message Handler thread to determine the cause. One possible reason for this message is that the thread did initialize, but took longer to do so than the Command Server waited. If possible, resolve the problem and (re-)issue the ISQXIII request for the target system. If the problem persists, stop the PSM using the STOPALL request, and restart the PSM virtual machine. Then reissue the ISQXIII command for the target system. If the problem persists report the problem to your system programmer.

System programmer response: Contact IBM for service.

ISQCS0235E Message Handler initialized, but not Message Server.

Explanation: As part of the initialization of a connection to a PSM, an internal INITCOM request was sent to the PSM. The PSM Message Handler thread was initialized but its required Message Server thread appeared not to initialize.

System action: No further action is taken.

Operator response: Investigate other messages issued by the PSM, specifically any from the Message Server thread to determine the cause. One possible reason for this message is that the thread did initialize, but took

longer to do so than the Command Server waited. If possible, resolve the problem and (re-)issue the ISQXIII request for the target system. If the problem persists, stop the PSM using the STOPALL request, and restart the PSM virtual machine. Then reissue the ISQXIII command for the target system. If the problem persists report the problem to your system programmer.

System programmer response: Contact IBM for service.

ISQCS0236E Neither Message Server nor Message Handler initialized.

Explanation: As part of the initialization of a connection to a PSM, an internal INITCOM request was sent to the PSM. Neither the PSM Message Handler nor the Message Server thread appeared to be initialized correctly.

System action: No further action is taken.

Operator response: Investigate other messages issued by the PSM, to determine the cause. One possible reason for this message is that the threads did initialize, but took longer to do so than the Command Server waited. If possible, resolve the problem and (re-)issue the ISQXIII request for the target system. If the problem persists, stop the PSM using the STOPALL request, and restart the PSM virtual machine. Then reissue the ISQXIII command. If the problem persists report the problem to your system programmer.

System programmer response: Contact IBM for service.

ISQCS0237W TERMCOM may not have succeeded.

Explanation: As part of the termination of a connection to a PSM, an internal TERMCOM request was sent to the PSM. Either the PSM Message Handler, or the Message Server thread, or both appeared to be not terminated correctly. This message may also be issued when a connection to a PSM is initialized, because any existing message processing is terminated first.

System action: No further action is taken.

Operator response: Investigate other messages issued by the PSM, to determine the cause. One possible reason for this message is that the threads did terminate but that the indications of their termination were not set correctly. Another possible reason is that one or both threads were not active. If this is part of a full termination (STOPALL) check that all threads terminate on the VM machine and that it returns to a CMS prompt. In this case the threads did terminate correctly.

Otherwise, at the VM console of the PSM, issue the HX command repeatedly to terminate any threads until the CMS prompt can be achieved with a BEGIN command. The PSM can then be restarted if desired.

If this is part of an ISQXCLS or ISQSTOP request, enter an INITCOM request followed by a TERMCOM request and check that these commands execute correctly.

If this is part of an ISQXIII request, it is followed by an attempt to start the Message Handler and Message Server threads. If these initialize correctly, all is well, and the message can be ignored.

If the problem persists report the problem to your system programmer.

System programmer response: Contact IBM for service.

ISQCS0238W ttttttt thread may not have stopped.

Explanation: As part of the termination of a connection to a PSM, an internal TERMCOM request was sent to the PSM. Either the PSM Message Handler, or the Message Server thread appeared not to terminate correctly. This message may be issued when a connection to a PSM is initialized, because any existing communication is then terminated first.

System action: No further action is taken.

Operator response: Investigate other messages issued by the PSM, to determine the cause. One possible reason for this message is that the threads did terminate but that the indications of their termination were not set correctly. If this is part of a full termination (STOPALL) check that all threads terminate on the VM machine and that it returns to a CMS prompt. In this case the threads did terminate correctly.

Otherwise, at the VM console of the PSM, issue the HX command repeatedly to terminate any threads until the CMS prompt can be achieved with a BEGIN command. The PSM can then be restarted if desired.

If this is part of an ISQXCLS or ISQSTOP request, enter an INITCOM request followed by a TERMCOM request and check that these commands execute correctly.

If the problem persists report the problem to your system programmer.

System programmer response: Contact IBM for service.

ISQCS0239E QueueReceiveImmed request for COMMAND queue gave result: rrrr

Explanation: Internally within the PSM, any PSM request entered at the VM console is first entered on a COMMAND queue and later read from the queue to be executed. The read from the queue failed.

The variable *rrrr* gives the reason code returned by the call to QueueReceiveImmed.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer. Restart the PSM.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual. Use this to resolve the problem, if possible. Otherwise report the problem to your IBM Service Center.

ISQCS0240E Invalid operand: 0000. Enter Q (Message Queue) or E (Events).

Explanation: An ISQQUERY special request was issued to the PSM, but the operand specified was not valid.

The variable *0000* is the invalid operand.

System action: The request is not done.

Operator response: Correct the syntax and reenter the request.

System programmer response: None.

ISQCS0242W Command SET SECUSER gave rc = rrrr

Explanation: As part of the processing for an ISQXIII or ISQXCLS request, the CP SET SECUSER command is used to change the secondary user specification of a guest machine that is a target system. The command gave an unexpected return code. (This may also be seen when ISQSTART and ISQSTOP are done.)

The variable *rrrr* is the return code from the CP SET SECUSER command.

System action: Operation continues.

Operator response: See the *CP Command and Utility Reference* manual for the explanation of the return code. (This is the numeric part of the HCP*nnn* message.) If necessary, inform your system programmer.

System programmer response: None.

ISQCS0244E Authorization token does not match. Request not done.

Explanation: A request was sent to the PSM from a ProcOps NetView. The authorization token embedded in the request did not match that expected by the PSM.

System action: The request is not done.

Operator response: Inform your system programmer.

System programmer response: Check that the authorization token for the PSM, which is specified in the ProcOps customization dialog, matches that specified for the SECURITY keyword in the ISQPARM DATA file on the PSM.

If the change is needed on ProcOps, correct the dialogs, rebuild the ProcOps control file and restart ProcOps using the new control file.

If the change is needed on the PSM, stop the PSM by

ISQCS0245E • ISQMS0301I

issuing ISQXCLS for all its active target systems. Correct the ISQPARM DATA file and restart the PSM. Then issue ISQXIII for one or more of its target systems, as desired.

ISQCS0245E Cannot close PSM itself.

Explanation: An internal CLOSE request was sent to the PSM. The guest machine name in the CLOSE request was the name of the PSM. This is not valid.

System action: The request is not done.

Operator response: None. This is an invalid request.

System programmer response: Report the problem to your IBM Service Center.

ISQCS0246E Request rejected. Remote node cannot INITCOM for another node.

Explanation: The PSM has received an internal INITCOM request from a ProcOps NetView. The INITCOM has requested initialization for a node other than its own. This is not valid.

System action: The request is not done.

Operator response: Inform your system programmer.

System programmer response: Report the problem to your IBM Service Center.

ISQCS0248W Waiting for TCPIP virtual machine <vvvvvvvv> to be active.

Explanation: The PSM has started up. Before it can do its TCPIP initialization, it checks whether the desired TCPIP virtual machine is running. The desired virtual machine is not running.

The variable *vvvvvvvv* is the name of the desired TCPIP virtual machine.

System action: The PSM control program continues to check, at regular intervals, whether the desired TCPIP virtual machine is active. When it finds the machine active, normal initialization is continued. No further ISQCS0248W messages are issued

Operator response: This message should normally only be issued when the VM system is starting up and a PSM is started before TCPIP becomes active.

If it occurs at other times log on to the VM system and check that the desired TCPIP virtual machine is active by issuing a CP QUERY command (for example, CP QUERY TCPIP).

The desired machine name is set in the ISQPARM DATA file. If no value is set in this file, a default name of TCPIP is used.

If necessary, stop the PSM by issuing the HX command repeatedly until the CMS prompt is displayed.

System programmer response: Check that the value

of the TCPIPNAME parameter in the ISQPARM DATA file is set correctly. If necessary, correct it and restart the PSM.

ISQCS0249I Guest...Status...

Explanation: The PSM has received an internal PSMDATA request. This gives status of the PSM and guest machines. This is the heading line of the response and is followed by ISQCS0250I data messages.

System action: None.

Operator response: None.

System programmer response: None.

ISQCS0251I Request <rrrrrrrr> received from remote NetView.

Explanation: The Command Server has received a request from a remote ProcOps NetView.

The variable *rrrrrrrr* is the full text of the command executed, including all blanks and special characters.

System action: None.

Operator response: None.

System programmer response: None.

ISQCS0252I Component... Status...

Explanation: The PSM has received an ISQSTATUS request. This acts as a heading line for subsequent ISQCS0253I messages that give the component name and status of parts of the PSM application.

System action: None.

Operator response: None.

System programmer response: None.

ISQCS0254I Invalid value <vvvvv> for ISQPARM parameter <ppppp>.

Explanation: As part of PSM initialization, the ISQPARM file has been read. The value specified for a parameter is invalid.

System action: The PSM terminates.

Operator response: Report the problem to your system programmer

System programmer response: Correct the parameter in error. Restart the PSM program.

ISQMS0301I Message Server waiting for connection to NetView.

Explanation: The Message Server thread in the PSM has been started and has issued a TCPIP LISTEN

request. It awaits a ProcOps NetView CONNECT request.

System action: The PSM awaits a ProcOps NetView CONNECT request.

Operator response: When an ISQXIII or ISQSTART is issued on the controlling ProcOps NetView, messages indicating that a connection has been established should be issued by the PSM. If they are not, check for other error messages on the ProcOps NetView. If necessary, inform your system programmer.

System programmer response: Check the NetView NETLOG for any messages indicating why the connection is not made.

ISQMS0302E ACCEPT issued by Message Server gave result: rrrrr

Explanation: In issuing an ACCEPT for an incoming CONNECT the Message Server received an unexpected result and was unable to complete the connection.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The Message Server thread terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program.

ISQMS0303I Session accepted by Message Server. New session socket: ss

Explanation: The Message Server thread received an incoming CONNECT.

The variable *ss* is the TCPIP socket number assigned to the new connection created by the CONNECT.

System action: The Message Server continues to create the connection.

Operator response: None.

System programmer response: None.

ISQt0304W SETSOCKOPT issued by cccccc thread for socket ss gave result rrrr

Explanation: As part of the process of initializing a TCPIP socket, a thread has issued a TCPIP SETSOCKOPT call to set the options for the connection. An unexpected result has been returned.

The variable *ccccc* is the name of the PSM thread that was running.

The variable *ss* is the socket number in use.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The thread continues. The connection does not have all its desired options. This can cause other errors, especially if the thread or the PSM is stopped and restarted within a short period of time.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program.

ISQMS0305I Handshake complete for Message Server connection.

Explanation: As part of the process of setting up a incoming connection, the Message Server receives and checks an authorization token from the ProcOps NetView. This has been done successfully. There is now a fully established connection between the PSM and ProcOps NetView for the receipt of message traffic.

System action: The PSM begins to send any messages it receives to ProcOps NetView.

Operator response: None.

System programmer response: None.

ISQMS0306I TCPIP error on Message Server connection. Result rrrr. Restarting.

Explanation: In attempting to maintain a connection between its Message Server and a ProcOps NetView, the Message Server has sustained a TCPIP error. Other error messages from the Message Server appear before this.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The Message Server attempts to restart the session to the same ProcOps NetView.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible.

ISQMS0307E QueueOpen for queue qqqq gave RC = cccc. Reason = rrrrrr

Explanation: In attempting to maintain a connection between its Message Server and a ProcOps NetView, the Message Server has sustained a TCPIP error. Other error messages from the Message Server appear before this.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The Message Server attempts to restart the session to the same ProcOps NetView.

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible.

ISQtt0308E LISTEN by cccccc on socket ss failed.
Result: rrrrr

Explanation: As part of the process of setting up a incoming connection, a thread has issued a TCPIP LISTEN call to await a connection. An unexpected result has been returned.

The variable *ccccc* is the name of the PSM thread that was running.

The variable *ss* is the socket number in use.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The thread terminates. (If the thread is the Command Server, the PSM terminates.)

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program. Otherwise, contact IBM for service.

ISQMS0309E GETHOSTID issued by cccccc failed.
Result: rrrr

Explanation: As part of the process of initializing TCPIP, a thread has issued a TCPIP GETHOSTID call. An unexpected result has been returned.

The variable *ccccc* is the name of the PSM thread that was running.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The thread terminates. (if the thread is the Command Server, the PSM terminates.)

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program. Otherwise, contact IBM for service.

ISQtt0310E SOCKET issued by cccccc failed. Result:
rrrrr

Explanation: As part of the process of obtaining a TCPIP socket, a thread has issued a TCPIP SOCKET call. An unexpected result has been returned.

The variable *ccccc* is the name of the PSM thread that was running.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The thread terminates. (If the thread is the Command Server, the PSM terminates.)

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program. Otherwise, contact IBM for service.

ISQtt0311E BIND issued by cccccc for socket ss failed. Result: rrrrr

Explanation: As part of the process of obtaining a TCPIP socket, a thread has issued a TCPIP BIND call. An unexpected result has been returned.

The variable *ccccc* is the name of the PSM thread that was running.

The variable *ss* is the socket number.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The thread terminates. (If the thread is the Command Server, the PSM terminates.)

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program. Otherwise, contact IBM for service.

ISQMS0312I QueueReceiveImmed request for MESSAGES queue gave result: rrrr

Explanation: The Message Server has issued a QueueReceiveImmed call to obtain messages from the Messages Queue. An unexpected result has been returned.

The variable *rrrr* is the reason code.

System action: The Message Server thread terminates.

Operator response: Report the problem to your system programmer.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual. Use this to resolve the problem, if possible, then restart the message processing threads. If this is not successful, stop and restart the PSM. In both cases messages stored on the queue will be lost. Report the problem to your IBM Service Center.

ISQt0313E FCNTL issued by cccccc for socket ss failed. Result: rrrrr

Explanation: As part of the process of obtaining a TCPIP socket, a thread has issued a TCPIP FCNTL call to set the socket in non-blocking mode. An unexpected result has been returned.

The variable *cccccc* is the name of the PSM thread that was running.

The variable *ss* is the socket number.

The variable *rrrrr* is the TCPIP result (this is displayed as a number and text description).

System action: The thread terminates. (If the thread is the Command Server, the PSM terminates.)

Operator response: Report the problem to your system programmer.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible. When corrected, restart the PSM Control Program. Otherwise, contact IBM for service.

ISQMS0314E Message Handler has failed.

Explanation: The Message Server has received a signal that the Message Handler has failed. This message is sent as an unsolicited message to the ProcOps NetView.

System action: The message is used by ProcOps NetView to set the status of the target systems managed by the PSM.

Operator response: Check why the Message Handler has failed. (Possible reasons include the filling of the MESSAGES Queue.) To restart the Message Handler use the ISQACT MH command from the PSM.

System programmer response: None.

ISQMS0315E CONNECT received from node nn.nn.nn.nn that is not current FP: mm.mmm.mmm

Explanation: The Message Server has received a request to CONNECT from a node that is not the current ProcOps focal point. That is, it is not the node that last issued a INITCOM request.

This can occur if an ISQSTART or ISQXIII request is issued from one ProcOps NetView without issuing a

ISQXCLS or ISQSTOP command on another ProcOps NetView that is the current focal point.

The variable *nn.nn.nn* is the dotted decimal IP address of the node attempting to CONNECT.

The variable *mm.mmm.mmm* is the dotted decimal IP address of the node that last issued an INITCOM request to this PSM. (An INITCOM request is issued when a ISQXIII command is first issued for a target system on that PSM. It can also be issued when ISQSTART command is issued, if there are target systems defined to initialize automatically.)

System action: The request to connect is denied.

Operator response: Check why the other PSM is trying to connect. If the wrong ProcOps NetView is the current focal point, issue a ISQSTOP request (or ISQXCLS requests for the PSM's target systems) from that NetView. Then issue ISQSTART (or ISQXIII for one or more target systems on that PSM) from the NetView that is to be the focal point.

System programmer response: None.

ISQMS0316I Message Handler has started.

Explanation: The Message Server has received a signal that the Message Handler has started. This message is sent as an unsolicited message to the ProcOps NetView.

System action: The message is used by ProcOps NetView to set the status of the target systems managed by the PSM.

Operator response: None.

System programmer response: None.

ISQMS0317E WRITE for socket ss failed with TCPIP error: rrrrr

Explanation: The Message Server thread has issued a TCPIP WRITE request to send data to the ProcOps NetView. The request failed.

The variable *ss* is the socket number.

The variable *rrrrr* is the return code from the request

System action: The Message Server thread terminates. Target systems on the PSM are closed.

Operator response: Report the problem to your system programmer. Issue ISQXIII requests for all target systems on the PSM that were active at the time of the failure. Ensure that the requests are successful.

System programmer response: Refer to the "REXX Sockets Return Codes" section of the *REXX/VM Reference* manual for a description of the return code given. Use this to resolve the error, if possible.

ISQMS0318E WRITE for socket *ss* timed out.

Explanation: The Message Server thread has issued a TCPIP WRITE request to send data to the ProcOps NetView. The request could not be satisfied immediately. The Message Server waited, but after an interval of time the WRITE request was still not possible.

The variable *ss* is the socket number.

System action: The Message Server thread terminates. Target systems on the PSM are closed.

Operator response: Report the problem to your system programmer. Issue ISQXIII requests for all target systems on the PSM that were active at the time of the failure. Ensure that the requests are successful.

System programmer response: None.

ISQt0401E QueueSend for *qqqqq* failed with RC = *cccc*. Reason: *rrrr*

Explanation: A PSM thread issued a QueueSend to put an entry (message or command) on to the indicated queue. It received an unexpected return code.

The variable *qqqq* is the name of the queue.

The variable *cccc* is the return code.

The variable *rrrr* is the reason code.

System action: The thread terminates.

Operator response: Report the problem to your system programmer.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual. Use this to resolve the problem, if possible. Otherwise report the problem to your IBM Service Center.

ISQt0402I Command <*cccccc*> entered.

Explanation: The Console Server thread received a request and put it on the Command Queue.

The variable *cccccc* is the command, entered in full.

System action: The thread continues. The command should be then be processed by the Command Server thread.

Operator response: None.

System programmer response: None.

ISQLG0505E Logger failed to retrieve traced data RC = *cccc*. Reason = *rrrr*.

Explanation: The Logger thread issued an EventRetrieve to obtain details of an event that was signaled to it. It received an unexpected return code.

The variable *cccc* is the return code.

The variable *rrrr* is the reason code.

System action: The Logger thread ignores the event and it is not logged.

Operator response: None.

System programmer response: None.

ISQMH0601E Unrecoverable error for Message Handler pipe stage. *MSG Service may be damaged and not restartable.

Explanation: The Message Handler PIPE stage (ISQRGIUC) has encountered a problem that means it cannot continue. Other error messages should precede this, indicating details of the actual error.

System action: The stage completes. This, in turn, should cause the Message Handler thread to complete. Because of the nature of the failure it may not be possible to recycle the message processing threads, and it may be necessary to stop and restart the PSM virtual machine.

Operator response: See the earlier messages to determine the actual problem.

System programmer response: See the earlier messages to determine the actual problem.

ISQMH0602E MESSAGES Queue has reached its maximum size. Queuing stopped.

Explanation: The Message Handler PIPE stage (ISQRGIUC) has determined that the number of messages in the Messages Queue exceeds the threshold value specified by the installation in the ISQPARM DATA file.

System action: Queuing of messages is halted. The PIPE stage completes. This should cause the Message Handler thread to complete.

Operator response: Determine whether the cause lies elsewhere. (For example, a connection to the ProcOps NetView may be lost.) Restart message processing when the problem is cleared, by entering the ISQXIII command for all target systems on the PSM that should be active. This will delete any messages in the queue at the time.

System programmer response: Review the threshold number of messages specified. Change the value specified in ISQPARM DATA and recycle the PSM, if necessary.

ISQt0701E CREATE_EVENT_MONITOR routine called with no events for *mmmmmmmmmm*

Explanation: A call was made to the internal CREATE_EVENT_MONITOR routine, but no event names were specified.

The variable *mmmmmmmmmm* is the name used within the thread to identify the event monitor to be created.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0702E EventMonitorCreate request for *mmmmmmmmmm* ended with RC = *cccc*. Reason: *rrrr*.

Explanation: A call was made to the internal CREATE_EVENT_MONITOR routine, but the EventMonitorCreate call it issued gave an unexpected result.

The variable *mmmmmmmmmm* is the name used within the thread to identify the event monitor.

The variable *cccc* is the return code from the EventMonitorCreate call.

The variable *rrrr* is the reason code.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0703E EventWait request for *mmmmmmmmmm* resulted in RC= *cccc*. Reason = *rrrr*

Explanation: A call was made to the internal WAIT_FOR_EVENT routine, but the EventWait call it issued gave an unexpected result.

The variable *mmmmmmmmmm* is the name used within the thread to identify the event monitor.

The variable *cccc* is the return code from the EventWait call.

The variable *rrrr* is the reason code.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0704E EventWait call for *mmmmmmmmmm* resulted in event flag setting *ssss* for *eeeeeeee*

Explanation: A call was made to the internal WAIT_FOR_EVENT routine, but the EventWait call that it issued resulted in an error setting for one of the events that were waited upon.

The variable *mmmmmmmmmm* is the name used within the thread to identify the event monitor.

The variable *ssss* is the unexpected setting that resulted.

The variable *eeeeeeee* is the name of the event.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0705E CREATE_EVENT routine called with too many or no operands: <*0000*>.

Explanation: A call was made to the internal CREATE_EVENT routine, but either no event name was specified, or more than one event name was specified.

The variable *0000* is the operands, if any, specified in the call.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0706E EventCreate request for event *eeeeeeee* ended with RC = *cccc*. Reason: *rrrr*

Explanation: A call was made to the internal CREATE_EVENT routine, but the EventCreate call it issued gave an unexpected result.

The variable *eeeeeeee* is the event name.

The variable *cccc* is the return code from the EventCreate call.

The variable *rrrr* is the reason code

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQtt0707E *ccccccc* thread has suffered a fatal error and is ending.

Explanation: A thread indicated has suffered a fatal error that means that it cannot continue.

The variable *ccccccc* is the name of the thread.

System action: The thread terminates.

Operator response: This message follows other messages that detail the actual problem. If necessary, report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQtt0708I *ccccccc* thread canceled.

Explanation: A request has been entered that cancels the indicated thread. The request has been signaled to the thread and it is beginning its processing for an orderly termination.

The variable *ccccccc* is the name of the thread.

System action: The thread terminates.

Operator response: None.

System programmer response: None.

ISQtt0709I *ccccccc* thread ending...

Explanation: Either as a result of a fatal error or a cancel request, a thread has ended. (This message is issued by the thread itself as its last action, so the thread should be deleted immediately afterwards.)

The variable *ccccccc* is the name of the thread.

System action: The thread terminates.

Operator response: If this is issued as a result of a fatal error, earlier messages detail the error.

System programmer response: None.

ISQtt0710I *ccccccc* thread entered.

Explanation: A thread has begun execution.

The variable *ccccccc* is the name of the thread.

System action: The thread continues.

Operator response: None.

System programmer response: None.

ISQtt0711I *ccccccc* thread initialized.

Explanation: A thread has completed its initialization.

The variable *ccccccc* is the name of the thread.

System action: The thread continues.

Operator response: None.

System programmer response: None.

ISQtt0712W EvMonReset *mmmmmmmmmm* issued by *ccccccc* gave RC = *cccc*. Reason *rrrrr*

Explanation: A call was made to the internal DELETE_EVENT_MONITOR routine. The EventMonitorReset call it issued gave an unexpected result.

The variable *mmmmmmmmmm* is the name used within the thread to identify the event monitor.

The variable *ccccccc* is the name of the thread.

The variable *cccc* is the return code from the EventMonitorReset call.

The variable *rrrrr* is the reason code.

System action: The thread continues.

Operator response: Report the problem to your system programmer.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application Multitasking* manual. Use this to resolve the problem, if possible. Otherwise report the problem to your IBM Service Center.

ISQtt0713W EvMonDelete *mmmmmmmmmm* issued by *ccccccc* gave RC = *cccc*. Reason *rrrrr*

Explanation: A call was made to the internal DELETE_EVENT_MONITOR routine. The EventMonitorDelete call it issued gave an unexpected result.

The variable *mmmmmmmmmm* is the name used within the thread to identify the event monitor.

The variable *ccccccc* is the name of the thread.

The variable *cccc* is the return code from the EventMonitorDelete call.

The variable *rrrrr* is the reason code.

System action: The thread continues.

Operator response: Report the problem to your system programmer.

System programmer response: The explanation of the reason code is found in the appendix "Return and Reason Code Values" in the *CMS Application*

Multitasking manual. Use this to resolve the problem, if possible. Otherwise report the problem to your IBM Service Center.

ISQt0714W EventSignal request by *ccccccc* for *eeeeeee* gave RC = *ccc*. Reason: *rrrr*

Explanation: A call was made to the internal SIGNAL_EVENT routine. The EventSignal call it issued gave an unexpected result.

- The variable *eeeeeee* is the event name.
- The variable *cccccc* is the name of the thread.
- The variable *ccc* is the return code from the EventMonitorDelete call.
- The variable *rrrr* is the reason code.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0715W QueueCreate for *qqqqqqqq* failed with RC = *ccc*. Reason: *rrrr*

Explanation: A call was made to an internal routine to create a multi-tasking queue. The QueueCreate call it issued gave an unexpected result.

- The variable *qqqqqqqq* is the name of the queue.
- The variable *cccccc* is the name of the thread.
- The variable *ccc* is the return code from the QueueCreate call.
- The variable *rrrr* is the reason code.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0717E NOVALUE error in line *llll*

Explanation:

Note: Always followed by message ISQt0718E.

- The variable *llll* is the line number in the REXX program that gave the condition.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0718E Statement in error is: *stmt*

Explanation:

- The variable *stmt* is the REXX statement the condition occurred in.

ISQt0719E SYNTAX error *tttt* in line *llll*: *xxxxx*

Explanation:

Note: Always followed by Message ISQt0718E.

- The variable *tttt* is the REXX error type.
- The variable *llll* is the line number in the REXX program the condition occurred in.
- The variable *xxxxx* is an explanation of the REXX error type.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0720E ERROR condition raised in statement *llll* RC: *ccc*

Explanation:

Note: Always followed by Message ISQt0718E.

- The variable *llll* is the line number of the statement in the REXX program that caused the error.
- The variable *ccc* is the return code from the function or command in error.

System action: The thread terminates.

Operator response: Report the problem to your system programmer. If the thread is the Logger or Console Server, message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0721E Recursive error in *ttttttt*. Thread ending.

Explanation: An recursive error has occurred in a thread. The thread is ending.

The variable *ttttttt* is the name of the thread.

System action: The thread terminates. Note that this message is issued only in its English language version. This is because a likely cause of recursive errors is when error messages are created.

Operator response: Report the problem to your system programmer. Earlier error messages may be seen that identify the original problem. If the failing thread is the Logger or Console Server, message processing can continue, but the facilities of the failed thread are not available. If any other thread fails, stop the PSM (if it has not stopped) and restart it.

System programmer response: Report the problem to your IBM Service Center.

ISQt0722E In ISQADDRS file, *xxxxxx:xxxxxx* is not a valid IPv6 hexadecimal address.

Explanation: As part of the initialization of the Command Server thread, the ISQADDRS DATA file was read in order to create a list of valid IP addresses for ProcOps NetView focal points. The entry quoted was entered in IPv6 hexadecimal form, but is not a valid IP address.

The variable *xxxxxx:xxxxxx* is the entry as interpreted from the file.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Correct the entry in the ISQADDRS DATA file. Restart the PSM control program in its virtual machine.

ISQt0723I In ISQADDRS file, *aaaaa* could not be resolved as a valid hexadecimal address in IPv6 network.

Explanation: As part of the initialization of the Command Server thread in the IPv6 environment, the ISQADDRS DATA file was read in order to create a list of valid IP addresses for ProcOps NetView focal points. The entry quoted cannot be resolved as a valid hexadecimal IPv6 address. Host names are not yet supported in the IPv6 environment.

The variable *aaaaa* is the entry as read from the file.

System action: The PSM control program terminates.

Operator response: Report the problem to your system programmer.

System programmer response: Correct the entry in the ISQADDRS DATA file. Use the correct hexadecimal IPv6 address or force the IPv4 protocol by defining the entry PSMIPv4=Y in the ISQPARM DATA file to resolve host names to IPv4 addresses. Restart the PSM control program in its virtual machine.

Appendix A. Return Codes, Status Codes, and Reason Codes

I/O Operations Return Codes

I/O operations returns the following return codes (RCs) to indicate the completion status of its commands. As you can see from the following table, they adhere to conventional practice.

Return Code	Explanation
0	Command completed successfully.
4	Command completed with a warning.
8	Command failed.
9	GETMAIN failure occurred. No storage is available at the host to send command output to the I/O operations workstation feature.
12	I/O operations was busy.
16	Command parameter error.
20	I/O operations system error.

I/O Operations Status Codes for a Dynamic Switch

Status Code	Explanation
00	Operational.
04	In contention (another user has changed information at the switch).
08	Read only (I/O operations can provide configuration information about this switch but cannot change switch configuration settings).
1C	Switch not open.
20	Allocate problem.
24	GETMAIN failure.
28	FREEMAIN failure.
3C	Page fix failure.
44	Page unfix failure.
48	I/O error.
4C	Hardware error (switch unit check).
52	UCB lookup failure.
58	Swapping failure.

I/O Operations Reason Codes

Together with a return code, I/O operations sends an 8-character reason code in hexadecimal notation. The following tables list the reason codes indexed by their corresponding return codes.

Reason Codes Associated With Several Return Codes

Reason Code	Explanation	See Message
20xx0000	Ready to accept commands.	IHVO000I
20xx0001	Empty string or empty parameter list.	IHVO001I

Return Codes, Status Codes, and Reason Codes

Reason Code	Explanation	See Message
20xx0002	Command name not valid.	IHVO002I
20xx0003	Command entered with no operands or options.	IHVO003I
20xx0004	Parameter greater than 24 characters.	IHVO004I
20xx0005	Option not valid.	IHVO005I
20xx0006	Conflicting options (for example, Force and Noforce specified with the same command).	IHVO006I
20xx0007	Too many parameters or operands.	IHVO007I
20xx0008	1st and 2nd operands both equal * (for example, Display Name *)	IHVO008I
20xx0009	Port address not valid.	IHVO009I
20xx000A	Operand not valid.	IHVO010I
20xx000B	Required operands missing.	IHVO011I
20xx000C	A name of asterisk (*) cannot be assigned to a port.	IHVO012I
20xx000E	Command not valid on this interface.	IHVO014I
20xx0010	Connectivity option specified with a nonconnectivity command.	IHVO016I
20xx0011	Combination not valid. Novary and Noforce specified with the same command.	IHVO017I
20xx0012	Combination not valid. Backout, Force, and Vary specified with the same command.	IHVO018I
20xx0013	Combination not valid. The Only option cannot be specified with the Allow command when specifying a 2nd operand of * or (*).	IHVO019I
20xx0014	Combination not valid. The options Only and Also can be specified with the Allow command only.	IHVO020I
20xx0015	Combination not valid. The options Disable and Nodisable can be specified with the Remove Switch command only.	IHVO021I
20xx0016	Combination not valid. The options Nocheck and Novary cannot be specified with the same command.	IHVO022I
20xx0017	Combination not valid. The options Process and Noprocess can be specified with the Sync Switch command only.	IHVO023I
20xx0018	Combination not valid. The options Sync and Fail can be specified with the Display Vary command only.	IHVO024I
20xx0019	Combination not valid. The options Force, Novary, and Backout cannot be specified with the same command.	IHVO025I
20xx0020	Combination not valid. The options Block and Connect can be specified with the Display Switch, Display Port, or Display Name commands only.	IHVO032I
20xx0021	Combination not valid. The options Force and Novary cannot be used on a Remove Chp or Restore Chp command, or a Pathtest Nopathtest option was used on a command other than an allowable Query Relation command.	IHVO033I
20xx0030	Combination not valid. The options On, Off, and Purge can be specified with the Reset Host command only.	IHVO048I
20xx0040	I/O operations already started.	IHVO064I
20xx0041	I/O operations never started.	IHVO065I
20xx00FC	Bad ESTAE environment.	IHVO252E
20xx00FD	Bad communication input buffer.	IHVO253E
20xx00FE	GETMAIN failed.	IHVO900I
20xx00FF	System operator interface cannot process any more commands.	IHVO255E
A0000005	Message ID does not exist.	IHVL001I
A0000006	Missing variable in message.	None
A0000007	Variable type not valid.	IHVL003I

Return Codes, Status Codes, and Reason Codes

Reason Code	Explanation	See Message
A000008	Parameter list sent to the central message facility was not valid.	IHVL002I

Reason Codes Associated With RC=0

Reason Code	Explanation	See Message
20xx0000	Ready to accept commands.	IHVO000I
40000001	Unlock request for process already unlocked.	None
40000003	Getlock request, but you have already locked command processing.	None
40000004	No one currently owns the lock.	None
D0xx0000	Command completed successfully.	IHVC000I

Reason Codes Associated With RC=4 or RC=8

Reason Code	Explanation	See Message
15xx000A	(RC=8)	
15xx0022	(RC=4)	
15xx0023	(RC=8)	
30xx0003	I/O operations is not logged on.	IHVV003I IHVV029I
30xx0005	GETMAIN failed.	IHVV027I
30xx0006	IUCV path to I/O operations failed.	IHVV004I
30xx0007	User not IUCV authorized.	IHVV005I
30xx0008	Data overflow from IUCV.	IHVV
30xx0010	IUCV reply message not found.	IHVV023I
30F0xxxx	RACROUTE failed. xxxx = return code from RACROUTE.	IHVV011I
30FCxxxx	ESTAE failed. xxxx = return code from the ESTAE.	IHVV011I
30B0xxxx	TSO service failed. xxxx = return code from TSO.	IHVV012I
30B1xxxx	TSO service abend. xxxx = abend code from TSO.	IHVV012I
50000001	Name in request not valid.	None
50000002	User failed RACF request.	IHVC100I
50000003	Link failed.	None
50000004	Command code not valid.	None
50000005	Requested message not found.	IHVL001I IHVZ999I
50000006	Specified switch not found.	IHVC821I
50000007	Specified VTAM name not found.	IHVC802I
50000008	No switches defined.	IHVC400I
50000009	Specified switch was not affected.	IHVC403I
5000000A	One or more hosts rejected the command.	IHVC401I
5000000B	No switches were affected by the command.	IHVC402I
5000000C	Switch data changed; database not refreshed.	IHVC428I
5000000D	Specified CHPID not found.	IHVC810I
5000000E	Specified device not found.	IHVC820I
5000000F	Specified port name not found.	IHVC840I
50000011	Switch data changed; database refreshed.	IHVC421I

Return Codes, Status Codes, and Reason Codes

Reason Code	Explanation	See Message
50000012	Specified port not found.	IHVC850I
50000013	Purge failed for resident host.	IHVC905I
50000014	One or more switches rejected the command.	IHVC406I
50000015	One or more hosts and switches failed.	IHVC407I
50000016	Incorrect use of connectivity command.	IHVC418I
50000017	Port name was not found on a switch.	IHVC427I
50000018	Tried Display Vary outside the range of 1-1500.	IHVC870I
50000019	No blocked port names.	IHVC841I
5000001A	No connected port names.	IHVC842I
5000001B	No blocked ports.	IHVC851I
5000001C	No connected ports.	IHVC852I
5000001D	Specified device not found on the specified switch.	IHVC823I
5000001E	There are no port names on the switch.	IHVC242I
5000001F	Switch not opened.	IHVC902I
50000020	Combination of request bits not valid.	IHVC602I
50000021	No Varys to display.	IHVC270I
50000022	FREEMAIN failure.	IHVC990I
50000023	GETMAIN failure.	IHVC900I
50000025	One or more switches had an error on a Display command.	IHVC230I
50000026	A user entered Getlock and processed a command.	IHVC422I
50000027	Channel subsystem configuration data changed.	IHVC423I
50000028	Channel subsystem configuration and switch data changed.	IHVC424I
50000029	Channel subsystem refresh failed.	IHVC425I
5000002A	Unable to obtain current switch and channel configuration data.	IHVC426I
5000002B	Remove not enabled.	IHVC429I
5000002C	Switch was removed.	IHVC430I
5000002D	Switch was restored.	IHVC432I
5000002E	No Sync Varys to display.	IHVC878I
5000002F	No Sync or Fail Varys to display.	IHVC876I
50000030	Hosts issued warnings.	IHVC615I
50000031	Not all switches were affected by the command.	IHVC617I
50000032	No Writewch request bits were set.	IHVC628I
50000033	More than one * assigned in name.	IHVC618I
50000034	More than 512 ports affected by the command.	IHVC619I
50000035	The user is not the lock owner.	IHVC630I
50000036	Data overflowed on the command.	IHVC881I
50000037	Starting number is greater than the ending number.	IHVC882I
50000038	SA z/OS system error.	None
50000039	No device numbers in specified range.	IHVC883I
5000003A	User does not own the lock and the data overflowed.	IHVC631I
5000003B	Switch does not support the required function.	IHVC640I
5000003C	File name not valid.	IHVC642I
5000003D	Switch file in use.	IHVC644I
5000003E	File descriptor not valid.	IHVC643I
5000003F	Error in Writefile data block.	IHVC650I
50000040	Ports not implemented.	None

Return Codes, Status Codes, and Reason Codes

Reason Code	Explanation	See Message
50000041	Maximum number of files saved at switch.	IHVC654I
50000042	File name does not exist on switch.	IHVC648I
50000043	Required resources not available to access file.	IHVC655I
50000044	Cannot delete IPL file at switch.	IHVC656I
50000045	VTAM name is reset off.	IHVC540I
50000046	Switch not initialized.	None
50000047	Command, operand, or option not supported on this host's ESCM level.	None
50000048	The command failed because I/O operations can provide configuration information about this switch but cannot change switch configuration settings.	IHVC437I
50000049	Command affects Inter-Switch-Link port.	IHVC913I
5000004A	Attempt to "allow" Control Unit Port (X'FE') to itself, API input data has been corrected.	None
51001004	Some host responded with a warning.	None
A5010001	All buffer slots are full, complete wrapping has occurred.	ING001I
A5020008	I/O operations API command failed.	ING001I
A5020012	I/O operations returned a 'busy' on an API command request.	ING001I
A5020016	I/O operations API rejected command.	ING001I
A5020020	I/O operations returned a system error on an API command request.	ING001I
A5030001	Storage request failed.	ING001I
A5030004	FREEMAIN failed.	ING001I
A5040001	Storage request failed.	ING001I
A5040008	I/O operations API command failed.	ING001I
A5210020	Release of I/O operations API module failed.	ING001I
A5240001	Storage request failed.	ING001I
D0xx0089	On a call from a REXX EXEC, insufficient storage is available to return the output of the command in IHVRESP. I/O operations returns a 24-character field where the leftmost 10 bytes contain the decimal number of bytes of command output that I/O operations attempted to return.	None
D5xx0003	FREEMAIN failed (RC=4).	
D5xx0004	Invalid input header area (RC=8).	
D5xx0005	Input buffer too small (RC=8).	
D5xx0006	Invalid command code (RC=8).	
D5xx0007	GETMAIN failed for block (RC=8).	
D5xx0008	GETMAIN failed for a CMPK 'pool' (RC=8).	
D5xx000A	Maximum number of storage pools obtained (RC=8).	
D5xx000B	External control block overflow (RC=4 or 8).	
D5xx000C	Some information not returned for the graphic view request (RC=4).	
D5xx000D	No information returned for the graphic view request (RC=8).	
E0xx0001	Connection between I/O operations and *CONFIG CP system services failed.	IHVZ720I
FFxx0001	MODESET to user key failed.	IHVU011I
FFxx0002	FREEMAIN failed.	IHVU011I
FFxx0003	MVCL instruction failed.	IHVU011I
FFxx0004	User canceled the Wait state.	IHVU011I

Return Codes, Status Codes, and Reason Codes

Reason Code	Explanation	See Message
FFxx0005	SCHEDULE failed.	IHVU011I
FFxx0006	MVCK instruction failed.	IHVU011I
FFxx0007	GETMAIN failed.	IHVU011I
FFxx0008	MODESET to key 0 failed.	IHVU011I
FFxx0009	RACROUTE failed.	IHVU011I
FFxxFFFF	Main I/O operations program failed.	IHVD999E

Reason Codes Associated With RC=12

Reason Code	Explanation	See Message
30xx0001	Another user processing a command.	IHVC514 IHVU019I
40000001	Another user processing a command.	IHVS001I
40000002	Another user locked command processing.	IHVS002I
40000003	Another user locked command processing through Getlock.	IHVS003I

Reason Codes Associated With RC=16

Reason Code	Explanation	See Message
D0xx0001	Empty string or parameter list not valid.	IHVO001I
D0xx0002	Command not valid. (See Note 1 on page 393.)	IHVO002I
D0xx0003	No operands or options specified with the command.	IHVO003I
D0xx0004	Parameter greater than 24 characters. (See Note 1 on page 393.)	IHVO004I
D0xx0005	Option not valid. (See Note 1 on page 393.)	IHVO005I
D0xx0006	Conflicting options, such as Force and Noforce, were specified in the same command. In the caller's response area, I/O operations returns each conflicting option in a 24-character field, left-justified and padded to the right with X'40's (blanks).	IHVO006I
D0xx0007	Too many parameters.	IHVO007I
D0xx0008	1st and 2nd operands both equal * (for example, Display Name * * or Block * *).	IHVO008I
D0xx0009	Port address is not valid. (See Note 1 on page 393.)	IHVO009I
D0xx000A	Operand is not valid. (See Note 1 on page 393.)	IHVO010I
D0xx000B	Required operand is missing.	IHVO011I
D0xx000C	A port name of * is not valid with the Write command.	IHVO012I
D0xx0010	Connectivity option used with a non-connectivity command. (See Note 1 on page 393.)	IHVO016I
D0xx0011	Combination not valid. Novary and Noforce cannot be specified in the same command.	IHVO017I
D0xx0012	Combination not valid. Backout, Force, and Vary specified with the same command.	IHVO018I
D0xx0013	Combination is not valid. 2nd operand of * or (*) cannot be specified with the Only option on the Allow command. (See Note 1 on page 393.)	IHVO019I
D0xx0014	Combination not valid. The Only or Also option can be specified with the Allow command only.	IHVO020I
D0xx0015	Combination not valid. Disable Nodisable option set is valid only with the Remove Switch command. (See Note 1 on page 393.)	IHVO021I
D0xx0016	Combination not valid. The options Nocheck and Novary cannot be specified with the same command.	IHVO022I

Return Codes, Status Codes, and Reason Codes

Reason Code	Explanation	See Message
D0xx0017	Combination not valid. Process Noprocess option set is only valid with the Sync Switch command. (See Note 1 on page 393.)	IHVO023I
D0xx0018	Combination not valid. Sync and Fail options are valid only with the Display Vary command. (See Note 1 on page 393.)	IHVO024I
D0xx0019	Combination not valid. The options Force, Novary, and Backout cannot be specified with the same command.	IHVO025I
D0xx0020	Combination not valid. Block or Connect can be specified on the Display Switch, Display Port, or Display Name commands only. (See Note 1 on page 393.)	IHVO032I
D0xx0021	I/O operations requested a message with too many variables.	IHVO033I
D0xx0030	Combination not valid. The On, Off, and Purge options can be specified with the Reset Host command only. (See Note 1 on page 393.)	IHVO048I
D0xx0040	The Datablock parameter of the Writeswch command is not a valid entry.	None
D0xx0045	On a call from a REXX EXEC, the length of the Writefile data block is not valid. The length must be 20480 bytes, which is 256 ports x 80 bytes (QSWT size) per port. In IHVRESP, I/O operations returns a 24-character field where the leftmost 8 decimal bytes contain the length specified in the input followed by 16 blanks (X'40').	None
D0xx0046	A switch number in a QSWT in the datablock on a Writefile command is different from the switch number in the first QSWT. All QSWTs in the datablock must contain the same switch device number. (See Note 2 on page 393.)	None
D0xx0047	The file name on a Writefile command is not valid. It must be one to eight characters in length and can only contain upper case A-Z, 0-9, underscore, or hyphen.	None
D0xx0048	The file descriptor on a Writefile command is not valid. It must be twenty-four characters in length and can contain any character starting with X'40' up to and including X'FE'.	None
D0xx0049	Sequence of ports in the datablock on a Writefile command is not valid. Ports in the QSWTs in the datablock must be in ascending order starting with X'00' up to and including X'FF'. (See Note 2 on page 393.)	None
D0xx004A	On a call from a REXX EXEC, the length of the Writefile file descriptor length is not valid. The length must be 24 bytes. In IHVRESP, I/O operations returns a 24-character field where the leftmost 8 decimal bytes contain the incorrect length specified in the input followed by 16 blanks (X'40').	None
D0xx0050	On a call from a REXX EXEC, the Writeport mask length is not valid. The length must equal 256. In IHVRESP, I/O operations returns a 24-character field where the leftmost 8 decimal bytes contain the incorrect length specified in the input followed by 16 blanks (X'40').	None
D0xx0055	Combination not valid. On a Writeport command with Connect or Disconnect as the 4th operand, the 5th operand must contain a port or port name.	None
D0xx0056	Combination not valid. On a Writeport command with a blank as the 4th operand, the 5th operand must also be blank.	None
D0xx0060	Combination not valid. Relations between like entities are not allowed on the Query Relation command except for hosts and switches. (See Note 1 on page 393.)	None
D0xx0070	Parameter type not valid. On a Query Entity command, the only valid parameter types are Value, Range, Array, and Table. On a Query Relation command and Query Interface Control Unit command, the only valid parameter types are Value, Range, and Array. (See Note 1 on page 393.)	None

Return Codes, Status Codes, and Reason Codes

Reason Code	Explanation	See Message
D0xx0071	The Scope operand is not valid. I/O operations accepts either no more operands or the following operands: Pathtest, Nopathtest, or Scope. (See Note 1 on page 393.)	None
D0xx0072	Query option not valid. Pathtest Nopathtest option set is not valid in this command. (See Note 1 on page 393.)	None
D0xx0073	An object (entity) array element is in error.	None
D0xx0074	The first value of the range parameter is missing.	None
D0xx0075	The range parameter is missing a hyphen.	None
D0xx0076	The first value of the range parameter is greater than the second value.	None
D0xx0077	The first value of the range parameter is not valid.	None
D0xx0078	The second value of the range parameter is not valid.	None
D0xx0079	The second value of the range parameter is missing.	None
D0xx0083	A Scope array element is in error.	None
D0xx0088	REXX data overflowed. The data could not be contained in the REXX variable because it exceeded the allowable maximum of 16 meg.	None
D0xx0090	The Array or Table parameter of this Query command is less than the minimum allowable size.	None
D0xx0091	Length of response area is not valid. This error can occur if an application called IHVAPI2 and one of the following conditions have been detected: <ul style="list-style-type: none"> • The application passed its own response area and specified a length that is less than zero. • The application passed a response area managed by I/O operations. I/O operations either found an invalid length in the prefix area of the response area, or the pointer to the start of the start of the prefix area is invalid. This may occur if the application has modified the prefix area unintentionally. 	None
D0xx00A0	Array or Table parameter of this Query command exceeds the maximum size allowed.	None
D0xx00B0	The number of elements found in an array, or the number of rows found in a table, is inconsistent with the actual size of the array or table.	None
D0xx00C0	Format identifier specified for a Query, Writeswch, or Writefile command is not valid. (See Note 2 on page 393.)	None
D0xx00D0	An error in a Query, Writeswch, or Writefile command is not valid. (See Note 2 on page 393.)	None
D0xx00FE	GETMAIN failed.	IHVO900I
D0xx00FF	Unrecoverable I/O operations error	IHVO255E
D0xx0101	Port address is not valid. In the caller's response area, I/O operations returns the 80-byte control block in error.	None
D0xx0102	Switch number not valid. In the caller's response area, I/O operations returns the 80-byte control block in error.	None
D0xx0103	Static port address is not valid. In the caller's response area, I/O operations returns the 80-byte control block in error.	None
D0xx0104	Logical name not valid. In the caller's response area, I/O operations returns the 80-byte control block in error.	None

Return Codes, Status Codes, and Reason Codes

Reason Code	Explanation	See Message
Notes: 1. In the caller's response area, I/O operations returns the parameter in error in a 38-character field, left-justified and padded to the right with blanks (X'40'). (Only if the parameter in error is a Writeport mask parameter, is the field 32 characters instead of 38.) 2. I/O operations returns a 24-character field in the caller's response area. Except for the leftmost bytes noted below, the remaining bytes are blanks (X'40'). <ul style="list-style-type: none"> • On a Query command, the leftmost byte of hex data represents the format identifier in error. • On a Writeswch or Writefile command by a REXX EXEC caller, the leftmost 8 bytes are in decimal. • On a Writeswch or Writefile command by a BAL caller, the leftmost 4 hexadecimal bytes represent the first format identifier in error. 		

Reason Codes Associated With RC=20

Report these reason codes with the associated return code of 20, related messages, and other data to your IBM Support Center.

Reason Code	Explanation	See Message
xxxx0002	Main I/O operations program has ended.	None
xxxx0004	I/O operations is not operational.	None
10xxnnnn 15xxnnnn 20xxnnnn 30xxnnnn 40xxnnnn 50xxnnnn 51xxnnnn 60xxnnnn 70xxnnnn 80xxnnnn 90xxnnnn 95xxnnnn A0xxnnnn B0xxnnnn C0xxnnnn	A system service or instruction required by I/O operations has failed.	IHVD005I ING001I
D0xx00AA	Input array data cannot be contained in the command control block. This block cannot exceed $(2^{32})-1$, or 4,294,967,295 bytes.	None
D50x000x	x=X'1'-X'F'.	
D0xxnnnn D5xxnnnn E0xxnnnn FFxxnnnn	A system service or instruction required by I/O operations has failed.	IHVD005I

Return Codes, Status Codes, and Reason Codes

Appendix B. Abend Codes and Reason Codes Issued By the Automation Manager

Communication DST and Automation Manager Abend Codes

This section describes the abend codes and reason codes that the automation manager may issue using the ABEND macro. If one of these abend codes occurs, contact your IBM Support Center.

With **ABEND CODE 3998** the following reason code can be issued:

X'01' The DST abends due to a severe MQSeries related problem during MQSeries outbound or MQSeries receive. The DST is intentionally terminated and will be restarted by SA z/OS due to message trapping of message DSI172I.

With **ABEND CODE 3999** the following reason codes can be issued:

- X'01'** PLX call stack storage cannot be obtained because no storage is available
- X'02'** PLX call stack storage is too small. PLX call stack storage has been obtained but a nested program does not get enough storage from the stack.
- X'0A'** List of ECB to be posted exceeds the limit (16). Module INGPXIMV detected an error while processing a time exit.
- X'0B'** List of ECB to be posted exceeds the limit (16). Module HSAAPIMV detected an error while processing a time exit.
- X'10'** The threshold for the loop detector has been exceeded.
- X'11'** There has been an attempt to add a duplicate list element.

Appendix C. Response Messages, Error Strings, Condition Codes

Response Messages (AOFA0000–AOFA0099)

Automation returns the following messages to indicate command-invocation, parameter-list, or parameter-resolution problems.

AOFA0000

Explanation: This response message is returned as an indicator of problems with command invocation, parameter lists, or parameter resolution. If the AOFA0000 response message is returned from the INGHWCMD command list, its data portion is an error string as listed below. If the AOFA0000 response message is returned from the INGHWCOM communication task command processor, its data portion contains a condition code from 001 through 033 (see “Hardware Communication Task Condition Codes "00B00xxx"” on page 434).

Error type

Error strings

Environment error

Required_System_Automation_Environment_is_not_complete
NetView_task_environment_must_be_CMD_LOW_for_opid()

Parameter error

Proc_or_Sys_name_and_HW_function_name_is_required
p_session_type_not_SYNC/ASYNC/blank
Timeout_range_tttt_already_defined.
Timeout_tt_out_of_range_1-59.
Timeout_specification_range_tt_is_not_valid.
Timeout_specification_tt_is_not_valid.
Timeout_specification_tttt_ends_invalid.
TRACE_option_must_be_ON_or_OFF
hwcmd_with_FORCE_operand_is_not_valid.
hwcmd_is_not_a_supported_HW_function.
p_must_be_a_decimal_integer_value_or_ALL
p_EXTERNAL_CPU_definition_error
CN_Activation_profile_name_not_alphanumeric
CN_Profile_name_is_a_positional_parm
p_does_not_support_target_wildcard.
p_Parm_is_in_wrong_position.
lparm_load_parm_length_must_be_8.
devnum_device_address_not_hexadecimal.
devnum_mandatory_load_address_invalid.
lval_Load_value_definition_error.
lval_Load_value_is_a_positional_parm.
pn_load_profile_name_not_alphanumeric.
pn_Load_profile_definition_error.
pn_Profile_name_is_a_positional_parm.
spc_P_and_LV_specs_are_mutually_exclusive.
name_invalid_chars_in_proc_or_sys_name.
name_name_longer_than_8_characters.
parm_Parm_is_unknown_or_in_wrong_position.
evt_is_an_invalid_event_type
evt_defined_more_than_once
ALL_must_be_the_1st_or_unique_event_parm
Event_specification_is_required
p_variable_name_APNAME_required
p_APNAME_data_len_must_be_1-16_chars
p_APNAME_data_must_not_contain_blanks
p_Applid_definition_error
p_ON/OFF_missing_error
p_invalid_image_cntl_varname

AOFA0001

p_YES/NO_missing_error
p_variable_is_READONLY
p_numeric_value_out_of_range
p_value_not_numeric
p_value_for_variable_missing
p_command_parm_not_supported
p_command_function_not_supported
p_profile_name_blank_or_gt_16char
p_act_profile_varname_missing
p_act_profile_vardata_def_error

Resolve error

sysname_for_CFs_LOAD/SYSRESET_are_not_supported.
sysname_type_specification_missing_or_invalid.
sysname_null_string_BCP_command_error
pname_has_invalid_CPC_address_format.
pname_has_no_IP_address_defined.
pname_has_no_processor_address_defined.
name_name_is_not_defined.
pname_name_not_valid_for_CPC_command.
pname_has_no_AUTHTKN_defined.
No_HWOPER_task_defined.

Check Task

hwtask_reached_QueueLimit qlim
hwtask_task_msqueue_data_is_invalid
hwtask_task_is_not_available
hwtask_task_module_INGHWCOM_not_running

Hardware Interface

BCP_internal_interface_is_disabled_or_not_active
BCP_internal_interface_status_cannot_be_determined

Authorization error

hwcmd_has_undefined_access_level
accllevel_to_resname_not_allowed_for_user
BadRC_during_access_chk_for_resname

Examples:

1. The INGHWCMD command failed returning an AOFA0000 error string:

```
AOFA0000 Resolve error:  
"MYSYS"_is_not_a_predefined_system_or_CF-name
```
2. The INGHWCMD command failed. Message AOFA0000 was returned by INGHWCOM command processor. The condition code 00B00003 indicates that an unknown communication interface name was passed from INGHWCMD to INGHWCOM.

```
AOFA0000 GETISTAT STATUS(REJECTED) CONDITION(00B00003) SENSE() CPCSNAME()  
TSTIME(020111073708)
```

AOFA0001

Explanation: This response message is returned from a request of the following hardware command functions: ACTIVATE, DEACTIVATE, LOAD, OOCOD, RESTART, SYSRESET, START, STOP, CBU, EXTERNAL, RESERVE

Examples:

1. The hardware function STOP was successfully performed for system KEY7:

```
AOFA0001 STOP KEY7 STATUS(SUCCESS)  
CPCSNAME(DEIBMD1.X7E1FA0A) TSTIME(020111135810)
```
2. The hardware function SYSRESET was rejected by INGHWCOM. The condition code 00B00056 indicates that system KEY6 is still operational and cannot be disrupted. In order to perform a disruptive hardware operation, the FORCE option must be specified:

```
AOFA0001 SYSRESET KEY6  
CONDITION(00B00056)SENSE()CPCSNAME(DEIBMD1.X7E1FA0A) TSTIME(020111142827)
```
3. The first command successfully sets a reserve for application ID IPSFO. The second command lists the reserve ID as stored in the Support Element. The third command fails because it tries to release the reserve with an invalid

reserve ID (that is, it is not used for the reserve). The fourth command is identical to a RESERVE LIST command. It shows a reserve ID that is identical to the CPCSNAME. This indicates that on the SE, EXCLUSIVE CONTROL has been set manually. In order to successfully release a reserve, the RESERVE OFF request must be made, using the same application name (reserve ID) as for the RESERVE ON request.

```
AOFA0001 RESERVE FREEWAY STATUS(SUCCESS)
          CPCSNAME(DEIBMIP1.IP3TVMA0)
          TSTIME(040828172045)

INGHWCMDFREEWAY RESERVE LIST

AOFA0001 RESERVE FREEWAY STATUS(SUCCESS)
          APPLNAME(0AUTOOPS.IPSFO)
          CPCSNAME(DEIBMIP1.IP3TVMA0) TSTIME(040828172315)

INGHWCMDFREEWAY RESERVE WHOKNOWS OFF

AOFA0001 RESERVE FREEWAY STATUS(FAILED) CONDITION(0B20000)
          SENSE(0838001B 0000 00000000)
          CPCSNAME(DEIBMIP1.IP3TVMA0) TSTIME(040828172445)

INGHWCMDFREEWAY RESERVE

AOFA0001 RESERVE FREEWAY STATUS(SUCCESS)
          APPLNAME(DEIBMIP1.IP3TVMA0)
          CPCSNAME(DEIBMIP1.IP3TVMA0) TSTIME(040828173117)
```

AOFA0002

Explanation: This message is the response to an INITCOM request. INITCOM establishes a session between INGHWCOM and the Processor Support Element of the addressed hardware.

Example: The session between the INGHWCOM and the processor Support Element of the CPC DEIBMD1.X7E1FA0A, configured with the hardware name "YORAMA," is established successfully:

```
AOFA0002 INITCOM YORAMA STATUS(SUCCESS)
          CPCSNAME(DEIBMD1.X7E1FA0A)TSTIME(020111143851)
```

| **Note:** For Processor Operations connections, INITCOM is executed as part of the host-based command ISQXIII
| (Initialize target connection).

AOFA0003

Explanation: This message is the response to an INITCOM request to an HMC. INITCOM establishes a session between INGHWCOM and the addressed CPC. In case the CPC is defined over an HMC, for each CPC managed by the HMC, one extra line is shown in the report.

Example: The session between INGHWCOM and the HMC where SERVER1 is a member, is established successfully. Implicitly, the sessions to the other CPCs of that HMC are also established:

```
AOFA0003 INITCOM SERVER1 STATUS(SUCCESS) TSTIME(030117084549)
AOFA0003 INITCOM CPCSNAME(DEIBMIPS.IP3T1000)
AOFA0003 INITCOM CPCSNAME(AUIBMQXP.QXPthes1)
AOFA0003 INITCOM CPCSNAME(AUIBMQXP.QXPthes9)
AOFA0003 INITCOM CPCSNAME(DEIBMD1.X7E1FA0A)
AOFA0003 INITCOM CPCSNAME(DEIBMD1.X7F1E30A)
AOFA0003 INITCOM CPCSNAME(DEIBMD1.X7F1F20A)
AOFA0003 INITCOM REPORT COMPLETE
```

| **Note:** For Processor Operations connections, INITCOM is executed as part of the host-based command ISQXIII
| (Initialize target connection).

AOFA0004

Explanation: This report is the response to a TERMCOM request. TERMCOM ends a session between INGHWCOM and the Processor Support Element of the addressed hardware.

Examples:

1. The session between the INGHWCOM and the Processor Support Element of the CPC USIBMSC.SCZP107 configured with the hardware name "P701" is terminated successfully:

```
AOFA0004 TERMCOM P701 STATUS(SUCCESS)
CPCSNAME(USIBMSC.SCZP701) TSTIME(020111090930)
```

2. The session termination between the INGHWCOM and the Processor Support Element of the CPC USIBMSC.SCZP701 configured with the hardware name "P701" was rejected. Condition code 00B00033 indicates that no session existed to terminate:

```
AOFA0004 TERMCOM P701 STATUS(REJECTED) CONDITION(00B00033) SENSE()
CPCSNAME(USIBMSC.SCZP701) TSTIME(020111091447)
```

| **Note:** For Processor Operations connections, TERMCOM is executed as part of host-based command ISQXCLS (Close
| target connection).

AOFA0005

Explanation: This report is the response to a TERMCOM request for a CPC that is defined over an HMC connection. In this case, TERMCOM terminates the session between INGHWCOM and the HMC.

Example: The session between INGHWCOM and the HMC where SERVER1 is defined, is terminated successfully. Implicitly, the sessions the other CPCs of that HMC are also terminated.

```
AOFA0005 TERMCOM SERVER1 STATUS(SUCCESS)
TSTIME(030117085107)
AOFA0005 TERMCOM CPCSNAME(DEIBMIPS.IP3T1000)
AOFA0005 TERMCOM CPCSNAME(AUIBMQXP.QXPTHES1)
AOFA0005 TERMCOM CPCSNAME(AUIBMQXP.QXPTHES9)
AOFA0005 TERMCOM CPCSNAME(DEIBMD1.X7E1FA0A)
AOFA0005 TERMCOM CPCSNAME(DEIBMD1.X7F1E30A)
AOFA0005 TERMCOM CPCSNAME(DEIBMD1.X7F1F20A)
AOFA0005 TERMCOM REPORT COMPLETE
```

| **Note:** For Processor Operations connections, TERMCOM is executed as part of host-based command ISQXCLS (Close
| target connection).

AOFA0006

Explanation: This report is the response to a CCNTL APNAME request for a CPC that displays or sets the name of an activation profile (reset or load) for the next CPC activation. APNAME is the only variable for a CPC that can be set with INGHWCMD.

Examples:

1. The current setting of the CPC activation profile is displayed. Profile names can be up to 16 characters in length.

```
AOFA0006 CCNTL FREEWAY STATUS(SUCCESS) APNAME(DEFAULT)
CPCSNAME(DEIBMIP1.IP3TVMA0) TSTIME(040825101045)
```

2. The profile name is changed to DEFAULTLOAD. Profile names can be up to 16 characters in length.

```
AOFA0006 CCNTL FREEWAY STATUS(SUCCESS) APNAME(DEFAULTLOAD)
CPCSNAME(DEIBMIP1.IP3TVMA0) TSTIME(040825101104)
```

AOFA0007

Explanation: This multiline report is the response to a ICNTL request. It returns either a list of all available image variables with their current settings, or the response to a query or set request for a single image variable. Note that the set of image variables available may be different, depending on the processor type, the image is running on.

Examples:

1. An ICNTL LIST request returns a report with the available image variables and their current settings for image KEY4 of CPC FREEWAY. Because the ICNTL command is entered without a specific processor type, the default is set to GPP (General Purpose Processor).

```
AOFA0007 ICNTL FREEWAY.KEY4 STATUS(ACCEPTED)
          CPCSNAME(DEIBMIP1.IP3TVMA0) PT(GPP) TSTIME(040911090931)
AOFA0007 DEFCAP(0)
AOFA0007 PWI(30)
AOFA0007 PWIC(NO)
AOFA0007 PWMN(10)
AOFA0007 PWMX(100)
AOFA0007 PWC(30)
AOFA0007 PWCC(NO)
AOFA0007 WLME(NO)
AOFA0007 ICNTL REPORT COMPLETE
```

2. Another LIST request output is shown. Note that LIST is the default request if no other ICNTL parameter is specified. The report shows the available image variables for image KEY1 of CPC SAFOS. This processor is of an older type than CPC FREEWAY from example 1. Only the initial processor weight (PWI) and the initial processor weight capping (PWIC) can be manipulated on this hardware type.

```
AOFA0007 ICNTL SAFOS.KEY1 STATUS(ACCEPTED)
          CPCSNAME(DEIBMIP1.IP3TKY10) PT(GPP) TSTIME(040911091922)
AOFA0007 PWI(30)
AOFA0007 PWIC(NO)
AOFA0007 ICNTL REPORT COMPLETE
```

3. In this example, z/OS Workload Manager (WLM) Control is enabled for the CP resources assigned to image KEY4 of CPC FREEWAY.

```
AOFA0007 ICNTL FREEWAY.KEY4 STATUS(ACCEPTED)
          CPCSNAME(DEIBMIP1.IP3TVMA0) PT(GPP) TSTIME(040911090931)
AOFA0007 WLME(YES)
AOFA0007 ICNTL REPORT COMPLETE
```

AOFA0016

Explanation: This report is the response to a CPCDATA request. It returns a report consisting of multiple AOFA0016 messages. The CPCDATA request combines the GETSINFO request for a CPC with the list of GETIINFO request, one for each image of the CPC.

Examples:

1. The AOFA0016 report message consists of three record line types. The first type is the CPC report starting with CPCDATA followed by the CPC's name. Depending on the amount of CPC information, multiple lines of this type are provided in the report. The last line completes the report. In between, 1-n IMAGE report lines may be displayed, depending on the number of images that are defined for the CPC.

CPC Report:

```
AOFA0016 CPCDATA R35 STATUS(OPERATING)
          PDATA(TYPE(2817),MODEL(M49),S/N(000020089F25))
          MODEL(LPAR) APROF(DEFAULT) CPCSNAME(IBM390PS.R35)
          NAME(R35)
          CPS(GPP(22),SAP(9),AAP(2),IFL(22),ICF(0),ZIIP(1))
AOFA0016 CPCDATA R35 TCAP(YES) TCAP_RECS(2)
          MSU(PERM(1084),PERM+BILL(2224) PERM+ALL_TEMP(2224))
          SW_MDL(PERM(615),PERM+BILL(722),PERM+ALL(722))
          PROCSTAT(DEFECTIVE(0),SPARE(2) PENDING(0))
AOFA0016 CPCDATA R35 ZBX_CHASSIS(2) TSTIME(110316135817)
```

CPC image (LPAR) reports:

```
AOFA0016 CPCDATA CPCINAME(LP01) STATUS(OPERATING) INUMBER(01)
          IDATA(OSTYPE(LINUX),OSLEVEL(Linux_2.6.32))
          MODE(LINUXONLY)
AOFA0016 CPCDATA CPCINAME(LP11) STATUS(EXCEPTIONS) INUMBER(0B)
          IDATA(OSNAME(z/VSE),OSTYPE(VSE),OSLEVEL(z/VSE_V9R1))
          MODE(ESA390)
AOFA0016 CPCDATA CPCINAME(LP12) STATUS(OPERATING) INUMBER(0C)
          IDATA(OSNAME(IRD7),OSTYPE(MVS),OSLEVEL(z/OS_V1R13),
```

AOFA0016

```
|      SYSPLEX(IRD7PLEX)) MODE(ESA390)
| AOFA0016 CPCDATA CPCINAME(LP57) STATUS(OPERATING) INUMBER(3C)
|      IDATA(OSNAME(TRX2),OSTYPE(MVS),OSLEVEL(z/OS_V1R13)
|      MODE(ESA390)
| AOFA0016 CPCDATA CPCINAME(LP59) STATUS(OPERATING) INUMBER(3E)
|      IDATA(OSNAME(BOEVM59),OSTYPE(VM),
|      OSLEVEL(z/VM_V5R4M0_SL1003)) MODE(ZVM)
| AOFA0016 CPCDATA CPCINAME(LP60) STATUS(NOT_ACTIVATED)
|      INUMBER() IDATA() MODE()
```

```
| Report completion:
| AOFA0016 CPCDATA REPORT COMPLETE
```

The STATUS field of line one, the CPC status, can have the following values:

```
OPERATING
NOT_OPERATING
NO_POWER
STATUS_CHECK
EXCEPTIONS
POWERSAVE
SERVICE
LINKNOTACTIVE
SERVICE_REQ
UNKNOWN
```

The PDATA field of line one contains the type, model, and serial number of the CPC.

The APROF field of line one contains the last activation profile name used to activate the CPC.

The MODE field of line one, the CPC mode, can have the following values:

```
ESA390
S370
FM
FMAE
HM
HMEA
LPAR
ESA390TPF
CF
FMEX
HMAS
LINUXONLY
```

For each identified CPC image, an AOFA0016 report line is generated.

The CPCINAME field of an image report line contains the image name of an identified image.

The INUMBER field contains the two hex digit partition number. For processor hardware supporting a single channel subsystem, the first digit is always zero. The second digit contains a partition number from 1-F. For processor hardware supporting multiple channel subsystems, the first hex digit contains the channel subsystem number, starting with zero and the second digit contains the partition number 1-F.

The IDATA field contains a collection of the available information supplied by the image BCP. This information can be: OSNAME, OSTYPE, or OSLEVEL; for BCPs of type MVS it can be SYSPLEX. Note that one or more IDATA fields may not be available in the AOFA0016, AOFA0017 response reports. This is because not all BCPs may supply the complete field set. If the OSLEVEL field is not shown in the response report, the BCP did not provide this information to the hardware.

The STATUS field of an image report line contains the same status values as supplied with the GETISTAT report message AOFA0017.

2. On processor H05 (which is System z10 hardware or later), more AOFA0016 reports are provided because more CPC information is available on this hardware. The CPC report is split in two.

```
AOFA0016 CPCDATA H05 STATUS(OPERATING)PDATA(TYPE(2097),MODEL(E26),
S/N(00002001EBAE)) MODE(LPAR) APROF(DEFAULT) CPCNAME(IBM390PS.H05) NAME(H05)
CPS(GPP(3),SAP(6),AAP(1),IFL(18),ICF(1),ZIIP(1))
```

```
AOFA0016 CPCDATA H05 TCAP(YES) TCAP_RECS(2) MSU(PERM(312),PERM+BILL(312),PERM+ALL_TEMP(312))
SW_MDL(PERM(703),PERM+BILL(703),PERM+ALL(703))
PROCSTAT((DEFECTIVE(0),SPARE(2),PENDING(0))
TSTIME(080619175410)
```

```
AOFA0016 CPCDATA CPCINAME(...)
```

```
..
```

```
AOFA0016 CPCDATA REPORT COMPLETE
```

The additional information in detail is:

CPS Shows the different types and numbers of processors:

GPP

The number of General Purpose Processors.

SAP

The number of Service Assist Processors.

AAP

The number of Application Assist processors.

IFL

The number of Integrated Facility for Linux processors.

ICF

The number of Internal Coupling Facility processors.

ZIIP

The number of Integrated Information Processors.

TCAP Indicates whether a temporary capacity change is allowed (YES) or not (NO).

TCAP_RECS

The number of temporary capacity records found for that CPC.

MSU Shows the various temporary capacity MSU values that are associated with software models based on:

PERM Permanent processors.

PERM+BILL

Permanent plus billable processors.

PERM+ALL_TEMP

Permanent plus all temporary processors.

SW_MDL

Shows the various Software Models related to temporary capacity, based on:

PERM Permanent processors.

PERM+BILL

Permanent plus billable processors

PERM+ALL_TEMP

Permanent plus all temporary processors.

PROCSTAT

Shows processor status information:

DEFECTIVEPROC

The number of defective processors.

SPAREPROC

The number of spare processors.

PENDINGPROC

The number of pending processors.

AOFA0017

Explanation: This report is the response to the following requests: GETSSTAT, GETSDGR, GETSINF, GETISTAT,GETIINFO, OOCOD STATUS, and CBU STATUS request.

GETISTAT Queries the status of an image object.

AOFA0017

GETIINFO	Queries the status of an image object and lists the available image information (OSname, OStype, OSlevel, SysplexName).
GETSSTAT	Queries the status of a CPC object.
GETSDGR	Queries the degraded reason indicator of a CPC supporting the DEGRADED status.
GETSINFO	Queries the status of a CPC object and lists the available CPC information (machine type and model, CPC serial number, last used activation profile name).
CBU STATUS	The status function of the CBU command returns the determined status of the optional capacity backup processor HW feature.
OOCOD STATUS	The status function of the OOCOD command returns the determined status of the On/Off Capacity on Demand HW feature.

On successful completion, the status field of message AOFA0017 may have one of the following values:

GETIINFO	CBU STATUS	OOCOD STATUS	GETSDGR
-----	-----	-----	-----
OPERATING	NOT_INSTALLED	ENABLED	NOT_DEGRADED
NOT_OPERATING	NOT_ACTIVATED	NOT_ENABLED	MEM_REDUCED
NOT_ACTIVATED	NOT_ENABLED	INSTALLED	MEM_BUS_FAILURE
STATUS_CHECK	UNAVAILABLE	NOT_INSTALLED	NODE_NOT_RUNNING
EXCEPTIONS	AVAILABLE	ACTIVATED	RING_OPEN
POWERSAVE	ACTIVATED	NOT_ACTIVATED	CBU_EXPIRATION
			MRU_FAILURE
			TEMPERATURE_PROBLEM
			IML_WAS_IN_DEGRADED_MODE

Examples:

1. The system (*), which runs on LPAR A3 of CPC USIBMSC.SCZP801, has a status of OPERATING and its system name is SC50.

```
AOFA0017 GETISTAT SC50 STATUS(OPERATING) CPCINAME(A3) CPCSNAME(USIBMSC.SCZP801)
        BUSY(NO) TSTIME(020111095940)
```

2. On processor YORAMA, the logical partition KEY7 has a status of OPERATING:

```
AOFA0017 GETISTAT YORAMA.KEY7 STATUS(OPERATING) CPCINAME(KEY7)
        CPCSNAME(DEIBMD1.X7E1FA0A) BUSY(NO) TSTIME(020204130403)
```

3. On processor FREEWAY (which is a System z9 hardware or older), Capacity Backup Upgrade is installed and enabled (AVAILABLE). It was ACTIVATED on January 18th 2008 and the activation period expires on January 28th 2008. There are four test activations left for processing.

```
AOFA0017 CBU FREEWAY STATUS(AVAILABLE,ACTIVATED)
        ACTIVATION(18/01/08)
        A-EXPIRATION(28/01/08)
        TESTSLEFT(4)
        CPCSNAME(DEIBMD1.X7E1FA0A)
        TSTIME(080204130403)
```

On processor H05 (which is a System z10 hardware or later), Capacity Backup Upgrade is installed and enabled (AVAILABLE). It was ACTIVATED on March 20th 2008 and the CBU record expires on December 31st 2009. There are three test activations left for processing.

```
AOFA0017 CBU H05 STATUS(AVAILABLE,ACTIVATED)
        ACTIVATION(20/03/08)
        R-EXPIRATION(31/12/09) TESTSLEFT(3)
        CPCSNAME(DEIBMD1.H05)
        TSTIME (080320110503)
```

4. The system defined as KEY6 to SA z/OS, which runs on logical partition KEY6 (cpciname), whose partition number cannot be determined (inumber field is empty), is an MVS OS type with an OS defined name KEY6. It runs as a member of the sysplex KEY6PLEX. The LPAR runs in ESA mode and is running on CPC DEIBMD1.X7E1FA0A.


```
AOFA0017 GETIINFO KEY6 STATUS(OPERATING) CPCINAME(KEY6)
INUMBER(0A) IDATA(OSNAME(KEY6),OSTYPE(MVS),SYSPLEX(KEY6PLEX))
MODE(ESA) CPCSNAME(DEIBMD1.X7E1FA0A) BUSY(NO)
TSTIME(020923110403)
```

5. Processor T60 is enabled for On/Off Capacity on Demand and the feature is installed and active. It was ACTIVATED on April 3rd 2007.

```
AOFA0017 OOCOD T60 STATUS(ENABLED,INSTALLED,ACTIVATED)
ACTIVATION(20070403090556) CPCSNAME(IBM390PS.T60)
TSTIME(070410152248)
```

6. CPC T29 is a System z9, machine type 2094, model S18. It is running in LPAR mode and has activation profile DEFAULT set to be used for CPC activation. The CPC's Support Element SNA style address is ZZZZZZZZZ.T29. The configured CPC name is T29. The CPC object T29 is in a BUSY state. This means that a task is currently active that may prevent other CPC tasks from being accepted for execution.

Applications should repeatedly check the BUSY state using GETSSTAT or GETSINFO until a BUSY(NO) is returned in the AOFA0017 report, before issuing an operations management command against the object.

```
AOFA0017 GETSINFO T29 STATUS(OPERATING)
PDATA(TYPE(2094),MODEL(S18),S/N(0000200D6AAD))
MODE(LPAR)
APROF(DEFAULT)
CPCSNAME(ZZZZZZZ.T29)
NAME(T29)
BUSY(YES)
TSTIME(060908171141)
```

7. CPC R35 is a System z196, machine type 2817, model M49. It is running in LPAR mode and has set activation profile DEFAULT to be used for CPC activation. The SE SNA style address for the CPC is IBM390PS.R35. The configured CPC name is R35. The CPC is not in BUSY state. Installed processors:

- General Purpose Proc (GP) - 15
- Service Assist Proc (SAP) - 9
- Application Assist Proc (AAP) - 1
- Integrated Facility for Linux Proc (IFL) - 22
- Integrated Coupling Facility Proc (ICF) - 0
- Integrated Information Proc (IIP) - 1

On R35, temporary capacity changes are allowed - TCAP(YES). There is one temporary capacity record installed for full or partial activation. With this record you can temporarily upgrade R35 from a model 615 to a model machine increasing the available MSUs from 1084 to 1648.

Currently 10 of the 15 GP processors are not assigned to any work. R35 has a Blade Center Rack with two chassis ZBX_CHASSIS attached.

```
AOFA0017 GETSINFO R35 STATUS(OPERATING) PDATA(TYPE(2817),
MODEL(M49),S/N(000020089F25)) MODE(LPAR) BUSY(NO)
APROF(DEFAULT) CPCSNAME(IBM390PS.R35) NAME(R35)
CPS(GP(15),SAP(9),AAP(1),IFL(22),ICF(0),IIP(1))
AOFA0017 GETSINFO R35 TCAP(YES) TCAP RECS(1)
MSU(PERM(1084),PERM+BILL(1648),PERM+ALL_TEMP(1648))
SW_MDL(PERM(615),PERM+BILL(715),PERM+ALL(715))
PROCSTAT(DEFECTIVE(0),SPARE(10),PENDING(0))
AOFA0017 GETSINFO R35 ZBX_CHASSIS(2) TSTIME(101019100945)
```

AOFA0018

Explanation: This report is returned in response to a GETCLUSTER command.

Examples:

- From the system that you are running on (*), the CPC addresses list in PDATA are in your scope of control. With a BCP Internal Interface connections, this list is determined internally from the local SE by contacting the HMC in your processor LAN that has the "Change Management" function enabled. The content of the Defined CPCs group of this HMC represents the CPCs that you can contact through this BCP Internal Interface session. Each scope list is terminated with a "report complete" message. The PDATA field of the AOFA0018 message contains

AOFA0019 • AOFA0020

CPC-related information. The first PDATA entry is always the fully qualified address of the CPC (cpcsname). Other PDATA information may be added in the future, separated by a comma.

```
AOFA0018 GETCLUSTER SC50 STATUS(SUCCESS)
CPCSNAME(USIBMSC.SCZP801)TSTIME(020112054842)
AOFA0018 GETCLUSTER PDATA(USIBMSC.SCZP801)
AOFA0018 GETCLUSTER PDATA(USIBMSC.SCZP701)
AOFA0018 GETCLUSTER PDATA(USIBMSC.SCZP702)
AOFA0018 GETCLUSTER PDATA(USIBMSC.SCZP601)
AOFA0018 GETCLUSTER REPORT COMPLETE
```

2. This GETCLUSTER request failed with a condition code of 0B100224 representing a BCP Internal Interface transport timeout condition:

```
AOFA0018 GETCLUSTER SC50 STATUS(FAILED) CONDITION(0B100224)
SENSE(00000000 0000 00000000)
CPCSNAME(USIBMSC.SCZP801)TSTIME(020111085916)
```

AOFA0019

Explanation: This report is returned in response to a FILTER LIST command.

Examples:

1. The filter list report shows the filters that are in place for SERVER1 CPC. The defined events in the list (EN) are only forwarded to the specified NetView operators or operator group, if at least one filter is set. If a prefix field PFX was specified with a filter SET command, its text is placed in front of every event message (ISQ900I).

```
AOFA0019 FILTER SERVER1 STATUS(SUCCESS) CPCSNAME(DEIBMD1.X7F1F20A) TSTIME(030128090248)
AOFA0019 FILTER CPC PDATA(EN(ST,HW) OP(TIL) PFX(ISQ900I))
AOFA0019 FILTER CPC PDATA(EN(CC,ST,HW,ALRT) OP(+A00001S) PFX(ISQ900I))
AOFA0019 FILTER CPC PDATA(EN(CC,ST,HW) OP(+000001S) PFX(ISQ900I))
AOFA0019 FILTER REPORT COMPLETE
```

If a prefix field PFX was specified with a filter SET command, its text is placed in front of every event message. If no prefix is given with a FILTER SET command, the default prefix AOFA0900 is used.

2. The filter list report shows the filters that are in place for image KEY2 on CPC SERVER1.

The first filter set sends all event reports to group +GEOOPER in a event message prefixed with GEO001I. The second filter sends the event messages with the default prefix AOFA0900, because in the preceding filter set command PFX was not specified.

```
AOFA0019 FILTER SERVER1.KEY2 STATUS(SUCCESS) CPCSNAME(DEIBMD1.X7F1F20A) TSTIME(030119070203)
AOFA0019 FILTER INAME(KEY2) PDATA(EN(CC,ST,HW,BCP,ALRT) OP(+GEOOPER) PFX(GEO001I))
AOFA0019 FILTER INAME(KEY2) PDATA(EN(CC,ST,HW,BCP) OP(TIL) PFX(AOFA0900))
AOFA0019 FILTER REPORT COMPLETE
```

AOFA0020

Examples:

1. In this example the APROF OPEN request returns report AOFA0020 with the number of profiles stored for each activation profile type, in the second line. If no profile is stored for a activation profile type, the value 0 is returned for that type.

```
AOFA0020 APROF FREEWAY STATUS(ACCEPTED)
CPCSNAME(DEIBMIP1.IP3TVMA0) TSTIME(040920100843)
AOFA0020 PROFILES(RESET(1),IMAGE(16),LOAD(1))
AOFA0020 APROF REPORT COMPLETE
```

2. In this example the APROF CLOSE request returned a successful CLOSE of the activation profiles. Note that activation profiles must have been opened using the APROF OPEN request, prior issuing other profile requests.

```
AOFA0020 APROF FREEWAY STATUS(ACCEPTED)
CPCSNAME(DEIBMIP1.IP3TVMA0) TSTIME(040920101023)
AOFA0020 APROF REPORT COMPLETE
```

3. This example shows the response to an Activation Profile List command for image profiles of CPC FR07. For each profile name listed, a new AOFA0020 report line is generated.

```

AOFA0020 APROF FR07 STATUS(ACCEPTED)
          CPCSNAME(DEIBMIP1.IP3TVMA0) TSTIME(040920101055)
AOFA0020 NAME(DEFAULT)
AOFA0020 IMAGE(VMA)
AOFA0020 IMAGE(CF1)
AOFA0020 IMAGE(SYSA)
AOFA0020 IMAGE(SYSB)
AOFA0020 IMAGE(SYSC)
AOFA0020 IMAGE(SYSD)
AOFA0020 IMAGE(SYSE)
AOFA0020 IMAGE(SYSF)
AOFA0020 IMAGE(KEY3)
AOFA0020 IMAGE(KEY4)
AOFA0020 IMAGE(CIM7)
AOFA0020 IMAGE(CIM8)
AOFA0020 IMAGE(CFD)
AOFA0020 IMAGE(CFE)
AOFA0020 IMAGE(CFF)
AOFA0020 APROF REPORT COMPLETE

```

4. In this example the contents of image profile KEY3 is read. For each profile keyword variable one report line is generated. Keyword variable OSSLP (operating system specific load parameter) is a special case. The data length of this variable, together with the other associated report data, exceeds the maximum length supported for this interface. For this reason, the output for OSSLP is split into 8 segments. The segment number is attached to the keyword, separated by a period from the keyword name. Because the APROF command is entered without a specific processor type, the default is set to GPP (General Purpose Processor).

```

AOFA0020 APROF FR07 STATUS(ACCEPTED) CPCSNAME(DEIBMIP1.IP3TVMA0) PT(GPP) TSTIME(041002111100)
AOFA0020 IMAGE(KEY3) BPS( )
AOFA0020 IMAGE(KEY3) BRLBA(0000000000000000)
AOFA0020 IMAGE(KEY3) DEFCAP(0)
AOFA0020 IMAGE(KEY3) IPLADR(NEXT)
AOFA0020 IMAGE(KEY3) IPLPRM(NEXT)
AOFA0020 IMAGE(KEY3) IPLTYP(NORMAL)
AOFA0020 IMAGE(KEY3) LUN(0000000000000000)
AOFA0020 IMAGE(KEY3) OSSLP.1( )
AOFA0020 IMAGE(KEY3) OSSLP.2( )
AOFA0020 IMAGE(KEY3) OSSLP.3( )
AOFA0020 IMAGE(KEY3) OSSLP.4( )
AOFA0020 IMAGE(KEY3) OSSLP.5( )
AOFA0020 IMAGE(KEY3) OSSLP.6( )
AOFA0020 IMAGE(KEY3) OSSLP.7( )
AOFA0020 IMAGE(KEY3) OSSLP.8( )
AOFA0020 IMAGE(KEY3) PWI(30)
AOFA0020 IMAGE(KEY3) PWIC(NO)
AOFA0020 IMAGE(KEY3) PWMN(10)
AOFA0020 IMAGE(KEY3) PWMX(100)
AOFA0020 IMAGE(KEY3) WLME(NO)
AOFA0020 IMAGE(KEY3) WWPN(0000000000000000)
AOFA0020 APROF REPORT COMPLETE

```

5. In this example a single variable, the IPLTYP of load profile DEFAULTLOAD is retrieved. In the response report, the variable name, with its current value in parenthesis is returned.

```

AOFA0020 APROF FR07 STATUS(ACCEPTED) CPCSNAME(DEIBMIP1.IP3TVMA0) PT(GPP) TSTIME(041002141030)
AOFA0020 LOAD(DEFAULTLOAD) IPLTYP(NORMAL)
AOFA0020 APROF REPORT COMPLETE

```

AOFA0021

Explanation: This report is returned in response to a CTRLCONS request. CTRLCONS is used to shut down or restart a hardware console (either HMC or SE).

Examples:

1. This example shows the successful execution of a Restart for the hardware console that the target system is connected to.

AOFA0022 • AOFA0023

```
AOFA0021 CTRLCONS(KEYHMC2) SUBMITTED TSTIME(070321171541)
AOFA0021 CTRLCONS T60 STATUS(SUCCESS) CPCSNAM(IBM390PS.T60) TSTIME(070321171541)
AOFA0021 CTRLCONS REPORT COMPLETE
```

2. This example shows that the CTRLCONS command rejected because the session mode for the target hardware is ASYNY, but must be SYNC (use the STATCOM command to verify).

```
AOFA0021 CTRLCONS(KEYHMC2) SUBMITTED TSTIME(070321171541)
AOFA0021 CTRLCONS T60 STATUS(FAILED) CONDITION(00B00082) SENSE() CPCSNAM(IBM390PS.T60) TSTIME(070321154745)
AOFA0021 CTRLCONS REPORT COMPLETE
```

AOFA0022

Explanation: This report is returned in response to a CONDATA query request.

Example: Query the Support Element console of a System z10:

```
AOFA0022 CONDATA H05 TSTIME(080708103727)
AOFA0022 CONDATA NAME(H05) VER(2.10.0) TYPE(SE)
AOFA0022 CONDATA IPADDR(9.152.90.35
                fe80:0:0:0:21a:6bff:fe38:738 0.0.0.0
                fe80:0:0:0:21a:6bff:fe38:738)
AOFA0022 CONDATA ECLEVEL(F85906.108)
AOFA0022 CONDATA AUTOSWITCH(YES)
AOFA0022 CONDATA REPORT COMPLETE
```

The console application version detected is 2.10.0 (2097 GA1) and it runs on a Support Element. The configured set of IPv4 and IPv6 addresses is also returned in the console data query. In addition the Engineering change ID (F85906) of the console application and its active Micro Code Level (MCL) 108 is shown. Finally, this Support Element is configured to switch automatically to its alternate, in case of failure.

```
| AOFA0022 CONDATA RAP2 TSTIME(101019095046)
| AOFA0022 CONDATA NAME(KEYHMC2) VER(2.11.0) TYPE(HMC)
| AOFA0022 CONDATA IPADDR(9.152.93.21 fe80:0:0:0:.....)
| AOFA0022 CONDATA ECLEVEL(N29809.72)
| AOFA0022 CONDATA VMS(8)
| AOFA0022 CONDATA REPORT COMPLETE
```

| The console application version detected is 2.11.0 (2817 GA1) and it runs on a HMC. The configured set of IPv4 and IPv6 addresses is also returned with this query. The Engineering Change ID (N29809) of the console application and its active Micro Code Level (MCL) 72 is shown. Finally, this HMC has 8 virtual machine servers (VMS) defined that can be managed from this console. Note that CONDATA responses of a TYPE(SE) connection cannot show virtual machine servers. VMS endpoint is always a HMC.

AOFA0023

Explanation: These reports are returned in response to a TCDATA query request.

Examples:

1. Query the list of the installed capacity records:

```
AOFA0023 TCDATA ECL2 TSTIME(090610101910)
AOFA0023 TCDATA RECORDID(M0002460)
AOFA0023 TCDATA RECORDID(CR7LYKLY)
AOFA0023 TCDATA REPORT COMPLETE
```

2. Query the details of OOCOD record with ID 05961411:

```
AOFA0023 TCDATA OBJECT(TEMPORARY CAPACITY RECORD) OBJECT
                ID(05961411) TSTIME(081015162834)
AOFA0023 TCDATA RECORDTYPE(OOCOD)
AOFA0023 TCDATA STATUS(Rea1)
AOFA0023 TCDATA PROCESSORINFO(1) TYPE(CP)
AOFA0023 TCDATA PROCESSORINFO(1) PROCSTEP(2)
AOFA0023 TCDATA PROCESSORINFO(1) MAX(-1)
AOFA0023 TCDATA PROCESSORINFO(1) REMAININGPROCDAYS(-1)
AOFA0023 TCDATA PROCESSORINFO(1) REMAININGMSUDAYS(-1)
AOFA0023 TCDATA PROCESSORINFO(2) TYPE(IFL)
AOFA0023 TCDATA PROCESSORINFO(2) PROCSTEP(0)
```

```

AOFA0023 TCDATA PROCESSORINFO(2) MAX(2)
AOFA0023 TCDATA PROCESSORINFO(2) REMAININGPROCDAYS(-1)
AOFA0023 TCDATA PROCESSORINFO(3) TYPE(AAP)
AOFA0023 TCDATA PROCESSORINFO(3) PROCSTEP(0)
AOFA0023 TCDATA PROCESSORINFO(3) MAX(2)
AOFA0023 TCDATA PROCESSORINFO(3) REMAININGPROCDAYS(-1)
AOFA0023 TCDATA PROCESSORINFO(4) TYPE(IIP)
AOFA0023 TCDATA PROCESSORINFO(4) PROCSTEP(0)
AOFA0023 TCDATA PROCESSORINFO(4) MAX(2)
AOFA0023 TCDATA PROCESSORINFO(4) REMAININGPROCDAYS(-1)
AOFA0023 TCDATA MAXREALDAYS(0)
AOFA0023 TCDATA MAXTESTDAYS(0)
AOFA0023 TCDATA TARGET(1) PROCSTEP(0)
AOFA0023 TCDATA TARGET(1) SPEEDSTEP(0)
AOFA0023 TCDATA TARGET(1) SOFTWAREMODEL(705)
AOFA0023 TCDATA TARGET(1) BILLABLEMSUCOST(176)
AOFA0023 TCDATA TARGET(1) BILLABLEMSUDELTA(0)
AOFA0023 TCDATA TARGET(2) PROCSTEP(-1)
AOFA0023 TCDATA TARGET(2) SPEEDSTEP(0)
AOFA0023 TCDATA TARGET(2) SOFTWAREMODEL(704)
AOFA0023 TCDATA TARGET(2) BILLABLEMSUCOST(89)
AOFA0023 TCDATA TARGET(2) BILLABLEMSUDELTA(-87)
AOFA0023 TCDATA TARGET(3) PROCSTEP(-2)
AOFA0023 TCDATA TARGET(3) SPEEDSTEP(0)
AOFA0023 TCDATA TARGET(3) SOFTWAREMODEL(703)
AOFA0023 TCDATA TARGET(3) BILLABLEMSUCOST(0)
AOFA0023 TCDATA TARGET(3) BILLABLEMSUDELTA(-176)
AOFA0023 TCDATA REPORT COMPLETE

```

To learn more about the TCDATA records shown, see *System z Application Programming Interface*, SB10-7030, Appendix F and *IBM System z10 Capacity on Demand*.

AOFA0024

Explanation: This report is returned in response to a STPDATA query request.

Example:

```

AOFA0024 STPDATA CTNID STPID(1nxhmc2)
AOFA0024 STPDATA PREFERRED NODENAME(T63)
AOFA0024 STPDATA PREFERRED NODEID TYPE(002094)
AOFA0024 STPDATA PREFERRED NODEID MODEL(S38)
AOFA0024 STPDATA PREFERRED NODEID MANUF(IBM)
AOFA0024 STPDATA PREFERRED NODEID POMANUF(02)
AOFA0024 STPDATA PREFERRED NODEID SEQNUM(00000002D09E)
AOFA0024 STPDATA BACKUP NODENAME(H05)
AOFA0024 STPDATA BACKUP NODEID TYPE(002097)
AOFA0024 STPDATA BACKUP NODEID MODEL(E26)
AOFA0024 STPDATA BACKUP NODEID MANUF(IBM)
AOFA0024 STPDATA BACKUP NODEID POMANUF(02)
AOFA0024 STPDATA BACKUP NODEID SEQNUM(00000001EBAE)
AOFA0024 STPDATA CURRENTTIMESERVER(Preferred)
AOFA0024 STPDATA REPORT COMPLETE

```

This report shows information about a coordinated timer network with the name 1nxhmc2. It has configured a preferred (T63) and a backup (H05) node. Both nodes are System z mainframes. For each of them the machine type and model information, manufacturing information, and the CPC serial numbers are provided. Finally it is indicated that the current time server role is assigned to the preferred time server, T63.

For more information about STP see the IBM Redbooks®, *Server Time Protocol Planning Guide* (SG24-7280) and *Server Time Protocol Implementation Guide* (SG24-7281).

AOFA0099

Explanation: This report is returned as the response to a connection status request command.

Examples:

1. The CPC SERVER1 is connected to this NetView using task AUTHW007. The connection is made either through INTERNAL or SNMP transport. The BCP Internal Interface uses the internal transport and always has the Support Element (SE) as target. For the SNMP transport, either SE or HMC can be the target. A TRANSPORT value of A-INTERNAL indicates an asynchronous session, whereas S-INTERNAL indicates a synchronous session. A value of SNMP indicates that a standard SNMP session is established, which is always asynchronous.

```
AOFA0099 STATCOM SERVER1 STATUS(CONNECTED)
      TASK(AUTHW007)
      TRANSPORT(A-INTERNAL)
      TARGET(SE)
      CPCSNAME(IBM390PS.P1234567)
      TSTIME(030104091131)
```

2. The CPC SAFOS is currently not connected to the INGHWCMD hardware interface.

```
AOFA0099 STATCOM SAFOS STATUS(NOT_CONNECTED)
      CPCSNAME(DEIBMD1.X7F1F20A)
      TSTIME(030104091145)
```

3. The CPC YORAMA is currently connected to the INGHWCMD hardware interface with a synchronous SNMP session. Synchronous sessions (SNMP or INTERNAL) allow the polling of CPC and image status and the retrieval of object information. HW events, such as status changes or messages from the HW or BCPs, cannot be processed with synchronous sessions.

```
AOFA0099 STATCOM SAFOS STATUS(CONNECTED)
      TASK(ISQCM001)
      TRANSPORT(S-SNMP)
      TARGET(SE)
      CPCSNAME(DEIBMIP1.IP3T1100)
      TSTIME(030104091145)
```

Asynchronous Response Messages (AOFA0100–AOFA0999)

Events from CPCs or CPC images arrive as asynchronous messages in NetView. The following event types are supported by the INGHWCMD interface:

- Messages from the Operating System (BCP messages)
- Status Changes of CPC or CPC image objects
- Hardware Messages from the CPC and its associated CPC images
- SNA Alert data from the CPC and its associated CPC images

AOFA0100 NEWSTATUS(obj_status_number) OLDSTATUS(obj_status_number)

Explanation: This message indicates a status change event of a CPC or CPC image object. Before you can receive this type of message, you must have successfully initialized an asynchronous session. Secondly, you must have specified a FILTER command with a NetView operator or group name as the receiver for each image or CPC that you target. See “Hardware Object Status Summary” on page 456 for a list of the supported object status numbers and their meaning.

Example: Partition KEY4 of FREEWAY was successfully CP stopped. The status changed from OPERATING (0001) to NOT_OPERATING (0002).

After the LPAR was CP started again, its status went back to OPERATING. Note that Prefix message AOFA0900 was used, because the FILTER command was specified without a PFX. No operating system messages are shown either, because the event notification that was enabled with the FILTER request was only ST, for object status change events.

```
INGHWCMD FREEWAY INITCOM ASYNC
INGHWCMD FREEWAY.KEY4 FILTER SET(ST) OP(TIL)
INGHWCMD FREEWAY.KEY4 STOP
```

```
AOFA0001 STOP FREEWAY.KEY4 STATUS(ACCEPTED) CPCINAME(KEY4) CPCSNAME(DEIBMD1.X7F1E30A)
      TSTIME(030208143117)
```

Asynchronous Response Messages (AOFA0100–AOFA0999)

AOFA0900 FREEWAY.KEY4 SC **AOFA0100 NEWSTATUS(0002) OLDSTATUS(0001)**

INGHWCMD FREEWAY.KEY4 START

AOFA0001 START FREEWAY.KEY4 STATUS(ACCEPTED) CPCINAME(KEY4) CPCNAME(DEIBMD1.X7F1E30A)
TSTIME(030208143134)

AOFA0900 FREEWAY.KEY4 SC **AOFA0100 NEWSTATUS(0001) OLDSTATUS(0002)**

AOFA0200 *CpcName.ImageName ConID Command_completion_response*

Explanation: This response message uses the following variables:

CpcName.ImageName

Specifies the name of the processor as defined in the SA z/OS policy and the image or LPAR name that this asynchronous command completion message originates from.

ConID

Always SC for System Console.

Command_completion_response

Message string providing the command name, completion status, and possible condition code information of an HW command.

AOFA0300 **Power Mode Change occurred, Old Status** (*Old-value*) **New Status** (*New-value*) **Object** (*CpcName*)

Explanation: This message indicates a change related to the CPC's Power[®] Mode settings. This message uses the following variables:

Old-value

Value before Power Mode was changed.

New-value

New Power Mode value.

CpcName

Specifies the name of the processor as defined in the SA z/OS policy.

AOFA0400 *CpcName ConID HW_message*

Explanation: This response message uses the following variables:

CpcName

Specifies the name of the processor as defined in the SA z/OS policy.

ConID

Always SC for System Console.

HW_message

The message text is identical to the short message text displayed on the SE or HMC that an HW message is waiting for, when a CPC object has been marked. Note that the message detail information can only be accessed using the HMC or SE GUI.

AOFA0500 *CpcName ConID* **Temporary Capacity Record** *record_id* **Change occurred** (*change_code*)

Explanation: This message indicates a change related to a Temporary Capacity record. It uses the following variables:

CpcName

Specifies the name of the processor as defined in the SA z/OS policy.

ConID

Always SC for System Console.

record_id

An 8 digit record number of the temporary capacity record.

change_code

A numeric code that indicates the type of change. Possible values are:

- 0** Capacity Record 'Add' occurred
- 1** Capacity Record: Delta
- 2** Capacity Record 'Delete' occurred
- 3** Capacity Record Accounting Data change
- 4** Capacity Record Activation Level change
- 5** Capacity Record: Priority Pending
- 6** Capacity Record: Other changes

Asynchronous Response Messages (AOFA0100–AOFA0999)

| **AOFA0600** *CpcName Cons_designator* **Temporary Capacity Change occurred**(*change_code*)

| **Explanation:** This message indicates a change related to Temporary Capacity management of the Target Hardware. It uses the following variables:

| *CpcName* Specifies the name of a processor as defined in the SA z/OS policy.

| *Cons_designator* Is always SC for System Console.

| *change_code* A numeric code that indicates the type of change. Possible values are:

- | **0** Capacity: Fenced Book
- | **1** Capacity: Defective Processor
- | **2** Capacity: Concurrent Book Replace
- | **3** Capacity: Concurrent Book Add
- | **4** Capacity: Capacity Check Stop
- | **5** Temporary Capacity Changes Allowed
- | **6** Temporary Capacity Changes Not Allowed

| **AOFA0700** **APPL_ENDED CONSOLE**(*con_name*) **REASON**(*reason*) **TYPE**(*type*) **COMPONENT**(*comp*)

| **Explanation:** This message indicates the end of the console application operation. This is the software on the Support Elements (SEs)/ Hardware Management Consoles (HMCs) that permits Operations Management tasks. SA z/OS requires this component to be operational in order to communicate with the SE/HMC.

| *con_name* Name of the affected console. For SE connections, this name is identical to the NAU name part of the CPC network address. For HMC connections this name is identical to the Console Name defined with the Customize Network Settings task from the HMC Console Settings Work Area.

| *reason* Issuer that caused the console application to end.

| **USER**
A logged on user

| **AUTOMATION**
An automation application.

| **OTHER**
Remote action such as SE reboot from an HMC.

| **UNKNOWN**
The reason code in the event data is unknown.

| **type** Specifies the action that caused the console application to end. Depending on the type given, a possible console application outage time can be predicted.

| **SHUTDOWN_CONSOLE**
Long outage, manual intervention is required to recover.

| **RESTART_APPLICATION**
Medium outage of 10 - 20 minutes can be expected until the console application is fully operational again.

| **RESTART_CONSOLE**
Medium outage of 10 -20 minutes can be expected until the console application is fully operational again.

| **UNKNOWN**
The type code in the event data is unknown. Long outage, manual intervention may be required to cover.

| **comp** Names the component that is involved in this event. In the case of reason code AUTOMATION, the involved component may be API. Note that the component information given in this report does not exceed more than 20 characters.

Asynchronous Response Messages (AOFA0100–AOFA0999)

AOFA0800 **CONSOLE**(*con_name*) **SECLOG REPORT** *date_time_stamp* *security_log_message*

Explanation: This message report is issued each time a new message is written to the Security Log of the Support Element (SE)/ Hardware Management Console (HMC) that is in session with SA z/OS. Depending on the length of the *security_log_message*, this report may have more than three AOFA0800 report lines. With a processor operations SNMP connection, you can use the ISQCCMD SECLOG common command to turn this function on or off.

con_name Name of the affected console. For SE connections, this name is identical to the NAU name part of the CPC network address. For HMC connections this name is identical to the Console Name defined with the Customize Network Settings task from the HMC Console Settings Work Area.

date_time_stamp Indicates the date and time, the record was written in the log. Date format is month-day-year and time format is hh:mm:ss:hss.

security_log_message

For more information about what types of messages are written to the console security log, use the 'View Security Logs' task of your SE/HMC console.

AOFA0900 *CpcName.ImageName* *ConID* *Event_message_string*

Explanation: This response message uses the following variables:

CpcName.ImageName

Specifies the name of the processor as defined in the SA z/OS policy and the image or LPAR name that this event message originates from.

ConID

For BCP message events the ID is OC, indicating a message from the operator console. For all other event messages, the ID is SC, indicating a message from the system console.

Event_message_string

Specifies the message text from the operating system as shown on the SE or HMC, or the message string specific to the other events type.

AOFA0996 **WAIT** *sec* **SECOND FOR TERMINATION OF A COMMAND SESSION: OPID**(*opid*) **CPCNAME**(*cpc*) **TSTIME**(*time*) **COMMAND**(*command*).

Explanation: The HW event handler for the CPC (*cpc*) session returned an error and the asynchronous session terminates. To perform a save memory cleanup, an HW command that is still active for the CPC running in task (*opid*) must be terminated first. Wait for (*sec*) seconds for normal command termination.

AOFA0997 **COMMAND SESSION WAS CANCELLED: OPID**(*opid*) **CPCNAME**(*cpc*) **TSTIME**(*time*) **COMMAND**(*command*) **NV STOP FORCE RC**(*rc*).

Explanation: The HW event handler for the CPC (*cpc*) session returned an error and the asynchronous session terminates. To perform a save memory cleanup, an HW command that is still active for the CPC running in task (*opid*) must be terminated first. A NetView STOP FORCE was executed with a return code of (*rc*).

AOFA0998 **EVENT HANDLER WARNING: ASYNCHRONOUS SESSION HAS BEEN REESTABLISHED**

Explanation: The HW event handler for this SE/HMC recovered a missing heartbeat condition signaled by the BCP Internal Interface. Some events might have been lost. Applications should query status information of the CPC or image in order to determine the current status. Note that BCP messages that are transported as HW events are not buffered and cannot be recovered if they are lost.

AOFA0999 **EVENT HANDLER ERROR RC**(*rc*) , **ASYNCHRONOUS SESSION IS TERMINATING.**

Explanation: The HW event handler for this SE or HMC session returned an error return code (*rc*). The asynchronous session terminates. An INITCOM ASNC must be issued to reactivate the session. Refer to "Data Exchange Services "0B100xxx"" on page 438 if the return code number is less than 100. Refer to "Internal Transport Services "0Bx00xxx"" on page 442 for return codes greater than 100. Note that only if you set a CPC event filter will you receive this event handler termination report.

This report might be issued in case of an unplanned or planned outage of the hardware management console (SE or HMC). For planned outages (maintenance), it is strongly recommended to first close the ongoing active target

Asynchronous Response Messages (AOFA0100–AOFA0999)

hardware connections to that hardware management console using command TERMCOM, or for Processor Operations users, command ISQXCLS.

Note: For Processor Operations connections, INITCOM is executed as part of host-based command ISQXIII (Initialize target connection).

Response Messages (AOFB0000-AOFB0099)

Automation returns the following messages to indicate command-invocation, parameter-list, or parameter-resolution problems or to report command execution results from the ProcOps ensemble commands

AOFB0000

Explanation: This response message is returned as an indicator for command-invocation, parameter-list, or parameter-resolution problems. If the AOFB0000 response message is returned from the ISQBXCMD command list, its data portion is an error string (see Table 9). If the AOFB0000 response message is returned from INGBXCOM or INGBCLS command processor, its data portion contains a condition code in form 00B600xx (see “Ensemble Communication Task Condition Codes “00B600xxx” on page 449 in *IBM Tivoli System Automation for z/OS Messages and Codes*).

Table 9. Error Strings

Error Type	Error Strings
Environment error	NetView_task_environment_must_be_CMD_LOW_for_OPID
Parm error	Timeout_specification_tttt_ends_invalid, Timeout_specification_tt_is_not_valid, Timeout_specification_range_tt_is_not_valid, Timeout_tttt_is_not_a_integer_value, Timeout_tt_out_of_range_1-65535, Timeout_range_tttt_already_defined
Resolve error	No_HWOPER_task_defined
Check task Check task	hwtask_reached_QueueLimit_qlim, hwtask_task_is_not_available

AOFB0001

Explanation: This response message is returned from a request of the following asynchronous zBX hardware command functions: ACTIVATE, DEACTIVATE and shows status of the command and optionally condition and sense codes for REJECTED or FAILED requests.

Example:

1. Activatate request accepted:

```
AOFB0001 ACTIVATE ENSR35 HMC(hmctrx) STATUS(ACCEPTED)
TSTIME(111013161610)
```

2. Activate request rejected with condition code 0B300019:

```
AOFB0001 DEACTIVATE ENSR35 HMC(hmctrx) STATUS(REJECTED)
CONDITION(0B300019) SENSE (0000 000 0000) TSTIME(120127174533)
```

AOFB0002

Explanation: This message is the response to a SUBSCRIBE request to Ensemble Notifications.

Response Messages (AOFB0000-AOFB0099)

AOFB0003

Explanation: This message is the response to an INITCOM request to an Ensemble. INITCOM establishes a session between INGBXCOM and the ensemble HMC of the addressed ensemble.

Example: The session between INGBXCOM and the Ensemble HMC where ENSR35 is an ensemble name, is established successfully.

```
AOFB0003 INITCOM ENSR35 HMC(hmctrx) STATUS(SUCCESS)
TSTIME(111013155723)
AOFB0003 ENSEMBLE(R35zBX) CPCS(1) ZBXS(1)
AOFB0003 RACKS(2) BCS(3) BLS(11) VHS(11)
AOFB0003 CPCSNAM(E)R35)
AOFB0003 INITCOM REPORT COMPLETE
```

The report shows the ensemble HMC hostname/IP address (HMC parameter), real ensemble name (ENSEMBLE parameter), number of discovered objects (CPCS, BLS, ZBXS, RACKS, BCS, VHS parameters) and the names of all CPCs (CPCNAME) nodes added to the data model.

AOFB0004

Explanation: This report is the response to a TERMCOM request scheduled by the INGBXCOM exception handler. The TERMCOM is scheduled internally in case of session problems, for example caused by TCPIP problems or temporary unavailability of the ensemble HMC of the addressed ensemble.

AOFB0005

Explanation: This report is the response to a TERMCOM request. TERMCOM ends a session between INGBXCOM and the ensemble HMC of the addressed ensemble.

Example: The session between INGBXCOM and the Ensemble HMC where ENS1 is an Ensemble name, is terminated successfully.

```
ISQBXCMD ENSR35 TERMCOM
AOFB0005 TERMCOM ENSR35 HMC(hmctrx) STATUS(SUCCESS)
TSTIME(111013155723)
AOFB0005 CPCSNAM(E)R35)
AOFB0005 TERMCOM REPORT COMPLETE
```

AOFB0011

Explanation: This report is the response to a ZBXDATA request. It returns a report consisting of multiple AOFB0011 messages. The ZBXDATA request retrieves the properties of a single ZBX object. An asterisk(*) in the reported property means that the property value is not defined.

Table 10. Reported Properties for AOFB0011

Reported property	Description
CPC	The name of the managing CPC.
NAME	The name of the zBX.
MTYPE	zBX Machine type.
MMODEL	zBX Machine model.
MSERIAL	zBX Machine serial.
DESC	Description of the zBX.
CDPXI50E	Number of blades entitled as DPXI50Z blades.
CISAOPTE	Number of blades entitled as ISAOPT blades.
CXE	Number of blades entitled as System x [®] IBM blades.
CPOWERE	Number of blades entitled as POWER [®] IBM blades.
MAXDPXI50E	Maximum licensed to be entitled as DPXI50Z blades.

Table 10. Reported Properties for AOFB0011 (continued)

Reported property	Description
MAXISAOPTE	Number of blades entitled as ISAOPT blades.
MAXPOWERE	Number of blades entitled as POWER IBM blades.
MAXXE	Maximum licensed to be entitled as System x blades.

Example: Example:

```
AOFB0011 ENSR35 CPC(R35) ZBX NAME(2458-002-000020000101)
AOFB0011 MTYPE(2458) MMODEL(002) MSERIAL(000020000101)
DESC(Represents one zBX)
AOFB0011 CDPXI50E(0) CISAOPTE(24) CPOWERE(9) CXE(2)
AOFB0011 MAXDPXI50E(112) MAXISAOPTE(112) MAXPOWERE(112) MAXXE(112)
AOFB0011 ZBXDATA REPORT COMPLETE
```

AOFB0012

Explanation: This report is the response to a BCDATA request. It returns a report consisting of multiple AOFB0012 messages. The BCDATA request retrieves the properties of a single blade center (BC). An asterisk (*) in the reported property means that the property value is not defined.

Table 11. Reported Properties for AOFB0012

Reported property	Description
CPC	The name of the managing CPC.
BC NAME	The Unified Resource Manager-assigned name of the zBX blade center.
MTYPE	BC type.
MMODEL	BC model.
MSERIAL	BC serial.
DESC	BC description.
LOCKD	The object is locked and thus disruptive actions or tasks cannot be performed on it.
STAT	The status of the blade center. Values are: <ul style="list-style-type: none"> "no-power" "operating"
ACCSTAT	The set of operational status values that the managed resource can be in and be considered to be in an acceptable (not alert causing) state.
UNACCSTAT	If true, the current operational status of the managed resource is not one of the acceptable statuses for this resource.
HASHWMSG	The BladeCenter® has a hardware message (true) or the BladeCenter does not have a hardware message (false).
LOC	Location code.
PCO	Specifies the current power consumption in watts(W) of this blade center. The blade center power consumption includes the power consumption of the blades contained within the blade center and the shared infrastructure.
AMBT	Specifies the input air temperature in degrees Celsius (°C) as measured by the system.

Response Messages (AOFB0000-AOFB0099)

Table 11. Reported Properties for AOFB0012 (continued)

Reported property	Description
EXHT	Specifies the exhaust air temperature in degrees Celsius (°C) as calculated by the system. This is useful in determining potential hot spots in the data center.

Example:

```
AOFB0012 ENSR35 CPC(R35) BC NAME(B.1)
AOFB0012 MTYPE(8852) MMODEL(PFM) MSERIAL(KQRZDTV)
DESC(Represents one BladeCenter)
AOFB0012 LOCKD(false) STAT(operating)
ACCSTAT(operating) UNACCSTAT(false) HASHMSG(false)
AOFB0012 LOC(B10B) PCO(832) AMBT(28.5) EXHT(32.5)
AOFB0012 BCDATA REPORT COMPLETE
```

AOFB0013

Explanation: This report is the response to a VHDATA request. It returns a report consisting of multiple AOFB0013 messages. The VHDATA request retrieves the properties of a single virtualization host object (VH). An asterisk (*) in the reported property means that the property value is not defined.

Table 12. Reported Properties for AOFB0013

Reported property	Description
CPC	The name of the managing CPC.
VH NAME	The display name of the virtual server. Names are unique to other existing virtual servers on the virtualization host.
TYPE	Type of virtual server. Values are: <ul style="list-style-type: none"> “power-vm” — a virtual server that has been defined on an IBM POWER7® blade. “x-hyp” — a virtual server that has been defined on an IBM System x blade.
LOCKD	The object is locked and thus disruptive actions or tasks cannot be performed on it.
STAT	The current operational status of the managed resource. Values are: <ul style="list-style-type: none"> “operating” — indicates the virtual server is deactivated, in the process of loading, or has an error “not-operating” — indicates the virtual server is deactivated, in the process of loading, or has an error “not-communicating”— indicates the HMC is not communicating with the support element. “status-check”- indicates the virtual server is powered on, but the virtualization host is not communicating so there is no status available. “definition-error” —indicates the specified zBX blade does not match the characteristics of the installed zBX blade. “no-power” — indicates the virtual server in the process of powering on. “stopped” — indicates the Support Element with the zBX blade is stopped.

Table 12. Reported Properties for AOFB0013 (continued)

Reported property	Description
ACCSTAT	The set of operational status values that the managed resource can be in and be considered to be in an acceptable (not alert causing) state.
UNACCSTAT	If true, the current operational status of the managed resource is not one of the acceptable statuses for this resource.
ASVS	Automatically start any virtual servers configured to start when the virtualization host is started.
INBMONSUP	If true, in-band monitoring is supporting for the virtualization host, allowing it to gather performance metrics on its virtual servers.
MIXEMODEBREST	Indicates if a virtual disk boot restriction exists for the virtualization host's virtual servers that would prevent booting from "virtio" virtual disks when both "ide" and "virtio" virtual disks are defined. In such a case, the virtual server will always boot from an "ide" virtual disk.
VSSHDWNTOUT	Amount of time, in seconds, to allow a virtual server to cleanly shutdown. After the elapsed time has passed, the virtual server will be forcefully stopped. The value may be -1 to indicate waiting "forever" or any integer value to specify an exact wait time in seconds.
MAXDEDP	The maximum number of dedicated processors that can be configured for a virtual server.
MAXIDED	The maximum number of IDE devices that can be configured for a virtual server.
MAXCPU	The maximum number of processing units that can be configured for a virtual server. Processing units are a unit of measure for shared processing power across one or more virtual processors. One shared processing unit on one virtual processor accomplishes approximately the same work as one dedicated processor.
MAXVCP	The maximum number of virtual processors that can be configured for a virtual server.
MAXMS4VS	The maximum memory size, in megabytes, that can be configured for a virtual server.
HOSTE NAME	The name of the hosting environment (blade) of the virtualization host.
HOSTE TYPE	The type of the hosting environment (blade) of the virtualization host.

Example:

```
AOFB0013 ENSR35 CPC(R35) VH NAME(C.2.02)
AOFB0013 TYPE(power-vm) LOCKD(false) STAT(operating)
ACCSTAT(operating) UNACCSTAT(true)
AOFB0013 ASVS(true) INBMONSUP(true) MIXEMODEBREST(*) VSSHDWNTOUT(300)
AOFB0013 MAXDEDP(7) MAXIDED(*) MAXCPU(*) MAXVCP(64) MAXMS4VS(27392)
AOFB0013 HOSTE NAME(C.2.02) HOSTE TYPE(power)
AOFB0013 VHDATA REPORT COMPLETE
```

Response Messages (AOFB0000-AOFB0099)

AOFB0014

Explanation: This report is the response to a VSDATA request. It returns a report consisting of multiple AOFB0014 messages. The VSDATA request retrieves the properties of a single virtual server (VS). An asterisk (*) in the reported property means that the property value is not defined.

Table 13. Reported Properties for AOFB0014

Reported property	Description
CPC	The name of the managing CPC.
VS NAME	The display name of the virtual server. Names are unique to other existing virtual servers on the virtualization host.
TYPE	Type of virtual server. Values are: <ul style="list-style-type: none"> “power-vm” — a virtual server that has been defined on an IBM POWER7 blade. “x-hyp” — a virtual server that has been defined on an IBM System x blade.
LOCKD	The object is locked and thus disruptive actions or tasks cannot be performed on it.
STAT	The current operational status of the managed resource. Values are: <ul style="list-style-type: none"> “operating” — indicates the virtual server is deactivated, in the process of loading, or has an error. “not-operating” — indicates the virtual server is deactivated, in the process of loading, or has an error. “not-communicating” — indicates the HMC is not communicating with the support element. “exceptions” — indicates the virtual server has a problem of some sort, such as not being able to access storage. “status-check”- indicates the virtual server is powered on, but the virtualization host is not communicating so there is no status available. “migrating” —indicates the virtual server is in the process of migrating to another virtualization host. “starting” — indicates the virtual server in the process of powering on. “stopping” — indicates the virtual server in the process of powering off.
ACCSTAT	The set of operational status values that the managed resource can be in and can be considered in an acceptable (not alert causing) state.
UNACCSTAT	If true, the current operational status of the managed resource is not one of the acceptable statuses for this resource.
AUTO	If true, this virtual server is automatically started when its hosting virtualization host is started.

Table 13. Reported Properties for AOFB0014 (continued)

Reported property	Description
BOOTMODE	The boot mode of the virtual server. Values are: <ul style="list-style-type: none"> • "normal" - The virtual server boots in normal mode. • "sms" - The virtual server boots to the System Management Services (SMS) menu. • "diagnostic-default-boot-list" - The sequence of devices read at startup. • "diagnostic-stored-boot-list" - The virtual server performs a service mode boot using the service mode boot list saved in NVRAM. • "open-firmware-prompt" - The virtual server boots to the open firmware prompt.
BOOTSEQ	List of boot sources. Values are: <ul style="list-style-type: none"> • "virtual-disk" – virtual-disks will be tried in order. • "network-adapter" – the first network-adapter will be used as a boot source. • "virtual-media" – virtual media will be used as a boot source.
DESC	Description of the virtual server.
BNACLNIP	The IP address of the virtual server used during network boot in dotted-decimal form.
BNAGATIP	The IP address of the gateway system used during network boot of the virtual server in dotted-decimal form.
BNASRVIP	The IP address of the boot server on which the disk boot source files reside in dotted-decimal form.
BNASUBIP	The IPv4 subnet mask of the network used during network boot of the virtual server in dotted-decimal form.
CPME	If true, management of processor performance is enabled for this virtual server if management of processor performance is enabled at the ensemble level.
DLPARA	If true, DLPAR support is enabled and specific virtual server properties may be updated when the virtual server status is "operating".
DLPARE	If true, this virtual server is configured for Dynamic Logical Partitioning.
INBMONE	If true, in-band monitoring support is enabled, allowing the hypervisor to gather performance metrics for the virtual server.
GPMPE	If true, guest platform management provider support is enabled, allowing the GPMP to gather performance data for work running on the virtual server.

Response Messages (AOFB0000-AOFB0099)

Table 13. Reported Properties for AOFB0014 (continued)

Reported property	Description
GPMP	Status of the System z Guest Platform Management Provider (GPMP). Values are: <ul style="list-style-type: none"> "unknown" – Guest performance agent status could not be determined. "not-operating" – Guest performance agent is not operating. "operating" – Guest performance agent is operating. "status-check" – There is a failure in communications between the virtual server and the guest performance agent.
GPMPV	Guest Platform Performance Manager version or keyword "unavailable" if not known.
HOSTNAME	Virtual server host name. This data is only available if the guest platform management provide is running on the virtual server.
MAC	The MAC address.
OSNAME	The name given to this system by its operating system. This data is only available if the guest platform management provider is running on the virtual server.
OSTYPE	The type of operating system that is running on the virtual server. This data is only available if the guest platform management provider is running on the virtual server.
OSLVL	The release level of the operating system, as reported by the OS itself. This data is only available if the guest platform management provider is running on the virtual server.
MEDIA	The display name of the mounted ISO or null if no media is mounted.
PMODE	The manner in which virtual processors are associated with the physical processors available on the virtualization host. Values are: <ul style="list-style-type: none"> "shared" - In shared mode, the virtual servers can use fractions of physical processors. The processor capacity is assigned in 0.1 units of physical processor, equivalent to 1.0 virtual processing unit. The virtual processor units can be shared among multiple virtual servers. "dedicated" - In dedicated mode, the processors are assigned in whole units of physical processors. The processors that are dedicated to the virtual server cannot be used by other virtual servers. The virtual server cannot use any processors other than its own dedicated processors.
INDEDP	Defines the initial number of dedicated processors that the virtual server can use; the number of dedicated processors to be provided to the virtual server when it is next activated.
INMEM	Initial memory value for use by the virtual server, specified in MB.

Table 13. Reported Properties for AOFB0014 (continued)

Reported property	Description
INPU	The number of processing units representing the initial processor scheduling target for this virtual server. If resources are available, the virtual server may receive more than this amount.
INVVP	Defines the initial number of virtual processors that the virtual server can use; the number of virtual processors to be provided to the virtual server when it is next activated.
MINDEDP	Defines the minimum number of dedicated processors that the virtual server can use.
MINMEM	Minimum memory value for use by the virtual server, specified in megabytes (MB).
MINPU	The minimum number of processing units required for this virtual server to start running.
MINVP	Defines the minimum number of virtual processors that the virtual server can use.
MAXDEDP	The upper limit for the number of dedicated processors to for the virtual server to consume.
MAXMEM	Maximum memory value for use by the virtual server, specified in megabytes (MB).
MAXPU	The maximum number of processing units that will be allocated to the virtual server (processor utilization capping).
MAXVP	The upper limit for the number of virtual processors to for the virtual server to consume.
WL NAME	The name of each workload resource group to which the virtual server is assigned.

Example:

```

AOFB0014 ENSR35 CPC(R35) VS NAME(SA34TST)
AOFB0014 TYPE(power-vm) LOCKD(false) STAT(not-operating) ACCSTAT()
UNACCSTAT(true)
AOFB0014 AUTO(true) BOOTMODE(normal) BOOTSEQ()
DESC(Test SA34 Test Server)
AOFB0014 BNACLNIP(null) BNAGATIP(null) BNASRVIP(null) BNASUBIP(null)
AOFB0014 CPME(true) DLPARA(false) DLPARE(false) INBMONE(false)
AOFB0014 GPMPE(not-operating) GPMPs(true) GPMPV(unavailable)
HOSTNAME(null) MAC(02:2d:c9:44:e1:00,40)
AOFB0014 OSNAME(null) OSTYPE(null) OSLVL(null) MEDIA(null)
PMODE(shared)
AOFB0014 INDEDP(null) INMEM(1024) INPU(0.1) INVVP(1)
AOFB0014 MINDEDP(null) MINMEM(1024) MINPU(0.1) MINVP(1)
AOFB0014 MAXDEDP(null) MAXMEM(1024) MAXPU(7.0) MAXVP(7)
AOFB0014 WL NAME(SA34)
AOFB0014 VSDATA REPORT COMPLETE

```

AOFB0015

Explanation: This report is the response to a WLDAPATA request. It returns a report consisting of multiple AOFB0015 messages. The WLDAPATA request retrieves the properties of a single workload resource group (WL). An asterisk (*) in the reported property means that the property value is not defined.

Response Messages (AOFB0000-AOFB0099)

Table 14. Reported Properties for AOFB0015

Reported property	Description
WL NAME	The display name specified for the workload resource group.
CAT	Up to 32 characters used to categorize the workload object, often with other workload resource groups within the ensemble.
ISDEF	This value identifies the default workload object. It is true for the default workload resource group and false for all other (custom) workload resource group in the ensemble.
PANC	The number of nodes (ensemble members) that have successfully activated the active performance policy for the workload resource group.
PASTAT	The status of the last performance policy activation. The possible values are: <ul style="list-style-type: none"> • "initializing" - The workload resource group performance function is being initialized and status is not yet known. • "in-progress"- Performance policy activation is in progress and any new activation request is rejected. • "active" - Performance policy activation is complete
DESC	Arbitrary text describing the workload resource group.
ACTPP NAME	Display name of the active performance policy in the workload resource group.
ACTPP ASTA	The activation status of the active performance policy object. The value is one of the following: <ul style="list-style-type: none"> • "not-active" - The performance policy is not currently the active policy in the workload resource group. • "in-progress" - The performance policy is currently being activated in the workload resource group. • "active" - The performance policy is currently the active policy in the workload resource group and its activation has completed.
DEFPP NAME	The name of the default performance policy object in the workload resource group.
DEFPP ASTA	The activation status of the default performance policy object. The value is one of the following: <ul style="list-style-type: none"> • "not-active" - The performance policy is not currently the active policy in the workload resource group. • "in-progress" - The performance policy is currently being activated in the workload resource group. • "active" - The performance policy is currently the active policy in the workload resource group and its activation has completed.
VS	The display name of the virtual server placed in the workload.
TYPE	The type of the virtual server placed in the workload.
STATUS	The status of the virtual server placed in the workload.

Table 14. Reported Properties for AOFB0015 (continued)

Reported property	Description
ITYPE	The reason(s) this virtual server is included in the workload resource group. The possible values are as follows: <ul style="list-style-type: none"> “direct”: The virtual server was added directly to the workload resource group. “custom-group”: The virtual server was added to the workload resource group due to its membership in one or more user-defined groups which belong to the workload resource group.
DM	If Y (yes), the virtual server is available in the ProcOps datamodel and can be managed using ensemble commands. If N (no), the server is unknown and should be discovered using DISCOVERY VS command if required for management.

Example:

```
AOFB0015 ENSR35 WL NAME(SA34)
AOFB0015 NAME(SA34) CAT() ISDEF(false) PANC(1) PASTAT(active)
DESC(SA34 TEST Workload)
AOFB0015 ACTPP NAME(Default) ASTA(active)
AOFB0015 DEFPP NAME(Default) ASTA(active)
AOFB0015 VS(SA34TST) TYPE(power-vm) STATUS(not-operating) ITYPE(*)
DM(Y)
AOFB0015 VS(SA34DEV) TYPE(power-vm) STATUS(starting) ITYPE(*) DM(Y)
AOFB0015 VS(SA34XSRV) TYPE(x-hyp) STATUS(operating) ITYPE(*) DM(Y)
AOFB0015 VS(SA34TEMP) TYPE(power-vm) STATUS(starting) ITYPE(*) DM(Y)
AOFB0015 WLDATA REPORT COMPLETE
```

AOFB0016

Explanation: This report is the response to a BLDATA request. It returns a report consisting of multiple AOFB0016 messages. The BLDATA request retrieves the properties of a single blade (BL). An asterisk (*) in the reported property means that the property value is not defined.

Table 15. Reported Properties for AOFB0016

Reported property	Description
CPC	The name of the managing CPC.
BL NAME	The Unified Resource Manager-assigned name of the blade.
MTYPE	Blade machine type.
MMODEL	Blade machine model.
MSERIAL	Blade machine serial
TYPE	Type of the blade. Values are: <ul style="list-style-type: none"> “power” — the System Z Power blade “system-x” — the System X blade “isaopt” — the IBM Smart Analytics Optimizer blade “dpxi50z” — the DataPower® X150 blade.
LOCKD	The object is locked and thus disruptive actions or tasks cannot be performed on it.

Response Messages (AOFB0000-AOFB0099)

Table 15. Reported Properties for AOFB0016 (continued)

Reported property	Description
STAT	The status of the blade. Possible values are: <ul style="list-style-type: none"> "no-power" — the blade is powered off "status-check" — the blade and the console are not communicating "not-operating"- the blade is powered on and communicating with the console but is not running work "stopped" — operations on the blade are quiesced "definition-error" — an error has occurred when loading the blade with the firmware "operating" — blade is operating normally
ACCSTAT	The set of operational status values that the managed resource can be in and be considered to be in an acceptable (not alert causing) state.
UNACCSTAT	If true, the current operational status of the managed resource is not one of the acceptable statuses for this resource.
HASHWMSG	The Blade has a hardware message (true) or the Blade does not have a hardware message (false).
LOC	Location code.
PCO	Specifies the current power consumption in watts(W) of this blade. The blade center power consumption includes the power consumption of the blades contained within the blade center and the shared infrastructure.
PROC	The number of zBX blade processors.
MEM	Memory size of the zBX blade specified in the MB.
VH NAME	The name of the virtualization host being hosted by the blade.
VH TYPE	The type of the virtualization host being hosted by the blade.

Example: Example:

```
AOFB0016 BLDATA ENSR35 CPC(R35) BL NAME(C.2.01)
AOFB0016 MTYPE(8406) MMODEL(71Y) MSERIAL(06F00FA) TYPE(power)
AOFB0016 LOCKD(false) STATUS(operating) ACCSTAT(operating)
UNACCSTAT(false) HASHWMSG(false)
AOFB0016 LOC(C01BBS01) PCO(155) PROC(1) MEM(32768)
AOFB0016 VH NAME(C.2.01) VH TYPE(power-vm)
AOFB0016 BLDATA REPORT COMPLETE
```

AOFB0017

Explanation: This report is the response to an ENSDATA request. It returns a report consisting of multiple AOFB0017 messages. The ENSDATA request retrieves the properties of the ensemble. An asterisk (*) in the reported property means that the property value is not defined.

Table 16. Reported properties for AOFB0017

Reported property	Description
NAME	The display name specified for the ensemble.
DESC	Text describing the Ensemble.

Table 16. Reported properties for AOFB0017 (continued)

Reported property	Description
LOCKD	The object is locked and thus disruptive actions or tasks cannot be performed on it.
ST	The status of the ensemble representing the current communication status between the primary and alternate HMC: <ul style="list-style-type: none"> "alternate-communicating" – The primary is communicating to the alternate. "alternate-not-communicating" - The primary is not communicating to the alternate.
UNACCSTAT	If true, the current operational status of the managed resource is not one of the acceptable statuses for this resource.
ACCSTAT	The set of operational status values that the managed resource can be in and be considered to be in an acceptable (not alert causing) state.
MGMTELVL	The Unified Resource Manager management enablement level for this ensemble. The level determines which Unified Resource Manager advanced management functions are available for use. Values are: <ul style="list-style-type: none"> "manage"- Gives you the basics for managing an ensemble. It includes HMC operational controls for IBM zEnterprise BladeCenter Extension(zBX), change management of firmware across the ensemble, energy monitoring, virtual networking with automated provisioning, virtual server management, and a base level of performance management. "automate"- Provides more leverage from the ensemble by managing workloads and energy. This level of support includes power capping, power savings mode, and performance management.
LBE	If true, Load Balancing is enabled for this ensemble.
LBIP	The IPV4 address or IPV6 addresses of Load Balancers allowed access to the Load Balancing function.
LBP	The Load Balancing port value in the range 1024-65534. The default port is 3860.
MAC	The ensemble MAC prefix.
RMAP	The list of reserved MAC address prefixes.
ULUP	Unique local address (ULA) prefix applied to addresses used for management communication between virtual servers and their virtualization hosts.
PCO	Specifies the current power consumption in watts (W) for this ensemble.
PR	Specifies the maximum power usage in watts (W) of this ensemble. This is a calculated value as indicated by the electrical rating labels or system rating plates of the ensemble components.
CPMEPVM	If true, management of processor performance is enabled for PowerVM® virtualization hosts. Management of processor performance is also available for virtual servers.

Response Messages (AOFB0000-AOFB0099)

Table 16. Reported properties for AOFB0017 (continued)

Reported property	Description
CPMEZVM	If true, management of processor performance is enabled for z/VM virtualization hosts. Management of processor performance is also available for virtual servers.

Example:

```
AOFB0017 NAME(R35zBX) DESC(ISA0) LOCKD(false)
AOFB0017 ST(alternate-communicating) UNACCSTAT(false)
ACCSTAT(alternate-communicating)
AOFB0017 MGMTLVL(automate) LBE(false) LBIP() LBP(3860)
AOFB0017 MAC(02:00:00:00:00:00) RMAP(02:b7:00:00:00:00)
ULUP(fd6d:dba4:3f11:0:0:0:0)
AOFB0017 PCO(28209) PR(65644) CPMEPVM(false) CPMEZVM(false)
AOFB0017 ENSDATA REPORT COMPLETE
```

AOFB0019

Explanation: This report is the returned response to a FILTER LIST command.

Table 17. Reported Properties for AOFB0019

Reported property	Description
FILTER	A list of URM notification types currently subscribed to.
CLASSES	A list of object classes currently subscribed to.
SELECTOR	URM API JMS Selector.
OP	The defined notifications are only forwarded to the specified operators or operator group, if at least one filter is set. If a prefix field PFX was specified with a filter SET command, its text is placed in front of every event message (ISQ800I).
PFX	Prefix for each notifications for specific operator or operators group.

Example:

```
AOFB0019 FILTER(job-completion,status-change)
AOFB0019 CLASSES(ENS,NODE,CPC,ZBX,RACK,BC,BL,VH,VS,WL)
AOFB0019 SELECTOR()
AOFB0019 OP(+B00001S) PFX(ISQ800I)
AOFB0019 OP(+X00001S) PFX(ISQ801I)
AOFB0019 FILTER REPORT COMPLETE
```

AOFB0020

Explanation: This report is the response to the following requests: GETESTAT, GETBLSTAT, GETBCSTAT, GETVHSTAT and GETVSSTAT request.

GETESTAT queries the state of the ensemble.

GETBCSTAT queries the status of one or multiple blade center objects.

GETBLSTAT queries the status of one or multiple blade objects.

GETVHSTAT queries the status of one or multiple virtualization host objects.

GETVSSTAT queries the status of one or multiple virtual servers.

It returns a report consisting of multiple AOFB0020 messages. The first message shows the data model filtering criteria used for the command, followed by multiple messages showing status of the required objects. The status field of the message may have one of the following issues:

Table 18. Request Values for AOFB0020

Request	Values
GETESTAT	alternate-communicating, alternate-not-communicating
GETBCSTAT	no-power, operating
GETBLSTAT	operating, status-check, not-operating, stopped, definition-error, no-power, stopped
GETVHSTAT	operating, not-operating, status-check, not-communicating, definition-error, no-power, stopping
GETVSSTAT	operating, not-operating, not-communicating, exceptions, migrating, starting

On successful completion, the status field of message AOFA0017 may have one of the following values:

GETESTAT	GETBCSTAT	GETBLSTAT
-----	-----	-----
ALTERNATE-COMMUNICATING	NO-POWER	OPERATING
ALTERNATE-NOT-COMMUNICATING	OPERATING	STATUS-CHECK
		NOT-OPERATING
		STOPPED
		DEFINITION-ERROR
		NO-POWER
		STOPPING
GETVHSTAT	GETVSSTAT	
-----	-----	
OPERATING	OPERATING	
NOT-OPERATING	NOT-OPERATING	
STATUS-CHECK	NOT-COMMUNICATING	
NOT COMMUNICATING	EXCEPTIONS	
DEFINITION-ERROR	MIGRATING	
NO-POWER	STARTING	
STOPPING		

Example:

1. GETBLSTAT sample report.

```
AOFB0020 ENSR35 BL NAME(*) CPC(*) TYPE(*)
AOFB0020 NAME(B.1.03) TYPE(isaopt) STATUS (not-operating)
AOFB0020 NAME(B.1.04) TYPE(isaopt) STATUS (not-operating)
AOFB0020 NAME(C.1.01) TYPE(system-x) STATUS (operating)
AOFB0020 NAME(C.1.14) TYPE(power) STATUS (operating)
AOFB0020 NAME(C.1.11) TYPE(power) STATUS (operating)
AOFB0020 GETBLSTAT REPORT COMPLETE
```

2. GETVSSTAT BL(C.2.02) sample report:

```
AOFB0020 ENSR35 VS NAME(*) CPC(*) BL(C.2.02) VH(C.2.02) TYPE(*)
AOFB0020 NAME(SA34TST) TYPE(power-vm) STATUS(not-operating)
AOFB0020 NAME(SA34DEV) TYPE(power-vm) STATUS(starting)
AOFB0020 NAME(SA34TMP) TYPE(power-vm) STATUS(operating)
AOFB0020 GETVSSTAT REPORT COMPLETE
```

3. GETESTAT sample report:

```
AOFB0020 NAME(R35zBX) TYPE(*) STATUS(alternate-communicating)
AOFB0020 GETESTAT REPORT COMPLETE
```

AOFB0021

Explanation: This report is the response to a LIST request. It returns a report consisting of multiple AOFB0021 messages. The LIST request retrieves the base properties of the required objects available in the ProcOps data model. An asterisk (*) in the reported property means that the property value is not defined.

Example:

1. LIST CPC sample output

Response Messages (AOFB0000-AOFB0099)

```

| AOFB0021 ENSR35 CPC(R35)
| AOFB0021 LIST REPORT COMPLETE
| 2. LIST ZBX sample output
| AOFB0021 ENSR35 CPC(R35) ZBX(2458-002-000020000101)
| AOFB0021 LIST REPORT COMPLETE
| 3. LIST VH sample output
| AOFB0021 ENSR35 CPC(R35) BL(C.1.01) VH(C.1.01) TYPE(x-hyp)
| AOFB0021 ENSR35 CPC(R35) BL(C.1.02) VH(C.1.02) TYPE(x-hyp)
| AOFB0021 ENSR35 CPC(R35) BL(C.1.11) VH(C.1.11) TYPE(power-vm)
| AOFB0021 ENSR35 CPC(R35) BL(C.1.12) VH(C.1.12) TYPE(power-vm)
| AOFB0021 LIST REPORT COMPLETE
| 4. LIST BL sample output
| AOFB0021 ENSR35 CPC(R35) BL(B.1.03) TYPE(isaopt)
| AOFB0021 ENSR35 CPC(R35) BL(C.1.01) TYPE(system-x)
| AOFB0021 ENSR35 CPC(R35) BL(C.1.02) TYPE(system-x)
| AOFB0021 ENSR35 CPC(R35) BL(C.1.11) TYPE(power)
| AOFB0021 ENSR35 CPC(R35) BL(C.1.12) TYPE(power)
| AOFB0021 ENSR35 CPC(R35) BL(C.2.03) TYPE(power)
| AOFB0021 LIST REPORT COMPLETE
| 5. LIST VS sample output
| AOFB0021 ENSR35 CPC(R35) VH(C.1.13) TYPE(power-vm)
| AOFB0021 VS(pbxericstst1)
| AOFB0021 ENSR35 CPC(R35) VH(C.1.13) TYPE(power-vm)
| AOFB0021 VS(pbxericstst2)
| AOFB0021 ENSR35 CPC(R35) VH(C.1.01) TYPE(x-hyp)
| AOFB0021 VS(sazbx114)
| AOFB0021 ENSR35 CPC(R35) VH(C.1.02) TYPE(x-hyp)
| AOFB0021 VS(sazbx121)
| AOFB0021 LIST REPORT COMPLETE

```

AOFB0022

Explanation: This report is the response to a CONDATA request. It returns a report consisting of multiple AOFB0022 messages. The CONDATA request retrieves the properties of the ensemble primary Hardware Management Console. An asterisk (*) in the reported property means that the property value is not defined.

Table 19. Reported Properties for the Console Object in AOFB0022

Reported Property	Description
NAME	The installation assigned name.
VER	The version number for the Console object.
ROLE	An indication ("primary" or "alternate") of how the Console object functions in a pairing, or null if this Console object is not paired.
AUTOSWTCH	Automatic switching between primary and alternate Hardware Management Consoles is enabled (true), or is not enabled (false).
IPSWP	Whether the current primary HMC IP addresses will be moved when making the alternate HMC the new primary (true) or not (false).
MMODEL	The model of the machine on which the Hardware Management Console is executing.
MSERIAL	The serial number of machine on which the Hardware Management Console is executing.
MTYPE	The type of machine on which the Hardware Management Console is executing.
ACTN	The pending action activation.

Response Messages (AOFB0000-AOFB0099)

Table 19. Reported Properties for the Console Object in AOFB0022 (continued)

Reported Property	Description
TYPE	The pending action type.
PENDING	Is the action pending (true) or not pending (false).
THIS HMC	The collection of network information for the local Hardware Management Console.
DOMAIN	The domain name configured for the network interface.
NAME	The Hardware Management Console name.
INTERFACE	The network interface name.
ISPRIVATE	Whether the interface is private(true) or public(false).
MAC	The MAC address of the network interface.
IPV4	The IPV4 address.
SUBNET	The IP mask value.
IPV6	The IPv6 address.
PFX	The number of leading bits of the IPv6 address that represent the network prefix.
PAIRED HMC	Describes the IP information for the paired Hardware Management Console known to the local HMC.
NAME	The Hardware Management Console name.
IPV4	The IPv4 address.
IPV6	The IPv6 address.
EC	Engineering Change information (SYSTEM stream only).
NUM	Engineering Change stream identifier.
PNUM	Engineering Change stream part number.
TYPE	Engineering Change stream name.
DESCR	Engineering Change stream descriptive text.
MCL	The list of activated or applied MicroCode Levels for this Engineering Change.
Level	Microcode level.
Type	Microcode level type.
Last	Time stamp of the last update, in the number of milliseconds since Midnight January 1, 1970 UTC.

Example:

```

AOFB0022 ENSR35 NAME(hmctst) VER(2.11.1) ROLE(primary) AUTOSWCH(true)
IPSWP(true)
AOFB0022 MMODEL(C1G) MSERIAL(99A3239) MTYPE(7327)
AOFB0022 ACTN(*) TYPE(*) PENDING(*)
AOFB0022 THIS HMC(DOMAIN(boeblingen.de.ibm.com) NAME(hmctest)
INTERFACE(eth0))
AOFB022 THIS HMC(ISPRIVATE(false) MAC(0012345678CD))
AOFB022 THIS HMC(IPV4(192.168.237.110) SUBNET(255.255.255.224))
AOFB022 THIS HMC(IPV6(fe80:0:0:0:b1fc:ea0:81e3:8bd5) PFX(64))
AOFB022 PAIRED HMC(NAME(hmctsta))
AOFB022 PAIRED HMC(IPV4(192.168.237.111))
AOFB022 PAIRED HMC(NAME(HMCTSTA))
AOFB022 PAIRED HMC(IPV6(fe80:0:0:0:b1fc:fcff:fe27:888b%2))

```

Response Messages (AOFB0000-AOFB0099)

```
| AOFB022 EC(NUM(N48180) PNUM(45D8928) TYPE(SYSTEM)
| DESCR(Hardware Management Console Framework))
| AOFB0022 MCL(LEVEL(300) TYPE(activated) LAST(1330362078000))
| AOFB0022 CONDATA REPORT COMPLETE
```

AOFB0030

Explanation: This report is the response to a DISCOVERY or DROP request. It returns the number of objects added or removed from the ProcOps data model.

Example:

```
| 1. DISCOVERY VS sample output
|   AOFB0030 ENSR35 DISCOVERED VS(20)
|   AOFB0030 DISCOVERY REPORT COMPLETE
| 2. DROP WL sample object
|   AOFB0030 ENSR35 DROPPED WL(2)
|   AOFB0030 DROP REPORT COMPLETE
```

AOFB0097

Explanation: This report is returned in response to a RESUME command request.

Example: 1:

```
| AOFB0097 RESUME ENSR35 STATUS(SUCCESS) REASON(MANUAL) ENSEMBLE(ENSR35) TSTIME(090530134423)
```

2:

```
| AOFB0097 RESUME ENSR35 STATUS(REJECTED) REASON(NOT_SUSPENDED) ENSEMBLE(ENSR35) TSTIME(090530134423)
```

In the first example, the connection to ensemble ENSR35 ECL2 was successfully resumed. The returned resume reason MANUAL indicates that this session/connection state was set manually using the RESUME command. In the second example you can see how the response looks, if you try to resume a session/connection which is not suspended.

AOFB0098

Explanation: This report is returned in response to a SUSPEND command request.

Example: 1:

```
| AOFB0098 SUSPEND ENSR35 STATUS(SUCCESS) REASON(MANUAL) ENSEMBLE(ENSR35) TSTIME(090530134423)
```

2:

```
| AOFB0098 SUSPEND ENSR35 STATUS(REJECTED) REASON(ALREADY_SUSPENDED) ENSEMBLE(ENSR35) TSTIME(090530134423)
```

In the first example the connection to ensemble ENSR35 was successfully suspended. The returned suspend reason MANUAL indicates that this session/connection state was set manually using the SUSPEND command. In the second example you can see how the response looks like, if you try to suspend an already suspended session/connection.

AOFB0099

Explanation: This report is returned as the response to a connection status request command.

Example:

```
| AOFB0099 STATCOM ENSR35 STATUS(CONNECTED) TASK(ISQEM001) HMC(hmcstat)
| HMCVER(2.11.1) MINAPIVER(1) MAXAPIVER(1) TST(120206165704)
```

Asynchronous Response Messages (AOFB0100-AOFB0999)

The Web Services API includes an asynchronous notification facility by which client applications may subscribe to and receive notification messages regarding a set of predefined management events. These events include:

- Addition and removal of managed objects to/from the inventory of resources that are managed by the HMC
- Changes to specified properties of managed object instances
- Changes to the operational status of managed objects
- Completion of asynchronously processed jobs.

The notification messages arrive as asynchronous messages in NetView:

- AOFB0100 - for the status change notifications
- AOFB0200 - for property change notifications
- AOFB0300 - for the completion of asynchronously processed jobs
- AOFB0400 - for inventory change notifications
- AOFB099x - for errors in the event notification session causing session termination and requiring recovery operations.

AOFB0100 **short_class CLASS(class) CPC(cpc) VH(vh) NAME(name) TYPE(type) NEWSTATUS(new_status) OLDSTATUS(old_status)**

Explanation: This message indicates a status change notification of an object. Before you can receive this type of message, you must have successfully initialized an ensemble session. Secondly, you must have been subscribed for the STATUS change notifications for the class with the SUBSCRIBE command.

Example: Virtual server SA34DEV is stopping.

```
AOFB0100 VS CLASS(virtual-server) CPC(R35) VH(C.2.01) NAME(SA34DEV)
TYPE(power-vm) NEWSTATUS(stopping) OLDSTATUS(starting)
```

AOFB0200 **AOFB0200 short_class CLASS(class) CPC(cpc) VH(vh) NAME(name) TYPE(type) PROPERTY(property) NEWVALUE(new_value) OLDVALUE(old_value)**

Explanation: This message indicates a property change notification of an object. Before you can receive this type of message, you must have successfully initialized an ensemble session. Secondly, you must have been subscribed for the PROPERTY change notifications for the class with the SUBSCRIBE command.

Example: 1: The description of the virtual server SA34DEV is changed.

```
AOFB0200 VS CLASS(virtual-server) CPC(R35) VH(C.2.01) NAME(SA34DEV)
TYPE(power-vm) PROPERTY(description) NEWVALUE(Development server for
SA34) OLDVALUE(Initial server setup)
```

2: SA34DEV is locked for disruptive task.

```
AOFB0200 VS CLASS(virtual-server) CPC(R35) VH(C.2.02) NAME(SA34TST)
TYPE(power-vm) PROPERTY(is-locked) NEWVALUE(true) OLDVALUE(false)
```

AOFB0400 **CLASS(class) INVENTORY(inv)**

Explanation: This message indicates the addition or removal of a managed object to/from the current inventory of resources that are being managed by the Unified Resource Manager. This occurs when managed resources are created or deleted, but also may occur in other situations, such as when Unified Resource Manager reestablishes its inventory of (already-existing) managed resources upon restart of the primary HMC. Before you can receive this type of message, you must have successfully initialized an ensemble session. Secondly, you must have been subscribed for the INVENTORY change notifications for the class with the SUBSCRIBE command.

Example: New virtual server added to the ensemble.

```
AOFB0400 CLASS(virtual-server) INVENTORY (add)
```

Asynchronous Response Messages (AOFB0100-AFB0999)

AOFB0996 WAIT *sec* SECOND FOR TERMINATION OF A COMMAND SESSION OPID(*opid*) TSTIME(*time*)
COMMAND(*command*)

Explanation: The HW event handler for ensemble session returned an error and then terminates. In order to perform a save memory cleanup, a still active HW command for the ensemble running in task (*opid*) needs to be terminated first. Wait for (*sec*) seconds for normal command termination.

AOFB0999 EVENT HANDLER ERROR RC(*rc*), ENSEMBLE IS TERMINATING

Explanation: The HW event handler for this ensemble HMC session returned an error return code (*rc*). The ensemble session terminates. An INITCOM must be issued to reactivate the session.

Condition Codes

This section gives further information about the condition codes for errors that are associated with the following:

- "Hardware Communication Task Condition Codes "00B00xxx""
- "Data Exchange Services "0B100xxx"" on page 438
- "Command Services "0B200xxx"" on page 440
- "Internal Transport Services "0Bx00xxx"" on page 442
- "Ensemble HMC API Exchange Services Condition Codes" on page 446
- "Ensemble Communication Task Condition Codes "00B600xxx"" on page 449

Hardware Communication Task Condition Codes "00B00xxx"

Table 20 lists the condition codes for Hardware Communication Task "00B00xxx".

Table 20. Hardware Communication Task Condition Codes

Reason Code	Error String	Error Description
001	ING_invalid_HLL_buffer	INGHWCOM was invoked, but the NetView HLL buffer found for C/C++ is not valid.
002	ING_origuser_invalid	The user ID and output correlator passed to INGHWCOM are not valid.
003	ING_interface_invalid	The hardware interface name passed to INGHWCOM is not valid. Allowed interface names are INTERNAL, SNMP, or SNA.
004	ING_interface_missing	No hardware interface name is passed to INGHWCOM.
005	ING_tgt_length-error	Parsing Error: The target object name (processor or image name) has an invalid length. It must be 1 to 8 characters.
006	ING_tgt_missing	Parsing Error: The target object name (processor or image name) is not specified.
007	ING_cpc_length_error	Parsing Error: The CPC address specification netid.nau has an invalid length. It must not exceed 17 characters.
008	ING_cpc_missing	Parsing Error: The CPC address specification, which is a required parameter for the request, is missing.
009	ING_imgname_length_err.	Parsing Error: The image name (LPAR name) parameter has an invalid length.
00A	ING_imgname_missing	Parsing Error: The image name (LPAR name) is a required parameter for the request, but has not been specified.
00B	ING_force_invalid	Parsing Error: The FORCE option is specified in the request but is not supported for the HW function. The following HW functions allow the FORCE option: ACTIVATE, DEACTIVATE, SYSRESET, LOAD
00C	ING_force_missing	Parsing Error: The FORCE option is required for the request, but has not been specified.
00D	ING_auth_missing	Parsing Error: The AUTHENTICATION specification that is required for each request is missing.
00E	ING_timeout_missing	Parsing Error: The required TIMEOUT parameter is missing in the request.

Table 20. Hardware Communication Task Condition Codes (continued)

Reason Code	Error String	Error Description
00F	ING_OCFCMD_truncated	Parsing Error: The HW function (OCFCMD) exceeds the maximum allowed length, which is 40 characters.
010	ING_OCFCMD_missing	Parsing Error: No HW function (OCFCMD) was specified in the request.
020	ING_SNMP_noIP_address	Parameter Resolution Error: The SNMP interface was specified for the HW request, but no IP address information is available.
021	ING_OCF_resolve_failed	Parameter Resolution Error: No HW function name to resolve, same as error 010.
022	ING_OCF_not_resolved	Parameter Resolution Error: An invalid HW function name was detected.
023	ING_BCP_null_cmd	Parameter Resolution Error: The BCP command retrieved from NetView common global variable ING.xxxx.COMDTEXT was empty.
024	ING_BCP_cmd_acc	Parameter Resolution Error: The NetView common global variable containing the BCP command could not be retrieved.
025	ING_session_type_missing_or_invalid	Parameter Resolution Error: Session type values SYNC or ASYNC missing, or an invalid session type value was passed.
026	ING_active_session_type_mismatch	Parameter Resolution Error: A session was requested having a different session type SYNC/ASYNC than the active session. Reissue the INITCOM request with the correct session type or terminate the active session prior to requesting a new one.
030	ING_nt_alloc_error	Storage Allocation Error: The Netid base table could not be allocated using CNMNAMS services.
031	ING_img_alloc_error	Storage Allocation Error: The storage for a system image could not be allocated using CNMNAMS services.
032	ING_img_locate_error	Storage Allocation Error: The previously allocated storage for a system image could not be located using CNMNAMS services.
033	ING_notinit_error	Storage Allocation Error: An HW function request was issued for a processor or system image without having allocated storage for that processor or system image. This happens if no INITCOM request was made prior to the first a HW function request.
050	ING_notinitialized_error	HW Function Error: An HW function request was issued for a processor or system image without having done an INITCOM. Same as error 033.
051	ING_imgnotfound_error	HW Function Error: An HW function request was issued for a system image that could not be located as an image belonging to the addressed CPC. Possible causes are: <ol style="list-style-type: none"> 1. The image name may be misspelled. 2. If the name was taken from configuration data, this may be invalid. 3. If, since the last INITCOM, the CPC configuration was changed with a Power On Reset, but no INITCOM was made afterwards, new CPC images are not recognized. 4. A INGHWCMD image request was entered without a required image name as the target.
052	ING_funcunknown_error	HW Function Error: An unknown HW function name was requested. Same as error 022.
053	ING_nocpobject_error	HW Function Error: GETCLUSTER failed. Cluster list attribute not resolved by the processor support element.
054	ING_nocluster_error	HW Function Error: GETCLUSTER failed. The cluster list returned by the processor support element was empty.
055	ING_nohwstatus_error	HW Function Error: An HW function that requires the determination of the status of the object prior to execution cannot be processed because the object status cannot be determined. This error is valid only for processors where the FORCE option has to be emulated by INGHWCOM.
056	ING_disruptive_cmd	HW Function Error: A disruptive HW function was requested without the FORCE option and the processor/image object is in an operational state. INGHWCOM uses FORCE(NO) (allow no disruptive commands) as default. If you want to allow disruptive commands you must specify the FORCE option in the INGHWCMD request.

Condition Codes

Table 20. Hardware Communication Task Condition Codes (continued)

Reason Code	Error String	Error Description
057	ING_noistatus_error	HW Function Error: CBU or OOCOD failed. A CBU or OOCOD function was requested but the current CBU / OOCOD status cannot be determined.
058	ING_noiobject_error	HW Function Error: Command failed. The processor hardware does not support the specified object attribute or the object attribute has a blank value.
059	ING_cbustatus_error	HW Function Error: CBU failed. A CBU status was returned that does not allow the request.
060	ING_filter_error	HW Function Error: A filter SET/UNSET command failed. Either a filter table is full and no new filters can be set, or a specified image name does not exist on the CPC. A maximum of 10 filter entries can be set per image. This error is also shown when doing a SET for an event type of BCP for a CPC object in LPAR mode. For CPC objects, this type of event is only supported when running in a BASIC processor mode.
061	ING_filter_not_async	HW Function Error: A filter command was entered for a SYNCHRONOUS session of the BCP Internal Interface, where filter commands are not supported. Synchronous sessions over the BCP Internal Interface are established if the SE MCL does not support asynchronous sessions.
062	ING_noobjvar_error	CCNTL, ICNTL, APROF error. The variable name that represents an object attribute for a CPC or image was not specified. The execution of the command is rejected. Contact your IBM Support Center.
063	ING_objvar_invalid_error	CCNTL, ICNTL, APROF error. The variable name that was specified in the HW command is not supported; the command is rejected. Specify a valid variable name and retry the command.
064	ING_objvar_opcode_error	CCNTL, ICNTL, APROF error. The internal opcode indicating a GET or a SET operation was not recognized. The command failed. Contact your IBM Support Center.
065	ING_noobjvar_value_error	CCNTL, ICNTL, APROF error. The required value operand for a SET operation was not specified. The command is rejected. Contact your IBM Support Center.
066	ING_aprf_error	APROF OPEN error. The allocation of storage for a activation profile list failed. Contact your IBM Support Center.
067	ING_isopen_error	APROF OPEN error. Activation profiles are already open, the request was rejected. Issue a APROF CLOSE request first.
068	ING_ptype_error	APROF error. The specified activation profile type is wrong. Allowed profile types are: RESET, IMAGE, or LOAD
069	ING_not_open_error	APROF error. Profiles are not OPEN. The requested profile function is rejected. Perform APROF OPEN and repeat the request.
06A	ING_profile_not_found	APROF READ/UPDATE error. The profile name that was specified in the request was not found. Verify the named activation profile and repeat the request. If the profile was newly created since the last APROF OPEN, close and open the profiles again to pick up the new profile name.
06B	ING_profupd_op_error	APROF UPDATE error. The update operator in the APROF request parameter list was not recognized. The request is rejected due to an invalid parameter list. Contact your IBM Support Center.
06C	ING_list_of_vars_error	APROF UPDATE error. The variable name list for this profile type could not be located. Contact your IBM Support Center.
06D	ING_variable_name_not_found_error	APROF UPDATE error. The specified variable name could not be found in the list of variables for that profile type. Use a valid variable name for the profile type and reissue the request.
06E	ING_variable_value_error	APROF UPDATE error. The value specified for the variable was not accepted. Correct the value and reissue the request.
06F	ING_OSSLP_acc_error	APROF UPDATE error. The common global access to retrieve the OSSLP data failed. Contact your IBM Support Center.
06G	ING_noproctype_error	ICNTL and APROF error. The processor type value of the request is invalid. The request is rejected.

Table 20. Hardware Communication Task Condition Codes (continued)

Reason Code	Error String	Error Description
070	ING_hmccpc_tbl_error	Initialization Error: An internal HMC/CPC/IMG table error occurred during INITCOM processing. Enable AOCTRACE and rerun INITCOM to get additional information about the problem.
071	ING_cpcimg_alloc_error	Initialization Error: For an image, the dynamic storage allocation request failed during INITCOM processing. Enable AOCTRACE and rerun INITCOM to get additional information about the problem.
072	ING_cpc_config_mismatch	Initialization Error: SNMP connections only. For SE connections, the CPC SNA address returned from the SE is different to the CPC SNA address configured for the CPCname used in the INITCOM request. For HMC connections, the configured SNA address of the CPCname of the INITCOM request is not defined on this HMC. For both connection types the communication is terminated.
080	ING_applname_error	RESERVE error. The application name specified to request or release a RESERVE, is not valid. The name must be one to eight characters long. The strings 'LIST,' 'ON', or 'OFF' are not allowed as application names.
081	ING_rsvparm_error	RESERVE error. The parameter values following the application name, must be either ON or OFF.
082	ING_session_not_sync	CTRLCONS error. Invalid session type of ASYNC encountered. It must be SYNC.
083	ING_oocodcmd_error	OOCOD error. Invalid OOCOD command provided (it must be STATUS, ACTIVATE, or UNDO).
084	ING_oocodprm_error	OOCOD error. No Order Number provided for the OOCOD ACTIVATE command.
085	ING_oocodstatus_error	OOCOD error. The current ooCoD status of the target does not allow the action (for example, an UNDO without a previous ACTIVATE).
0A0	ING_invalid_task	HW Task Error: The HW communication interface is running on a NetView task that is not the configured task. Module INGHWCOM terminates. For SA z/OS, make sure that you have defined autotask names with keyword HWOPERxx in your active automation policy. Use the SA z/OS Config Dialog to verify this.
0A2	ING_config_error	HW Task Error: The configuration information about the NetView autotask names to be used for the HW communication interface cannot be retrieved. This happens if the interface is called but SA z/OS initialization is not complete. This error also happens if the autotasks are not defined. See error code 0A0 for additional information.
090	ING_tcmnolist_error	No temporary capacity record list found.
091	ING_notobject_error	Object not found. In this case, the temporary capacity record object was not found.
092	ING_tcmrecfnd_error	No matching record found for the provided record ID.
093	ING_xml_parse_error	Error found while parsing temporary capacity record XML data.
094	ING_xml_compose_error	Error found while composing temporary capacity add or remove XML data.
095	ING_cpctcm_alloc_error	Error occurred while allocating storage for TCM records data.
096	ING_stpnodata_error	STP error. STP information could not be retrieved from the HW. Either the HW does not support STP or the XML data is invalid.
097	ING_pwrnostat_error	POWERMOD error. Current power mode value could not be retrieved.
098	ING_pwrnolist_error	POWERMOD error. Supported power modes list could not be retrieved.
099	ING_pwrnoallw_error	POWERMOD error. Power mode change allowed flag could not be retrieved.
100	ING_pwrcommnd_error	POWERMOD error. Invalid command used with POWERMOD.
101	ING_pwrattrib_error	POWERMOD error. Invalid UPDATE parameter.
0AA	ING_session_suspended	HW Task Error: Connections or sessions to the addressed CPC have been previously suspended. In this state any HW request or session initialization request is rejected. You must use the RESUME command to restore the normal operation mode for this CPC connection.

Condition Codes

Table 20. Hardware Communication Task Condition Codes (continued)

Reason Code	Error String	Error Description
0B0	ING_suspend_set_error	SUSPEND error: While executing a SUSPEND request an error occurred. (1) The suspend state is controlled with the common global variable xxx.0SUSPEND.cpcnetid.cpcnau (where xxx is either ING or ISQ) and there was an error reading or setting this variable. Look for additional messages in the netlog. It may be necessary to correct the suspend flag common global variable manually.
0B1	ING_resume_set_error	RESUME error: While executing a RESUME request an error occurred. (1) A suspend state is controlled with the common global variable xxx.0SUSPEND.cpcnetid.cpcnau (where xxx is either ING or ISQ) and there was an error resetting this variable. Look for additional messages in the netlog. It may be necessary to correct the suspend flag common global variable manually to resume the session.
0B2	ING_suspend_term_error	SUSPEND error: While executing a SUSPEND request an error occurred. The session termination failed. The suspend state is set. If a RESUME is executed, it does <i>not</i> attempt to reinitialize the connection.
0B3	ING_resume_init_error	RESUME error: While executing a RESUME request an error occurred. The session reinitialization failed. The suspend state is unset. You must perform session reinitialization using INITCOM.
0B4	ING_suspend_prohibited_for_HMC	SUSPEND error: A Suspend operation was attempted while an HMC connection was being used to communicate with the addressed CPC. Only SE connections are supported.

Data Exchange Services "0B100xxx"

Table 21 on page 439 lists the condition codes that are returned if there is an error with the following INGHWCMD functions:

- ACTIVATE
- APROF
- CCNTL
- CBU
- CTRLCONS
- DEACTIVATE
- EXTERNAL
- GETISTAT
- GETSINFO
- GETSSTAT
- ICNTL
- INITCOM
- LOAD
- OOCOD
- RESERVE
- RESTART
- START
- STOP
- SYSRESET
- TERMCOM

The condition code data "xxx" prefixed by 0B100 is returned as part of the following response messages, with a status value of REJECTED or FAILED:

- AOFA0001
- AOFA0002
- AOFA0004
- AOFA0006
- AOFA0007
- AOFA0017

- AOFA0018
- AOFA0020

For additional return code information see the section “Data Exchange API Call Return Codes”, in the appendix “API Return Codes” of the most current release of *System z Application Programming Interface*, SB10-7030.

Table 21. Data Exchange Services Condition Codes

Condition Code	Error String
001	<p>HWMCA_DE_NO_SUCH_OBJECT</p> <p>This condition indicates that an object (CPC, Image, Profile) or an associate attribute cannot be found for this request in the SNMP MIB data base of the SE or HMC. A possible cause can be that an object has been requested that is not available on this CPC's type or CPC's micro code level.</p> <p>Additional information for INITCOM requests:</p> <p>If this condition code is returned from an INITCOM request, a Set request for the console object itself did not work. A possible cause is an API configuration problem on the target SE or HMC.</p> <ol style="list-style-type: none"> 1. The community name settings in the SE or HMC may not allow access from your location due to an invalid netmask specification. 2. (BCPii only) Make sure that the community name that is specified in the Enable API window is also defined in the SNMP Settings window and has the IP address of the first LAN adapter of the SE defined with it. This information is valid only for SE Workplace versions 1.6.2-1.8.2.
002	HWMCA_DE_INVALID_DATA_TYPE
003	HWMCA_DE_INVALID_DATA_LENGTH
004	HWMCA_DE_INVALID_DATA_PTR
005	<p>HWMCA_DE_INVALID_DATA_VALUE</p> <p>If this condition code is returned from an ICNTL request, the variable value is not accepted by the HW. This means either the value is below or above a range of accepted values (such as a negative value, where only 0 or positive values are allowed), or a certain variable value conflicts with the setting of a related variable. If, for example, the value of the initial processing weight, PWI, is set below the value of the minimum processing weight, PWMN this request fails with a condition code of 005.</p>
006	HWMCA_DE_INVALID_INIT_PTR
007	HWMCA_DE_INVALID_ID_PTR
008	HWMCA_DE_INVALID_BUF_PTR
009	HWMCA_DE_INVALID_BUF_SIZE
010	HWMCA_DE_INVALID_DATATYPE_PTR
011	HWMCA_DE_INVALID_TARGET
012	HWMCA_DE_INVALID_EVENT_MASK
013	HWMCA_DE_INVALID_PARAMETER
014	HWMCA_DE_READ_ONLY_OBJECT
015	<p>HWMCA_DE_SNMP_INIT_ERROR</p> <p>This is a condition code that can be retried.</p>
016	HWMCA_DE_INVALID_OBJECT_ID
017	HWMCA_DE_REQUEST_ALLOC_ERROR
018	<p>HWMCA_DE_REQUEST_SEND_ERROR</p> <p>For SNMP sessions only: If possible, enable the INGHWCOM trace and repeat the request. Additional TCP/IP error information will be in an error message (EDC) from the LE runtime environment. Have the <i>z/OS LE Run-Time Messages</i> manual available for reference.</p>

Condition Codes

Table 21. Data Exchange Services Condition Codes (continued)

Condition Code	Error String
019	HWMCA_DE_TIMEOUT For SNMP sessions only: If possible, enable the INGHWCOM trace and repeat the request. Additional TCP/IP error information will be in an error message (EDC) from the LE runtime environment. Have the <i>z/OS LE Run-Time Messages</i> manual available for reference. Verify if the network address and the network mask specified in the SE/HMC for the community name are valid. Address field and mask field are logically ANDed. If the result is 0, the request may fail with a TIMEOUT. Note that address and mask fields positions that are both 0 are not ANDed.
020	HWMCA_DE_REQUEST_RECV_ERROR
021	HWMCA_DE_SNMP_ERROR Check that the SNMP API is enabled on the SE or HMC.
022	HWMCA_DE_INVALID_TIMEOUT
028	HWMCA_DE_INVALID_HOST
029	HWMCA_DE_INVALID_COMMUNITY
030	HWMCA_DE_INVALID_QUALIFIER
098	HWMCA_DE_REQUIRES_QUALIFIER
099	HWMCA_DE_TRANSPORT_ERROR Check that the BCP Internal Interface modules are correctly installed. Module or module version error detected in LPA lib or LNKLIST.

Command Services "0B200xxx"

Table 22 lists the condition codes that are returned if there is an error with the following hardware functions:

- ACTIVATE
- CBU
- DEACTIVATE
- EXTERNAL
- LOAD
- OOCOD
- RESTART
- START
- STOP
- SYSRESET
- TCM
- POWERMOD

The condition code data "xxx" prefixed by 0B200 is returned as part of the AOFA0001 response message with a status value of REJECTED or FAILED.

For additional return code information refer to the section "Data Exchange API Call Return Codes", in the appendix "API Return Codes" of the most current release of *System z Application Programming Interface, SB10-7030*.

Table 22. Command Services Condition Codes

Condition Code	Error String
000	HWMCA_CMD_STARTED_BUT_COMPLETION_MAY_HAVE_FAILED An HW command was successfully started. For SYNC sessions check the AOFA0001 command completion report for additional sense information about the actual command completion. With ASYNC sessions, the command completion event report contains this information, and applications can register for it with the FILTER command.

Table 22. Command Services Condition Codes (continued)

Condition Code	Error String
001	HWMCA_CMD_NO_SUCH_OBJECT An HW command has been requested that is not available on this HW type or microcode level.
002	HWMCA_CMD_INVALID_DATA_TYPE
003	HWMCA_CMD_INVALID_DATA_LENGTH
004	HWMCA_CMD_INVALID_DATA_PTR
005	HWMCA_CMD_INVALID_DATA_VALUE
006	HWMCA_CMD_INVALID_INIT_PTR
007	HWMCA_CMD_INVALID_ID_PTR
010	HWMCA_CMD_INVALID_DATATYPE_PTR
011	HWMCA_CMD_SNMP_ERROR_INCONSISTENT_NAME A command request was received while a prior request was just finished. The prior task is still not complete, but it has released the RESERVE already. This is a temporary condition and the request can be repeated.
013	HWMCA_CMD_INVALID_PARAMETER
017	HWMCA_CMD_REQUEST_ALLOC_ERROR
018	HWMCA_CMD_REQUEST_SEND_ERROR For SNMP sessions only: If possible, enable the INGHWCOM trace and repeat the request. Additional TCP/IP error information will be in an error message (EDC) from the LE runtime environment. Have the <i>z/OS LE Run-Time Messages</i> manual available for reference.
019	HWMCA_CMD_TIMEOUT For SNMP sessions only: If possible, enable the INGHWCOM trace and repeat the request. Additional TCP/IP error information will be in an error message (EDC) from the LE runtime environment. Have the <i>z/OS LE Run-Time Messages</i> manual available for reference.
020	HWMCA_CMD_REQUEST_RECV_ERROR
021	HWMCA_CMD_SNMP_ERROR
022	HWMCA_CMD_INVALID_TIMEOUT
023	HWMCA_CMD_INVALID_CMD
024	HWMCA_CMD_OBJECT_BUSY
025	HWMCA_CMD_INVALID_OBJECT
026	HWMCA_CMD_COMMAND_FAILED
027	HWMCA_CMD_INITTERM_OK
028	HWMCA_CMD_CBU_DISRUPTIVE_OK
029	HWMCA_CMD_CBU_PARTIAL_HW
030	HWMCA_CMD_CBU_NO_SPARES
031	HWMCA_CMD_CBU_TEMPORARY
032	HWMCA_CMD_CBU_NOT_ENABLED
033	HWMCA_CMD_CBU_NOT_AUTHORIZED
034	HWMCA_CMD_CBU_FAILED
035	HWMCA_CMD_CBU_ALREADY_ACTIVE
036	HWMCA_CMD_CBU_INPROGRESS
038	HWMCA_CMD_INVALID_MACHINE_STATE The command request cannot be performed at this time because the target object is currently not in an appropriate state (that is, it is not powered on).
039	HWMCA_CMD_NO_RECORDID The command request was unsuccessful because the specified capacity record with the specified identifier does not exist.

Condition Codes

Table 22. Command Services Condition Codes (continued)

Condition Code	Error String
040	HWMCA_CMD_NO_SW_MODEL The command request was unsuccessful because the specified software model is invalid for the target object.
041	HWMCA_CMD_NOT_ENOUGH_RESOURCES The command request was unsuccessful because the request specifies more resources than are currently available on the target object.
042	HWMCA_CMD_NOT_ENOUGH_ACTIVE_RESOURCES The command request was unsuccessful because the request specifies more resources than are currently active on the target object.
043	HWMCA_CMD_ACT_LESS_RESOURCES The command request for additional resources was unsuccessful because the request specifies a net decrease in the resources for the target object.
044	HWMCA_CMD_DEACT_MORE_RESOURCES The command request for the removal of resources was unsuccessful because the request specifies a net increase in the resources for the target object.
045	HWMCA_CMD_ACT_TYPE_MISMATCH The command request was unsuccessful because the type value specified (real or test) was not valid for the type of capacity record being used.
046	HWMCA_CMD_API_NOT_ALLOWED The command request was unsuccessful because the target Defined CPC was configured not to allow capacity changes via the Console Application Programming Interfaces.
047	HWMCA_CMD_CDU_IN_PROGRESS The command request cannot be performed at this time because a concurrent driver upgrade operation is being performed.
048	HWMCA_CMD_MIRRORING_RUNNING The command request cannot be performed at this time because a support element mirror operation is being performed.
049	HWMCA_CMD_COMMUNICATIONS_NOT_ACTIVE The command request cannot be performed at this time because communications with the Defined CPC object are not active.
050	HWMCA_CMD_RECORD_EXPIRED The command request was unsuccessful because the capacity record being used for the operation has expired.
051	HWMCA_CMD_PARTIAL_CAPACITY The command request was successful, but not all of the resources that were requested could be made available.
052	HWMCA_CMD_ALREADY_ACTIVE
071	HWMCA_CMD_PROCESSOR_POWER_MODE_NOT_ENTITLED
072	HWMCA_CMD_PROCESSOR_POWER_MODE_NOT_ALLOWED
073	HWMCA_CMD_PROCESSOR_POWER_MODE_GROUP_CONTROLLED
099	HWMCA_TRANSPORT_ERROR Before the HwmcaCmd was invoked, an internally issued data entry Hwmca call terminated with CC 99. Same as HWMCA_DE_TRANSPORT_ERROR condition code 0B100099.

Internal Transport Services "0Bx00xxx"

Note that this set of condition codes applies to BCP Internal Interface connections *only*.

Table 23 on page 443 lists the condition codes that are returned if there is an error with the following INGHWCMD functions:

- ACTIVATE
- CBU

- CTRLCONS
- DEACTIVATE
- EXTERNAL
- GETISTAT
- GETSSTAT
- INITCOM
- LOAD
- OOCOD
- RESTART
- START
- STOP
- SYSRESET
- TERMCOM

The condition code data "xxx" prefixed by 0B100 or 0B200 is returned as part of the following response messages, with a status value of REJECTED or FAILED:

- AOFA0001
- AOFA0002
- AOFA0004
- AOFA0017
- AOFA0018

Table 23. Internal Transport Services Condition Codes

Reason Code	Error Description
100	A problem was encountered prior to sending the request to the HSAET32 API for processing. This is likely due to a failure to an environmental error. Check whether the Support Element is fully operational. A running reboot of the SE may have caused this problem.
101	A request incomplete condition occurred.
102	A report list overflow occurred. This return code should not currently be issued for SNMP requests, however is included for OCF query (Query-Read-Cluster) compatibility.
110	The issuer of the request is not (RACF) authorized to the requested function. Note that (like HCD) the HSAET32 services require that RACF or a compatible SAF product be installed and operational. An RACF system message ICH408I is issued with additional information. If no ICH408I message is issued, make sure the class FACILITY is RACLISTed and repeat the request.
111	The control block ID or version of the HSDB passed to HSAET32 services is invalid. For hwmcaapi requests, this indicates that HSAPHCPI is incompatible with the supporting HSAPHARI module.
112	The requested function is invalid or not supported by the current level of HSAET32 services. For hwmcaapi requests, this indicates an incompatibility between HSAPHCPI and the supporting HSAPHARI module.
113	The control block ID of the request list passed to HSAET32 services is invalid or inappropriate for the requested function. For hwmcaapi requests, this indicates a problem in module HSAPHCPI.
114	The request list entry count passed to HSAET32 services is invalid or inappropriate for the requested function. For hwmcaapi requests, this indicates a problem in module HSAPHCPI.
115	The request list entry pointer passed to HSAET32 services is null and therefore invalid. For hwmcaapi requests, this indicates a problem in module HSAPHCPI.
116	Some of the input areas passed to HSAET32 services exist in a storage area that the caller does not have authority to fetch or update.
117	The input parameter list generated by the HSAXHARI (or CBDIHSD) macro does not have the correct version ID or type, or does not point to an HSDB.

Condition Codes

Table 23. Internal Transport Services Condition Codes (continued)

Reason Code	Error Description
118	The control block ID of the Output Report request list passed to HSAET32 services is invalid or inappropriate for the requested function. For hwmcaapi requests, this indicates a problem in module HSAPHCPI.
119	The session token is invalid. This is probably due to a previous failure of the hwmcainitialize request, an hwmcatерminate request being issued for the session, or improper modification of the HWMCA_SCLP_TARGET_INFO structure.
120	The host environment does not support HSAET32 services. HSAET32 services are not currently on VM hosts.
121	An address space resource manager could not be established.
122	A task resource manager could not be established.
123	The HSAET32 associated recovery routine (HSAPHARR) was entered due to an unexpected error processing the request.
124	The CBDMHWA CSECT could not be found in the nucleus.
125	The HSAET32 monitor exit (HSAPHMON) could not be established as the secondary ET32 listner exit for the application.
126	The system date and time could not be obtained to correlate HRE and associated MDS-MU's.
127	A failure occurred attempting to access the HWAX.
129	An attempt to send the MDS_MU requests across the BCP Internal Interface interface failed.
130	Either the EP_OPERATIONS_MGMT vector (9F22) from the event type 30 data was not available or its length was invalid.
131	Either the application name-group for the EP_OPERATIONS_MGMT application (event type 30 data) was not returned in the 9F22 vector or its length was invalid.
132	Either the NetID of the local support element was not returned in the application name-group for the EP_OPERATIONS_MGMT application (event type 30 data) or its length was invalid.
133	Either the NAU of the local support element was not returned in the application name-group for the EP_OPERATIONS_MGMT application (event type 30 data) or its length was invalid.
134	Either the CPC image name vector (9F70) from the event type 30 data was not available or its length was invalid.
135	Error in ET30 ESTAE routine. Module HSAPHSDI was unable to establish an ESTAEX recovery environment.
136	Either the primary OCF name vector (9F81) from the event type 30 data was not available or its length was invalid.
140	BCP Internal Interface Access Error. HSAET32 services have been disabled or were not been started.
148	BCP Internal Interface Session Error. The hardware session with the target CPC has terminated. This condition is raised due to a missing heartbeat from the ET32 agent code running on the target SE. To recover from this situation the application should perform a TERMCOM followed by an INITCOM in order to reestablish session communication. The occurrence of this RC indicates that possibly events have been lost.
150	Error in ET32 ESTAE routine.
151	HSAPHSPI identified a parameter that is not contextually valid.
152	HSAPHSPI identified a missing parameter that is contextually required.
153	HSAPHSPI identified a parameter value that is syntactically incorrect.
160	IEAMSCHD_Error. An error has occurred attempting to schedule HSAPHDSC for execution. An IEAMSCHD return code is also provided to further explain the cause of the error.

Table 23. Internal Transport Services Condition Codes (continued)

Reason Code	Error Description
161	CSS_Error. An error condition was raised using callable supervisor services facilities. The IEAVxxxx return code is also provided to further explain the cause of the error.
162	CPSS_Error. An error condition was raised using cellpool services facilities. The CSRPExxx return code is also provided to further explain the cause of the error.
163	HSAPHDSC_Error. An error condition was raised using callable supervisor services facilities. The IEAVxxxx return code is also provided to help identify the cause of the error.
164	CTRACE_Error. An error condition was encountered by module HSAPHDSC, and is further identified by the return code also provided.
165	An error condition was encountered while attempting define (or delete) the application to Component Trace. The CTRACE return code is also returned to help identify the cause of the error.
166	HSAPHDSC Error. The Access List Service macro ALESERV returned an error condition. Report the available sense data of the AOFA0001 response when contacting your IBM Support Center.
167	HSAPHDSC Error. The TCB Token Service macro TCBTOKEN returned an error condition. Report the available sense data of the AOFA0001 response when contacting your IBM Support Center.
168	HSAPHDSC/HSAPHMNX Error. The Data Space Service macro DSPSERV returned an error condition. Report the available sense data of the AOFA0001 response when contacting your IBM Support Center.
169	HSAPHARI Error. The Component Trace Service returned an error condition. Report the available sense data of the AOFA0001 response when contacting your IBM Support Center.
170	HSAPHARI/HSAPHMNX Error. The Callable CellPool Service CSRPEXP returned an error condition. Report the available sense data of the AOFA0001 response when contacting your IBM Support Center.
171	HSAPHARI Error. The Name Token Service IEANTCR returned an error condition. Report the available sense data of the AOFA0001 response when contacting your IBM Support Center.
204	The request was accepted by the local support element and will be processed asynchronously. No further reason code is provided. (This function is not currently used by the hwmcaapi implementation).
208	Execution of request was failed by the target support element. This indication is normally accompanied by a condition report that is returned as the error reason, and may also be accompanied by sense data further identifying the cause of the failure.
212	<p>The request was rejected by the local support element. This indication is normally accompanied by a condition report that is returned as the error reason.</p> <p>This condition, together with sense code 081C00BA, may occur if the addressed CPC is not identical with your own CPC and is correctly defined to the SA z/OS BCPII, but is not defined as a CPC to the master HMC. The master HMC is used by the HW to route a request to a remote CPC.</p> <p>In the processor LAN environment, an HMC acts as a master HMC if it has the LIC Change services flag enabled. If you have several master HMCs in your environment, each master HMC should have the same set of CPCs defined for it that you want to access over the BCPII. If you are connected to your own CPC, you can use the INGHWCMD GETCLUSTER command to find out which other CPCs are defined to the master HMC that the BCPII is currently using.</p>
216	An MDS-MU error message was received from the target support element. The condition report code is returned as the error reason.

Condition Codes

Table 23. Internal Transport Services Condition Codes (continued)

Reason Code	Error Description
220	HSAPHMNX detected a structural error while processing the incoming report from the target Support Element. If the SENSE field contains data, check whether the first byte has a value of X'27'. In this case, a request length difference was detected between the data coming from the OS and what was returned from the target Support Element. Note that in this case a LOGREC software symptom record is written containing additional data. (HSAET32)
224	No response was received within the time interval designated for the request. No further reason code provided.
228	An error was detected in a request list entry. An internal reason code is generated to identify the field in error. (This code is not used for hwmcaapi requests).
232	A routing error has occurred while forwarding the requests for processing. This indication is normally accompanied by a condition report that is returned as the error reason.

Ensemble HMC API Exchange Services Condition Codes

This section gives further information about the condition codes for the HMC API exchange services. The API exchange services are used by the ProcOps ensemble commands to communicate with the HMC Web Services API over HTTP protocol. For more information about the API, refer to the *System z Hardware Management Console Web Services API Version 2.11.1 (SC27-2616-00)*. The condition codes are usually accomplished with a SENSE code, which consists of three parts:

- Error number (ERRNO) for socket related problems.
- HTTP status code returned by the HMC API
- HTTP reason code returned by the HMC API

Most common HTTP status codes are described in Table 25 on page 448.

Table 24. Ensemble HMC API exchange Services Condition Codes

Reason Code	REST API	Error Code/Description
00B30000	BXMCA_NO_ERR	A HMC API exchange service call has completed successfully.
00B30002	BXMCA_INVALID_PARM	A HMC API exchange service call specified an invalid parameter.
00B30003	BXMCA_ERR_NOT_SUBSCRIBED	A HMC API exchange service call 'WaitEvent' was executed without correct subscription to a JMS Topic.
00B30004	BXMCA_INVALID_DATA_PTR	A HMC API exchange service command call specified a data pointer that is not appropriate for the corresponding data type.
00B30005	BXMCA_ERR_ALREADY_INITIALIZED	A HMC API exchange service initialization was requested for an already initialized session.
00B30006	BXMCA_INVALID_INIT_PTR	A HMC API exchange service call specified null as the pointer to the initialize structure.
00B30007	BXMCA_NOT_INITIALIZED	A HMC API exchange service call was made to a not-initialized target.
00B30008	BXMCA_INVALID_BUF_PTR	A HMC API exchange service call specified null as the pointer to the output buffer.
00B30009	BXMCA_WRONG_SESSION	A HMC API exchange service call 'WaitEvent' was executed in a command session.
00B30011	BXMCA_INVALID_TARGET	A HMC API exchange service initialization call specified null as the pointer to the target data structure.

Table 24. Ensemble HMC API exchange Services Condition Codes (continued)

Reason Code	REST API	Error Code/Description
00B30012	BXMCA_INVALID_HMC_ADDR	A HMC API exchange service initialization call specified an invalid host name or internet address for the target console.
00B30013	BXMCA_INVALID_HMC_CREDENTIAL	A HMC API exchange service initialization API call specified an invalid userid and/or password for the target console.
00B30017	BXMCA_REQUEST_ALLOC_ERROR	A HMC API exchange service command call encountered an error trying to allocate some temporary storage for internal use.
00B30018	BXMCA_REQUEST_SEND_ERROR	A HMC API exchange service command call encountered an error trying to send a request to the target console. This is typically due to a network error or HMC API problem of some sort.
00B30019	BXMCA_REQUEST_SEND_TIMEOUT	A HMC API exchange service command timed out while attempting to send a request to the target console. This is typically due to a SSL setup problem, network error or HMC API problem of some sort.
00B30020	BXMCA_REQUEST_RECV_ERROR	A HMC API exchange service command encountered an error trying to receive a response from the target console. This is typically due to a network error or HMC API problem of some sort.
00B30021	BXMCA_REQUEST_RECV_TIMEOUT	A HMC API exchange service command timed out while waiting for a response. The response was not received within the specified time period.
00B30022	BXMCA_REQUEST_HTTP_CORRUPTED	A HMC API exchange service command encountered an error trying to parse received HTTP response. The data contains no valid HTTP header.
00B30023	BXMCA_REQUEST_HTTP_ERROR	The HMC API command request failed. The HTTP status code is greater or equal to 400.
00B30024	BXMCA_PARSER_ALLOC_ERROR	JSON Parser encountered an error trying to allocate some temporary storage for internal use.
00B30025	BXMCA_BUFFER_TOO_SMALL	The buffer provided for a URI is too small.
00B30030	BXMCA_STOMP_ALLOC_ERROR	A HMC API exchange service call encountered an error trying to allocate some temporary storage for internal use for event notifications.
00B30031	BXMCA_STOMP_SEND_ERROR	A HMC API exchange service call encountered an error trying to send a request to the target console JMS message broker. This is typically due to a network error or HMC API problem of some sort.
00B30032	BXMCA_STOMP_SEND_TIMEOUT	A HMC API exchange service timed out while attempting to send a request to the target console JMS message broker. This is typically due to a SSL setup problem, network error or HMC API problem of some sort.
00B30033	BXMCA_STOMP_RECV_ERROR	A HMC API exchange service encountered an error trying to receive a response from the target console JMS message broker. This is typically due to a network error or HMC API problem of some sort.
00B30034	BXMCA_STOMP_RECV_TIMEOUT	A HMC API exchange service timed out while waiting for an event. The response was not received within the specified time period. This simply means that no events were received within the specified time period, so the calling application should proceed accordingly.
00B30035	BXMCA_STOMP_FRAME_CORRUPTED	A HMC API exchange service encountered an error trying to parse received STOMP notification.

Condition Codes

Table 24. Ensemble HMC API exchange Services Condition Codes (continued)

Reason Code	REST API	Error Code/Description
00B30036	BXMCA_STOMP_FRAME_ERROR	A HMC API exchange service encountered an error trying to parse received STOMP frame. The Frame ID cannot be identified.
00B30037	BXMCA_STOMP_FRAME_UNKNOWN	A HMC API exchange service received a STOMP Frame with unsupported Frame ID.
00B30046	BXMCA_ERR_STOMP_CONNECT	A HMC API exchange service encountered an error trying to connect to the target console JMS message broker.
00B30047	BXMCA_ERR_XAPISESSION	A HMC API logon returned no valid session id.
00B30048	BXMCA_ERR_XAPITOPIC	A HMC API logon command returned no JMS topic for notifications.
00B30049	BXMCA_ERR_XAPITOPIC_JOBS	A HMC API logon command returned no JMS topic for asynchronous job completion notifications.
00B30050	BXMCA_ERR_UNSUPP_HMC_VER	The HMC Web Services API available on the target HMC is not compatible with the current SA z/OS release.
00B30051	BXMCA_ERR_APIVER	Unable to identify current version of the HMC Web Services API.
00B30052	BXMCA_ERR_LOGON	The logon to the HMC failed. Possible HTTP sense codes: 0000 403 0000: User authentication failed on the HMC. 0000 403 0040: The user is disabled on the HMC. 0000 403 0041: The user is not authorized to use the HMC Web Services interface.
00B30053	BXMCA_JSON_PARSE_ERR	A HMC API exchange service command encountered an error trying to parse received response data.
00B30054	BXMCA_ERR_LOGON_EXPIRED	The logon to the HMC failed. The user's password has expired.
00B30097	BXMCA_INVALID_STACKNAME	A HMC API exchange service initialization call specified an invalid TCPIP stack for the ensemble HMC API communication.
00B30099	BXMCA_INVALID_CHARSET	A HMC API exchange service specified an invalid codeset for an operation.

Table 25. Common HTTP status codes reported by the HTTP API.

HTTP status code	Description
200 (OK)	The request has succeeded completely.
201 (Created)	The request has succeeded completely and resulted in the creation of a new managed resource/object.
202 (Accepted)	The request was successfully validated and has been accepted to be carried out asynchronously.
204 (No Content)	The request succeeded completely, and no additional response information is provided.
400 (Bad Request)	The request was missing required input, had errors in the provided input, or included extraneous input. Additional information regarding the error is provided in an error response body that includes a reason code with additional information.

Table 25. Common HTTP status codes reported by the HTTP API. (continued)

HTTP status code	Description
403 (Forbidden)	Multiple error conditions result in this status code: <ul style="list-style-type: none"> The request requires authentication but no X-API-Session header was provided, or one was provided but the session ID was invalid. The user under which the API request was authenticated is not authorized to perform the requested operation.
404 (Not Found)	Multiple error conditions result in this status code: <ul style="list-style-type: none"> The URI does not designate an extant resource, or designates a resource for which the API user does not have object-access permission. The URI designates a resource or operation that is not supported by the HMC because it is currently the alternate HMC.
405 (Method Not Allowed)	The request specifies a method that is not valid for the designated object.
409 (Conflict)	The managed resource is in an incorrect state (status) for performing the requested operation. Additional information regarding the error is provided in an error response body that includes a reason code with additional information.
500 (Server Error)	A server error occurred during processing of the request.
503 (Service Unavailable)	The request could not be carried out by the HMC due to some temporary condition.

Ensemble Communication Task Condition Codes "00B600xxx"

Table 26. Ensemble Communication Task Condition Codes

Reason Code	REST API Error Code	Error Description
00B60001	ING_invalid_HLL_buffer	Ensemble communication service was invoked, but the NetView HLL buffer found for C/C++ is not valid.
00B60002	ING_origuser_invalid	The user ID and output correlator passed to the ensemble communication service are not valid.
00B60005	ING_tgt_length_error	Parsing Error: The target ensemble name has an invalid length. It must be 1 to 8 characters.
00B60006	ING_tgt_missing	Parsing Error: The target ensemble name is not specified.
00B60007	ING_cpc_length_error	Parsing Error: The CPC address specification netid.nau has an invalid length. It must not exceed 17 characters.
00B60008	ING_cpc_missing	Parsing Error: The CPC address specification, which is a required parameter for the request, is missing.
00B6000B	ING_force_invalid	Parsing Error: The FORCE option is specified in the request but is not supported for the ensemble function. The following ensemble functions allow the FORCE option: DEACTIVATE BL, DEACTIVATE VS.
00B6000C	ING_force_missing	Parsing Error: The FORCE option is required for the request, but has not been specified.
00B6000E	ING_timeout_missing	Parsing Error: The required TIMEOUT parameter is missing in the request.

Condition Codes

Table 26. Ensemble Communication Task Condition Codes (continued)

Reason Code	REST API Error Code	Error Description
00B6000F	ING_OCFCMD_truncated	Parsing Error: The ensemble HW function (OCFCMD) exceeds the maximum allowed length of 256 characters.
00B60010	ING_OCFCMD_missing	Parsing Error: No ensemble HW function (OCFCMD) was specified in the request.
00B6001A	ING_invalid_target	The target object class specified in ACTIVATE or DEACTIVATE request is not supported. Supported are: BL for blades, VS for virtual servers.
00B6001B	ING_ambiguous_obj_error	More than one resource in the ProcOps internal data model corresponds to the specified command parameters. The command requires a single object; only full specification of all command parameters insures selection of the single object from the data model for execution.
00B6001C	ING_no_obj_error	No resources in the ProcOps internal data model correspond to the specified command parameters. You may use LIST ensemble command to see all objects and their specification discovered for the target ensemble.
00B6001D	ING_no_obj_properties	The HMC Web Services API returned no properties for the query.
00B6001E	ING_session_mismatch	Parameter Resolution Error: A session was requested having a different target ensemble name than the active session. Terminate the active session prior to requesting a new one.
00B6001F	ING_empty_inventory	The HMC Web Services API has no inventory data.
00B60020	ING_REST_noIP_address	Parameter Resolution Error: The HTTP interface was specified for the ensemble HW request, but no IP address information is available.
00B60021	ING_OCF_resolve_failed	Parameter Resolution Error: No ensemble HW function name to resolve, same as error 010.
00B60022	ING_OCF_not_resolved	Parameter Resolution Error: An invalid ensemble HW function name was detected.
00B60023	ING_var_no_IP	Configuration Error: Unable to retrieve the IP address or host name of the primary HMC. ISQVARS keyword "enspaddr" cannot be retrieved.
00B60024	ING_var_no_USER	Configuration Error: Unable to retrieve userid. ISQVARS keyword "ensuser" cannot be retrieved.
00B60025	ING_var_no_PASSWORD	Configuration Error: Unable to retrieve the password. ISQVARS keyword "enspw" cannot be retrieved.
00B60026	ING_var_no_IPSTACK	Configuration Error: Unable to retrieve the TCP/IP stack. ISQVARS keyword "ipstack" cannot be retrieved.

Table 26. Ensemble Communication Task Condition Codes (continued)

Reason Code	REST API Error Code	Error Description
00B60027	ING_var_no_SAFPW	Configuration Error: Unable to retrieve the password. NetView command "GETPW <userid> <ensemble entry name>,READ" failed. <userid> is the name of the HMC user associated with the session. <ensemble entry name> is the name of the entry as used by SA z/OS dialogs for the zEnterprise ensemble. Note: The predefined value SAFPW is used to allow maintaining the password for the HMC user <userid>. In this case the ensemble's entry name <ensemble> is used to setup the SAF resource identifier in the VSAM file representing the NetView password dataset. Be aware of this when changing, adding or deleting ensemble entries. For more information, see the appendix "Security and Authorization" in <i>IBM Tivoli System Automation for z/OS Planning and Installation</i> .
00B60028	ING_init_GETPW_failed	Configuration Error: Unable to generate new password. NetView command "GETPW <userid> <ensemble entry name>,REGEN" failed. <userid> is the name of the HMC user associated with the HMC session. <ensemble entry name> is the name of the entry as used by SA z/OS dialogs for the zEnterprise ensemble.
00B60031	ING_alloc_error	Storage Allocation Error: The storage for the ProcOps internal data model could not be allocated using DSIGET services.
00B60032	ING_locate_error	Configuration Error: Unable to retrieve session anchor. ISQVARS keyword "ensanch" cannot be retrieved.
00B60033	ING_notinit_error	Storage Allocation Error: An ensemble HW function request was issued for a target without having allocated storage for that ensemble. This happens if no initialization request (ISQXIII) for the ensemble was made prior to the first function request.
00B60040	ING_nocpc_error	Configuration Error: No ensemble nodes (CPCs) were discovered for the target ensemble. Please verify your ensemble configuration and the CPCs selected in the customization dialogs of SA z/OS for the target ensemble. Please check the "Managed Resource Roles" of the userid specified for the ensemble on the ensemble HMC.
00B60041	ING_noens_error	Configuration Error: No ensembles were discovered. Please verify your ensemble configuration and the HMC connection parameters specified in the customization dialogs of SA z/OS for the target ensemble.
00B60042	ING_nozbx_error	Configuration Error: No zBX Blade Extensions were discovered for the target ensemble. At least one zBX is required in order to use ensemble HW functions. Please check the "Managed Resource Roles" of the userid specified for the ensemble on the ensemble HMC.
00B60043	ING_cpc_mismatch	Configuration Error: The CPCs discovered on the target ensemble do not match to the CPCs selected for the ensemble in the customization dialogs of SA z/OS.

Condition Codes

Table 26. Ensemble Communication Task Condition Codes (continued)

Reason Code	REST API Error Code	Error Description
00B60044	ING_ens_mismatch	Configuration Error: The ensemble name does not match to the name specified for the ensemble in the customization dialogs of SA z/OS.
00B60045	ING_lock_error	ENQ Error: The ENQ/DEQ request failed for the ProcOps data model. Major: INGBXENS minor: <target>.
00B60050	ING_notinitized_error	Storage Allocation Error: An ensemble HW function request was issued for a target without having allocated storage for that ensemble. This happens if no initialization request (ISQXIII) for the ensemble was made prior to the first function request. Same as 033.
00B60052	ING_funcunknown_error	Parameter Resolution Error: An invalid ensemble HW function name was detected. Same as 022.
00B60053	ING_noensobject_error	Configuration Error: No ensembles were discovered. Please verify your ensemble configuration and the HMC connection parameters specified in the customization dialogs of SA z/OS for the target ensemble. Same as 041.
00B60055	ING_nohwstatus_error	Ensemble HW Function Error: An function that requires the determination of the status of the object prior to execution cannot be processed because the object status cannot be determined
00B60056	ING_disruptive_cmd	Ensemble HW Function Error: A disruptive ensemble HW function was requested without the FORCE option and the BI/VS object is in an operational state. ISQECMD FORCE(NO) (allow no disruptive commands) as default. If you want to allow disruptive commands you must specify the FORCE option in the ISQECMD request.
00B60060	ING_filter_error	Ensemble HW Function Error: A filter SET/UNSET command failed. The filter table is full and no new filters can be set, a maximum of 10 filter entries can be set per ensemble.
00B600A0	ING_invalid_task	Ensemble HW Task Error: The ensemble communication interface is running on a wrong NetView task. ISQEMxxx is required.
00B600A1	ING_not_authorized	Authorization error: The ensemble HW function is called by an unauthorized user. The operator executing the command has no required authorization for the facility "ISQ.ENS.<target>": Access level READ is required for all query commands. Access level CONTROL is required for initialization, termination, discovery and drop commands. Access level ALTER is required for ACTIVATE and DEACTIVATE commands.
00B600A2	ING_config_error	Ensemble HW Task Error: The configuration information about the NetView autotask names to be used for the ensemble communication interface cannot be retrieved. This happens if the interface is called but SA z/OS initialization is not complete.
00B600AA	ING_session_suspended	Ensemble HW Task Error: Connections or sessions to the addressed ensemble HMC have been previously suspended. In this state any ensemble HW request or session initialization request is rejected. You must use the RESUME command to restore the normal operation mode for this ensemble connection.

Table 26. Ensemble Communication Task Condition Codes (continued)

Reason Code	REST API Error Code	Error Description
00B600B0	ING_suspend_set_error	SUSPEND error: While executing a SUSPEND request an error occurred. (1) The suspend state is controlled with the common global variable ISQBX.0SUSPEND.ensemble and there was an error reading or setting this variable. Look for additional messages in the netlog. It may be necessary to correct the suspend flag common global variable manually.
00B600B1	ING_resume_set_error	RESUME error: While executing a RESUME request an error occurred. (1) The suspend state is controlled with the common global variable ISQBX.0SUSPEND.ensemble and there was an error reading or setting this variable. Look for additional messages in the netlog. It may be necessary to correct the suspend flag common global variable manually to resume the session.
00B600B2	ING_suspend_term_error	SUSPEND error: While executing a SUSPEND request an error occurred. The session termination failed. The suspend state is set. If a RESUME is executed, it does not attempt to reinitialize the connection.
00B600B3	ING_resume_init_error	RESUME error: While executing a RESUME request an error occurred. The session re-initialization failed. The suspend state is unset. You must perform session re-initialization using ISQXCLS/ISQXIII.

Condition Codes

Appendix D. Sense Codes, Hardware Object Status Summary

Sense Codes

Note that for BCP Internal Interface connections the sense codes are copied from the request response report information into the AOFAXxxx messages. For detailed sense data information refer to the appendix “HWMCA_EVENT_COMMAND_RESPONSE Return Codes” in the most current release of *System z Application Programming Interface*, SB10-7030.

To display online help for a sense code, use the NetView SENSE command followed by the 8-character sense code parameter from the AOFAXxxx report string that is returned.

0806000A RESOURCE UNKNOWN

Explanation: The profile name (CNAME) specified in a operations command is not recognized by the receiving node.

System programmer response: Correct the configuration identifier and resend the request.

08090000 Mode inconsistency: The requested function cannot be performed in the present state of the receiver.

Explanation: This command is prohibited because the target is in an incompatible mode. For example, an ITIMER request is not accepted when the system is power-on reset in LPAR mode.

System programmer response: This function cannot be performed in the present state of the receiver. Retry the request after the target mode status has changed.

08090001 Mode inconsistency: The requested function cannot be performed in the present state of the receiver.

Explanation: Acceptance of the command is prohibited because the target is in an incompatible mode. For example, an ITIMER request is not accepted when the system is power-on reset in LPAR mode.

System programmer response: None. This function cannot be performed in the present state of the receiver.

08090027 Mode inconsistency: The requested function cannot be performed in the present state of the receiver.

Explanation: The receiving Hardware Management Console is not in the correct state to automatically dial out using the attached modem.

System programmer response: Ensure the receiving Hardware Management Console is customized to use

the autodial and RSF functions.

08090051 Mode inconsistency: The requested function cannot be performed in the present state of the receiver.

Explanation: Operations management control is not enabled.

System programmer response: Enable the system for automated operations and resend the request. Ensure that the Emergency Power Off switch is on.

080A000A Permission rejected: The receiver has denied an implicit or explicit request of the sender.

Explanation: A STATLEV request was rejected because it was not compatible with the status reporting values set in the receiver.

System programmer response: Correct the STATLEV value and resend the request.

080A000C Permission rejected: The receiver has denied an implicit or explicit request of the sender.

Explanation: A SETCLOCK request has failed because it required that a clock be set in a configuration where a dominant timing source has priority.

System programmer response: If the Sysplex Timer is the dominant timing source the SOURCE, TIME, UTCO, and OFFSET operands cannot be used in the command string. Remove these operands and resend the request.

080C0005 Procedure not supported: A procedure specified is not supported in the receiver.

Explanation: The command is not supported.

Sense Codes, Hardware Object Status Summary

System programmer response: Resend the request using a supported command, if possible.

080C0007 Procedure not supported: A procedure specified is not supported in the receiver.

Explanation: A request for a function is supported by the receiver, but the resource identified in the request does not support that function. This function cannot be canceled.

System programmer response: None.

080F0001 End-user not authorized: The requesting end-user does not have access to the requested resource.

Explanation: Authorization checks have not been successfully passed.

System programmer response: Correct the command authorization-token and resend the request.

08120000 Insufficient resource: The receiver cannot act on the request because of a temporary lack of resource.

Explanation: System resources are temporarily busy.

System programmer response: Resend command if required.

08120011 Insufficient resource: The receiver cannot act on the request because of a temporary lack of resource.

Explanation: Insufficient storage is available to the target component to satisfy the request.

System programmer response: Resend command.

08120012 Insufficient resource: The receiver cannot act on the request because of a temporary lack of resource.

Explanation: A timed command was rejected because the OCF timed operations queue was full.

System programmer response: Cancel any unnecessary scheduled requests and resend the command.

08150001 Function active: A request to activate an element or procedure was received, but the element or procedure was already active.

Explanation: Unable to perform the command because the target CPC Subset or CPC Image is operational and the force operand has not indicated the override selection.

System programmer response: Put the system in the appropriate state and resend the command.

081A0000 Request sequence error.

Explanation: Unable to perform the command because the target partition is in the deactivated state.

System programmer response: Activate the logical partition, then resend the original request.

Hardware Object Status Summary

Table 27 lists the status values for CPC and image objects provided by the z900 API. The status description was taken from the HMC online help because the API documentation does not provide this information. Note that the status numbers are displayed with the asynchronous report message AOFA0100.

Table 27. Status Values for CPC and Image Objects Provided by the z900 API

Status Value	Number	Description
OPERATING	0001	Image: All of the image's processors are operating. CPC: All of the CPC's processors are operating.
NO POWER	0004	CPC: CPC power is off.
NOT OPERATING	0002	Image: None of the image's processors are operating, but the exact status of the processors vary. CPC: <i>If a power-on reset has not been performed:</i> The CPC's processors cannot operate until a power-on reset of the CPC is performed. <i>If a power-on reset was performed:</i> None of the CPC's processors are operating, but the exact status of the processors vary.
NOT ACTIVATED	0008	Image: The image is defined in the CPC's current input/output (I/O) configuration, but is not activated.

Hardware Object Status Summary

Table 27. Status Values for CPC and Image Objects Provided by the z900 API (continued)

Status Value	Number	Description
EXCEPTIONS	0010	<p>Image: At least one of the image's processors is operating, and at least one processor is not operating, but the exact status of the processors vary.</p> <p>CPC: At least one of the CPC's processors is operating, and at least one processor is not operating, but the exact status of the processors vary.</p>
STATUS CHECK	0020	<p>Image: The CPC is not communicating with the support element. The status of the image and its CPs cannot be determined.</p> <p>CPC: The CPC is not communicating with the support element.</p>
POWERSAVE	0100	<p>CPC: Utility power for the CPC failed, and one or more of its active control programs put the CPC in a power save state. The CPC is using only enough power from its alternate, temporary power source to preserve data for the control programs that put it in the power save state.</p> <p>Image: The image cannot operate until power for the CPC is restored.</p>
LINK NOT ACTIVE	0080	<p>CPC: The CPC's support element is not communicating with this HMC. The status of the CPC cannot be determined.</p>
SERVICE	0040	<p>CPC: A console operator enabled service status for the CPC (ordinarily done at the request of a service representative to allow providing service for the CPC).</p>
SERIOUSALERT	0200	No explanation found on the HMCs.
ALERT	0400	No explanation found on the HMCs.
ENVALERT	0800	No explanation found on the HMCs.
SERVICE REQUIRED	1000	The next disruption will result in the CPC operating in degraded capacity, or it will fail to operate.
DEGRADED	2000	The CPC was found operating in a degraded state. Note that this status is set on specific hardware. Use the HW command GETSGDR to determine the reason.

Hardware Object Status Summary

Appendix E. Automation Manager Reason Codes

These reason codes listed in Table 28 are given in message INGY1004I to indicate that an error occurred in communicating with the automation manager.

Table 28. Automation Manager Reason Codes

Description	Code
PREPROCESSOR_FAILURE	0x00008000
NO_RESOURCES_SELECTED	0x00001004
NO_MATCHES	0x00001008
BROKEN_RELATIONSHIPS	0x0000100C
NO_AGENTS_SELECTED	0x00001010
ROLE_NOT_FOUND	0x00001014
AGENT_NOT_FOUND	0x00001018
NOT_AN_AGENT	0x0000101C
MULTIPLE_AGENTS_FOUND	0x00001020
TABLE_OVERRIDE	0x00001024
TABLE_NOT_FOUND	0x00001028
UNKNOWN_TABLE_SCOPE	0x0000102C
PROCEDURE_NOT_FOUND	0x00001030
INTERNAL_AGENT_SELECTED	0x00001034
REQUEST_SOURCE_NOT_FOUND	0x00001038
REQUEST_NOT_FOUND	0x0000103C
ORDER_IS_NOT_SUPPORTED	0x00001040
RELATIONSHIP_COULD_NOT_BE_ADDED	0x00001044
RELATIONSHIP_NOT_FOUND	0x00001048
TRIGGER_NOT_FOUND	0x0000104C
TRIGGER_LINK_NOT_FOUND	0x00001050
AGENT_LINK_NOT_FOUND	0x00001054
NON_GROUP_AGENT_SELECTED	0x00001058
GROUP_MEMBER_NOT_FOUND	0x0000105C
NOT_A_GROUP_MEMBER	0x00001060
GROUP_POLICY_COULD_NOT_BE_SET	0x00001064
GROUP_MEMBER_POLICY_COULD_NOT_BE_SET	0x00001068
CONFIGURATION_UPDATE_MODE_IS_NOT_SET	0x0000106C
PROCEDURE_COULD_NOT_BE_ADDED	0x00001070
ORDER_NOT_FOUND	0x00001074
PROPAGATION_NOT_FOUND	0x00001078
REQUIRED_TRIGGER_NOT_FOUND	0x0000107C
RESOURCE_COULD_NOT_BE_CREATED	0x00001080
GROUP_COULD_NOT_BE_CREATED	0x00001084
AGENT_COULD_NOT_BE_CREATED	0x00001088
UNKNOWN_GROUP_TYPE	0x00001090
SUPPORTING_RESOURCE_NOT_FOUND	0x00001094
CONFIGURATION_UPDATE_MODE_IS_STILL_SET	0x00001098

Automation Manager Reason Codes

Table 28. Automation Manager Reason Codes (continued)

Description	Code
CONFIGURATION_SOURCE_NOT_FOUND	0x0000109C
CONDITION_COULD_NOT_BE_ADDED	0x000010A0
CONDITION_NOT_FOUND	0x000010A4
UNSUPPORTED_INSTRUCTION	0x000010A8
UNKNOWN_INSTRUCTION	0x000010AC
NO_HISTORIES_FOUND	0x000010B0
INVALID_TABLE_SCOPE	0x000010B4
VARIABLE_HISTORY_NOT_FOUND	0x000010B8
VSAM_ERROR	0x000010BC
RESOURCE_NOT_IN_LIST	0x000010C0
SCHEDULE_COULD_NOT_BE_CREATED	0x000010C4
GROUP_MEMBER_COULD_NOT_BE_ADDED	0x000010C8
INVALID_CONFIG_LOCATION	0x000010D0
CONFIG_MAP_COULD_NOT_BE_OPENED	0x000010D4
CONFIG_MAP_READ_ERROR	0x000010D8
CONFIG_MAP_IS_NOT_VALID	0x000010DC
CONFIG_ID_NOT_FOUND	0x000010E0
CONFIG_SET_COULD_NOT_BE_OPENED	0x000010E4
CONFIG_SET_MISSING_PART_ID	0x000010E8
CONFIG_SET_MISSING_PART_TYPE	0x000010EC
CONFIG_SET_INVALID_PART_TYPE	0x000010F0
CONFIG_SET_MISSING_PART_SIGNATURE	0x000010F4
CONFIG_SET_READ_ERROR	0x000010F8
CONFIG_PART_DUPLICATE_SKIP	0x00001104
CONFIG_PART_NO_UPDATE_SKIP	0x00001108
CONFIG_SET_NO_PARTS_FOUND	0x0000110C
CONFIG_PART_COULD_NOT_BE_OPENED	0x00001110
CONFIG_PART_IS_NOT_VALID	0x00001114
CONFIG_PART_READ_ERROR	0x00001118
NO_CONFIG_PARTS_FOUND_FOR_UPDATE	0x0000111C
PROCESSING_CONFIG_PART	0x00001120
DELETING_CONFIG_PART	0x00001124
MESSAGE_NOT_FOUND	0x00001128
NO_MESSAGE_CLASSES_DEFINED	0x0000112C
MESSAGE_IS_NOT_VALID	0x00001130
MESSAGE_COULD_NOT_BE_LOAD	0x00001134
CONFIG_SET_NO_CONFIG_LEVEL_DEFINED	0x00001138
QUALIFIER_IS_NOT_SUPPORTED	0x0000113C
COLUMN_IS_NOT_SUPPORTED	0x00001140
UNREFERENCED_BRANCH_LABEL	0x00001144
BRANCH_LABEL_NOT_FOUND	0x00001148
ARRAY_IS_EMPTY	0x0000114C
TIMER_NOT_FOUND	0x00001150
TIMER_IS_NOT_RUNNING	0x00001154
TIMER_OVERRIDE	0x00001158

Automation Manager Reason Codes

Table 28. Automation Manager Reason Codes (continued)

Description	Code
TIMER_RESOURCE_NOT_FOUND	0x0000115C
QUALIFIED_TABLE_WITH_BLOCK_COLUMNS	0x00001160
INCOMPATIBLE_PERSISTENT_STATE_IMAGE	0x00001164
PERSISTENT_STATE_IMAGE_PART_COULD_NOT_BE_OPENED	0x00001168
PERSISTENT_STATE_IMAGE_PART_IS_NOT_VALID	0x0000116C
PERSISTENT_STATE_IMAGE_PART_READ_ERROR	0x00001170
PERSISTENT_STATE_IMAGE_PART_WRITE_ERROR	0x00001174
PERSISTENT_STATE_QUEUE_CORRUPTED	0x00001178
OVERRIDE_FILE_NOT_AVAILABLE	0x0000117C
RELATIONSHIP_CLASS_NOT_FOUND	0x00001180
DST_DEACTIVATION_OCCURED	0x00001184
RELATIONSHIP_LOOP_DETECTED	0x00001188
DIAG_INFO_RECORDING_FAILURE	0x0000118C
ALREADY_IN_DIAG_INFO_RECORDING_MODE	0x00001190
NOT_IN_DIAG_INFO_RECORDING_MODE	0x00001194
TIMEREFS_NOT_FOUND	0x00001198
AGENT_COMMS_ADDRESS_NOT_SET	0x0000119C
REQUEST_NOT_LOCKED	0x000011A0
DEPENDENCY_NOT_FOUND	0x000011A4
INVALID_STATE_MODEL	0x000011A8

Automation Manager Reason Codes

Appendix F. Sysplex Communication Services Return and Reason Codes

Table 29 and Table 30 on page 465 list the return and reason codes associated with Sysplex Communications Services internal APIs. These are displayed in messages INGY1000I and AOF350E.

INGY1000I INGPXxxx FAILED IN MODULE *modname*, RC=x'nnnn', REASON=x'nnnn'

Where RC=*return code* and REASON=*function code* from Table 29 (depending on the failing INGPXxxx module).

Note: AOF355E will be issued with RC=40 (invalid call to routine) if any of the following functions are called with invalid parameters.

Table 29. Sysplex Communication Manager API Function Codes

Module INGPXRPC		
Return Code	Function Code	Explanation
0	X'00' 0	OK, command and data accepted and queued. A handle is returned if the target parameter was specified.
0	X'18' 24	Target name is not active or not in the member table. The member table is maintained by the Sysplex Communication Manager and can be displayed via the REXX program INGRXQRY.
0	X'20' 32	REXX variable pool problem (IRXEXCOM error).
0	X'24' 36	Any other internal error.
0	X'28' 40	Sysplex Communication Manager Environment is not available. This may happen, for example, if the Sysplex Communication Manager DST is not started.
0		An ABEND occurred. The function was recovered, but returned unsuccessfully.
0	X'34' 52	Invalid request type was specified.
0	X'38' 56	The is no Primary Automation Manager (PAM) active.
Module INGPXRCV		
Return Code	Function Code	Explanation
0	X'00' 0	Successful completion. For a complete receive request all output was completely received and placed into stem data. Data.0 specifies the number of stem elements. If the response was an empty stem data.0 contains 0. For a partial receive request the requested number or less than the requested number of command output records were received and placed into stem data. Data.0 will contain the number of output records that were received. This might be zero.
0	X'04' 4	Successful completion, but more data is available. (Applicable only to a partial receive).
0	X'08' 8	Data is not yet completely received. (Applicable only to a complete receive).
0	X'0C' 12	Invalid handle. Handle is currently in use or has been deleted or is being deleted.

Table 29. Sysplex Communication Manager API Function Codes (continued)

0	X'10'	16	Processing of the RPC request failed. The Communication Manager detected an error while processing the pipe identified by the handle. This error took place either on the local system or on the remote system. The Communication Manager issued message AOF350E to explain the error in more detail (see Table 30 on page 465).
0	X'20'	32	REXX variable pool problem (IRXEXCOM error).
0	X'24'	36	Any other internal error.
0	X'28'	40	Sysplex Communication Manager Environment is not available. This may happen, for example, if the Sysplex Communication Manager DST is not started.
0	X'2C'	44	An ABEND occurred. The function was recovered, but returned unsuccessfully.
0	X'30'	48	The third parameter (number of records) was specified as zero.
Module INGPXSND			
Return Code	Function Code		Explanation
0	X'00'	0	Successful completion. All command output has been accepted and queued to be sent to the RPC originator.
0	X'0C'	12	Invalid handle. Handle is currently in use or has been deleted or is being deleted.
0	X'10'	16	Processing of the RPC request failed. The Communication Manager detected an error while processing the pipe identified by the handle. This error took place either on the local system or on the remote system. The Communication Manager issued message AOF350E to explain the error in more detail (see Table 30 on page 465).
0	X'20'	32	REXX variable pool problem (IRXEXCOM error).
0	X'24'	36	Any other internal error.
0	X'28'	40	Sysplex Communication Manager Environment is not available. This may happen, for example, if the Sysplex Communication Manager DST is not started.
0	X'2C'	44	An ABEND occurred. The function was recovered, but returned unsuccessfully.
Module INGPXDEL			
Return Code	Function Code		Explanation
0	X'00'	0	Successful completion. Handle has been deleted.
0	X'0C'	12	Invalid handle. Handle is currently in use or has been deleted or is being deleted.
0	X'10'	16	Processing of the RPC request failed. The Communication Manager detected an error while processing the pipe identified by the handle. This error took place either on the local system or on the remote system. The Communication Manager issued message AOF350E to explain the error in more detail (see Table 30 on page 465).
0	X'20'	32	REXX variable pool problem (IRXEXCOM error).
0	X'24'	36	Any other internal error.
0	X'28'	40	Sysplex Communication Manager Environment is not available. This may happen, for example, if the Sysplex Communication Manager DST is not started.
0	X'2C'	44	An ABEND occurred. The function was recovered, but returned unsuccessfully.
Module INGPXWAT			
Return Code	Function Code		Explanation
0	X'00'	0	Successful completion. Wait time has expired.

Table 29. Sysplex Communication Manager API Function Codes (continued)

0	nn	STIMER error code.
Module INGPXMEM		
Return Code	Function Code	Explanation
0	X'00' 0	Successful completion.
0	X'20' 32	REXX variable pool problem (IRXEXCOM error).
0	X'24' 36	Any other internal error.
0	X'28' 40	Sysplex Communication Manager Environment is not available. This may happen, for example, if the Sysplex Communication Manager DST is not started.
0	X'2C' 44	An ABEND occurred. The function was recovered, but returned unsuccessfully.
0	X'34' 52	Invalid function specified.
Module INGPXUTI		
Return Code	Function Code	Explanation
0	X'00' 0	Successful completion.
0	X'1C' 28	Invalid request specified as the first parameter.
0	X'20' 32	REXX variable pool problem (IRXEXCOM error).
0	X'28' 40	Sysplex Communication Manager Environment is not available. This may happen, for example, if the Sysplex Communication Manager DST is not started.
0	X'2C' 44	An ABEND occurred. The function was recovered, but returned unsuccessfully.
Module INGPXQRY		
Return Code	Function Code	Explanation
0	X'00' 0	Successful completion.
0	X'190' 400	Any REXX signal on failure. first parameter.
0	X'194' 404	Receiving the command failed.
0	X'198' 408	Receiving the input data for the command failed.
0	X'19C' 412	Sending the command response back to the originating RPC failed.
0	X'1A0' 416	NetView PIPE failed.
Module INGPXCAM		
The command handler INGPXCAM issues message AOF356 with RC= <i>nn</i> if any of the following errors are detected.		
Return Code	Function Code	Explanation
4		Error, invalid RPC type.
8		Error, REXX function failed.
12		Error, NetView PIPE failed.
16		Error, buffer.0 is wrong.

**AOF350E SYSPLEX COMMUNICATION ERROR: SYSTEM=*sysname*
ERRTYPE=*nn* ERRCODE=*nn* DIAG='INGPX*xxx* '**

Where ERRTYPE=*function code* from Table 29 on page 463 (depending on the failing DIAG=INGPX*xxx* module) and ERRCODE=*error code* from Table 30.

Table 30. Error Codes from Message AOF350E

Error Code	Description
1	Read detects queue is empty.

Table 30. Error Codes from Message AOF350E (continued)

Error Code	Description
4	TOD damaged.
8	CommEnv block is invalid.
12	Cannot add buffer to outbound queue.
16	Cannot create new package buffer.
20	Cannot pack data into package buffer.
24	Cannot unpack package buffer.
28	More buffer expected but not available.
32	Cannot initialize package buffer (read).
36	Cannot release package buffer (read).
40	Dequeue outBndQ failed.
44	Cannot initialize package buffer (outbnd).
48	Cannot release package buffer (outbnd).
52	Cannot write buffer to outbound queue.
56	Cannot add message to inbound queue.
60	Cannot send message.
64	Cannot free pipe into buffer pool.
68	Cannot make buffer from pipe object.
72	Cannot initialize package buffer (close).
76	Remote pipe is damaged.
80	Delete pipe failed.
84	Terminate (DST) was requested.
88	SynchOutbound.
92	SynchOutbound add to queue.
96	SynchOutbound null buffer.
100	SynchOutbound get buffer.
104	SynchOutbound add inbound queue.
108	SynchOutbound reuse buffer.
112	SynchOutbound pipe not synch.
116	Outbound fails with retry (outbound).
120	Outbound retry limit exceeded (outbound).
124	Remote pipe signals error.
128	Remote pipe leave XCF group.
136	Unlock from pipelist error.
144	Inbound queue is out of synch.
148	Pipe cannot be removed.
152	Dequeue outbound failed.
156	Cannot initialize package buffer (copy).
160	Cannot release package buffer (copy).
164	Outbound fails with retry (copy).
168	Outbound retry limit exceeded (copy).
172	Cannot send (copy).
176	Cannot add FIFO into PPI list.

Appendix G. End-to-End Automation Adapter Messages

The following messages are issued by the end-to-end automation adapter.

Note that within NetView an additional * may be appended to the end of the message text.

EEZA0001E Syntax error on line *line number*

Explanation: A syntax error has occurred in the configuration file, for example, a leading = on a line.

System action: The automation adapter stops.

Operator response: Analyze the configuration file for invalid syntax.

EEZA0002E Wrong datatype in key *the key*. Expected *the desired type*, found value "*the value that was found*"

Explanation: The value of the given key cannot be interpreted as the desired type. For example, the system expected a boolean value but found the string "hello".

System action: The automation adapter stops.

Operator response: Analyze the configuration file for invalid key/value pairs.

EEZA0003E The key "*the key that was not found*" was not found and no default value was given

Explanation: The system wanted to retrieve from the configuration file a value that did not exist and no default value was given.

System action: The automation adapter stops.

Operator response: Supply a value for the key in the configuration file.

EEZA0004E Integer out of bounds in key "*the key*". Expected value between *the lower bound expected* and *the upper bound expected*, found *the value parsed*

Explanation: The system expected an integer value between the given bounds (inclusive) for the given key, but found a value outside these bounds.

System action: The automation adapter stops.

Operator response: Supply a value within the given bounds for the key.

EEZA0006E Cannot create an instance of the class because class not found: *class name*

Explanation: The automation adapter cannot load the class.

System action: The automation adapter rejects the request.

Operator response: Check whether the class name is valid and is available in the corresponding classpath.

EEZA0007E Cannot create an instance of the class because method not found: *class name*

Explanation: The automation adapter can load the class but cannot create an instance.

System action: The automation adapter rejects the request.

Operator response: Check whether the class is valid.

EEZA0008E Cannot create an instance of the class because of an unknown error: *class name*

Explanation: The automation adapter cannot load the class or create an instance.

System action: The automation adapter rejects the request.

Operator response: Check whether the class is valid and analyze the attached original exception.

EEZA0009E Invocation of adapter plug-in failed: *plug-in=plug-in name, method=method name, internalRetcode=internal return code, taskRetcode=task return code*

Explanation: The automation adapter client API was called to execute a task on the remote automation adapter. There are three error categories:

- The client suffers an error on the connection
- The execution of the task within the automation adapter backend failed
- Execution failed in the automation adapter plug-in.

| Automation adapter internal return codes:
| 3 Backend detects that the request expired
| before it was started.
| 4 Backend cannot be loaded.
| 22 Backend cannot load the plug-in Java class.

- | 23 An exception occurred in the backend or the plug-in returns a non-zero task return code.
- | 24 Backend rejects the execution of the task. See the automation adapter trace for details.
- | 30 Backend detects invalid input arguments. Cannot execute task. This is an internal error.
- | 40 Authentication failed. The user ID or password has not been specified or an internal error occurred.
- | 41 Authentication failed. The user ID or password is invalid on the automation adapter host.
- | 42 Authentication failed. The user ID has been revoked. JAAS login modules may not provide this information or may also use it if the password has expired.
- | 43 Authentication failed. The password has expired. JAAS login modules may not provide this information.

The automation adapter client connection internal return codes:

- | 101 Connection timed out while open.
- | 102 Connection failed during open/read/write because peer closed.
- | 103 Not all data received. The data was read successfully but the number of bytes is too small. This might be an internal error.
- | 104 Unable to connect to the automation adapter.
- | 105 Unable to close connection. This might be an internal error.
- | 106 Unable to send data. Connection has been aborted due to, for example, SSL mismatch or the automation adapter stopped.
- | 107 Marshal error occurred while sending or receiving data. This is an internal error.
- | 108 Connection is invalid, for example, connection has not been opened or previously failed.
- | 109 Read response failed. Data is zero, for example, the connection has been aborted due to an SSL mismatch.
- | 110 Connection timed out during read/write.

Predefined task return codes from plug-in:

- | <0 Internally used task return codes have been mapped to one of the automation adapter return codes.
- | 0 The plug-in was successful.
- | 4004 The plug-in connection timed out.
- | 4008 The plug-in authorization is missing to execute the request.
- | 4009 The plug-in authentication failed.
- | 4012 The plug-in failed but might execute the next request successfully. This is the default return code if the plug-in throws an exception but the plug-in return code is zero.
- | 4016 The plug-in failed and might not execute the next request successfully.
- | 4020 The plug-in failed and forces the entire automation adapter to stop.

- | 4024 The plug-in cannot find the policy. For example, this may occur if the policy should be activated.
- | 4028 The plug-in has found an error in the automation policy. For example, a policy rule was violated.

System action: Execution of the remote task fails.

- | **Operator response:** Analyze the return code description. If it is an internal error, check IBM Electronic Support for additional information - <http://www.ibm.com/support/entry/portal/>

EEZA0010E Request expires before the adapter passes it to the adapter plug-in. Timeout period is *timeout value* seconds

Explanation: All requests have an associated expiration date. The request is scheduled to an execution thread that detected that the expiration time had expired.

System action: The automation adapter rejects the request.

Operator response: Analyze the reason (for example, high working load). Increase the timeout period if necessary.

EEZA0011E The backend program specification is invalid

Explanation: The backend program is not a Java program or the Java program name was not specified.

System action: The automation adapter rejects the request.

- | **Operator response:** Check the program that called the automation adapter client API.

EEZA0012E Invalid parameter list

Explanation: The automation adapter detected a request that is associated with an invalid parameter list.

System action: The automation adapter rejects the request.

- | **Operator response:** Check the program that called the automation adapter client API.

EEZA0013E Authentication for user ID *user name* failed

Explanation: The request is associated with a user ID and password that have been validated unsuccessfully.

System action: The automation adapter rejects the request.

Operator response: Check whether the user ID is authorized for the system and check the security policy.

EEZA0014E The original exception *original-class* needs to be transported to the remote caller

Explanation: An exception from an underlying component needs to be transported to the remote caller.

System action: None.

Operator response: Analyze the original exception attached with this message.

EEZA0015E Method not supported: *name of the missing method*

Explanation: The automation adapter detected an unknown method name. The list of all valid method names is defined in the EEZAdapterInteraction interface.

System action: The automation adapter rejects the request.

| **Operator response:** Check IBM Electronic Support for
| additional information - [http://www.ibm.com/
| support/entry/portal/](http://www.ibm.com/support/entry/portal/)

EEZA0017E Request not supported: *name of the unsupported request*

Explanation: The automation adapter plug-in does not support the specified request.

System action: The request might be rejected depending on the behavior of the plug-in.

| **Operator response:** Check if the automation domain
| supports this type of request.

EEZA0022E Adapter client is unable to connect to the adapter at *host:port* due to exception: *the exception that was caught*

Explanation: The automation adapter client cannot connect to the server at the given host and port. The original exception text is provided.

System action: The connection is not established.

Operator response: Analyze the original exception.

EEZA0023E Cache directory is invalid

Explanation: The EIF cache directory is not a directory.

System action: The automation adapter stops.

Operator response: Correct the configuration file.

EEZA0024E EIF sender and receiver must not be equal

Explanation: The EIF configuration parameters are not allowed to point to each other.

System action: The automation adapter stops.

Operator response: Correct the configuration file.

EEZA0025E Cannot find the plug-in configuration file: *configuration file name*

Explanation: The master configuration file contains the name of a plug-in configuration file that cannot be found.

System action: The automation adapter stops.

Operator response: Correct the configuration file.

EEZA0026E No plug-in configuration file was specified

Explanation: The master configuration file must contain at least one plug-in configuration file.

System action: The automation adapter stops.

Operator response: Correct the configuration file.

EEZA0027E Cannot load configuration file: *configuration file name*

Explanation: The specified configuration file cannot be loaded.

System action: The automation adapter stops.

Operator response: Correct the configuration file.

EEZA0028E Plug-in configuration file does not contain all mandatory parameters: *configuration file name*

Explanation: The specified configuration file does not contain all mandatory parameters. The plug-in is not used.

System action: The automation adapter does not deploy the plug-in.

Operator response: Correct the configuration file.

EEZA0029E Cannot create the first instance of the plug-in class: *class name*

Explanation: An attempt was made to create the first instance of the plug-in during initialization. Creation failed.

System action: The automation adapter does not deploy the plug-in.

Operator response: Correct the configuration file.

EEZA0030E Cannot set up event subscription list for plug-in configuration file: *plug-in configuration file name*

Explanation: The specification of the EIF event classes in the plug-in configuration file is invalid.

System action: The automation adapter does not deploy the plug-in.

Operator response: Correct the configuration file.

EEZA0031E Cannot load configuration file from: *plug-in configuration file name*

Explanation: The automation adapter cannot load the specified configuration file because either no configuration file or an invalid one was specified.

System action: The automation adapter stops.

Operator response: Check whether the name of the configuration file is correct.

EEZA0032E Initialization of the adapter failed: *original exception*

Explanation: An error occurred in the initialization step of the automation adapter.

System action: The automation adapter stops.

Operator response: Analyze the associated exception. If there is no exception text for this message, try to find additional messages that were sent previously.

EEZA0033E Unable to create *type of factory* SocketFactory

Explanation: The automation adapter server or client cannot create a socket factory for the remote contact.

System action: The automation adapter client cannot create a connection or the automation adapter server cannot receive connections.

Operator response: Analyze the reason using previous messages.

EEZA0036E The adapter suffered an unexpected interruption: *original exception*

Explanation: The automation adapter waits for a termination command. An unexpected interruption occurred.

System action: The automation adapter stops.

Operator response: Analyze the original exception.

EEZA0037E The adapter stops because no plug-in has been successfully initialized

Explanation: At least one plug-in must have been successfully initialized otherwise the automation adapter stops.

System action: The automation adapter stops.

Operator response: Analyze previous messages and exceptions issued by the failing plug-in.

EEZA0038E A (SSL) socket configuration error occurred: *exception text*

Explanation: An error occurred during the loading or processing of (SSL) socket-related configuration data. An SSL handshake exception will only be reported during initial contact.

System action: The automation adapter client cannot create a connection or the automation adapter server cannot receive connections.

Operator response: Analyze the exceptions text. Check the SSL configuration file, `ing.adapter.ssl.properties`, if necessary.

EEZA0039E Not all data was read from socket: *number of bytes read bytes read, number of bytes expected bytes expected to be read*

Explanation: The incoming request has a length in bytes, but not all bytes can be read.

System action: The automation adapter rejects the request.

Operator response: Check why the socket connection was broken while transferring data.

EEZA0040E The adapter client cannot establish connection to the adapter: *string representation of the connection*

Explanation: Opening the connection failed. A request cannot be sent to the automation adapter. The string representation of the connection contains details about the connection.

System action: The automation adapter frontend failed.

Operator response: Analyze the connection information.

EEZA0041E The adapter client cannot invoke an adapter request: *InternalRC=internal return code, TaskRC=task return code*

Explanation: A connection to the automation adapter has been successfully established. The automation adapter frontend might have sent a request to the automation adapter but the request failed. If the

internal or task return codes are not applicable (n/a), some other unexpected exception occurred.

System action: The automation adapter frontend failed.

Operator response: Analyze the internal and task return codes (see EEZA0009E for an explanation of the return codes).

EEZA0042E The adapter has thrown a remote exception: InternalRC=*internal return code*, TaskRC=*task return code*. The original message was: *message text*

Explanation: A connection to the automation adapter has been successfully established. The automation adapter frontend has sent a request to the automation adapter but the plug-in has thrown an exception.

System action: None.

Operator response: Analyze the internal and task return codes (see EEZA0009E for an explanation of the return codes).

EEZA0043E A required command line parameter is missing

Explanation: One of the required command line parameters is missing (such as -start,-stop or -terminate).

System action: The automation adapter frontend failed.

Operator response: Specify the required command-line parameters and try again.

EEZA0045E The adapter cannot establish a server socket due to illegal arguments: *exception text*

Explanation: The automation adapter cannot establish a receiver thread and cannot accept incoming connections.

System action: The automation adapter stops.

Operator response: Analyze the configuration file for an invalid IP address.

EEZA0047E The adapter is unable to accept connections due to socket exception "*exception*"

Explanation: An exception occurred as the automation adapter was about to accept an incoming connection.

System action: The automation adapter stops.

Operator response: Analyze the exception text.

EEZA0051W Termination of the adapter failed due to exception: *error message*

Explanation: The attempt to stop the receiver thread failed because an exception occurred.

System action: None.

Operator response: Analyze the exception text.

EEZA0052E Cannot create an in-storage EIF configuration file: *exception text*

Explanation: An instance of the Java class ByteArrayInputStream cannot be created or written.

System action: The automation adapter stops.

Operator response: This is probably an internal error. The exception text might give the reason for the problem.

EEZA0053E Missing argument for command line parameter "*the parameter*"

Explanation: A required argument for a command line parameter (such as -start) is missing. For example, AdapterCmd -start would be incorrect because -start requires an argument. A correct example would be: AdapterCmd -start com.ibm.ing.sapplugin.INGXPluginInvocation

System action: Processing of this command ends.

Operator response: Check the documentation for information about valid command line arguments and their parameters.

EEZA0055E Remote Contact inactivity threshold exceeded: elapsed seconds=*elapsed seconds* threshold=*threshold*

Explanation: The automation adapter calculates the elapsed time since the last synchronous request was received. The automation adapter stops itself if this time exceeds the number specified in the parameter eez-remote-contact-activity-interval-seconds. Any incoming event is used as a trigger for the calculation.

System action: The automation adapter stops.

Operator response: You might want to increase the number of seconds specified by the parameter eez-remote-contact-activity-interval-seconds. Setting this parameter to 0 (zero) means it never expires.

EEZA0056I Initial contact was enabled and the connection to the management server has been established

Explanation: The parameter eez-initial-contact was set to true and the automation adapter attempted to connect the management server. The handshake to the management server was successful.

EEZA0057E • EEZA0071E

System action: None.

| **Operator response:** No action required.

EEZA0057E The connection to the management server cannot be established

Explanation: The automation adapter stops attempting to connect to the management server because the timeout interval is over.

System action: The automation adapter stops.

Operator response: You might want to increase the number of minutes specified by the parameter `eez-initial-contact-retry-interval-minutes`. Specify the value 0 (zero) in order to retry forever.

EEZA0058E The plug-in has not been deployed or not yet started: *name of the Java plug-in class*

Explanation: An attempt was made to issue a request against a non-deployed plug-in or a plug-in that has not been started.

System action: The automation adapter rejects the request.

Operator response: Check the plug-in configuration file and deploy the missing plug-in class. Search for message EEZA0115I.

EEZA0059E An internal error occurred

Explanation: The automation adapter detected an internal error.

System action: None.

| **Operator response:** Contact your IBM Electronic Support for additional information -
| <http://www.ibm.com/support/entry/portal/>

EEZA0060I The termination of the adapter is delayed for *duration of the delay in seconds* seconds

Explanation: Stopping the automation adapter is delayed for a short while until it has sent the appropriate domain leave events. You can configure the duration of this delay with the `eez-stop-delay-seconds` parameter.

System action: The automation adapter attempts to send domain leave events.

| **Operator response:** No action required.

EEZA0061E Unable to bind a socket to address *eez-remote-contact-hostname* at port *eez-remote-contact-port*. Reason: *message of the exception*

Explanation: The automation adapter was unable to

use this address or port. Possible causes of the problem are:

- The port is already in use by another program.
- The address could not be assigned.

System action: The automation adapter stops.

Operator response: Make sure that no program is using this port (that is, an automation adapter that is already running). If another program needs this port configure the automation adapter to use another port (with the `eez-remote-contact-port` parameter in the master configuration file). Ensure that the address is valid.

EEZA0062I The start command of the automation plug-in *name of the Java plug-in class* was successful

Explanation: The selected automation plug-in was successfully started.

System action: The automation adapter has started the automation plug-in.

| **Operator response:** No action required.

EEZA0063I The stop command of the automation plug-in *name of the Java plug-in class* was successful

Explanation: The selected automation plug-in was successfully stopped.

System action: The automation adapter has stopped the automation plug-in.

| **Operator response:** No action required.

EEZA0064I The termination command for the adapter was successful

Explanation: The automation adapter was successfully stopped.

System action: The automation adapter stops.

| **Operator response:** No action required.

EEZA0070E The host name *eez-remote-contact-hostname* is unknown

Explanation: The automation adapter was unable to resolve the host name.

System action: The automation adapter stops.

Operator response: Specify a valid host name.

EEZA0071E The domain name is either null or empty

Explanation: The plug-in returned an invalid domain name since it is either null or empty.

System action: The plug-in cannot be started.

Operator response: Specify a valid domain name in the plug-in configuration file.

EEZA0100I The adapter has been started

Explanation: This is the first of a sequence of three messages until the automation adapter is ready. The automation adapter starts initialization and will try to connect to the management server if `eez-initial-contact=true`.

System action: None.

| **Operator response:** No action required.

EEZA0101I The adapter is active

Explanation: The automation adapter becomes *active* after a connection has been successfully established to the management server. The automation adapter continues initialization, finds and starts up all plug-ins.

System action: None.

| **Operator response:** No action required.

EEZA0102I The adapter is ready

Explanation: The automation adapter startup sequence is complete.

System action: None.

| **Operator response:** No action required.

EEZA0103I The adapter is stopping

Explanation: An internal or an external stop command has been received.

System action: The automation adapter is about to stop.

| **Operator response:** No action required.

EEZA0104I The adapter has been stopped

Explanation: The automation adapter termination is complete. All possible stop delay periods are over. The process stops immediately.

System action: The automation adapter has stopped.

Operator response: No action required.

EEZA0105I The adapter has been stopped due to a failure, rc=return code

Explanation: The automation adapter stopped because an error occurred. All possible stop delay periods are over. The process stops immediately. The return code might be:

12 if initial contact failed

13 if the remote contact activity threshold is exceeded

16 if a plug-in forced termination of the automation adapter

20 if initialization failed

24 if an error occurred after initialization was successful

28 if an unsupported environment is detected

System action: The automation adapter stops.

Operator response: Search for error messages that were issued previously. On z/OS return code 28 might be caused by the 64-bit JVM. You should use the 32-bit JVM instead.

EEZA0111I The plug-in is starting: name of the Java plug-in class

Explanation: The automation adapter has already successfully created an instance of the plug-in class and will now call function `INIT_DOMAIN`.

System action: None.

| **Operator response:** No action required.

EEZA0112I The plug-in has been started: name of the Java plug-in class

Explanation: The automation adapter plug-in has successfully initialized the domain (`INIT_DOMAIN`).

System action: None.

| **Operator response:** No action required.

EEZA0113I The plug-in is stopping: name of the Java plug-in class

Explanation: The automation adapter will call plug-in function `TERM_DOMAIN`.

System action: None.

| **Operator response:** No action required.

EEZA0114I The plug-in has been stopped: name of the Java plug-in class

Explanation: The automation adapter plug-in has successfully stopped the domain (`TERM_DOMAIN`).

System action: None.

| **Operator response:** No action required.

EEZA0115I The plug-in startup failed: name of the Java plug-in class

Explanation: This message might follow after `EEZA0111I`, but the attempt to start the plug-in via function `INIT_DOMAIN` failed. The automation adapter plug-in will not be started automatically.

System action: The plug-in will be disabled. A join event was not sent.

Operator response: You might want to restart the plug-in using the automation adapter start command. Analyze further plug-in messages.

EEZA0116I The status of the event sender changed:
Address=Address, Port=Port,
Status=Status

Explanation: This message occurs if the status of the EIF connection changed. The reason could be that a new EIF connection is created or an existing EIF connection is lost. The reason can be found in the status. A status='connection timed out' is expected if the SA Application Manager is stopped, for example when the SA Application Manager moves to another system and therefore the adapter needs to change the EIF sender destination. The *status* can be:

- 1 connection created
- 2 connection changed
- 4 connection closed
- 8 connection timed out

System action: None.

Operator response: No action required.

EEZA9991E The message file is not installed

Explanation: The English message file must be available.

System action: The automation adapter stops.

Operator response: Make sure that the message file is in the class path

EEZA9992E EEZAdapterLogger is not available

Explanation: The automation adapter logging component has not been initialized.

System action: The automation adapter stops. Other processes using the automation adapter client API will be unable to write messages to log and trace files.

Operator response: Check IBM Electronic Support for additional support - <http://www.ibm.com/support/entry/portal/>

EEZC0001I Setting up Tivoli Common Directory at location where Tivoli Common Directory is being set up.

Explanation: The Tivoli Common Directory path was set to its default value, as shown in the message text.

System action: No system action required.

Operator response: No operator action required.

EEZC0002I Unable to determine Tivoli Common Directory. Diverting serviceability related output to alternative location.

Explanation: The system was not able to determine the Tivoli Common Directory.

System action: Processing continues. The application will attempt to divert serviceability related output to another location for this session.

Operator response: In order to manage its serviceability related output, the application should be granted read/write permission to the location /etc/ibm/tivoli/common (UNIX) or <Program_Files_Dir>\ibm\tivoli\common (Windows).

EEZC0003I Base output directory for serviceability related files (for example, message log files and trace files) has been set to new output directory.

Explanation: The output directory for serviceability related files was set to its default value, as shown in the message text.

System action: From now on the application will write serviceability related information to the directory that is contained in the message text.

Operator response: No action is required if the base output directory for serviceability related files is acceptable. Otherwise, if it is required to relocate the base output directory, modify the entry in log.properties, which should be located at <Program_Files_dir>\ibm\tivoli\common\cfg (Windows) or /etc/ibm/tivoli/common/cfg/log.properties (UNIX). Changes to this file will take effect once the corresponding component is restarted.

EEZC0004I Changing base output directory for serviceability related files of name of logger from old output directory to new output directory.

Explanation: Due to changes in configuration settings the output directory of serviceability related files has been relocated.

System action: From now on the system will write serviceability related information to the new location.

Operator response: No action is required if the base output directory for serviceability related files is acceptable. Otherwise, if it is required to relocate the base output directory, modify the entry in log.properties, which should be located at <Program_Files_dir>\ibm\tivoli\common\cfg (Windows) or /etc/ibm/tivoli/common/cfg/log.properties (UNIX). Changes to this file will take effect once the corresponding component is restarted.

EEZK0003E String *someString* is too long: the maximum length of *nameOfTheString* Strings is *maxLength*.

Explanation: Setting the String to the specified value did not succeed due to string length.

System action: The current task ends.

Operator response: Verify the input parameters. If the problem persists, supply the trace file and this message text to your IBM Support Center.

EEZK0004E String named *someStringName* must not be null and must not exceed the maximum length of *maxLength*.

Explanation: Setting the String to null is not allowed.

System action: The current task ends.

Operator response: Verify the input parameters. If the problem persists, supply the trace file and this message text to your IBM Support Center.

EEZK0005E An exception, which is not an instance of `EEZApplicationException` has been passed to the `EEZApplicationTransientException`. The type of the message is *exceptionType*. The exception message is: *exceptionMessage*.

Explanation: This is an unexpected behavior.

System action: The current task will continue. The exception will be processed.

Operator response: Provide the logs and traces for more details, if any error other error occurs.

Glossary

This glossary includes terms and definitions from:

- The *IBM Dictionary of Computing* New York: McGraw-Hill, 1994.
- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies can be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018. Definitions are identified by the symbol (A) after the definition.
- The *Information Technology Vocabulary* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.

The following cross-references are used in this glossary:

Contrast with. This refers to a term that has an opposed or substantively different meaning.

Deprecated term for. This indicates that the term should not be used. It refers to a preferred term, which is defined in its proper place in the glossary.

See. This refers the reader to multiple-word terms in which this term appears.

See also. This refers the reader to terms that have a related, but not synonymous, meaning.

Synonym for. This indicates that the term has the same meaning as a preferred term, which is defined in the glossary.

Synonymous with. This is a backward reference from a defined term to all other terms that have the same meaning.

A

ACF. See automation configuration file.

ACF/NCP. Advanced Communications Function for the Network Control Program. See Advanced Communications Function and Network Control Program.

ACF/VTAM. Advanced Communications Function for the Virtual Telecommunications Access Method. Synonym for VTAM. See Advanced Communications Function and Virtual Telecommunications Access Method.

active monitoring. In SA z/OS automation control file, the acquiring of resource status information by soliciting such information at regular, user-defined intervals. See also passive monitoring.

adapter. Hardware card that enables a device, such as a workstation, to communicate with another device, such as a monitor, a printer, or some other I/O device.

adjacent hosts. Systems connected in a peer relationship using adjacent NetView sessions for purposes of monitoring and control.

adjacent NetView. In SA z/OS, the system defined as the communication path between two SA z/OS systems that do not have a direct link. An adjacent NetView is used for message forwarding and as a communication link between two SA z/OS systems. For example, the adjacent NetView is used when sending responses from a focal point to a remote system.

Advanced Communications Function (ACF). A group of IBM licensed programs (principally VTAM, TCAM, NCP, and SSP) that use the concepts of Systems Network Architecture (SNA), including distribution of function and resource sharing.

advanced program-to-program communication (APPC). A set of inter-program communication services that support cooperative transaction processing in a Systems Network Architecture (SNA) network. APPC is the implementation, on a given system, of SNA's logical unit type 6.2.

alert. (1) In SNA, a record sent to a system problem management focal point or to a collection point to communicate the existence of an alert condition. (2) In NetView, a high-priority event that warrants immediate attention. A database record is generated for certain event types that are defined by user-constructed filters.

alert condition. A problem or impending problem for which some or all of the process of problem determination, diagnosis, and resolution is expected to require action at a control point.

alert focal-point system. See NPDA focal point system.

alert threshold. An application or volume service value that determines the level at which SA z/OS changes the associated icon in the graphical interface to the alert color. SA z/OS may also issue an alert. See warning threshold.

AMC. (1) See Automation Manager Configuration. (2) The Auto Msg Classes entry type.

American Standard Code for Information Interchange

(ASCII). A standard code used for information exchange among data processing systems, data communication systems, and associated equipment. ASCII uses a coded character set consisting of 7-bit coded characters (8-bit including parity check). The ASCII set consists of control characters and graphic characters. See also Extended Binary Coded Decimal Interchange Code.

APF. See authorized program facility.

API. See application programming interface.

APPC. See advanced program-to-program communication.

application. In SA z/OS, applications refer to z/OS subsystems, started tasks, or jobs that are automated and monitored by SA z/OS. On SNMP-capable processors, application can be used to refer to a subsystem or process.

Application entry. A construct, created with the customization dialogs, used to represent and contain policy for an application.

application group. A named set of applications. An application group is part of an SA z/OS enterprise definition and is used for monitoring purposes.

application program. (1) A program written for or by a user that applies to the user's work, such as a program that does inventory or payroll. (2) A program used to connect and communicate with stations in a network, enabling users to perform application-oriented activities.

application programming interface (API). An interface that allows an application program that is written in a high-level language to use specific data or functions of the operating system or another program.

ApplicationGroup entry. A construct, created with the customization dialogs, used to represent and contain policy for an application group.

ARM. See automatic restart management.

ASCB. Address space control block.

ASCB status. An application status derived by SA z/OS running a routine (the ASCB checker) that searches the z/OS address space control blocks (ASCBs) for address spaces with a particular job name. The job name used by the ASCB checker is the job name defined in the customization dialog for the application.

ASCII. See American Standard Code for Information Interchange.

ASF. See automation status file.

authorized program facility (APF). A facility that permits identification of programs that are authorized to use restricted functions.

automated console operations (ACO). The use of an automated procedure to replace or simplify the action that an operator takes from a console in response to system or network events.

automated function. SA z/OS automated functions are automation operators, NetView autotasks that are assigned to perform specific automation functions. However, SA z/OS defines its own synonyms, or *automated function names*, for the NetView autotasks, and these function names are referred to in the sample policy databases provided by SA z/OS. For example, the automation operator AUTBASE corresponds to the SA z/OS automated function BASEOPER.

automatic restart management (ARM). A z/OS recovery function that improves the availability of specified subsystems and applications by automatically restarting them under certain circumstances. Automatic restart management is a function of the Cross-System Coupling Facility (XCF) component of z/OS.

automatic restart management element name. In MVS 5.2 or later, z/OS automatic restart management requires the specification of a unique sixteen character name for each address space that registers with it. All automatic restart management policy is defined in terms of the element name, including SA z/OS's interface with it.

automation. The automatic initiation of actions in response to detected conditions or events. SA z/OS provides automation for z/OS applications, z/OS components, and remote systems that run z/OS. SA z/OS also provides tools that can be used to develop additional automation.

automation agent. In SA z/OS, the automation function is split up between the automation manager and the automation agents. The observing, reacting and doing parts are located within the NetView address space, and are known as the *automation agents*. The automation agents are responsible for:

- Recovery processing
- Message processing

- Active monitoring: they propagate status changes to the automation manager

automation configuration file. The SA z/OS customization dialogs must be used to build the automation configuration file. It consists of:

- The automation manager configuration file (AMC)
- The NetView automation table (AT)
- The NetView message revision table (MRT)
- The MPFLSTSA member

automation control file (ACF). In SA z/OS, a file that contains system-level automation policy information. There is one master automation control file for each NetView system that SA z/OS is installed on. Additional policy information and all resource status information is contained in the policy database (PDB). The SA z/OS customization dialogs must be used to build the automation control files. They must not be edited manually.

automation flags. In SA z/OS, the automation policy settings that determine the operator functions that are automated for a resource and the times during which automation is active. When SA z/OS is running, automation is controlled by automation flag policy settings and override settings (if any) entered by the operator. Automation flags are set using the customization dialogs.

automation manager. In SA z/OS, the automation function is split up between the automation manager and the automation agents. The coordination, decision making and controlling functions are processed by each sysplex's *automation manager*.

The automation manager contains a model of all of the automated resources within the sysplex. The automation agents feed the automation manager with status information and perform the actions that the automation manager tells them to.

The automation manager provides *sysplex-wide* automation.

Automation Manager Configuration. The Automation Manager Configuration file (AMC) contains an image of the automated systems in a sysplex or of a standalone system. See also "automation configuration file."

Automation NetView. In SA z/OS the NetView that performs routine operator tasks with command procedures or uses other ways of automating system and network management, issuing automatic responses to messages and management services units.

automation operator. NetView automation operators are NetView autotasks that are assigned to perform specific automation functions. See also automated function. NetView automation operators may receive messages and process automation procedures. There are no logged-on users associated with automation

operators. Each automation operator is an operating system task and runs concurrently with other NetView tasks. An automation operator could be set up to handle JES2 messages that schedule automation procedures, and an automation statement could route such messages to the automation operator. Similar to *operator station task*. SA z/OS message monitor tasks and target control tasks are automation operators.

automation policy. The policy information governing automation for individual systems. This includes automation for applications, z/OS subsystems, z/OS data sets, and z/OS components.

automation policy settings. The automation policy information contained in the automation control file. This information is entered using the customization dialogs. You can display or modify these settings using the customization dialogs.

automation procedure. A sequence of commands, packaged as a NetView command list or a command processor written in a high-level language. An automation procedure performs automation functions and runs under NetView.

| **automation routines.** In SA z/OS, a set of
| self-contained automation routines that can be called
| from the NetView automation table, or from
| user-written automation procedures.

automation status file (ASF). In SA z/OS, a file containing status information for each automated subsystem, component or data set. This information is used by SA z/OS automation when taking action or when determining what action to take. In Release 2 and above of AOC/MVS, status information is also maintained in the operational information base.

automation table (AT). See NetView automation table.

autotask. A NetView automation task that receives messages and processes automation procedures. There are no logged-on users associated with autotasks. Each autotask is an operating system task and runs concurrently with other NetView tasks. An autotask could be set up to handle JES2 messages that schedule automation procedures, and an automation statement could route such messages to the autotasks. Similar to *operator station task*. SA z/OS message monitor tasks and target control tasks are autotasks. Also called *automation operator*.

available. In VTAM programs, pertaining to a logical unit that is active, connected, enabled, and not at its session limit.

B

Base Control Program (BCP). A program that provides essential services for the MVS and z/OS operating systems. The program includes functions that

manage system resources. These functions include input/output, dispatch units of work, and the z/OS UNIX System Services kernel. See also Multiple Virtual Storage and z/OS.

basic mode. A central processor mode that does not use logical partitioning. Contrast with logically partitioned mode.

BCP. See Base Control Program.

BCP Internal Interface. Processor function of CMOS-390 and System z processor families. It allows for communication between basic control programs such as z/OS and the processor support element in order to exchange information or to perform processor control functions. Programs using this function can perform hardware operations such as ACTIVATE or SYSTEM RESET.

beaconing. The repeated transmission of a frame or messages (beacon) by a console or workstation upon detection of a line break or outage.

| **blade.** A hardware unit that provides
| application-specific services and components. The
| consistent size and shape (or form factor) of each blade
| allows it to fit in a BladeCenter chassis.

| **BladeCenter chassis.** A modular chassis that can
| contain multiple blades, allowing the individual blades
| to share resources such as management, switch, power,
| and blower modules.

BookManager®. An IBM product that lets users view softcopy documents on their workstations.

C

central processor (CP). The part of the computer that contains the sequencing and processing facilities for instruction execution, initial program load (IPL), and other machine operations.

central processor complex (CPC). A physical collection of hardware that consists of central storage, one or more central processors, timers, and channels.

central site. In a distributed data processing network, the central site is usually defined as the focal point for alerts, application design, and remote system management tasks such as problem management.

CFR/CFS and ISC/ISR. I/O operations can display and return data about integrated system channels (ISC) connected to a coupling facility and coupling facility receiver (CFR) channels and coupling facility sender (CFS) channels.

channel. A path along which signals can be sent; for example, data channel, output channel. See also link.

channel path identifier. A system-unique value assigned to each channel path.

channel-attached. (1) Attached directly by I/O channels to a host processor (for example, a channel-attached device). (2) Attached to a controlling unit by cables, rather than by telecommunication lines. Contrast with link-attached. Synonymous with local.

CHPID. In SA z/OS, channel path ID; the address of a channel.

CHPID port. A label that describes the system name, logical partitions, and channel paths.

CI. See console integration.

CICS/VS. Customer Information Control System for Virtual Storage. See Customer Information Control System.

CLIST. See command list.

clone. A set of definitions for application instances that are derived from a basic application definition by substituting a number of different system-specific values into the basic definition.

clone ID. A generic means of handling system-specific values such as the MVS SYSCLONE or the VTAM subarea number. Clone IDs can be substituted into application definitions and commands to customize a basic application definition for the system that it is to be instantiated on.

CNC. A channel path that transfers data between a host system image and an ESCON control unit. It can be point-to-point or switchable.

command. A request for the performance of an operation or the execution of a particular program.

command facility. The component of NetView that is a base for command processors that can monitor, control, automate, and improve the operation of a network. The successor to NCCF.

command list (CLIST). (1) A list of commands and statements, written in the NetView command list language or the REXX language, designed to perform a specific function for the user. In its simplest form, a command list is a list of commands. More complex command lists incorporate variable substitution and conditional logic, making the command list more like a conventional program. Command lists are typically interpreted rather than being compiled. (2) In SA z/OS, REXX command lists that can be used for automation procedures.

command procedure. In NetView, either a command list or a command processor.

command processor. A module designed to perform a specific function. Command processors, which can be written in assembler or a high-level language (HLL), are issued as commands.

command processor control block. An I/O operations internal control block that contains information about the command being processed.

Command Tree/2. An OS/2-based program that helps you build commands on an OS/2 window, then routes the commands to the destination you specify (such as a 3270 session, a file, a command line, or an application program). It provides the capability for operators to build commands and route them to a specified destination.

common commands. The SA z/OS subset of the CPC operations management commands.

Common User Access (CUA) architecture. Guidelines for the dialog between a human and a workstation or terminal.

communication controller. A type of communication control unit whose operations are controlled by one or more programs stored and executed in the unit or by a program executed in a processor to which the controller is connected. It manages the details of line control and the routing of data through a network.

communication line. Deprecated term for telecommunication line.

connectivity view. In SA z/OS, a display that uses graphic images for I/O devices and lines to show how they are connected.

console automation. The process of having NetView facilities provide the console input usually handled by the operator.

console connection. In SA z/OS, the 3270 or ASCII (serial) connection between a PS/2 computer and a target system. Through this connection, the workstation appears (to the target system) to be a console.

console integration (CI). A hardware facility that if supported by an operating system, allows operating system messages to be transferred through an internal hardware interface for display on a system console. Conversely, it allows operating system commands entered at a system console to be transferred through an internal hardware interface to the operating system for processing.

consoles. Workstations and 3270-type devices that manage your enterprise.

Control units. Hardware units that control I/O operations for one or more devices. You can view information about control units through I/O

operations, and can start or stop data going to them by blocking and unblocking ports.

controller. A unit that controls I/O operations for one or more devices.

converted mode (CVC). A channel operating in converted (CVC) mode transfers data in blocks and a CBY channel path transfers data in bytes. Converted CVC or CBY channel paths can communicate with a parallel control unit. This resembles a point-to-point parallel path and dedicated connection, regardless whether it passes through a switch.

couple data set. A data set that is created through the XCF couple data set format utility and, depending on its designated type, is shared by some or all of the z/OS systems in a sysplex. See also sysplex couple data set and XCF couple data set.

coupling facility. The hardware element that provides high-speed caching, list processing, and locking functions in a sysplex.

CP. See central processor.

CPC. See central processor complex.

CPC operations management commands. A set of commands and responses for controlling the operation of System/390 CPCs.

CPC subset. All or part of a CPC. It contains the minimum *resource* to support a single control program.

CPCB. See command processor control block.

CPU. Central processing unit. Deprecated term for processor.

cross-system coupling facility (XCF). A component of z/OS that provides functions to support cooperation between authorized programs running within a sysplex.

CTC. The channel-to-channel (CTC) channel can communicate with a CTC on another host for intersystem communication.

Customer Information Control System (CICS). A general-purpose transactional program that controls online communication between terminal users and a database for a large number of end users on a real-time basis.

customization dialogs. The customization dialogs are an ISPF application. They are used to customize the enterprise policy, like, for example, the enterprise resources and the relationships between resources, or the automation policy for systems in the enterprise. How to use these dialogs is described in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

CVC. See converted mode.

D

DataPower X150z. See IBM Websphere DataPower Integration Appliance X150 for zEnterprise (DataPower X150z).

DASD. See direct access storage device.

data services task (DST). The NetView subtask that gathers, records, and manages data in a VSAM file or a network device that contains network management information.

data set. The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access.

data set members. Members of partitioned data sets that are individually named elements of a larger file that can be retrieved by name.

DBCS. See double-byte character set.

DCCF. See disabled console communication facility.

DCF. See Document Composition Facility.

DELAY Report. An RMF report that shows the activity of each job in the system and the hardware and software resources that are delaying each job.

device. A piece of equipment. Devices can be workstations, printers, disk drives, tape units, remote systems or communications controllers. You can see information about all devices attached to a particular switch, and control paths and jobs to devices.

DEVR Report. An RMF report that presents information about the activity of I/O devices that are delaying jobs.

dialog. Interactive 3270 panels.

direct access storage device (DASD). A device that allows storage to be directly accessed, such as a disk drive.

disabled console communication facility (DCCF). A z/OS component that provides limited-function console communication during system recovery situations.

disk operating system (DOS). (1) An operating system for computer systems that use disks and diskettes for auxiliary storage of programs and data. (2) Software for a personal computer that controls the processing of programs. For the IBM Personal Computer, the full name is Personal Computer Disk Operating System (PCDOS).

display. (1) To present information for viewing, usually on the screen of a workstation or on a hardcopy device. (2) Deprecated term for panel.

distribution manager. The component of the NetView program that enables the host system to use, send, and delete files and programs in a network of computers.

Document Composition Facility (DCF). An IBM licensed program used to format input to a printer.

domain. (1) An access method and its application programs, communication controllers, connecting lines, modems, and attached workstations. (2) In SNA, a system services control point (SSCP) and the physical units (PUs), logical units (LUs), links, link stations, and associated resources that the SSCP can control with activation requests and deactivation requests.

double-byte character set (DBCS). A character set, such as Kanji, in which each character is represented by a 2-byte code.

DP enterprise. Data processing enterprise.

DSIPARM. This file is a collection of members of NetView's customization.

DST. Data Services Task.

E

EBCDIC. See Extended Binary Coded Decimal Interchange Code.

ECB. See event control block.

EMCS. Extended multiple console support. See also multiple console support.

ensemble. A collection of one or more zEnterprise nodes (including any attached zBX) that are managed as a single logical virtualized system by the Unified Resource Manager, through the Hardware Management Console.

ensemble member. A zEnterprise node that has been added to an ensemble.

enterprise. The composite of all operational entities, functions, and resources that form the total business concern and that require an information system.

enterprise monitoring. Enterprise monitoring is used by SA z/OS to update the *NetView Management Console (NMC)* resource status information that is stored in the *Resource Object Data Manager (RODM)*. Resource status information is acquired by enterprise monitoring of the *Resource Measurement Facility (RMF) Monitor III* service information at user-defined intervals. SA z/OS stores this information in its operational information base, where it is used to update the information presented to the operator in graphic displays.

Enterprise Systems Architecture (ESA). A hardware architecture that reduces the effort required for managing data sets and extends addressability for system, subsystem, and application functions.

entries. Resources, such as processors, entered on panels.

entry type. Resources, such as processors or applications, used for automation and monitoring.

environment. Data processing enterprise.

error threshold. An automation policy setting that specifies when SA z/OS should stop trying to restart or recover an application, subsystem or component, or offload a data set.

ESA. See Enterprise Systems Architecture.

eServer™. Processor family group designator used by the SA z/OS customization dialogs to define a target hardware as member of the System z or 390-CMOS processor families.

event. (1) In NetView, a record indicating irregularities of operation in physical elements of a network. (2) An occurrence of significance to a task; for example, the completion of an asynchronous operation, such as an input/output operation. (3) Events are part of a trigger condition, such that if all events of a trigger condition have occurred, a startup or shutdown of an application is performed.

event control block (ECB). A control block used to represent the status of an event.

exception condition. An occurrence on a system that is a deviation from normal operation. SA z/OS monitoring highlights exception conditions and allows an SA z/OS enterprise to be managed by exception.

Extended Binary Coded Decimal Interchange Code (EBCDIC). A coded character set of 256 8-bit characters developed for the representation of textual data. See also American Standard Code for Information Interchange.

extended recovery facility (XRF). A facility that minimizes the effect of failures in z/OS, VTAM, the host processor, or high availability applications during sessions between high availability applications and designated terminals. This facility provides an alternate subsystem to take over sessions from the failing subsystem.

F

fallback system. See secondary system.

field. A collection of bytes within a record that are logically related and are processed as a unit.

file manager commands. A set of SA z/OS commands that read data from or write data to the automation control file or the operational information base. These commands are useful in the development of automation that uses SA z/OS facilities.

focal point. In NetView, the focal-point domain is the central host domain. It is the central control point for any management services element containing control of the network management data.

focal point system. (1) A system that can administer, manage, or control one or more target systems. There are a number of different focal point system associated with IBM automation products. (2) **NMC focal point system.** The NMC focal point system is a NetView system with an attached workstation server and LAN that gathers information about the state of the network. This focal point system uses RODM to store the data it collects in the data model. The information stored in RODM can be accessed from any LAN-connected workstation with NetView Management Console installed. (3) **NPDA focal point system.** This is a NetView system that collects all the NPDA alerts that are generated within your enterprise. It is supported by NetView. If you have SA z/OS installed the NPDA focal point system must be the same as your NMC focal point system. The NPDA focal point system is also known as the *alert focal point system*. (4) **SA z/OS Processor Operations focal point system.** This is a NetView system that has SA z/OS host code installed. The SA z/OS Processor Operations focal point system receives messages from the systems and operator consoles of the machines that it controls. It provides full systems and operations console function for its target systems. It can be used to IPL these systems. Note that some restrictions apply to the Hardware Management Console for an S/390 microprocessor cluster. (5) **SA z/OS SDF focal point system.** The SA z/OS SDF focal point system is an SA z/OS NetView system that collects status information from other SA z/OS NetViews within your enterprise. (6) **Status focal point system.** In NetView, the system to which STATMON, VTAM and NLDM send status information on network resources. If you have a NMC focal point, it must be on the same system as the Status focal point. (7) **Hardware Management Console.** Although not listed as a focal point, the Hardware Management Console acts as a focal point for the console functions of an S/390 microprocessor cluster. Unlike all the other focal points in this definition, the Hardware Management Console runs on a LAN-connected workstation,

frame. For a System/390 microprocessor cluster, a frame contains one or two central processor complexes (CPCs), support elements, and AC power distribution.

full-screen mode. In NetView, a form of panel presentation that makes it possible to display the contents of an entire workstation screen at once.

Full-screen mode can be used for fill-in-the-blanks prompting. Contrast with line mode.

G

gateway session. An NetView-NetView Task session with another system in which the SA z/OS outbound gateway operator logs onto the other NetView session without human operator intervention. Each end of a gateway session has both an inbound and outbound gateway operator.

generic alert. Encoded alert information that uses code points (defined by IBM and possibly customized by users or application programs) stored at an alert receiver, such as NetView.

group. A collection of target systems defined through configuration dialogs. An installation might set up a group to refer to a physical site or an organizational or application entity.

group entry. A construct, created with the customization dialogs, used to represent and contain policy for a group.

group entry type. A collection of target systems defined through the customization dialog. An installation might set up a group to refer to a physical site or an organizational entity. Groups can, for example, be of type STANDARD or SYSPLEX.

H

| **Hardware Management Console (HMC).** A user
| interface through which data center personnel
| configure, control, monitor, and manage System z
| hardware and software resources. The HMC
| communicates with each central processor complex
| (CPC) through the Support Element. On an IBM
| zEnterprise 196 (z196), using the Unified Resource
| Manager on the HMCs or Support Elements, personnel
| can also create and manage an ensemble.

Hardware Management Console Application (HWMCA). A direct-manipulation object-oriented graphical user interface that provides a single point of control and single system image for hardware elements. The HWMCA provides grouping support, aggregated and real-time system status using colors, consolidated hardware messages support, consolidated operating system messages support, consolidated service support, and hardware commands targeted at a single system, multiple systems, or a group of systems.

heartbeat. In SA z/OS, a function that monitors the validity of the status forwarding path between remote systems and the NMC focal point, and monitors the availability of remote z/OS systems, to ensure that status information displayed on the SA z/OS workstation is current.

help panel. An online panel that tells you how to use a command or another aspect of a product.

hierarchy. In the NetView program, the resource types, display types, and data types that make up the organization, or levels, in a network.

high-level language (HLL). A programming language that provides some level of abstraction from assembler language and independence from a particular type of machine. For the NetView program, the high-level languages are PL/I and C.

HLL. See high-level language.

host (primary processor). The processor that you enter a command at (also known as the *issuing processor*).

host system. In a coupled system or distributed system environment, the system on which the facilities for centralized automation run. SA z/OS publications refer to target systems or focal-point systems instead of hosts.

HWMCA. See Hardware Management Console Application.

| **Hypervisor.** A program that allows multiple instances
| of operating systems or virtual servers to run
| simultaneously on the same hardware device. A
| hypervisor can run directly on the hardware, can run
| within an operating system, or can be imbedded in
| platform firmware. Examples of hypervisors include
| PR/SM, z/VM, and PowerVM Enterprise Edition.

I

| **IBM blade.** A customer-acquired, customer-installed
| select blade to be managed by IBM zEnterprise Unified
| Resource Manager. One example of an IBM blade is a
| POWER7 blade.

| **IBM Smart Analyzer for DB2 for z/OS.** An optimizer
| that processes certain types of data warehouse queries
| for DB2 for z/OS.

| **IBM System z Application Assist Processor (zAAP).**
| A specialized processor that provides a Java execution
| environment, which enables Java-based web
| applications to be integrated with core z/OS business
| applications and backend database systems.

| **IBM System z Integrated Information Processor (zIIP).** A specialized processor that provides
| computing capacity for selected data and transaction
| processing workloads and for selected network
| encryption workloads.

| **IBM Websphere DataPower Integration Appliance
| X150 for zEnterprise (DataPower X150z).** A
| purpose-built appliance that simplifies, helps secure,
| and optimizes XML and Web services processing.

| **IBM Enterprise 196 (z196).** The newest generation of System z family of servers built on a new processor chip, with enhanced memory function and capacity, security, and on demand enhancements to support existing mainframe workloads and large scale consolidation.

| **IBM zEnterprise BladeCenter Extension (zBX).** A heterogeneous hardware infrastructure that consists of a BladeCenter chassis attached to an IBM zEnterprise 196 (z196). A BladeCenter chassis can contain IBM blades or optimizers.

| **IBM zEnterprise BladeCenter Extension (zBX) blade.** Generic name for all blade types supported in an IBM zEnterprise BladeCenter Extension (zBX). This term includes IBM blades and optimizers.

| **IBM zEnterprise System (zEnterprise).** A heterogeneous hardware infrastructure that can consist of an IBM zEnterprise 196 (z196) and an attached IBM zEnterprise BladeCenter Extension (zBX) Model 002, managed as a single logical virtualized system by the Unified Resource Manager.

| **IBM zEnterprise Unified Resource Manager.** Licensed Internal Code (LIC), also known as firmware, that is part of the Hardware Management Console. The Unified Resource Manager provides energy monitoring and management, goal-oriented policy management, increased security, virtual networking, and data management for the physical and logical resources of a given ensemble.

I/O operations. The part of SA z/OS that provides you with a single point of logical control for managing connectivity in your active I/O configurations. I/O operations takes an active role in detecting unusual conditions and lets you view and change paths between a processor and an I/O device, using dynamic switching (the ESCON director). Also known as I/O Ops.

I/O Ops. See I/O operations.

I/O resource number. Combination of channel path identifier (CHPID), device number, etc. See internal token.

images. A grouping of processors and I/O devices that you define. You can define a single-image mode that allows a multiprocessor system to function as one central processor image.

IMS. See Information Management System.

IMS/VS. See Information Management System/Virtual Storage.

inbound. In SA z/OS, messages sent to the focal-point system from the PC or target system.

inbound gateway operator. The automation operator that receives incoming messages, commands, and responses from the outbound gateway operator at the sending system. The inbound gateway operator handles communications with other systems using a gateway session.

Information Management System (IMS). Any of several system environments available with a database manager and transaction processing that are capable of managing complex databases and terminal networks.

Information Management System/Virtual Storage (IMS/VS). A database/data communication (DB/DC) system that can manage complex databases and networks. Synonymous with Information Management System.

INGEIO PROC. The I/O operations default procedure name. It is part of the SYS1.PROCLIB.

initial microprogram load. The action of loading microprograms into computer storage.

initial program load (IPL). (1) The initialization procedure that causes an operating system to commence operation. (2) The process by which a configuration image is loaded into storage at the beginning of a workday or after a system malfunction. (3) The process of loading system programs and preparing a system to run jobs.

initialize automation. SA z/OS-provided automation that issues the correct z/OS start command for each subsystem when SA z/OS is initialized. The automation ensures that subsystems are started in the order specified in the automation control files and that prerequisite applications are functional.

input/output configuration data set (IOCDs). A configuration definition built by the I/O configuration program (IOCP) and stored on disk files associated with the processor controller.

input/output support processor (IOSP). The hardware unit that provides I/O support functions for the primary support processor and maintenance support functions for the processor controller.

Interactive System Productivity Facility (ISPF). An IBM licensed program that serves as a full-screen editor and dialog manager. Used for writing application programs, it provides a means of generating standard screen panels and interactive dialogs between the application programmer and the terminal user. See also Time Sharing Option.

interested operator list. The list of operators who are to receive messages from a specific target system.

internal token. A *logical token* (LTOK); name by which the I/O resource or object is known; stored in IODF.

IOCDs. See input/output configuration data set.

IOSP. See input/output support processor..

IPL. See initial program load.

ISPF. See Interactive System Productivity Facility.

ISPF console. You log on to ISPF from this 3270-type console to use the runtime panels for I/O operations and SA z/OS customization panels.

issuing host. The base program that you enter a command for processing with. See primary host.

J

JCL. See job control language.

JES. See job entry subsystem.

JES2. An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In an installation with more than one processor, each JES2 processor independently controls its job input, scheduling, and output processing. See also job entry subsystem and JES3

JES3. An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In complexes that have several loosely coupled processing units, the JES3 program manages processors so that the global processor exercises centralized control over the local processors and distributes jobs to them using a common job queue. See also job entry subsystem and JES2.

job. (1) A set of data that completely defines a unit of work for a computer. A job usually includes all necessary computer programs, linkages, files, and instructions to the operating system. (2) An address space.

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.

job entry subsystem (JES). An IBM licensed program that receives jobs into the system and processes all output data that is produced by jobs. In SA z/OS publications, JES refers to JES2 or JES3, unless otherwise stated. See also JES2 and JES3.

K

Kanji. An ideographic character set used in Japanese. See also double-byte character set.

L

LAN. See local area network.

line mode. A form of screen presentation in which the information is presented a line at a time in the message area of the terminal screen. Contrast with full-screen mode.

link. (1) In SNA, the combination of the link connection and the link stations joining network nodes; for example, a System/370 channel and its associated protocols, a serial-by-bit connection under the control of synchronous data link control (SDLC). See synchronous data link control. (2) In SA z/OS, link connection is the physical medium of transmission.

link-attached. Describes devices that are physically connected by a telecommunication line. Contrast with channel-attached.

Linux on System z. UNIX-like open source operating system conceived by Linus Torvalds and developed across the internet.

local. Pertaining to a device accessed directly without use of a telecommunication line. Synonymous with channel-attached.

local area network (LAN). (1) A network in which a set of devices is connected for communication. They can be connected to a larger network. See also token ring. (2) A network that connects several devices in a limited area (such as a single building or campus) and that can be connected to a larger network.

logical partition (LP). A subset of the processor hardware that is defined to support an operating system. See also logically partitioned mode.

logical switch number (LSN). Assigned with the switch parameter of the CHPID macro of the IOCP.

logical token (LTOK). Resource number of an object in the IODF.

logical unit (LU). In SNA, a port through which an end user accesses the SNA network and the functions provided by system services control points (SSCPs). An LU can support at least two sessions, one with an SSCP and one with another LU, and may be capable of supporting many sessions with other LUs. See also physical unit and system services control point.

logical unit 6.2 (LU 6.2). A type of logical unit that supports general communications between programs in a distributed processing environment. LU 6.2 is characterized by:

- A peer relationship between session partners
- Efficient use of a session for multiple transactions
- A comprehensive end-to-end error processing

- A generic application program interface (API) consisting of structured verbs that are mapped to a product implementation

Synonym for advanced program-to-program communication.

logically partitioned (LPAR) mode. A central processor mode that enables an operator to allocate system processor hardware resources among several logical partitions. Contrast with basic mode.

LOGR. The sysplex logger.

LP. See logical partition.

LPAR. See logically partitioned mode.

LSN. See logical switch number.

LU. See logical unit.

LU 6.2. See logical unit 6.2.

LU 6.2 session. A session initiated by VTAM on behalf of an LU 6.2 application program, or a session initiated by a remote LU in which the application program specifies that VTAM is to control the session by using the APPCCMD macro. See logical unit 6.2.

LU-LU session. In SNA, a session between two logical units (LUs) in an SNA network. It provides communication between two end users, or between an end user and an LU services component.

M

MAT. Deprecated term for NetView automation table.

MCA. See Micro Channel architecture.

MCS. See multiple console support.

member. A specific function (one or more modules or routines) of a multisystem application that is defined to XCF and assigned to a group by the multisystem application. A member resides on one system in the sysplex and can use XCF services to communicate (send and receive data) with other members of the same group.

message automation table (MAT). Deprecated term for NetView automation table.

message class. A number that SA z/OS associates with a message to control routing of the message. During automated operations, the classes associated with each message issued by SA z/OS are compared to the classes assigned to each notification operator. Any operator with a class matching one of the message's classes receives the message.

message forwarding. The SA z/OS process of sending messages generated at an SA z/OS target system to the SA z/OS focal-point system.

message group. Several messages that are displayed together as a unit.

message monitor task. A task that starts and is associated with a number of communications tasks. Message monitor tasks receive inbound messages from a communications task, determine the originating target system, and route the messages to the appropriate target control tasks.

message processing facility (MPF). A z/OS table that screens all messages sent to the z/OS console. The MPF compares these messages with a customer-defined list of messages on which to automate, suppress from the z/OS console display, or both, and marks messages to automate or suppress. Messages are then broadcast on the subsystem interface (SSI).

message suppression. The ability to restrict the amount of message traffic displayed on the z/OS console.

Micro Channel architecture. The rules that define how subsystems and adapters use the Micro Channel bus in a computer. The architecture defines the services that each subsystem can or must provide.

microprocessor. A processor implemented on one or a small number of chips.

migration. Installation of a new version or release of a program to replace an earlier version or release.

MP. Multiprocessor.

MPF. See message processing facility.

MPFLSTSA. The MPFLST member that is built by SA z/OS.

multi-MVS environment. physical processing system that is capable of operating more than one MVS image. See also MVS image.

multiple console support (MCS). A feature of MVS that permits selective message routing to multiple consoles.

Multiple Virtual Storage (MVS). An IBM operating system that accesses multiple address spaces in virtual storage. The predecessor of z/OS.

multiprocessor (MP). A CPC that can be physically partitioned to form two operating processor complexes.

multisystem application. An application program that has various functions distributed across z/OS images in a multisystem environment.

multisystem environment. An environment in which two or more systems reside on one or more processors. Or one or more processors can communicate with programs on the other systems.

MVS. See Multiple Virtual Storage.

MVS image. A single occurrence of the MVS operating system that has the ability to process work. See also multi-MVS environment and single-MVS environment.

MVS/ESA. Multiple Virtual Storage/Enterprise Systems Architecture. See z/OS.

MVS/JES2. Multiple Virtual Storage/Job Entry System 2. A z/OS subsystem that receives jobs into the system, converts them to an internal format, selects them for execution, processes their output, and purges them from the system. In an installation with more than one processor, each JES2 processor independently controls its job input, scheduling, and output processing.

N

NAU. (1) See network addressable unit. (2) See network accessible unit.

NCCF. See Network Communications Control Facility.

NCP. (1) See network control program (general term). (2) See Network Control Program (an IBM licensed program). Its full name is Advanced Communications Function for the Network Control Program. Synonymous with ACF/NCP.

NCP/token ring interconnection. A function used by ACF/NCP to support token ring-attached SNA devices. NTRI also provides translation from token ring-attached SNA devices (PUs) to switched (dial-up) devices.

NetView. An IBM licensed program used to monitor a network, manage it, and diagnose network problems. NetView consists of a command facility that includes a presentation service, command processors, automation based on command lists, and a transaction processing structure on which the session monitor, hardware monitor, and terminal access facility (TAF) network management applications are built.

NetView (NCCF) console. A 3270-type console for NetView commands and runtime panels for system operations and processor operations.

NetView automation procedures. A sequence of commands, packaged as a NetView command list or a command processor written in a high-level language. An automation procedure performs automation functions and runs under the NetView program.

NetView automation table (AT). A table against which the NetView program compares incoming

messages. A match with an entry triggers the specified response. SA z/OS entries in the NetView automation table trigger an SA z/OS response to target system conditions. Formerly known as the message automation table (MAT).

NetView command list language. An interpretive language unique to NetView that is used to write command lists.

NetView Graphic Monitor Facility (NGMF). Deprecated term for NetView Management Console.

NetView hardware monitor. The component of NetView that helps identify network problems, such as hardware, software, and microcode, from a central control point using interactive display techniques. Formerly called *network problem determination application*.

NetView log. The log that NetView records events relating to NetView and SA z/OS activities in.

NetView Management Console (NMC). A function of the NetView program that provides a graphic, topological presentation of a network that is controlled by the NetView program. It provides the operator different views of a network, multiple levels of graphical detail, and dynamic resource status of the network. This function consists of a series of graphic windows that allows you to manage the network interactively. Formerly known as the NetView Graphic Monitor Facility (NGMF).

NetView message table. See NetView automation table.

NetView paths via logical unit (LU 6.2). A type of network-accessible port (VTAM connection) that enables end users to gain access to SNA network resources and communicate with each other. LU 6.2 permits communication between processor operations and the workstation. See logical unit 6.2.

NetView-NetView task (NNT). The task that a cross-domain NetView operator session runs under. Each NetView program must have a NetView-NetView task to establish one NNT session. See also operator station task.

NetView-NetView task session. A session between two NetView programs that runs under a NetView-NetView task. In SA z/OS, NetView-NetView task sessions are used for communication between focal point and remote systems.

network. (1) An interconnected group of nodes. (2) In data processing, a user application network. See SNA network.

network accessible unit (NAU). In SNA networking, any device on the network that has a network address, including a logical unit (LU), physical unit (PU), control point (CP), or system services control point (SSCP). It is

the origin or the destination of information transmitted by the path control network. Synonymous with network addressable unit.

network addressable unit (NAU). Synonym for network accessible unit.

Network Communications Control Facility (NCCF). The operations control facility for the network. NCCF consists of a presentation service, command processors, automation based on command lists, and a transaction processing structure on which the network management applications NLDM and NPDA are built. NCCF is a precursor to the NetView command facility.

Network Control Program (NCP). An IBM licensed program that provides communication controller support for single-domain, multiple-domain, and interconnected network capability. Its full name is Advanced Communications Function for the Network Control Program.

network control program (NCP). (1) A program that controls the operation of a communication controller. (2) A program used for requests and responses exchanged between physical units in a network for data flow control.

Network Problem Determination Application (NPDA). An NCCF application that helps you identify network problems, such as hardware, software, and microcode, from a central control point using interactive display methods. The alert manager for the network. The precursor of the NetView hardware monitor.

Networking NetView. In SA z/OS the NetView that performs network management functions, such as managing the configuration of a network. In SA z/OS it is common to also route alerts to the Networking NetView.

NGMF. Deprecated term for NetView Management Console.

NGMF focal-point system. Deprecated term for NMC focal point system.

NIP. See nucleus initialization program.

NMC focal point system. See focal point system

NMC workstation. The NMC workstation is the primary way to dynamically monitor SA z/OS systems. From the windows, you see messages, monitor status, view trends, and react to changes before they cause problems for end users. You can use multiple windows to monitor multiple views of the system.

NNT. See NetView-NetView task.

notification message. An SA z/OS message sent to a human notification operator to provide information

about significant automation actions. Notification messages are defined using the customization dialogs.

notification operator. A NetView console operator who is authorized to receive SA z/OS notification messages. Authorization is made through the customization dialogs.

NPDA. See Network Problem Determination Application.

NPDA focal-point system. See focal point system.

NTRI. See NCP/token ring interconnection.

nucleus initialization program (NIP). The program that initializes the resident control program; it allows the operator to request last-minute changes to certain options specified during system generation.

O

objective value. An average Workflow or Using value that SA z/OS can calculate for applications from past service data. SA z/OS uses the objective value to calculate warning and alert thresholds when none are explicitly defined.

OCA. In SA z/OS, operator console A, the active operator console for a target system. Contrast with OCB.

OCB. In SA z/OS, operator console B, the backup operator console for a target system. Contrast with OCA.

OCF. See operations command facility.

OCF-based processor. A central processor complex that uses an operations command facility for interacting with human operators or external programs to perform operations management functions on the CPC.

OPC/A. See Operations Planning and Control/Advanced.

OPC/ESA. See Operations Planning and Control/Enterprise Systems Architecture.

Open Systems Adapter (OSA). I/O operations can display the Open System Adapter (OSA) channel logical definition, physical attachment, and status. You can configure an OSA channel on or off.

operating system (OS). Software that controls the execution of programs and that may provide services such as resource allocation, scheduling, input/output control, and data management. Although operating systems are predominantly software, partial hardware implementations are possible. (T)

operations. The real-time control of a hardware device or software function.

operations command facility (OCF). A facility of the central processor complex that accepts and processes operations management commands.

Operations Planning and Control/Advanced (OPC/A). A set of IBM licensed programs that automate, plan, and control batch workload. OPC/A analyzes system and workload status and submits jobs accordingly.

Operations Planning and Control/Enterprise Systems Architecture (OPC/ESA). A set of IBM licensed programs that automate, plan, and control batch workload. OPC/ESA analyzes system and workload status and submits jobs accordingly. The successor to OPC/A.

operator. (1) A person who keeps a system running. (2) A person or program responsible for managing activities controlled by a given piece of software such as z/OS, the NetView program, or IMS. (3) A person who operates a device. (4) In a language statement, the lexical entity that indicates the action to be performed on operands.

operator console. (1) A functional unit containing devices that are used for communications between a computer operator and a computer. (T) (2) A display console used for communication between the operator and the system, used primarily to specify information concerning application programs and I/O operations and to monitor system operation. (3) In SA z/OS, a console that displays output from and sends input to the operating system (z/OS, LINUX, VM, VSE). Also called *operating system console*. In the SA z/OS operator commands and configuration dialogs, OC is used to designate a target system operator console.

operator station task (OST). The NetView task that establishes and maintains the online session with the network operator. There is one operator station task for each network operator who logs on to the NetView program.

operator view. A set of group, system, and resource definitions that are associated together for monitoring purposes. An operator view appears as a graphic display in the graphical interface showing the status of the defined groups, systems, and resources.

OperatorView entry. A construct, created with the customization dialogs, used to represent and contain policy for an operator view.

| **optimizer.** A special-purpose hardware component or
| appliance that can perform a limited set of specific
| functions with optimized performance when compared
| to a general-purpose processor. Because of its limited
| set of functions, an optimizer is an integrated part of a
| processing environment, rather than a stand-alone unit.
| One example of an optimizer is the IBM Smart
| Analytics Optimizer for DB2 for z/OS.

OS. See operating system.

OSA. See Open Systems Adapter.

OST. See operator station task.

outbound. In SA z/OS, messages or commands from the focal-point system to the target system.

outbound gateway operator. The automation operator that establishes connections to other systems. The outbound gateway operator handles communications with other systems through a gateway session. The automation operator sends messages, commands, and responses to the inbound gateway operator at the receiving system.

P

page. (1) The portion of a panel that is shown on a display surface at one time. (2) To transfer instructions, data, or both between real storage and external page or auxiliary storage.

panel. (1) A formatted display of information that appears on a terminal screen. Panels are full-screen 3270-type displays with a monospaced font, limited color and graphics. (2) By using SA z/OS panels you can see status, type commands on a command line using a keyboard, configure your system, and passthru to other consoles. See also help panel. (3) In computer graphics, a display image that defines the locations and characteristics of display fields on a display surface. Contrast with screen.

parallel channels. Parallel channels operate in either byte (BY) or block (BL) mode. You can change connectivity to a parallel channel operating in block mode.

parameter. (1) A variable that is given a constant value for a specified application and that may denote the application. (2) An item in a menu for which the user specifies a value or for which the system provides a value when the menu is interpreted. (3) Data passed to a program or procedure by a user or another program, specifically as an operand in a language statement, as an item in a menu, or as a shared data structure.

partition. (1) A fixed-size division of storage. (2) In VSE, a division of the virtual address area that is available for program processing. (3) On an IBM Personal Computer fixed disk, one of four possible storage areas of variable size; one can be accessed by DOS, and each of the others may be assigned to another operating system.

partitionable CPC. A CPC that can be divided into 2 independent CPCs. See also physical partition, single-image mode, MP, and side.

partitioned data set (PDS). A data set in direct access storage that is divided into partitions, called *members*, each of which can contain a program, part of a program, or data.

passive monitoring. In SA z/OS, the receiving of unsolicited messages from z/OS systems and their resources. These messages can prompt updates to resource status displays. See also active monitoring

PCE. A processor controller. Also known as the support processor or service processor in some processor families.

PDB. See policy database.

PDS. See partitioned data set.

physical partition. Part of a CPC that operates as a CPC in its own right, with its own copy of the operating system.

physical unit (PU). In SNA, the component that manages and monitors the resources (such as attached links and adjacent link stations) of a node, as requested by a system services control point (SSCP) through an SSCP-PU session. An SSCP activates a session with the physical unit to indirectly manage, through the PU, resources of the node such as attached links.

physically partitioned (PP) configuration. A mode of operation that allows a multiprocessor (MP) system to function as two or more independent CPCs having separate power, water, and maintenance boundaries. Contrast with single-image mode.

POI. See program operator interface.

policy. The automation and monitoring specifications for an SA z/OS enterprise. See *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

policy database. The automation definitions (automation policy) that the automation programmer specifies using the customization dialog is stored in the policy database. Also known as the PDB. See also automation policy.

POR. See power-on reset.

port. (1) System hardware that the I/O devices are attached to. (2) In an ESCON switch, a port is an addressable connection. The switch routes data through the ports to the channel or control unit. Each port has a name that can be entered into a switch matrix, and you can use commands to change the switch configuration. (3) An access point (for example, a logical unit) for data entry or exit. (4) A functional unit of a node that data can enter or leave a data network through. (5) In data communication, that part of a data processor that is dedicated to a single data channel for the purpose of receiving data from or transmitting data to one or more external, remote devices.

power-on reset (POR). A function that re-initializes all the hardware in a CPC and loads the internal code that enables the CPC to load and run an operating system. See initial microprogram load.

PP. See physical partition.

PPI. See program to program interface.

PPT. See primary POI task.

PR/SM. See Processor Resource/Systems Manager.

primary host. The base program that you enter a command for processing at.

primary POI task (PPT). The NetView subtask that processes all unsolicited messages received from the VTAM program operator interface (POI) and delivers them to the controlling operator or to the command processor. The PPT also processes the initial command specified to execute when NetView is initialized and timer request commands scheduled to execute under the PPT.

primary system. A system is a primary system for an application if the application is normally meant to be running there. SA z/OS starts the application on all the primary systems defined for it.

problem determination. The process of determining the source of a problem; for example, a program component, machine failure, telecommunication facilities, user or contractor-installed programs or equipment, environment failure such as a power loss, or user error.

processor. (1) A device for processing data from programmed instructions. It may be part of another unit. (2) In a computer, the part that interprets and executes instructions. Two typical components of a processor are a control unit and an arithmetic logic unit.

processor controller. Hardware that provides support and diagnostic functions for the central processors.

processor operations. The part of SA z/OS that monitors and controls processor (hardware) operations. Processor operations provides a connection from a focal-point system to a target system. Through NetView on the focal-point system, processor operations automates operator and system consoles for monitoring and recovering target systems. Also known as ProcOps.

Processor Resource/Systems Manager™ (PR/SM). The feature that allows the processor to use several operating system images simultaneously and provides logical partitioning capability. See also logically partitioned mode.

ProcOps. See processor operations.

ProcOps Service Machine (PSM). The PSM is a CMS user on a VM host system. It runs a CMS multitasking application that serves as "virtual hardware" for ProcOps. ProcOps communicates via the PSM with the VM guest systems that are defined as target systems within ProcOps.

product automation. Automation integrated into the base of SA z/OS for the products CICS, DB2, IMS, TWS (formerly called *features*).

program operator interface (POI). A NetView facility for receiving VTAM messages.

program to program interface (PPI). A NetView function that allows user programs to send or receive data buffers from other user programs and to send alerts to the NetView hardware monitor from system and application programs.

protocol. In SNA, the meanings of, and the sequencing rules for, requests and responses used for managing the network, transferring data, and synchronizing the states of network components.

proxy resource. A resource defined like an entry type APL representing a processor operations target system.

PSM. See ProcOps Service Machine.

PU. See physical unit.

R

RACF. See Resource Access Control Facility.

remote system. A system that receives resource status information from an SA z/OS focal-point system. An SA z/OS remote system is defined as part of the same SA z/OS enterprise as the SA z/OS focal-point system to which it is related.

requester. A workstation from that user can log on to a domain from, that is, to the servers belonging to the domain, and use network resources. Users can access the shared resources and use the processing capability of the servers, thus reducing hardware investment.

resource. (1) Any facility of the computing system or operating system required by a job or task, and including main storage, input/output devices, the processing unit, data sets, and control or processing programs. (2) In NetView, any hardware or software that provides function to the network. (3) In SA z/OS, any z/OS application, z/OS component, job, device, or target system capable of being monitored or automated through SA z/OS.

Resource Access Control Facility (RACF). A program that can provide data security for all your resources. RACF protects data from accidental or deliberate unauthorized disclosure, modification, or destruction.

resource group. A physically partitionable portion of a processor. Also known as a *side*.

Resource Measurement Facility (RMF). A feature of z/OS that measures selected areas of system activity and presents the data collected in the format of printed reports, System Management Facility (SMF) records, or display reports.

Resource Object Data Manager (RODM). In NetView for z/OS, a component that provides an in-memory cache for maintaining real-time data in an address space that is accessible by multiple applications. RODM also allows an application to query an object and receive a rapid response and act on it.

resource token. A unique internal identifier of an ESCON resource or resource number of the object in the IODF.

restart automation. Automation provided by SA z/OS that monitors subsystems to ensure that they are running. If a subsystem fails, SA z/OS attempts to restart it according to the policy in the automation configuration file.

Restructured Extended Executor (REXX). A general-purpose, high-level, programming language, particularly suitable for EXEC procedures or programs for personal computing, used to write command lists.

return code. A code returned from a program used to influence the issuing of subsequent instructions.

REXX. See Restructured Extended Executor.

REXX procedure. A command list written with the Restructured Extended Executor (REXX), which is an interpretive language.

RMF. See Resource Measurement Facility.

RODM. See Resource Object Data Manager.

S

SAF. See Security Authorization Facility.

SA IOM. See System Automation for Integrated Operations Management.

SAPlex. SAPlex or "SA z/OS Subplex" is a term used in conjunction with a sysplex. In fact, a SAPlex is a subset of a sysplex. However, it can also be a sysplex. For a detailed description, refer to "Using SA z/OS Subplexes" in *IBM Tivoli System Automation for z/OS Planning and Installation*.

SA z/OS. See System Automation for z/OS.

SA z/OS customization dialogs. An ISPF application through which the SA z/OS policy administrator

defines policy for individual z/OS systems and builds automation control data and RODM load function files.

SA z/OS customization focal point system. See focal point system.

SA z/OS data model. The set of objects, classes and entity relationships necessary to support the function of SA z/OS and the NetView automation platform.

SA z/OS enterprise. The group of systems and resources defined in the customization dialogs under one enterprise name. An SA z/OS enterprise consists of connected z/OS systems running SA z/OS.

SA z/OS focal point system. See focal point system.

SA z/OS policy. The description of the systems and resources that make up an SA z/OS enterprise, together with their monitoring and automation definitions.

SA z/OS policy administrator. The member of the operations staff who is responsible for defining SA z/OS policy.

SA z/OS satellite. If you are running two NetViews on an z/OS system to split the automation and networking functions of NetView, it is common to route alerts to the Networking NetView. For SA z/OS to process alerts properly on the Networking NetView, you must install a subset of SA z/OS code, called an *SA z/OS satellite* on the Networking NetView.

SA z/OS SDF focal point system. See focal point system.

SCA. In SA z/OS, system console A, the active system console for a target hardware. Contrast with SCB.

SCB. In SA z/OS, system console B, the backup system console for a target hardware. Contrast with SCA.

screen. Deprecated term for panel.

screen handler. In SA z/OS, software that interprets all data to and from a full-screen image of a target system. The interpretation depends on the format of the data on the full-screen image. Every processor and operating system has its own format for the full-screen image. A screen handler controls one PS/2 connection to a target system.

SDF. See status display facility.

SDLC. See synchronous data link control.

SDSF. See System Display and Search Facility.

secondary system. A system is a secondary system for an application if it is defined to automation on that system, but the application is not normally meant to be

running there. Secondary systems are systems to which an application can be moved in the event that one or more of its primary systems are unavailable. SA z/OS does not start the application on its secondary systems.

Security Authorization Facility (SAF). An MVS interface with which programs can communicate with an external security manager, such as RACF.

server. A server is a workstation that shares resources, which include directories, printers, serial devices, and computing powers.

service language command (SLC). The line-oriented command language of processor controllers or service processors.

service period. Service periods allow the users to schedule the availability of applications. A service period is a set of time intervals (service windows), during which an application should be active.

service processor (SVP). The name given to a processor controller on smaller System/370 processors.

service threshold. An SA z/OS policy setting that determines when to notify the operator of deteriorating service for a resource. See also alert threshold and warning threshold.

session. In SNA, a logical connection between two network addressable units (NAUs) that can be activated, tailored to provide various protocols, and deactivated, as requested. Each session is uniquely identified in a transmission header by a pair of network addresses identifying the origin and destination NAUs of any transmissions exchanged during the session.

session monitor. The component of the NetView program that collects and correlates session-related data and provides online access to this information. The successor to NLDM.

shutdown automation. SA z/OS-provided automation that manages the shutdown process for subsystems by issuing shutdown commands and responding to prompts for additional information.

side. A part of a partitionable CPC that can run as a physical partition and is typically referred to as the A-side or the B-side.

Simple Network Management Protocol (SNMP). A set of protocols for monitoring systems and devices in complex networks. Information about managed devices is defined and stored in a Management Information Base (MIB).

single image. A processor system capable of being physically partitioned that has not been physically partitioned. Single-image systems can be target hardware processors.

single-MVS environment. An environment that supports one MVS image. See also MVS image.

single-image (SI) mode. A mode of operation for a multiprocessor (MP) system that allows it to function as one CPC. By definition, a uniprocessor (UP) operates in single-image mode. Contrast with physically partitioned (PP) configuration.

SLC. See service language command.

SMP/E. See System Modification Program/Extended.

SNA. See Systems Network Architecture.

SNA network. In SNA, the part of a user-application network that conforms to the formats and protocols of systems network architecture. It enables reliable transfer of data among end users and provides protocols for controlling the resources of various network configurations. The SNA network consists of network addressable units (NAUs), boundary function components, and the path control network.

SNMP. See Simple Network Management Protocol.

solicited message. An SA z/OS message that directly responds to a command. Contrast with unsolicited message.

SSCP. See system services control point.

SSI. See subsystem interface.

start automation. SA z/OS-provided automation that manages and completes the startup process for subsystems. During this process, SA z/OS replies to prompts for additional information, ensures that the startup process completes within specified time limits, notifies the operator of problems, if necessary, and brings subsystems to an UP (or ready) state.

startup. The point in time that a subsystem or application is started.

status. The measure of the condition or availability of the resource.

status display facility (SDF). The system operations part of SA z/OS that displays status of resources such as applications, gateways, and write-to-operator messages (WTORs) on dynamic color-coded panels. SDF shows spool usage problems and resource data from multiple systems.

status focal-point system. See focal point system.

steady state automation. The routine monitoring, both for presence and performance, of subsystems, applications, volumes and systems. Steady state automation may respond to messages, performance exceptions and discrepancies between its model of the system and reality.

structure. A construct used by z/OS to map and manage storage on a coupling facility.

subgroup. A named set of systems. A subgroup is part of an SA z/OS enterprise definition and is used for monitoring purposes.

SubGroup entry. A construct, created with the customization dialogs, used to represent and contain policy for a subgroup.

| **subplex.** See SAplex.

subsystem. (1) A secondary or subordinate system, usually capable of operating independent of, or asynchronously with, a controlling system. (2) In SA z/OS, an z/OS application or subsystem defined to SA z/OS.

subsystem interface (SSI). The z/OS interface over which all messages sent to the z/OS console are broadcast.

support element. A hardware unit that provides communications, monitoring, and diagnostic functions to a central processor complex (CPC).

support processor. Another name given to a processor controller on smaller System/370 processors. See service processor.

SVP. See service processor.

switch identifier. The switch device number (swchdevn), the logical switch number (LSN) and the switch name

switches. ESCON directors are electronic units with ports that dynamically switch to route data to I/O devices. The switches are controlled by I/O operations commands that you enter on a workstation.

symbolic destination name (SDN). Used locally at the workstation to relate to the VTAM application name.

synchronous data link control (SDLC). A discipline for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges may be duplex or half-duplex over switched or nonswitched links. The configuration of the link connection may be point-to-point, multipoint, or loop. SDLC conforms to subsets of the Advanced Data Communication Control Procedures (ADCCP) of the American National Standards Institute and High-Level Data Link Control (HDLC) of the International Standards Organization.

SYSINFO Report. An RMF report that presents an overview of the system, its workload, and the total number of jobs using resources or delayed for resources.

SysOps. See system operations.

sysplex. A set of z/OS systems communicating and cooperating with each other through certain multisystem hardware components (coupling devices and timers) and software services (couple data sets).

In a sysplex, z/OS provides the coupling services that handle the messages, data, and status for the parts of a multisystem application that has its workload spread across two or more of the connected processors, sysplex timers, coupling facilities, and couple data sets (which contains policy and states for automation).

A Parallel Sysplex® is a sysplex that includes a coupling facility.

sysplex application group. A sysplex application group is a grouping of applications that can run on any system in a sysplex.

sysplex couple data set. A couple data set that contains sysplex-wide data about systems, groups, and members that use XCF services. All z/OS systems in a sysplex must have connectivity to the sysplex couple data set. See also couple data set.

Sysplex Timer. An IBM unit that synchronizes the time-of-day (TOD) clocks in multiple processors or processor sides. External Time Reference (ETR) is the z/OS generic name for the IBM Sysplex Timer (9037).

system. In SA z/OS, system means a focal point system (z/OS) or a target system (MVS, VM, VSE, LINUX, or CF).

System Automation for Integrated Operations

Management. (1) An outboard automation solution for secure remote access to mainframe/distributed systems. Tivoli System Automation for Integrated Operations Management, previously Tivoli AF/REMOTE, allows users to manage mainframe and distributed systems from any location. (2) The full name for SA IOM.

System Automation for OS/390. The full name for SA OS/390, the predecessor to System Automation for z/OS.

System Automation for z/OS. The full name for SA z/OS.

system console. (1) A console, usually having a keyboard and a display screen, that is used by an operator to control and communicate with a system. (2) A logical device used for the operation and control of hardware functions (for example, IPL, alter/display, and reconfiguration). The system console can be assigned to any of the physical displays attached to a processor controller or support processor. (3) In SA z/OS, the hardware system console for processor controllers or service processors of processors connected using SA z/OS. In the SA z/OS operator commands and configuration dialogs, SC is used to designate the system console for a target hardware processor.

System Display and Search Facility (SDSF). An IBM licensed program that provides information about jobs, queues, and printers running under JES2 on a series of panels. Under SA z/OS you can select SDSF from a pull-down menu to see the resources' status, view the z/OS system log, see WTOR messages, and see active jobs on the system.

System entry. A construct, created with the customization dialogs, used to represent and contain policy for a system.

System Modification Program/Extended (SMP/E). An IBM licensed program that facilitates the process of installing and servicing an z/OS system.

system operations. The part of SA z/OS that monitors and controls system operations applications and subsystems such as NetView, SDSF, JES, RMF, TSO, RODM, ACF/VTAM, CICS, IMS, and OPC. Also known as SysOps.

system services control point (SSCP). In SNA, the focal point within an SNA network for managing the configuration, coordinating network operator and problem determination requests, and providing directory support and other session services for end users of the network. Multiple SSCPs, cooperating as peers, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its domain.

System/390 microprocessor cluster. A configuration that consists of central processor complexes (CPCs) and may have one or more integrated coupling facilities.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks.

T

TAF. See terminal access facility.

target. A processor or system monitored and controlled by a focal-point system.

target control task. In SA z/OS, target control tasks process commands and send data to target systems and workstations through communications tasks. A target control task (a NetView autotask) is assigned to a target system when the target system is initialized.

target hardware. In SA z/OS, the physical hardware on which a target system runs. It can be a single-image or physically partitioned processor. Contrast with target system.

target system. (1) In a distributed system environment, a system that is monitored and controlled by the focal-point system. Multiple target systems can be controlled by a single focal-point system. (2) In SA z/OS, a computer system attached to the focal-point system for monitoring and control. The definition of a target system includes how remote sessions are established, what hardware is used, and what operating system is used.

task. (1) A basic unit of work to be accomplished by a computer. (2) In the NetView environment, an operator station task (logged-on operator), automation operator (autotask), application task, or user task. A NetView task performs work in the NetView environment. All SA z/OS tasks are NetView tasks. See also message monitor task, and target control task.

telecommunication line. Any physical medium, such as a wire or microwave beam, that is used to transmit data.

terminal access facility (TAF). (1) A NetView function that allows you to log onto multiple applications either on your system or other systems. You can define TAF sessions in the SA z/OS customization panels so you don't have to set them up each time you want to use them. (2) In NetView, a facility that allows a network operator to control a number of subsystems. In a full-screen or operator control session, operators can control any combination of subsystems simultaneously.

terminal emulation. The capability of a microcomputer or personal computer to operate as if it were a particular type of terminal linked to a processing unit to access data.

threshold. A value that determines the point at which SA z/OS automation performs a predefined action. See alert threshold, warning threshold, and error threshold.

time of day (TOD). Typically refers to the time-of-day clock.

Time Sharing Option (TSO). An optional configuration of the operating system that provides conversational time sharing from remote stations. It is an interactive service on z/OS, MVS/ESA, and MVS/XA.

Time-Sharing Option/Extended (TSO/E). An option of z/OS that provides conversational timesharing from remote terminals. TSO/E allows a wide variety of users to perform many different kinds of tasks. It can handle short-running applications that use fewer sources as well as long-running applications that require large amounts of resources.

timers. A NetView command that issues a command or command processor (list of commands) at a specified time or time interval.

Tivoli Workload Scheduler (TWS). A family of IBM licensed products that plan, execute and track jobs on several platforms and environments. The successor to OPC/A.

TOD. Time of day.

token ring. A network with a ring topology that passes tokens from one attaching device to another; for example, the IBM Token-Ring Network product.

TP. See transaction program.

transaction program. In the VTAM program, a program that performs services related to the processing of a transaction. One or more transaction programs may operate within a VTAM application program that is using the VTAM application program interface (API). In that situation, the transaction program would request services from the applications program using protocols defined by that application program. The application program, in turn, could request services from the VTAM program by issuing the APPCCMD macro instruction.

transitional automation. The actions involved in starting and stopping subsystems and applications that have been defined to SA z/OS. This can include issuing commands and responding to messages.

translating host. Role played by a host that turns a resource number into a token during a unification process.

trigger. Triggers, in combination with events and service periods, are used to control the starting and stopping of applications in a single system or a parallel sysplex.

TSO. See Time Sharing Option.

TSO console. From this 3270-type console you are logged onto TSO or ISPF to use the runtime panels for I/O operations and SA z/OS customization panels.

TSO/E. See Time-Sharing Option/Extended.

TWS. See Tivoli Workload Scheduler.

U

UCB. See unit control block.

unit control block (UCB). A control block in common storage that describes the characteristics of a particular I/O device on the operating system and that is used for allocating devices and controlling I/O operations.

unsolicited message. An SA z/OS message that is not a direct response to a command.

user task. An application of the NetView program defined in a NetView TASK definition statement.

Using. An RMF Monitor III definition. Jobs getting service from hardware resources (processors or devices) are **using** these resources. The use of a resource by an address space can vary from 0% to 100% where 0% indicates no use during a Range period, and 100% indicates that the address space was found using the resource in every sample during that period.

V

view. In the NetView Graphic Monitor Facility, a graphical picture of a network or part of a network. A view consists of nodes connected by links and may also include text and background lines. A view can be displayed, edited, and monitored for status information about network resources.

| **Virtual Server.** A logical construct that appears to
| comprise processor, memory, and I/O resources
| conforming to a particular architecture. A virtual server
| can support an operating system, associated
| middleware, and applications. A hypervisor creates and
| manages virtual servers.

| **Virtual Server Collection.** A set of virtual servers that
| supports a workload. This set is not necessarily static.
| The constituents of the collection at any given point are
| determined by virtual servers involved in supporting
| the workload at that time.

| **virtual Server Image.** A package containing metadata
| that describes the system requirements, virtual storage
| drives, and any goals and constraints for the virtual
| machine (for example, isolation and availability). The
| Open Virtual Machine Format (OVF) is a Distributed
| Management Task Force (DMTF) standard that
| describes a packaging format for virtual server images.

| **Virtual Server Image Capture.** The ability to store
| metadata and disk images of an existing virtual server.
| The metadata describes the virtual server storage,
| network needs, goals and constraints. The captured
| information is stored as a virtual server image that can
| be referenced and used to create and deploy other
| similar images.

| **Virtual Server Image Clone.** The ability to create an
| identical copy (clone) of a virtual server image that can
| be used to create a new similar virtual server.

Virtual Storage Extended (VSE). A system that consists of a basic operating system (VSE/Advanced Functions), and any IBM supplied and user-written programs required to meet the data processing needs of a user. VSE and the hardware that it controls form a complete computing system. Its current version is called VSE/ESA.

Virtual Telecommunications Access Method (VTAM). An IBM licensed program that controls communication and the flow of data in an SNA network. It provides single-domain, multiple-domain, and interconnected

network capability. Its full name is Advanced Communications Function for the Virtual Telecommunications Access Method. Synonymous with ACF/VTAM.

VM Second Level Systems Support. With this function, Processor Operations is able to control VM second level systems (VM guest systems) in the same way that it controls systems running on real hardware.

VM/ESA. Virtual Machine/Enterprise Systems Architecture. Its current version is called z/VM.

volume. A direct access storage device (DASD) volume or a tape volume that serves a system in an SA z/OS enterprise.

VSE. See Virtual Storage Extended.

VTAM. See Virtual Telecommunications Access Method.

W

warning threshold. An application or volume service value that determines the level at which SA z/OS changes the associated icon in the graphical interface to the warning color. See alert threshold.

workstation. In SA z/OS workstation means the *graphic workstation* that an operator uses for day-to-day operations.

write-to-operator (WTO). A request to send a message to an operator at the z/OS operator console. This request is made by an application and is handled by the WTO processor, which is part of the z/OS supervisor program.

write-to-operator-with-reply (WTOR). A request to send a message to an operator at the z/OS operator console that requires a response from the operator. This request is made by an application and is handled by the WTO processor, which is part of the z/OS supervisor program.

WTO. See write-to-operator.

WTOR. See write-to-operator-with-reply.

WWV. The US National Institute of Standards and Technology (NIST) radio station that provides standard time information. A second station, known as WWVB, provides standard time information at a different frequency.

X

XCF. See cross-system coupling facility.

XCF couple data set. The name for the sysplex couple data set prior to MVS/ESA System Product Version 5 Release 1. See also sysplex couple data set.

XCF group. A set of related members that a multisystem application defines to XCF. A member is a specific function, or instance, of the application. A member resides on one system and can communicate with other members of the same group across the sysplex.

XRF. See extended recovery facility.

Z

z/OS. An IBM mainframe operating system that uses 64-bit real storage. See also Base Control Program.

z/OS component. A part of z/OS that performs a specific z/OS function. In SA z/OS, component refers to entities that are managed by SA z/OS automation.

z/OS subsystem. Software products that augment the z/OS operating system. JES and TSO/E are examples of z/OS subsystems. SA z/OS includes automation for some z/OS subsystems.

z/OS system. A z/OS image together with its associated hardware, which collectively are often referred to simply as a system, or z/OS system.

| **z196.** See IBM Enterprise 196 (z196).

| **zAAP.** See IBM System z Application Assist Processor (zAAP).

| **zBX.** See IBM zEnterprise BladeCenter Extension (zBX).

| **zBX blade.** See IBM zEnterprise BladeCenter Extension (zBX) blade.

| **zCPC.** The physical collection of main storage, central processors, timers, and channels within a zEnterprise mainframe. Although this collection of hardware resources is part of the larger zEnterprise central processor complex, you can apply energy management policies to zCPC that are different from those that you apply to any attached IBM zEnterprise BladeCenter Extension (zBX) or blades. See also central processor complex.

| **zIIP.** See IBM System z Integrated Information Processor (zIIP).

| **zEnterprise.** See IBM zEnterprise System (zEnterprise).

Numerics

390-CMOS. Processor family group designator used in the SA z/OS processor operations documentation and in the online help to identify any of the following S/390 CMOS processor machine types: 9672, 9674, 2003, 3000, or 7060. SA z/OS processor operations uses the OCF facility of these processors to perform operations management functions. See OCF-based processor.