APL2

Messages and Codes

Version 2 Release 1

IBM

APL2

Messages and Codes

Version 2 Release 1

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

First Edition (March 1992)

This edition applies to APL2 Version 2 Release 1, Program Number 5688-228, APL2 Version 2 Release 1 Application Environment, Program Number 5688-229, and to any subsequent releases until otherwise indicated in new editions or technical newsletters.

Changes are made periodically to this publication; before using this publication in connection with the operation of IBM systems, consult the latest edition of the applicable IBM system bibliography for current information on this product.

Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality. If you request publications from the address given below, your order will be delayed because publications are not stocked there.

A Reader's Comment Form is provided at the back of this publication. If the form has been removed, comments may be addressed to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Contents

Notices

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

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

Trademarks and Service Marks

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

APL2

DB2

GDDM

MVS/ESA

MVS/XA

SQL/DS

VTAM

About This Book

This book, *APL2 Messages and Codes*, lists, explains, and suggests responses to messages and codes issued by APL2*. This book is intended for all users who deal with APL2's messages and codes.

The APL2 Library

Figure 1, below, shows which APL2 publications should be consulted for help with various tasks.

Figure 1. APL2 Licensed Program Library

Task	APL2 Publication	Publication Number
Evaluating APL2	*APL2 General Information*	GH21-1051
	APL2 Application Environment Licensed Program Specifications	GH21-1063
	APL2 Licensed Program Specifications	GH21-1070
Installing APL2	*APL2 Installation and Customization under CMS*	SH21-1062
	APL2 Installation and Customization under TSO	SH21-1055
Migrating to Version 2 Release 1	*APL2 Migration Guide*	SH21-1069
Finding Reference Information	*APL2 Programming: Language Reference*	SH21-1061
	APL2 Programming: Processor Interface Reference	SH21-1058
	APL2 Programming: System Services Reference	SH21-1054
	APL2 Reference Card	SH21-1071
	APL2 Reference Summary	SX26-3999
Programming	*An Introduction to APL2*	SH21-1073
	APL2 GRAPHPAK: User's Guide and Reference	SH21-1074
	APL2 Programming: Guide	SH21-1072
	APL2 Programming: Using Structured Query Language (SQL)	SH21-1057
	APL2 Programming: Using the Supplied Routines	SH21-1056
Diagnosing Problems	*APL2 Diagnosis*	LY27-9601
	APL2 Messages and Codes	SH21-1059

Related Publications

"Bibliography" on page 126 contains a listing of related publications which provide additional information on topics related to other products.

Chapter 1. Overview

In working with APL2, you may encounter various messages and codes. This book lists and explains those messages and codes.

This chapter contains an overview and introduction to the messages and codes in the following six chapters.

General Message Information

The following text explains how messages are displayed. For explanations and corrective actions, depending on the type of message you received, see one of the following chapters:

- Chapter 2, "APL2 Informational and Error Messages" on page 7, for language and system messages

- Chapter 3, "APL2 Workspace Messages" on page 70, for workspace messages.

For general information about language messages, see "APL2 Language Messages" on page 2; for information on system messages, see "APL2 System Messages" on page 4; and for general information about workspace messages, see "Workspace Messages" on page 5.

There is often more than one explanation for an error message. In this book, the order of explanation has no significance. When there is more than one explanation for an error message, determine what best applies to the condition under which the error occurred and take the corresponding action. APL2 messages may be displayed either with or without a leading message identifier. The identifier will be present if DEBUG (32) is in effect for the APL2 session.

Messages are shown in upper case English in this book. Depending on the setting of the $\Box NLT$ system variable, many messages may actually be displayed in lower case or national languages.

Each numbered error message consists of:

- The standard 3-character beginning "AP2".

- A code identifying the workspace or the module that detected the error. (In this book, three dots represent the place held for this code.)

    The first letter following "AP2" indicates:

A	Session Manager
I	Interpreter
T	TSO Executor
V	CMS Executor
X	Common Executor Services
W	Distributed Workspaces

    The remaining characters of this code uniquely identify the workspace or module that detected the error.

- A 3-digit number that is unique to each message.

- The text of the message itself.

1

- An optional + at the end. This indicates that a backup message is available and can be retrieved by entering)*MORE*.

Variable length messages may be truncated if they become too long. If this occurs, characters at the end of the message are discarded, and an ellipsis (...) is inserted at the *end* of the message.

Figure 2 illustrates the format of a message.

```
                            Standard 3 character beginning

                            Code identifying the workspace or
                            the entry point in the module that
                            detected the error, represented here
                            by three dots (...)

                            Message number

                            Message text




   AP2...414 INVALID COMMAND NAME
```
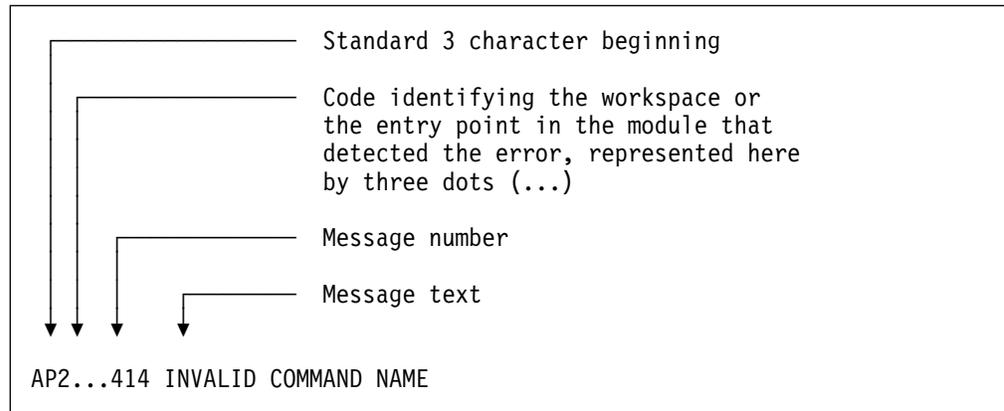
Figure 2. Display Format of a Message

APL2 messages may be displayed either with or without a leading message identifier. The identifier will be present if DEBUG (32) is in effect for the APL2 session.

In Chapter 2, "APL2 Informational and Error Messages" on page 7, the messages are ordered according to the 3-digit number. In Chapter 3, "APL2 Workspace Messages" on page 70, the messages are ordered alphabetically by workspace, and then numerically within each workspace. Each system message is accompanied by an explanation, a description of the system action, and a suggested user response. Each workspace message is accompanied by an explanation, and a suggested user response. Italic expressions in message text represent descriptive tokens, which the system inserts into the message you receive. All messages are indexed for cross-reference according to the message text.

APL2 Language Messages

APL2 displays language information and error messages as the next line of output, beginning at the left margin. Such messages indicate:

- An interrupt was signaled or an error occurred within an expression (for example, an incorrect number of arguments for a function, invalid arguments, or incorrect syntax).
- Successful or unsuccessful completion of actions initiated by system commands.

Interrupts and Errors in APL2 Expressions

Interrupts and errors in expressions generate the following:

- The error message
- The expression interrupted or in error
- Two carets pointing to the expression

Figure 3 shows the message displayed when an error occurs in the expression $2 \div 'X'$.

```
        2÷'X'   ◄─── Expression as entered
 DOMAIN ERROR  ◄─── Message
        2÷'X'   ◄─── Expression in error
        ∧∧      ◄─── Carets pointing to the expression
```

Figure 3. Display When an Error Occurs in an APL2 Expression

The left caret indicates how far APL2 interpreted the statement—from right to left. The right caret indicates where the error or interrupt occurred. In Figure 3, APL2 interpreted the entire expression. The error occurred with the divide function because the right argument was not numeric. Sometimes only one caret appears, because the point at which the error occurred and the point to which APL2 interpreted the expression are the same.

The error message, the expression interrupted or in error, and the two carets pointing to the expression can be retrieved using $\square EM$ (event message). Further information on the category of error can be obtained using $\square ET$ (event type). *APL2 Programming: Language Reference* discusses clearing the error from the state indicator.

Errors in Defined Functions or Operators: When execution of a defined operation results in an error, APL2 displays an error message similar to that generated by an error in immediate execution. The name of the operation and the line number precede the display of the statement in error. Figure 4 shows a message displayed when an error occurs within the defined function named $AVERAGE$.

```
        AVERAGE 4 9 'B'   ◄─ Function invoked
 DOMAIN ERROR             ◄─ Error classification
 AVERAGE[1]  Z←(+/X)÷ρX   ◄─ Statement causing the error
             ∧  ∧         ◄─ Pointers showing how far APL2
                             interpreted the statement and
                             where the error occurred
```

Figure 4. Display When an Error Occurs in a Defined Function or Operator

APL2 Programming: Language Reference describes clearing the state indicator when an error suspends execution of a defined operation.

System Command Messages

Messages generated as a result of system commands may indicate successful completion of an operation, or an error. For instance, issuing the $)CLEAR$ system command to clear the active workspace results in the display of the information message $CLEAR\ WS$:

```
        )CLEAR
CLEAR WS
```

The message indicates successful clearing of the active workspace. After receiving an information message, you can proceed as normal.

If the message indicates an error in the execution of a system command, the command is not executed.

APL2 System Messages

APL2 system messages result from problems with either an auxiliary processor, the APL2 session manager, or the executor under CMS or TSO. For example, the following message results from an invalid session manager command:

```
AP2ACXCM414  INVALID COMMAND NAME
```

If the session manager is in use, messages from the session manager can appear on the control line of the screen; the session manager command, preceded by an asterisk, appears on the command line. Most other system messages appear as part of the session log.

Installation and Service Messages

In general, messages issued by APL2 during installation and servicing of the product conform to the naming conventions used when otherwise running APL2.

Under VM, however, naming conventions for EXECs used to install and service licensed programs require that the APL2 and APL2 Application Environment (AE) EXECs have names beginning with characters other than AP2. The names of the installation EXECs consist of I followed by the seven-digit product number. The names of the verification EXECs consist of V followed by the seven-digit product number. The name of the service EXECs consist of the seven-digit product number followed by A.

Figure 5. APL2 and APL2 AE Installation and Verification EXEC Names

Product	Installation EXEC Name	Verification EXEC Name	Service EXEC Name
APL2	I5688228	V5688228	5688229A
APL2 AE	I5688229	V5688229	5688229A

When numbered error messages are issued by these three EXECs, they are not identified by an ID beginning with AP2. Instead they are assigned a message ID beginning with the EXEC name. This identifies the source of the messages. The EXEC name is followed by a three-digit number. These are the same numbers listed in the messages in Chapter 2, "APL2 Informational and Error Messages" on page 7, and that is where they are described.

For example, if you receive the following message during installation of APL2 on VM:

```
I5688228670  APL2 LOAD SUCCESSFUL
```

you can find a description of the message in Chapter 2, "APL2 Informational and Error Messages" on page 7 under 'AP2...670'.

Workspace Messages

Workspace message identifiers begin with the four characters "AP2W", followed by a name identifying the workspace and a message number. These message identifiers are not displayed with the message; they are contained in the global variable $MORE$. If you encounter an error when using a workspace, enter $MORE$ to view the message identifier and any additional diagnostic information available.

Figure 6 on page 5 illustrates the format of a workspace message. Remember, the identifier of the message, as shown in the example, does not appear unless you enter $MORE$.

```
                   ┌──────── Message
                   │
                   ▼
        OPEN ERROR

             MORE

        AP2WCMS7
        ▲ ▲ ▲ ▲
        │ │ │ └──────────── Message number
        │ │ │
        │ │ └────────────── Workspace name that generated the
        │ │                 error message
        │ │
        │ └──────────────── Code indicating that error message
        │                   is issued by a workspace
        │
        └────────────────── Standard 3-character beginning
```

Figure 6. Display Format of a Workspace Message

If the message table or text is not found, you will receive the code $AP2WMESSAGE999$. This is because the message table was not copied along with the workspace.

Codes Associated with APL2

Code Information

Chapter 4, "Abend Codes" on page 97, lists abend codes numerically by decimal number. The abend codes are also shown in hexadecimal form. The table gives issuing modules and explanations.

Chapter 5, "Dump Codes" on page 100, lists dump codes and the modules that issue them.

Chapter 6, "APL2 Auxiliary Processor Codes" on page 101, lists the return codes that emanate from the auxiliary processors associated with APL2. They are arranged in tables sequentially by processor number. Each consists of a brief description and appropriate action, where necessary.

Chapter 7, "Return Codes from APL2 Modules and EXECs" on page 118, lists the return codes from APL2 modules and installation and service EXECs. They are arranged in tables and contain the code number and a brief description.

Chapter 2. APL2 Informational and Error Messages

This chapter contains a listing of the system and interpreter messages you may receive from APL2 (messages issued from workspaces are in Chapter 3, "APL2 Workspace Messages" on page 70). These messages are listed in numerical order; for example:

`AP2...256`

where:

AP2... identifies the message as having been issued from APL2. The ... are place holders. When you see a message, these place holders will be filled in with 4 or 5 characters, depending on where (in which module or EXEC) APL2 detects the error.

256 is the message number of the error you are getting.

The messages in this chapter are indexed for cross-reference, according to the message text.

AP2...001 INSUFFICIENT STORAGE FOR SIGNON - PROCESSOR *number*

Explanation: There is not enough virtual storage for the auxiliary processor *number* to build the control blocks that it needs. The problem may occur either during APL2 invocation or when an offer is made to the processor.

System Action: If the problem occurs during invocation, the auxiliary processor is terminated.

If the problem occurs during a share offer, the offer is not accepted, but the auxiliary processor is still available. It will continue to respond for already shared variables, and it may be able to accept an offer later, if more storage becomes available.

User Response: Use one or more of the following approaches to correct the problem.

- Decrease WSSIZE if the present value is more than you will need to load and execute workspaces for the current session.

- Decrease SHRSIZE if you will not be passing large units of data through shared variables.

- Under CMS, release access to disks you don't need *or* use NUCXDROP to drop unneeded nucleus extensions *or* issue HX *or* re-IPL to reinitialize storage.

- Under CMS, increase your virtual machine size with the CP DEFINE STORAGE command.

  Warning: You will have to IPL CMS after issuing this command.

- Under TSO, increase your region size.

- Ask your system administrator to increase your CMS default virtual machine size or TSO region size.

You can also specify FREESIZE, but this should be used with caution:

- If you have already specified WSSIZE, giving a FREESIZE requirement will probably only change the symptom for failing invocation.

- If you have not specified WSSIZE, increasing FREESIZE will decrease your workspace size by an uncontrolled amount, and may cause WS FULL problems during execution.

- It is difficult to know how much FREESIZE is really needed. In addition to the APL2 invocation requirements, any non-APL2 programs used during your session will need storage, and APL2 itself will use dynamic storage in varying amounts.

For more about the FREESIZE, SHRSIZE and WSSIZE options, see *APL2 Programming: System Services Reference*.

AP2...002 UNRECOVERABLE ERROR FROM SSM. TERMINATION OF AP *num1* RC=*num2* RS=*num3*

Explanation: While passing a request to the shared variable processor, auxiliary processor *num1* received a return code and reason code indicating that the operation had failed. The auxiliary processor was unable to recover. The return code is *num2*, and *num3* is the reason code from the shared variable processor.

System Action: The auxiliary processor is terminated.

User Response: Report the problem to your system administrator.

AP2...003 ERROR IN TRANSLATE SERVICES - PROCESSOR *number*

Explanation: An error has occurred while translating characters between EBCDIC and the former VS APL internal representation.

System Action: Auxiliary processor *number* is terminated.

User Response: Report the problem to your system administrator.

AP2...004 INVALID SIGNON PARAMETER LIST - PROCESSOR *number*

Explanation: Auxiliary processor *number* has attempted to sign on to the APL2 shared variable processor with an invalid parameter list. This is an internal APL2 logic problem.

System Action: The auxiliary processor is terminated.

User Response: Report the problem to your system administrator.

AP2...005 ABEND. SHARED VARIABLE SET RETRACTED FOR AP *number*

Explanation: An abend has occurred.

System Action: The auxiliary processor terminates.

User Response: Report the problem to your system administrator.

AP2...006 PROCESSOR *number* ALREADY SIGNED ON

Explanation: An auxiliary processor has attempted to sign on to the shared variable processor as processor *number*, but another processor has already signed on with that number.

System Action: The attempted sign-on is rejected, and that version of the AP is terminated. Execution continues.

User Response: Check your APNAMES invocation option. If you did not specify APNAMES, either directly or in a CLIST or EXEC you created, contact your system programmer.

If you are using the APNAMES option to obtain a particular version of an auxiliary processor, use the EXCLUDE option to eliminate other versions of the processor which may have been specified in installation options.

AP2...007 *apname* **ABENDED AT** *offset* **RETURN CODE IS** *num1* **REASON CODE IS** *num2*

Explanation: An auxiliary processor distributed with APL2 received an unexpected return or reason code as the result of a shared variable operation.

apname identifies the auxiliary processor that received the unexpected code.

offset is a hexadecimal offset from the beginning of the auxiliary processor to the instruction at which the unexpected return or reason code was discovered.

num1 is a shared variable processor return code.

num2 is the reason code associated with the error.

System Action: The auxiliary processor terminates.

User Response: You should communicate the displayed information to your system administrator.

AP2...008 *CMSVSAM* **UNAVAILABLE, PROCESSOR** *number* **TERMINATING**

Explanation: Under CMS, a variable was shared with AP *number*, but the processor was unable to locate VSAM or its address overlaps with the virtual machine.

System Action: The auxiliary processor terminates. The APL session continues, but variables offered to the auxiliary processor are never matched.

User Response: See your system administrator.

AP2...010 (content varies)

Explanation: This message most frequently contains some other message returned to APL2 by another part of an MVS system. This can include REXX, VSAM, DAIR, or access method SYNADAF data. Each of those components generates a unique message identifier which will appear within the message text.

System Action: As defined by the component generating the message.

User Response: As defined by the component generating the message.

Explanation 2: Under TSO, an APL2 installation exit may issue this message. The format of the text, and any associated message identifier, are as defined by the installation.

System Action: As defined by the installation exit routine.

User Response: See specific instructions for your installations.

Explanation 3: This message is used as a normal response to some of the forms of the $)CHECK$ $SYSTEM\ ADDR$ diagnostics command (see *APL2 Diagnosis*).

System Action: None.

User Response: As directed by the *APL2 Diagnosis* manual or other instructions that led you to use the $)CHECK$ command.

AP2...011 *name* **UNAVAILABLE**

Explanation: APL2 could not access a particular resource that it needed while processing a $\Box NA$ (name association) system function. These conditions may also arise the first time a previously defined external object is used after loading it into the active workspace. The resource may be any of the following:

AP2TN*nn* **or AP2VN***nn*
 The system is unable to locate or call Processor *nn* (currently 10, 11, or 12).

EXECCOMM or IRXEXCOM
 Processor 10 is unable to communicate with REXX.

routine-name
 Processor 11 is unable to access the named routine, whose whose name was provided in the left argument to $\Box NA$ or in a names file explicitly or implicitly pointed to by that argument.

file_name
 Processor 12 is unable to access the named file. This will be the name specified in the second sub-item of the left argument to $\Box NA$.

System Action: The $\Box NA$ function will return a zero result, and no external association will be in affect.

User Response: Correct the $\Box NA$ argument, make the required resource available, or contact your system administrator.

Explanation 2: APL2 could not access a particular resource that it needed while processing an APL system command. The resource may be any of the following:

CPYSPILL
 $)COPY$, $)MCOPY$, or $)PCOPY$ needs additional storage beyond that available in the workspace and the unused portion of FREESIZE. For CMS it has attempted to use your private library disk as defined in LIBTAB APL2. For TSO it has tried to use a

CPYSPILL DDname or your private VSAM library as defined by an F0 or F*nnnn* DDname. These attempts have failed because the required pointers are not available, because there is not enough space for the needed file, because the private library already contains a CPYSPILL file, or because of I/O errors.

file_name

)*IN* or)*PIN* is unable to access the file from which you have requested objects. The name is shown fully qualified as it was presented to the operating system.

System Action: The system command fails, and control is returned to immediate execution mode.

User Response: Correct the command, provide the needed DDname, make the resource available, or for the CPYSPILL case, reinvoke APL2 with more FREESIZE.

Explanation 3: A request for help was made while using the *IDIOMS* facility, but the *language* APL2HELP file could not be located or read. The system will have tried to use both the help file for the current value of ⎕*NLT* and the ENP help file.

System Action: The *IDIOMS* facility continues, but no help is available.

User Response: For CMS, ensure that you are linked to the APL2 system disks where the help files (file type APL2HELP) are stored. For TSO, ensure that you have an APL2HELP DDname allocated which points to the APL2 help library. Contact your system administrator if you are unable to locate the proper disk or library.

AP2...012 *name* **ALREADY IN USE**

Explanation: This message indicates a problem processing a Processor 11 or 12 ⎕*NA* request or an AP 211 *USE* request.

For Processor 11:

Explanation: *name* is the value of a processor 11 *nick* entry which defines an initialization routine. That routine should be called only once. It has already been called once, but it is being called again.

System Action: SYSTEM LIMIT with ⎕*ET* set to 16.

User Response: Check the logic of your APL functions or contact the person responsible for the application you are using.

For Processor 12 or AP 211:

Explanation: *name* is a data set name or ddname. The data set (or data set allocated to the ddname) is already in use either through PDF EDIT or with write access through processor 12 or AP 211. The data set may be in use either by you or another user.

System Action: If in response to a processor 12 request, a 0 is returned from ⎕*NA*. The name association is not performed. If in response to an AP 211 *USE* request, a ¯17 is returned. The data set is not ready for use.

User Response: If the data set is in use by you, expunge the Processor 12 variable, issue an AP 211 *RELEASE* command, or END the PDF EDIT session. If you are not using the data set, try again later after the other user is done with the data set.

AP2...013 ERROR DETECTED WHILE PROCESSING ARGUMENT ITEM n

Explanation: This is a)*MORE* message which is queued in some cases after an external routine has found an error in its call.

System Action: An error (such as *LENGTH ERROR*, *DOMAIN ERROR*, etc.) is signalled. The type of error and the function call within which it occurred are placed in ⎕*EM*, and also displayed unless the application is using event control.

User Response: Check the argument parameter item identified by this message for the error which was reported in ⎕*EM*.

AP2...015 *name* **ALREADY EXISTS**

Explanation: *name* is a file indentifier passed to either processor 12 or AP 211 in a request to create the file. The file already exists.

System Action: A 0 is returned from ⎕*NA* or ¯17 is returned by AP 211. The file is neither created nor opened.

User Response: Change your AP 211 request to *USE* or your processor 12 request to open for write access if you want to use the existing dataset.

AP2...020 INCORRECT RECORD LENGTH

Explanation: Your record does not contain 80 bytes as required by the)*OUT* (TSO only) or)*IN* command you issued.

System Action: The command will not be processed.

User Response: Reallocate the file with a record length of 80.

AP2...021 DUPLICATE FILE NAME: *fn*

Explanation: When you issued an)*OUT* command, the temporary file AP2OUT AP2UT1 already existed.

System Action: The command will not be processed.

User Response: Either rename or delete the AP2OUT file.

AP2...022   PROCESSOR *num1* ERROR *num2*

Explanation: Auxiliary processor *num1* was being used internally by some other APL2 facility and returned unexpected error code *num2*. Possible facilities include)*IN* or)*PIN*,)*OUT*, ∇-edit when a system editor is being used, and the *IDIOMS* external function.

System Action: The system command or edit session is terminated, returning to immediate execution. If *IDIOMS* was being used, it terminates signalling *SYSTEM ERROR*.

User Response: Take appropriate action for the return code issued. For information about return codes, see *APL2 Programming: System Services Reference* or Chapter 6 of this book.

Explanation 2: TSO auxiliary processor 100 was being used either directly or through)*HOST*, and received an error return code from the TSOLNK (IKJEFTSR) facility. *num2* is the return code provided by TSOLNK.

System Action: Auxiliary processor 100 will set a return code of 1 in the shared variable.

User Response: Common return codes include 40 for command or CLIST not found, 44 for syntax error, and 52 for command not supported in batch. Consult *TSO/E programming Services* for full details.

AP2...023   PROCESSOR *number* NOT AVAILABLE

Explanation: Some APL2 facility needed to use auxiliary processor *num1*, but could not complete a share with it. Possible facilities include)*IN* or)*PIN*,)*OUT*, ∇-edit when a system editor is being used, and the *IDIOMS*, *SERVER*, or *RAPL2* external functions.

System Action: The system command, edit session, or *RAPL2* function is terminated, returning to immediate execution or its caller. If *IDIOMS* or *SERVER* was being used, it terminates signalling *SYSTEM LIMIT, INTERFACE UNAVAILABLE*.

User Response: Verify that the auxiliary processor identified by *num1* was started for the APL session, and is still active. One way to do this is the following:

```
⍝ Associate SVP information function
3 11 ⎕NA 'SVI'
⍝ List active local processors
SVI SVI←⎕AI
```

If the auxiliary processor is active, it may have already accepted as many shares as it is designed for. The limits for each processor are given in *APL2 Programming: System Services Reference*. One other possibility is that the session SVMAX (an invocation parameter) has been reached. The)*QUOTA* command can be used to determine this number.

Explanation 2: TSO auxiliary processor 101 was unable to obtain storage for the AISIZE stack during initialization.

System Action: The processor terminates, so is not available during the session.

User Response: This is probably an error in specifying the AISIZE invocation parameter, but if a very large AISIZE is needed then additional storage must be made available for it. See the user response to message AP2...001 for information about storage allocation.

AP2...024   INTERNAL ERROR *location*

Explanation: *location* identifies an associated processor, either as a name or as a CSECT + offset. An unexpected error has occurred at the specified location.

System Action: *SYSTEM ERROR* is signaled in the workspace. The processor remains available for subsequent calls.

User Response: Contact your system administrator.

AP2...025   *name* PARAMETER ERROR [*proc*]

Explanation:

This message is queued when an error is detected in an external object's routine description during a name association (⎕*NA*) with processor 11.

name is the external object name specified in the right argument to ⎕*NA*. *proc* is either unrecognized data or a tag which has invalid data.

System Action: The ⎕*NA* completes with result 0, and no external association will exist for the object.

User Response: If one exists, examine the NAMES file entry for the object. If the object is self-describing, verify that the routine description for the object was correct.

Explanation 2: An error was detected during a name association (⎕*NA*) with processor 12. *name* is the right argument to ⎕*NA*. Additional messages will have been queued giving more information.

System Action: The ⎕*NA* completes with result 0, and no external association will exist for the variable.

User Response: Check the left argument to ⎕*NA*, and use)*MORE* to obtain additional information if the source of the problem is not obvious.

AP2...026   NAMESPACE *name* MODIFIED SINCE LAST ACCESS

Explanation: The namespace has been modified since the active workspace was last saved. If objects were created or modified in the namespaces scope when it was last accessed, those objects are lost.

Note: For more information on name scopes, refer to the *APL2 Programming: System Services Reference*.

System Action: A fresh copy of the new namespace is made available to the user. The previous version of the namespace, saved as part of the user's workspace, is no longer available in the active workspace.

User Response: If no objects were created or modified in the previous version of the namespace, no action is necessary. If objects were created or modified in the packaged workspace name scope, they can only be reaccessed by gaining access to the previous version of the namespace. Make the previous version of the namespace available, re-$)LOAD$ the active workspace and retry. **Warning:** Do not re-$)SAVE$ the active workspace under the same name if you plan to reaccess the previous version of the package.

AP2...027   RECORD LENGTH MUST BE PROVIDED

Explanation: This message is queued in response to a Processor 12 $\square NA$ which has specified 'FC' to request creation of an operating system file. Processor 12 needs to know the maximum record length for the file, but it has not been provided.

System Action: The $\square NA$ is terminated with result 0 and the file is not created.

User Response: To create a fixed length record file, provide the length explicitly in the descriptor, for example '$G0 \ 1 \ *(C1 \ 1 \ 80)$' rather than '$G0 \ 1 \ *(C1 \ 1 \ *)$'. To create a variable length record file, leave the $*$ in the descriptor, but provide length information with the 'FC'. For example '$FC(80)$'.

AP2...030   CP/CMS COMMAND *name* ABENDED, CODE *number*

Explanation: CP/CMS command *name*, issued through AP 100 or by using $)HOST$, has failed. *nnn* indicates the abend code.

System Action: Execution continues. The return code $^-100$ is generated for the AP 100 request.

User Response: The failing command may have damaged the integrity of CMS or APL2, so proceed carefully. Report the problem to your system administrator.

AP2...031   INVALID FILE IDENTIFIER

Explanation: An invalid file identifier has been passed to either processor 12 or AP 211.

System Action: A 0 is returned from $\square NA$ or $^-14$ is returned by AP 211. The file is not opened.

User Response: Correct your file identifier so that it adheres to standard operating system file naming conventions or on TSO is a valid ddname prefixed with an equal sign.

AP2...032   DDNAME CREATION AND DELETION NOT SUPPORTED

Explanation: A request was been made to either processor 12 or AP 211 to create or drop a file. The supplied file identifier was prefixed with an equal sign (indicating it is a ddname.) APL2's file services do not support creation or deletion of ddnames.

System Action: A 0 is returned from $\square NA$ or 13 is returned by AP 211. The ddname is not allocated or freed.

User Response: Free the ddname using a TSO FREE command rather than processor 12 or AP 211.

AP2...033   INVALID RECORD LENGTH OR FORMAT FOR FILE CREATION. MAXIMUM LENGTH: *number*

Explanation: A request was been made through either processor 12 or AP 211 to create a file with longer records than is supported by the operating system. On CMS, the largest record length supported is 65535 characters. On TSO, the largest record length supported for variable record format data sets is 32752 characters. On TSO, the largest record length supported for fixed record format data sets is 32760 characters.

System Action: A 0 is returned from $\square NA$ or $^-8$ is returned by AP 211. The file is not created.

User Response: Specify a smaller record length.

AP2...034   UNSUPPORTED TYPE OF FILE [$word$]

Explanation: A request was been made through either processor 12 or AP 211 to open an unsupported type of file. If *word* appears in the message, it states the file attribute which is unsupported. Common types of unsupported files are members of partitioned datasets, files with invalid LRECL, or files that are neither RECFM V, VB, F, nor FB.

System Action: A 0 is returned from $\square NA$ or 2 or 4 is returned by AP 211. The file is not opened.

User Response: If you were trying to access the file using processor 12, use a processor 10 routine or an auxiliary processor. If you were trying to access the file using AP 211, it is not a valid AP 211 file.

AP2...035   FILE'S *word* IS INCOMPATIBLE WITH REQUEST *word* IS *value*.

Explanation: A request made through either processor 12 or AP 211 can not be processed because the request implies a record length or record format which is in conflict with the specified file.

System Action: A 0 is returned from $\square NA$ or 7 is returned by AP 211. The file is not opened.

User Response: If you were trying to access a record format V file with AP 211, copy it to a record format F file.

AP2...036 ATTEMPT TO ACCESS RECORD BEYOND END OF FILE. FILE CONTAINS *number* **RECORDS.**

Explanation: A request was made to read a record beyond the end of a file or to write a record more than 1 record beyond the end of the file. This message typically follows an *INDEX ERROR*.

System Action: The record is neither read nor written.

User Response: Correct the index used in the expression.

AP2...037 YOU MAY NOT MODIFY THIS FILE. IT IS OPEN FOR READ-ONLY ACCESS.

Explanation: An attempt was made to modify a file, either through processor 11 or AP 211, which is open for read-only access. If the file is being accessed using processor 12, this message typically follows a *SYNTAX ERROR*.

System Action: The file is not modified. If the attempt was made using AP 211, a 4 is returned.

User Response: If accessing the file with AP 211, *RELEASE* the file and *USE* with write access. If accessing the file with processor 12, expunge the associated variable and reassociate it using write access.

AP2...038 RECORDS IN THIS FILE ARE LIMITED TO *number* **CHARACTERS**

Explanation: For variable record length files, an attempt was made to add a record which is longer than the file supports. *number* is the maximum length supported by the file.

For fixed record length files, an attempt was made to add or replace a record whose length is not exactly the file's LRECL. *number* is the file's LRECL.

System Action: The file is not modified.

User Response: Pad or truncate the data you're writing or use a different file.

AP2...039 ANY REPLACEMENT RECORD *recno* **MUST BE** *reclen* **CHARACTERS LONG.**

Explanation: An attempt was made to replace a record in a variable record format file. The replacement record must be exactly the same length as the record it is to replace. *recno* is the number of the record being replaced. *reclen* is the length of record *recno*.

System Action: The file is not modified.

User Response: Pad or truncate the data you're writing or use a different file.

AP2...040 FILE FULL

Explanation: The file being accessed has reached the limit of space available for it. On CMS, this means that the disk containing the file is full. On TSO, this means that the amount of space allocated for the data set has all been used.

On CMS, this message will appear immediately when an attempt is made to write more data than will fit on the disk. On TSO, this message may not appear until a later access to the file is made. This is because APL uses buffers to manage the data being written to disk storage. The buffer containing a new or modified record may not actually be written to DASD (at which time the file full condition occurs) until the buffer is needed. This may not happen for quite a while.

The only way to force the data to be written to disk storage is to close the file. This error can happen as a result of almost any access of a processor 12 variable or AP 211 file.

System Action: If the error was detected while closing a file associated through Processor 12, the interpreter response depends on the user or application action which caused the file to be closed:

- If □*EX*, the explicit result is 0 and the file remains open.

- If exiting from a defined function or operator where the Processor 12 variable was localized, the function is suspended on line 0 and the file remains open.

- If due to)*ERASE*, or to)*COPY* or)*IN* of an object which would replace the Processor 12 variable, the response is *NOT ERASED* or *NOT COPIED* and the file remains open.

- If due to → with no arguments, or to any of the system commands)*CLEAR*,)*CONTINUE*,)*LOAD*,)*OFF*,)*RESET*, or)*SIC*, then the file is closed and processing continues without additional error indication.

User Response: Examine the file and determine if any data was lost. Acquire more disk space or allocate a larger data set.

If the file was left open, repeating the request will usually cause the file to be closed, and the request to be processed normally.

AP2...041   AP 101 STACK OVERFLOW, APL STACK PURGED

Explanation: More data has been placed into the alternate input stack area than is permitted for this invocation.

System Action: The entire alternate input stack is purged, to prevent the side effects that may occur in executing the existing stack items.

The system places the stack into *purging status*, which is cleared when the value of the shared variable is referenced to retrieve the return code. If the value is not referenced, the alternate input area remains in purging status until the shared variable is referenced or retracted.

User Response: Exit, then reinvoke APL2, using the AISIZE option to set a larger alternate input area for the application stack. The default size is 512 bytes.

AP2...042   AP 101 STACK CLEARED DUE TO INVALID DATA

Explanation: Invalid data has been specified into the variable shared with AP 101. Data is limited to character vectors less than 255 characters long.

System Action: The entire alternate input stack is purged to prevent the side effects that may occur by executing the existing stack items.

The system places the stack into *purging status*, which inhibits you from placing more input on the stack and prevents execution of any input.

User Response: Clear the purging status by either referencing the value of it or retracting the shared variable. Correct the APL2-defined function that passes data to AP 101. Limit character vectors to 255 bytes.

AP2...043   AP *num1* IS BEING MISUSED, AP RC = *num2*

Explanation: This message is displayed when the user or workspace function ignores the nonzero return code *number* from the auxiliary processor. For example, a file named in a shared variable initial value cannot be opened, but the workspace attempts to write records to the file without checking the return code.

System Action: No value is assigned to the record variable, nor is a value set by the workspace accepted by the auxiliary processor. This normally results in a shared variable interlock after the message is issued. The return code remains available in the control variable.

User Response: Issue a strong interrupt (by hitting the *attention key* more than once) if the session appears to be hung. (For more information on interrupts, see *APL2 Programming: System Services Reference*.) If you are using an APL2 function that you have written, change it to check for error return codes from the auxiliary processor.

If you are using an application package provided by someone else, report the error code from this message, together with the function name and line number at the point of *INTERRUPT*, to the person responsible for that application package.

AP2...044   UNUSUAL END TO FORMAT, *decnum* RECORDS FORMATTED, DCB ABEND *hexnum*

Explanation: This message is displayed when AP 210 is requested to format a BDAM data set with more records than will fit within the data set allocation.

System Action: AP 210 terminates the format operation, and sets a negative return code that indicates the number of records actually formatted on the data set.

User Response: Increase the data set allocation for the number of records being formatted, or reformat the file so that it has the number of records specified in the AP 210 return code.

AP2...045   *name* ABEND - SYSTEM CODE - x'*hexnum*'

Explanation: *name* begins with "@" and represents some APL2 auxiliary processor or service task which has terminated abnormally.

System Action: The APL2 session continues without the processor if possible.

User Response: Report the problem to the owner of the failing processor. If the processor was provided by IBM, report it as an error in the IBM product.

Explanation 2: *name* is AP2TN*nn*, where *nn* is an associated processor number (currently 10, 11, or 12).

System Action: A *SYSTEM ERROR* will be signalled.

User Response: Report the problem as a failure in the APL2 product.

Explanation 3: *name* represents a TSO command issued through AP 100 or)*HOST* which has terminated abnormally.

System Action: Auxiliary processor 100 assigns the decimal equivalent of the abend code to its shared variable.

User Response: Refer to MVS abend code documentation or report the problem to the owner of the failing command.

Explanation 4: *name* is "MVS," and the problem arose while using the APL2 external functions which access MVS/ESA* Data Window Services.

System Action: The external function ends by signalling the abend code and reason code as a □*ET* value.

User Response: Refer to MVS abend code documentation or *MVS/ESA Callable Services for High Level Languages*. This may represent either a problem with the MVS facility or an error in the APL application.

AP2...046   AP ENTRY UNRESOLVED IN MODULE
modname

Explanation: An auxiliary processor named in the APNAMES invocation option had not been link-edited correctly. *modname* is the name given in APNAMES.

System Action: The session continues without the auxiliary processor.

User Response: Contact the person responsible for the auxiliary processor (usually your system administrator). Either the auxiliary processor must have an entry point of APENTRY, or a link-edit CHANGE card must be used to redefine its entry point as APENTRY.

Note: The load module entry point must be AP2TASVP (under TSO) or AP2VASVP (under CMS). You must have had this correct originally to receive this message; just be careful when you relink.

AP2...047   NOT ENOUGH VIRTUAL STORAGE TO
SATISFY PROCESSOR *number* REQUEST

Explanation: There is not enough virtual storage for auxiliary processor *number* to build control blocks or provide buffers. The problem may occur:

1. During APL2 invocation
2. When an offer is made to the processor
3. When a service request is made

System Action: There may be some variations for individual processors, but in general, for the above categories:

1. The auxiliary processor is terminated.

2. The offer is not accepted, but the auxiliary processor is still available. It will continue to respond for already shared variables, and may be able to accept an offer later if more storage becomes available.

3. An error code is returned in the shared variable.

User Response: See the user response for message AP2...001.

AP2...048   UNEXPECTED PROCESSOR *num1*
ERROR: SERVICE=*sercode*, RC=*num2*

Explanation: The auxiliary processor has issued a service request to APL2. It received an error code that it was not programmed to expect.

System Action: An error code may be returned in the shared variable, the shared variable may be retracted,

or the auxiliary processor may terminate, depending on the severity of the error.

User Response: Contact the person responsible for the auxiliary processor (usually your system administrator). That person will need to know the service code (two alphabetic characters) and the return code.

AP2...049   PROCESSOR *number* TERMINATED

Explanation: The auxiliary processor had previously detected and probably reported an error from which it cannot recover.

System Action: The auxiliary processor terminates. Any subsequent offers in the same APL2 session will remain outstanding ($\Box SVO$ will return 1).

User Response: This message should follow one or more other error messages. If no other messages were displayed, and the text is followed by a +, then enter $)MORE$ to determine why the processor terminated. To regain use of the processor, you must use $)OFF$ or $)CONTINUE$ to exit APL2, and then reinvoke it.

AP2...051   IDIOM NUMBER: *nnn* ADDED TO
"IDIOM_LIST" FUNCTION

Explanation: The idiom number indicated by *nnn* has been added to the function.

User Response: This is an informational message only.

AP2...052   INVALID FUNCTION KEY PRESSED

Explanation: The function key pressed is not enabled on the current screen.

User Response: Press only the function keys indicated in the informational area at the bottom of the screen.

AP2...053   CURSOR MUST BE ON A SELECTION

Explanation: A function key was pressed, but the cursor was not on a valid selection for the action requested.

User Response: Position the cursor on one of the selections on the screen before pressing the function key.

AP2...055   SELECTION MUST BE A NUMBER
BETWEEN *nnn* and *mmm*

Explanation: A selection number is either out of range or not numeric.

User Response: Choose a number in the range indicated by *nnn* and *mmm*.

AP2...056 IDIOM NUMBER: *nnn* **ADDED TO OUTPUT**

Explanation: The idiom number indicated by *nnn* has been added to the explicit result.

User Response: This is an informational message only.

AP2...057 INVALID INDEX ORIGIN. IT HAS BEEN RESET TO *nnn*

Explanation: An index origin other than zero or one was specified.

User Response: Specify index origin zero or one.

AP2...058 INDEX ORIGIN *nnn* **SELECTED**

Explanation: The index origin has been set as indicated.

User Response: This is an informational message only.

AP2...059 *ccccc* **GROUP SELECTED**

Explanation: The indicated group has been selected. The items in the group are displayed on the screen.

User Response: This is an informational message only.

AP2...060 *nnn* **ITEMS FOUND FOR SEARCH ARGUMENT:** *ccccc*

Explanation: The indicated number of items were found for the indicated search argument. The items are displayed on the screen.

User Response: This is an informational message only.

AP2...061 NO SEARCHES HAVE BEEN MADE; THERE IS NOTHING IN THE WINDOW

Explanation: The user pressed the function key to enter the selection window, but it is empty.

User Response: Execute some searches before attempting to enter the selection window.

AP2...062 NOTHING NEW REQUESTED, NO SEARCH PERFORMED

Explanation: The search argument entered is the same as the previous search.

User Response: This is an informational message only.

AP2...063 SELECTION *nnn* **IS THE SAME; IT HAS BEEN RESELECTED**

Explanation: The search argument entered is the same as a previous search. The items selected for that search are displayed on the screen.

User Response: This is an informational message only.

AP2...064 SELECTION *nnn* **GIVES THE SAME RESULTS; SEARCH INSERTED AFTER IT**

Explanation: The search argument entered is different from all previous searches, but the items selected are the same as for a previous search. The selection is inserted after that one in the selection window.

User Response: This is an informational message only.

AP2...065 SELECTION 000 CANNOT BE ALTERED

Explanation: Selection 000 has been chosen to be moved or deleted, but this selection is the set of all idioms, and cannot be altered.

User Response: Do not attempt to alter selection 000.

AP2...066 *ccccc nnn* **DELETED**

Explanation: The phrase *ccccc* indicated by selection number *nnn* has been deleted from the selection list as requested.

User Response: This is an informational message only.

AP2...067 BOTH M AND A OR F ARE NEEDED TO MOVE A SELECTION

Explanation: The commands to move a selection in the window have been entered incorrectly. Either "M" and "A", or "M" and "F", must be entered.

User Response: Correct the command, and try again.

AP2...068 SELECTION *nnn* **MOVED AFTER** *mmm*

Explanation: The indicated selection has been moved as requested.

User Response: This is an informational message only.

AP2...069 WARNING: SEARCH TOO COMPLEX. *nnn* **IDIOMS FOUND, ARGUMENT TRUNCATED**

Explanation: The search argument is too long. It was truncated before the search.

User Response: The results of the local search are correct, but the search parameters are too long to store.

Future searches using these parameters may be inaccurate.

AP2...070   ERROR OCCURRED IN SHARED VARIABLE INTERFACE. EVENT TYPE: *et*

Explanation: The external function $RAPL2$ encountered an error in the shared variable interface with a remote interpreter. Either the degree of coupling changed unexpectedly, or an error occurred while awaiting a response from the interpreter, referencing or displaying the variable's value, or specifying a new value for the remote interpreter to process.

The message is prefixed with $RAPL2$ lid where lid is $\uparrow\square AI$ on the session where $RAPL2$ is running. This information allows you to determine what system issued the message.

System Action: If the degree of coupling changed unexpectedly, $RAPL2$ retracts the shared variable and terminates. No shutdown instruction is passed to the remote interpreter.

If an error occurred awaiting a response or referencing or displaying the variable, $RAPL2$ displays this message and waits for another response from the remote interpreter.

If an error occurred while specifying a new value, $RAPL2$ attempts to send a shutdown command to the remote interpreter. If the error occurs when a shutdown command has already been sent, $RAPL2$ displays this message and waits for another response from the remote interpreter.

User Response: Continue with the remote APL2 session. Examine recent commands to determine if any data seems to have been lost.

AP2...071   CONNECTION WITH PROCESSOR *id* ACTIVE.

Explanation: The external function $RAPL2$ has successfully established a shared variable connection with a remote interpreter using ID *id*. This message is issued when the connection is initially established and upon return from a local $\square$ prompt loop.

The message is prefixed with $RAPL2$ lid where lid is $\uparrow\square AI$ on the session where $RAPL2$ is running. This information allows you to determine what system issued the message.

System Action: $RAPL2$ prompts for input to be passed to the remote interpreter and displays the interpreter's results.

User Response: Respond to $RAPL2$'s prompts as you would for a local interpreter.

AP2...072   CONNECTION WITH PROCESSOR *ID* TERMINATED.

Explanation: $RAPL2$'s connection with a remote interpreter has been broken.

The message is prefixed with $RAPL2$ lid where lid is $\uparrow\square AI$ on the session where $RAPL2$ is running. This information allows you to determine what system issued the message.

System Action: $RAPL2$ terminates when the remote interpreter retracts the shared variable. This typically happens when the interpreter receives an $)OFF$ or $)CONTINUE$ command.

User Response: Continue with local interpreter session.

AP2...073   UNKNOWN RESPONSE RECEIVED FROM INTERPRETER. *cmd et*

Explanation: The external function $RAPL2$ has received an unknown value through the variable shared with a remote interpreter. The first two elements of the shared variable value are displayed in *cmd et*.

The message is prefixed with $RAPL2$ lid where lid is $\uparrow\square AI$ on the session where $RAPL2$ is running. This information allows you to determine what system issued the message.

System Action: $RAPL2$ waits for more data from the remote interpreter.

User Response: Continue with the remote-session if possible. Examine any further messages that may be issued.

AP2...078   INTERRUPT (ENTER), SWITCH (0),)CONTINUE (1):

Explanation: You have signalled an interrupt while using $RAPL2$ to conduct a session with a remote interpreter. $RAPL2$ is prompting you to indicate whether you want the interrupt sent on to the remote interpreter, want to enter a $\square$ prompt session with the local interpreter, or want a shutdown command sent to the remote interpreter.

The message is prefixed with $RAPL2$ lid where lid is $\uparrow\square AI$ on the session where $RAPL2$ is running. This information allows you to determine what system issued the message.

System Action: $RAPL2$ awaits your response.

User Response: Hit enter to send the interrupt to the remote interpreter, enter 1 to receive $\square$ prompts from the local interpreter, or enter 2 to shutdown the remote interpreter.

AP2...079 INTERRUPT AGAIN TO SWITCH BACK.

Explanation: You have signalled an interrupt while using *RAPL2* to conduct a session with a remote interpreter. You have then entered 1 to receive ⎕ prompts from the local interpreter. This message instructs you to signal interrupt again to resume your session with the remote interpreter.

The message is prefixed with *RAPL2 lid* where *lid* is +⎕AI on the session where *RAPL2* is running. This information allows you to determine what system issued the message.

System Action: The system issues ⎕ prompts.

User Response: Enter expressions to be executed by the local interpreter. To resume the remote session, signal interrupt again.

AP2...080 APL2PIE: UNEXPECTED ERROR, 3 11
⎕*NA 'OPTION'* **FAILED**

Explanation: The APL2 Program Interface facility is unable to access the *OPTION* external function which is distributed with APL2.

System Action: The function terminates with no result. If it was called by *APL2PI* or an APL application function, that function will suspend with *VALUE ERROR*.

User Response: Enter)*MORE* for further explanation if a "+" was displayed at the end of the message. The most likely sources of the problem are that the AP2TN11 or AP2VN11 names file has been modified incorrectly, or that the definition of *OPTION* has been overridden by a privately defined external object of the same name. Ensure that the APL2 session has access to a proper names file.

AP2...081 *program***: UNEXPECTED ERROR,** *code*
FROM *service*

Explanation: If *program* is P10 then *service* is the name of a REXX service, and *code* is the return code from that service.

System Action: *SYSTEM ERROR* is signalled to the APL application or session.

User Response: Report the problem to your installation or to IBM service. The error may be in either APL2 or REXX.

Explanation 2: If *program* is APL2PIE then *service* will be the name of an internal APL2 routine and *code* is the return code from that routine.

System Action: The *APL2PIE* function terminates with result 1 2, which will be displayed in the APL2 session or returned to the APL application which invoked *APL2PIE*.

User Response: Report the problem to your installation or to IBM service.

AP2...082 ENTER 'APL2PI' TO RETURN CONTROL

Explanation: This is an informational message which may be displayed when a non-APL program requests interactive APL session execution.

System Action: The message is returned as part of the explicit result of the *APL2PIE* or *APL2PI* function, and so is displayed if that function was invoked from an interactive APL2 session or during APL2 invocation.

User Response: On completion of any desired APL2 interactive work, type *APL2PI* to return control to the non-APL application.

**AP2...103 INSUFFICIENT STORAGE HAS BEEN
ALLOCATED BY THIS PROCESSOR**

Explanation: The amount of storage allocated by AP 127 is not large enough to process the internal calls in the processor.

System Action: AP 127 will abend.

User Response: Contact your system administrator.

**AP2...104 APL2 PROCESSOR ERROR IN DATA
FORMAT**

Explanation: The data passed to AP 127 by the common shared variable processor (CSVP) is not in standard APL2 format. This is an internal APL2 logic problem.

System Action: AP 127 will abend.

User Response: Contact your system administrator.

AP2...105 SQL DATA TYPE *number* **(COLUMN
name) NOT SUPPORTED BY AP127**

Explanation: The data passed to AP 127 by SQL contains a column with a data type not supported by AP 127.

System Action: The request will not be processed.

User Response: Rework the query so that it does not include the named column.

**AP2...106 AP 127 PROCESSOR ERROR - INVALID
INTERNAL CODE**

Explanation: The AP 127 SQL interface module has been called with an operation code it does not recognize. This is an internal AP 127 logic problem.

System Action: AP 127 will abend.

User Response: Contact your system administrator.

AP2...109 THE RESULT TABLE IS TOO LARGE FOR THE SHARED VARIABLE

Explanation: The result table retrieved by a *FETCH* request is too large to fit into the space allocated for the shared variable.

System Action: The result table will not be placed in the shared variable.

User Response: Reissue the *FETCH* request (or issue the *SETOPT* request), specifying fewer rows in the options list or increase the SHRSIZE invocation option. See message AP2...001 for SHRSIZE option user response.

AP2...110 THE ERROR CODE IS NOT A 5-ITEM NUMERIC VECTOR

Explanation: The return code item passed via a *'MSG'* operation or *MESSAGE* function is not a five-item numeric vector.

System Action: The command will not be processed.

User Response: Respecify the return code vector.

AP2...111 ATTEMPT TO SHARE MORE THAN THE ALLOWED LIMIT OF *number* VARIABLES WITH AP 127

Explanation: You have attempted to share more variables with AP 127 than it is prepared to accept at one time. (The current limit is 10 variables.)

System Action: The additional variable will not be matched.

User Response: Specify a variable already shared, or retract one or more variables before sharing again.

AP2...112 THE FIRST ITEM OF THE SHARED VARIABLE IS NOT A RECOGNIZED OPERATION

Explanation: The operation code passed to AP 127 is not one of the recognized character-string operation codes.

System Action: The command will not be processed.

User Response: Check the spelling of the operation code, and respecify the variable.

AP2...113 THE *operation* OPERATION EXPECTS *num1* ARGUMENTS BUT RECEIVED *num2*

Explanation: The number of arguments for the operation is too great or too small.

System Action: The command will not be processed.

User Response: Respecify the variable with the correct number of arguments for the operation.

AP2...114 SQL STATEMENT NOT CHARACTER VECTOR OR MATRIX

Explanation: An SQL statement was passed to AP 127 via the PREP or EXEC operation, but it was not a simple character vector or matrix.

System Action: The command will not be processed.

User Response: Resubmit the request with the statement in the correct form.

AP2...115 ATTEMPT TO USE OTHER THAN THE ACTIVE SHARED VARIABLE *name*

Explanation: An attempt has been made to specify a second shared variable before the first has been retracted. Up to 10 variables may be shared at one time, but only one can be *active* (specified) at a time. After a variable has become active, it remains so until it is retracted.

System Action: The command will not be processed.

User Response: Retract the first variable before specifying the second.

AP2...119 *number* IS NOT A RECOGNIZED ROWS VALUE

Explanation: The *number* entered for rows in an options list is not a positive integer.

System Action: The command will not be processed.

User Response: Respecify the number of rows.

AP2...120 ATTEMPT TO PREPARE MORE THAN THE ALLOWED LIMIT OF *number* STATEMENTS

Explanation: No more than *number* SQL statements can be processed via a *PREP* request before one or more of them must be purged. The current limit is 40 statements.

System Action: The command will not be processed.

User Response: Issue a *PURGE* request for one or more of the existing statements, or reuse a statement name in a *PREP* request.

AP2...121 INSUFFICIENT STORAGE TO *token1* *token2*

Explanation: AP 127 cannot obtain enough space to honor the named request. *token1* will be PREPARE, DECLARE, DESCRIBE, or EXECUTE. If *token1* is PREPARE, DECLARE, or DESCRIBE, *token2* will be the statement name. If *token1* is EXECUTE, *token2* will be IMMEDIATE.

System Action: The command will not be processed.

User Response: Issue a *PURGE* request for one or more SQL statements to make space available, or

either decrease the WSSIZE or increase the FREESIZE invocation option.

AP2...122 INDEX VALUE FOUND IN STATEMENT POSITION *number* **IS NOT OF A RECOG-NIZABLE TYPE**

Explanation: An index value in an SQL statement is not a positive integer.

System Action: The command will not be processed.

User Response: Change the vector index item to a positive integer.

AP2...125 INDEX MARKER FOUND IN POSITION *number* **OF AN EXEC STATEMENT**

Explanation: The SQL statement passed with an *EXEC* request contains a colon marking an item as a vector index. *EXEC* requests cannot contain vector indices.

System Action: The command will not be processed.

User Response: Correct the SQL statement and reissue the *EXEC* request.

AP2...126 NO ACTIVE STATEMENTS EXIST

Explanation: A *GET NAMES* request could not be completed because no SQL statements have been processed via a *PREP* request.

System Action: The command will not be processed.

User Response: Issue a *PREP* request.

AP2...127 UNKNOWN OPTION VALUE

Explanation: The options list contains a character value other than *'MATRIX'*, *'VECTOR'*, *'LENGTH'*, or *'NOLENGTH'*.

System Action: The command will not be processed.

User Response: Correct the options list, and reissue the request.

AP2...131 *number* **IS AN UNRECOGNIZED ERROR CLASS** *number* **is an unrecognized error class (AP2...131)**

Explanation: The fourth item of the numeric vector argument to the *'MSG'* operation is not 1 or 2, or the fourth item of the argument to the *MESSAGE* function is not 1, 2, or 3.

System Action: The command will not be processed.

User Response: Correct the fourth item of the argument, and reissue the request.

AP2...132 *number* **IS AN UNDEFINED MESSAGE NUMBER FOR AP 127**

Explanation: Although the fourth item is 1, the fifth item of the numeric vector argument to the *'MSG'* operation or *MESSAGE* function is not a valid AP 127 message number.

System Action: The command will not be processed.

User Response: Correct the fifth item of the argument, and reissue the request, or check the fourth item to be sure that it correctly identifies the source of the error—1 for AP 127, 2 for SQL, or 3 for a workspace function (permitted only with the *MESSAGE* function).

AP2...133 INSUFFICIENT STORAGE FOR VALUE-LIST OF *name*

Explanation: AP 127 cannot allocate enough space to store the value list passed on a *CALL* or *OPEN* request.

System Action: The command will not be processed.

User Response: Decrease the size of the value list, or either decrease the WSSIZE or increase the FREESIZE invocation option. See message AP2...001 for user response for the FREESIZE and WSSIZE options.

AP2...134 THE STATEMENT *name* **EXPECTS AT LEAST** *num1* **ITEMS IN ITS VALUE-LIST AND RECEIVED** *num2*

Explanation: The value list passed with a *CALL* or an *OPEN* request has fewer items than the highest index value passed with the statement.

System Action: The command will not be processed.

User Response: Either increase the number of items in the value list or change the vector index to reflect the actual number of items passed.

AP2...135 *name* **IS SELECT STATEMENT** *operation* **- OPERATION CANNOT BE COMPLETED**

Explanation: A *CALL* or *PUT* request has been issued for a cursor *SELECT* statement.

System Action: The command will not be processed.

User Response: Correct the name in the *CALL* or *PUT* request or issue an *OPEN* and *FETCH* request for the statement.

AP2...136 *name* **IS NOT A SELECT STATEMENT -** *operation* **OPERATION CANNOT BE COM-PLETED**

Explanation: A *DESCRIBE* request has been issued for a statement other than an SQL *SELECT* statement.

System Action: The command will not be processed.

User Response: Correct the name in the

DESCRIBE request or issue a *CALL* request for the statement.

AP2...137 *name* **IS NOT PREPARED -** *operation* **OPERATION CANNOT BE COMPLETED**

Explanation: A *CALL*, *OPEN*, or *DESCRIBE* request has been made for a statement that has not been processed via a *PREP* request.

System Action: The command will not be processed.

User Response: Issue a *PREP* request for the statement named.

AP2...139 *name* **IS NOT OPEN -** *operation* **CANNOT BE COMPLETED**

Explanation: A *FETCH* or *CLOSE* operation has been issued for a statement that has not been processed by an *OPEN* request.

System Action: The command will not be processed.

User Response: Issue the *OPEN* request before issuing the *FETCH* or *CLOSE*.

AP2...141 **THE VALUE PASSED IN POSITION** *number* **IS OF A TYPE NOT RECOGNIZED BY SQL**

Explanation: The item identified, for example, a complex number, cannot be processed by SQL.

System Action: The command will not be processed.

User Response: Correct the value list and reissue the request.

AP2...142 **THE VALUE-LIST PASSED IS IN AN UNRECOGNIZED FORMAT**

Explanation: The value list argument in a *CALL*, *PUT*, or *OPEN* request is not a numeric vector, a character vector, a numeric scalar, or a mixed vector of character vectors and numeric scalars. The depth must be two or less.

System Action: The command will not be processed.

User Response: Correct the value list and reissue the request.

AP2...143 **STATEMENT NAME OF LENGTH** *length* **IS TOO LONG**

Explanation: The name specified for a statement must fit in 18 bytes of storage after removing leading and trailing blanks.

System Action: The command will not be processed.

User Response: Shorten the name and reissue the *PREP* request.

AP2...144 **STATEMENT NAME NOT VALID**

Explanation: The name specified for an SQL statement on a *PREP* request is not a character vector with length 2 through 18 and a non-underlined alphabetic character as its first item.

System Action: The command will not be processed.

User Response: Correct the name, and reissue the *PREP* request.

AP2...145 **THE STATEMENT** *name* **DOES NOT EXIST**

Explanation: The SQL statement named has not been processed via a *PREP* request.

System Action: The command will not be processed.

User Response: Either correct the name or issue a *PREP* request for the name.

AP2...146 **WARNING - VALUE LIST LONGER THAN NECESSARY**

Explanation: A value list passed on an *OPEN* or *CALL* request has more items than the highest vector index in the statement. This is a warning only. It does not necessarily indicate an error.

System Action: The command is processed normally.

User Response: Verify that the value list is correctly formatted.

AP2...147 **INSUFFICIENT STORAGE TO INITIALIZE DATABASE**

Explanation: The AP 127 SQL interface module could not obtain enough storage to contain the initialization parameter area.

System Action: The data base connection is not processed.

User Response: Allocate a larger machine or region to your user id or increase the FREESIZE and/or decrease the WSSIZE APL2 invocation options.

AP2...148 **INSUFFICIENT STORAGE TO ALLOCATE DATA BUFFER OF LENGTH** *number*

Explanation: AP 127 could not obtain enough storage for the shared variable data buffer size required. This could be due to the size of either the user request or the result table.

System Action: The request will not be processed.

User Response: Decrease the size of the object being passed to or retrieved by AP 127, or increase the amount of storage available (see message AP2...147).

AP2...149 SQL STATEMENT OF LENGTH *number* **IS TOO LONG**

Explanation: The SQL statement passed is longer than the maximum allowed by the data base.

System Action: The request will not be processed.

User Response: Shorten the length of the SQL statement.

AP2...150 VALUE PASSED IN POSITION *num1* **OF LENGTH** *num2* **IS TOO LONG**

Explanation: One of the items in the value list passed on an *OPEN* or *CALL* request is larger than the maximum allowed by the data base.

System Action: The request will not be processed.

User Response: Check the value and correct it.

AP2...152 THE AP 127 - DB2 SUBTASK HAS ABENDED WITH CODE *number*

Explanation: The subtask attached by AP 127 to communicate with DB2* has abended.

System Action: All prepared statements are purged. The auxiliary processor may still be used. A reattach will be issued if needed.

User Response: See your system programmer.

AP2...153 ERROR DURING DB2 CALL - RETURN CODE: *num1* **REASON CODE:** *num2*

Explanation: An error was encountered during a DB2 call that could not be translated into an SQLCODE. The DB2 return code and reason codes are documented in *IBM DATABASE 2 Messages and Codes*.

System Action: The data base connection is not processed.

User Response: See your data base administrator.

AP2...154 *operation* OF AP 127 SUBTASK FAILED - RETURN CODE: *number*

Explanation: The subtask which calls DB2 could not be attached due to the failure of the indicated operation.

System Action: The data base connection is not processed.

User Response: Contact your system administrator.

AP2...155 DB2 OPERATOR STOPPING SUB- SYSTEM - LINK TERMINATED

Explanation: The DB2 operator is taking your subsystem down. You cannot continue processing.

System Action: The data base connection is terminated.

User Response: See your data base administrator.

AP2...156 DB2 STOPPING ABNORMALLY - LINK TERMINATED

Explanation: DB2 is terminating abnormally. You cannot continue processing.

System Action: The data base connection is terminated.

User Response: See your data base administrator.

AP2...157 DB2 OPERATOR STOPPING SUB- SYSTEM - PLEASE END YOUR SESSION

Explanation: The DB2 operator is taking your subsystem down but is delaying to allow you to end your application normally.

System Action: The data base connection will be terminated shortly.

User Response: Finish your unit of work, and issue a *COMMIT* as quickly as possible.

AP2...158 AP 127 IS RESTARTED DUE TO ABEND – CODE: *number* **TYPE:** *character*

Explanation: An ABEND has been issued for AP 127. *number* is the numeric code. *character* is the type of error:

P for programmer

S for system

U for user

System Action: AP 127 is restarted. All statements are purged and the connection with the data base is broken.

User Response: Contact your system administrator.

AP2...159 DATA BASE INTERFACE LINKAGE NOT INSTALLED WITH APL2

Explanation: AP 127 cannot be accessed because prerequisite parts of SQL/DS* have not been installed with APL2.

The text deck ARIRVSTC must be link-edited into the APL2 LOADLIB.

System Action: The command will not be processed.

User Response: Contact your system administrator.

AP2...161 *command* **IS NOT VALID IN THIS OPER-
ATING SYSTEM.**

Explanation: A syntactically correct command has
been issued, but it is not a legal command for the oper-
ating system. For example, the PUT command is not
allowed in TSO (DB2), and the SSID command is not
allowed in CMS (SQL/DS).

System Action: The command will not be processed.

User Response: Do not issue the command in that
environment.

AP2...162 THE *argument* **ARGUMENT TO CONNECT
MUST BE A CHARACTER VECTOR OF
LENGTH 1 TO** *number*

Explanation: An argument has been passed to
CONNECT that is either not a character string or not of
correct length.

System Action: The command will not be processed.

User Response: Correct the argument and reissue
the request.

AP2...163 UNEXPECTED RETURN CODE *number*
FROM DSNTIAR

Explanation: In processing a *MSG* request, AP 127
received an unexpected return code from the DB2
routine DSNTIAR. This is probably an internal AP 127
error.

System Action: This message is issued as a warning.
The *MSG* request will be completed, and the SQLCA
contents will still be returned. The message text,
however, will be missing.

User Response: Contact your system programmer.

AP2...164 ERROR LOADING DB2 PROGRAM *name*

Explanation: AP 127 was unable to load the DB2
program named.

System Action: The connection to DB2 is not estab-
lished.

User Response: Check to see that the appropriate
DB2 libraries have been allocated to your user ID,
either in LPALIB, your LOGON PROC, or the CLIST
used to invoke APL2.

AP2...165 INVALID PARAMETER TO *operation*
OPERATION

Explanation: The parameter passed to the ISOL oper-
ation is not 'RR' or 'CS'; the parameter to the
DESCRIBE operation is not 'NAMES', 'LABELS',
'ANY', or 'BOTH'; the parameter passed to the
DECLARE operation is not 'HOLD'; or the parameter
passed to the COMMIT or ROLLBACK operation is not
'RELEASE'.

System Action: The command will not be processed.

User Response: Correct the invalid parameter, and
reissue.

AP2...166 INVALID EXTENDED CHARACTERS

Explanation: The extended characters passed have
been determined to be invalid. This can occur if the
character set ID in the data does not match the char-
acter set ID set for the APL2 session (via the DBCS
invocation option), or if the data contains characters not
within the defined range for DBCS characters.

System Action: The request will not be processed.

User Response: Verify the data and resubmit the
request.

**AP2...167 CANNOT SWITCH SUBSYSTEMS WHEN
CONNECTED**

Explanation: The SSID command has been issued
when the DB2 connection is already active.

System Action: The command will not be processed.

User Response: Retract the shared variable or issue
a ROLLBACK or COMMIT with the RELEASE option
before issuing the SSID command.

AP2...180 NO VALUE-LIST PASSED FOR PUT *name
name* **(AP2...180)**

Explanation: A *PUT* command was issued without a
value-list. A value-list is required for this command.

System Action: The command is not processed.

User Response: Reissue the command with a value-
list.

AP2...200 ARRAY OR INTERNAL FUNCTION DEPTH

Explanation: An array has been created which is
nested too deeply for the function being applied to it, or
some other internal recursion limit has been reached.
The Match, Depth and external functions, as well as
shared variable and copy operations, are limited to
depth 181. Selective specification is limited to depth
100.

Note: This error may also be signalled by an external
processor, or by a defined function or operator. Consult
documentation for the application, external processor, or
processor 11 function routine which signalled the error.

System Action: *SYSTEM LIMIT* is signalled, with
□*ET* set to 1 10. Message 210 will have been
produced; the present message is normally queued for
display with)*MORE*.

User Response: Check for undesired recursion. Be
sure that the data you provided is in the format
expected by the function or application, and does not

exceed its design limits in length or complexity. You may have to modify the function to avoid certain primitive operations on deeply nested arrays.

If you are not using arrays that are too deeply nested, the message may be caused by an error in the APL2 product.)*SAVE* the active workspace if possible, and contact your system administrator.

AP2...201 PROMPT LENGTH

Explanation: During a prompt/response interaction (⎕ output followed by ⎕ input) the system determines that it is not able to display the prompt on the user's terminal in such a way that it can be properly merged with the response. The specific length limits are dependent on the terminal type and access method.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *SYSTEM LIMIT* is signalled, with ⎕*ET* set to 1 11. Message 210 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: Check for incorrectly formatted prompt output. If a long prompt is required you may be able to extend the limit by using the APL2 session manager.

AP2...202 VALUE UNREPRESENTABLE

Explanation: An array value provided by an auxiliary processor or associated processor is not representable in the APL2 workspace.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *SYSTEM LIMIT* is signalled, with ⎕*ET* set to 1 12. Message 210 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: Contact the person responsible for the auxiliary processor, associated processor, or processor 11 function routine that is being used. That program will have to be corrected, but you may be able to avoid the problem by using different data or a different sequence of operations.

AP2...210 *SYSTEM LIMIT*

Explanation: The user or an executing program has invoked an APL2 function in accordance with the language definition, but in a way that exceeds some resource of the implementation.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: ⎕*ET* is set to 1 *n*, where *n* identifies the resource more specifically. A secondary message is queued which describes the problem. The point of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement. If the message occurs while executing a system command, typically)*COPY*,)*MCOPY*, or)*PCOPY*, the system command is terminated but ⎕*ET* is not set.

User Response: Display ⎕*ET* or enter)*MORE* if you do not understand the nature of the problem. Specific discussions of the resources and possible responses to each limit are provided for the corresponding secondary messages. Those messages can be found in this chapter as numbers 200 through 209 and 214 through 219.

AP2...211 *INTERRUPT*

Explanation: An interrupt or attention has been signaled from the terminal during processing. Execution has been halted.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: ⎕*ET* is set to 1 1, the point of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement. If the message occurs while executing a system command, typically)*COPY*,)*MCOPY*, or)*PCOPY*, the system command is terminated but ⎕*ET* is not set.

User Response: Enter an APL2 statement or system command. Execution can be resumed with a branch statement such as →ι0.

The interrupted statement appears in the state indicator. If execution is not resumed, the state indicator should be cleared with → or)*RESET*, as described in *APL2 Programming: Language Reference*.

AP2...212 *SYSTEM ERROR*

Explanation: An error was detected by the host system editor interface during ∇-edit processing. This may reflect a problem in reading or writing the file used to transfer the object being edited to or from the editor.

System Action: The ∇-edit process is terminated without modifying the object.

User Response: Analyze any additional messages that may be displayed or queued, or unusual actions taken within the editor (such as a change in file organization).

Explanation 2: A non-recoverable problem was detected by APL2 system interface code, by an external processor, or by a Processor 11 external routine.

System Action: If a system command was being executed it terminates unsuccessfully. If an APL statement is being processed it is suspended and □*ET* is set to 1 2.

User Response: If additional messages are displayed, either automatically or in response to)*MORE*, correct whatever problem is indicated by those messages and retry. If there are no additional messages, or only a message about an ABEND, try to recreate the problem with an APLDUMP file allocated. If you see one or more messages about dumps being taken, report the symptoms and the dumps you get to IBM service (or the author of the program if a non-IBM external routine is involved).

Explanation 3: A fault has occured in the internal operation of the APL2 interpreter, or the active workspace has been damaged. This case can be recognized by the additional output displayed as described under "System Action," below.

System Action: The system does all of the following steps when processing the error.

*   Displays several lines of system-related information that may be helpful to the system administrator in reporting and correcting the problem.

*   Saves the damaged workspace in a workspace named *DUMPnnn*.

*   Clears the active workspace.

*   Resets □*ET* to 0 0.

*   Prompts with 6 blanks for the next input.

User Response: This problem should be reported to IBM service. If the session manager is in use, copy the screen where the *SYSTEM ERROR* message appeared. If the session manager is not in use and your terminal has a print screen function, print the screen. If you cannot copy or print the screen from the terminal, contact your system administrator to determine whether the information should be saved.

You may also want to do the following:

*   In a clear workspace, use the)*COPY* or)*PCOPY* command to attempt to restore objects from the *DUMPnnn* workspace. Examine and retest the copied objects to be sure that they have not been damaged.

*   Drop the *DUMPnnn* from the library with the)*DROP* system command.

*   For more information, see *APL2 Diagnosis*.

AP2...213 *WS FULL*

Explanation: An attempt has been made to execute an operation that requires more storage than is currently available.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: □*ET* is set to 1 2, the point of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement. If the message occurs while executing a system command, typically)*COPY*,)*MCOPY*, or)*PCOPY*, the system command is terminated but □*ET* is not set.

User Response: One of the following responses may be required.

*   Issue the system command)*RESET* to clear the state indicator.

*   Issue the system command)*ERASE* to remove unnecessary objects from the active workspace.

*   Revise calculations to use less space.

*   )*SAVE* the workspace and reinvoke APL2 with a larger *WSSIZE*

AP2...214 SYMBOL TABLE

Explanation: The symbol table limit has been exceeded. The APL2 symbol table is limited to 32767 entries. Entries are required for each unique name, each unique constant, and each ill-formed constant in the workspace. (The symbol table may be further constrained by the size of your workspace, but in that event error 213 will be signalled in place of 210 and this one.)

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *SYSTEM LIMIT* is signalled, with □*ET* set to 1 4. Message 210 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: Remove some objects from the active workspace. One of the following responses may be used to correct the problem.

- Unneeded objects may be deleted.

- The workspace may be divided into two or more smaller workspaces.

- Functions may be stored in a file, and read and fixed ($\Box FX$) as needed.

- Part of the workspace may be packaged and accessed via $\Box NA$.

AP2...215   NO SHARES

Explanation: Either the shared variable processor is not available, or your processor number (the first element of $\Box AI$) is already in use. The most common case is that you have specified an ID number during APL2 invocation that matches that used by one of the auxiliary processors started during invocation.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $SYSTEM\ LIMIT$ is signalled, with $\Box ET$ set to $1\ \ 5$. Message 210 will have been produced; the present message is normally queued for display with $)MORE$.

User Response: Reinvoke APL2 with a different ID number, or without an auxiliary processor that is using your number. If this does not solve the problem, contact your system programmer. Invoking APL2 with the TRACE(1) parameter may help the system programmer to isolate the problem.

AP2...216   INTERFACE QUOTA

Explanation: The active workspace is attempting to use more shared variables than the limit set by the SVMAX invocation option.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $SYSTEM\ LIMIT$ is signalled, with $\Box ET$ set to $1\ \ 12$. Message 210 will have been produced; the present message is normally queued for display with $)MORE$.

User Response: Retract a shared variable that is no longer needed, or reinvoke APL2 with a larger SVMAX value. The current SVMAX value is shown on the third line of the $)QUOTA$ command response.

AP2...217   INTERFACE CAPACITY

Explanation: This message indicates a problem with the use of shared variables. It may occur either while offering a variable for sharing ($\Box SVO$) or while assigning a value to a shared variable. There are two possible causes:

1. The value being assigned to a shared variable (or the value that the variable already contains, if it is being offered) is too large to fit in shared memory. This may be because the value is too large for the defined size of shared memory, or it may be that other shared variable values are currently using too much of the defined space. In either case, this is the logical equivalent in shared memory of $WS\ FULL$ in the workspace.

2. The shared variable processor is unable to accept additional processors. This can happen only during $\Box SVO$, and only the first time that an offer is made from the current active workspace. There are separate quotas for global shared memory and local shared memory. The local quota is based on the number of auxiliary processors started at APL2 invocation, and should not normally cause a problem. The global quota is set by your installation.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $SYSTEM\ LIMIT$ is signalled, with $\Box ET$ set to $1\ \ 7$. Message 210 will have been produced; the present message is normally queued for display with $)MORE$.

User Response: Try to offer a non-existent variable to a non-existent processor. For example,

   $1\ \Box SVO\ 'JUNK'$

If this fails in the same way, you are experiencing the second problem described above.

If you are authorized to use the global SVP, the problem is probably due to the global quota. It is likely to be transient, and resolved by the action of some other APL2 user. Use $\rightarrow''$ to resume execution at the shared variable offer which triggered the SIGNON. If the problem persists, contact your system programmer. The number of processors permitted is controlled by the MAXPROC keyword of the SVPPARMS file under TSO, and by the last parameter of the AP2CSVP START command under CMS.

If the problem occurs while signing on to the local SVP, you appear to be using an auxiliary processor which does multiple signons. Contact the owners of auxiliary processors that are started during APL2 invocation.

If you are able to offer a variable with no value, then you are experiencing the first problem described above.

If the problem occurred while assigning a value to a shared variable, assign that same value instead as a new array within the workspace. Now use that new name (or the name of the original variable if the problem occurred while offering a variable) as a quoted right argument to: 4 $\Box AT$ **'name'**. The first number returned will be the space required in shared memory to hold the variable value.

Now type the system command)$QUOTA$. If the last number returned ($SIZE$) is not considerably larger than the space needed by your variable, you will need to reinvoke APL2 with a larger SHRSIZE.

If the current SHRSIZE looks more than adequate, you may have other unreferenced variable values currently in shared memory. Your workspace may have failed to reference a large value when intended.

If the problem occurs while assigning a value to a variable shared with an independent processor (another user, for example) the storage involved is the global shared memory defined by your installation. The SHRSIZE invocation option has no effect on this storage. In this case, the problem is likely to be transient, and resolved by the action of some other APL2 user. Use →'' to resume execution at the shared variable specification. If the problem persists, contact your system programmer. The space available is controlled by the SMSIZE keyword of the SVPPARMS file under TSO, and by the size of the global shared storage saved segment under VM.

AP2...218 ARRAY RANK

Explanation: The maximum rank of an array in APL2 is 64. The interrupted statement is creating an array which would exceed this limit.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $SYSTEM\ LIMIT$ is signalled, with $\Box ET$ set to 1 8. Message 210 will have been produced; the present message is normally queued for display with)$MORE$.

User Response: The function probably contains an error in calculating the left argument to Reshape, or a loop involving lamination, outer product, or some other operation which increases the rank of the result. If you really need this many dimensions, consider restructuring the array to use depth, or a combination of depth and rank.

AP2...219 ARRAY SIZE

Explanation: The interrupted statement was attempting to create an array which either had a dimension that was too long or would have used too much storage.

The maximum length of an array along one axis (dimension) is $^-1+2*31$, or a little more than 2,000,000,000.

The maximum size of any simple array or simple item in a nested array is $2*24$ bytes of storage. APL2 uses a variety of internal representations for numbers and characters, so that it is difficult to state rules in terms of the number of elements in the array. However the condition can only arise in MVS/XA* systems with workspaces larger than 16 megabytes.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $SYSTEM\ LIMIT$ is signalled, with $\Box ET$ set to 1 9. Message 210 will have been produced; the present message is normally queued for display with)$MORE$.

User Response: Use $\Box L$ and $\Box R$ to help understand why the array has become so large. If the problem is with the length along one axis, there is probably an error in calculating the lengths of the dimensions, or a loop in catenating arrays. Only empty arrays can have this large a dimension, as other limits would be reached first on any non-empty arrays.

The second element of 4 $\Box AT$ can be used to determine the space required in bytes for a simple item. For performance reasons, APL2 does not continually remap data into the most compact representation possible. Boolean values, for example, can be represented more compactly than integers, and APL2 will use the most compact form for arrays that are created from boolean constants or as the result of boolean functions. But if non-boolean values are removed from an integer array, the interpreter has no way of knowing if this was fortuitous or represents a new view of the nature of the array. You may be able to force a more compact representation of an object by applying transformations that can only yield boolean, integer, or one-byte character results. For example:

```
BOOL←1=BOOL
INT←⌊INT
CHAR←⌷AV[⌷AVιCHAR]
```

Another solution is to restructure the simple array as a nested array.

AP2...220 *SYNTAX ERROR*

Explanation: The displayed APL2 statement is constructed improperly.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $\Box ET$ is set to 2 *n*, where *n* identifies the error more specifically. A secondary message is queued which describes the problem. The point of interruption is indicated, and the user is prompted (by 6 blanks) to enter some input statement.

User Response: Display $\Box ET$ or enter)*MORE* if you do not understand the nature of the problem. Specific discussions of the types of errors and possible corrections to each are provided for the corresponding secondary messages. Those messages can be found here as numbers 221 through 225.

AP2...221 NO ARRAY OR OPERAND

Explanation: Either a monadic or dyadic function was invoked without a right argument, or an operator was invoked without a required operand.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *SYNTAX ERROR* is signalled, with $\Box ET$ set to 2 1. Message 220 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: Check the argument or operand requirements of the primitive, derived, or defined operation. Note that functions may be left ambi-valent (left argument optional), but not right ambi-valent. Operators are not ambi-valent.

AP2...222 ILL-FORMED LINE

Explanation: The APL2 statement cannot be parsed properly. It may contain:

- unmatched parentheses, brackets, or apostrophes.
- improperly used "**:**" or "**;**."
- unsupported syntactic elements (such as ω or $\$$).

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *SYNTAX ERROR* is signalled, with $\Box ET$ set to 2 2. Message 220 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: Check the token pointed to by the caret. It is usually unmatched or used improperly. If that token is correct, check the one immediately to its right. For paired symbols, check for proper nesting as well as pairing. $A \leftarrow A[3 \times (B+C])$), for example, is not properly nested. Apostrophes within quoted strings must be doubled: '*DON''T FORGET*'. Consult *APL2 Programming: Language Reference* for more information. Read especially the chapter titled "Syntax and Expressions" and the appendix on "The APL2 Character Set."

AP2...223 NAME CLASS

Explanation: Assignment to something other than a variable or an unused name.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *SYNTAX ERROR* is signalled, with $\Box ET$ set to 2 3. Message 220 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: Be sure that the object of assignment is not a constant, or a name in use as a label, function, or operator.

AP2...224 INVALID OPERATION IN CONTEXT

Explanation: You are using ∘ or ← in an unsupported fashion, or are making unauthorized use of an internal system facility.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *SYNTAX ERROR* is signalled, with $\Box ET$ set to 2 4. Message 220 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: The rightmost or only caret should point to the operation which is unsupported. The problem can also arise when using selective specification. Only those functions specifically listed in *APL2 Programming: Language Reference* can be used in a parenthesized expression to the left of an assignment arrow.

AP2...225 SYNTAX PROHIBITED IN)*CS* MODE

Explanation: You are running APL2 in a VS APL compatibility mode, and have encountered a construction which would be valid in APL2 but was not in VS APL.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *SYNTAX ERROR* is signalled, with □*ET* set to 2 5. Message 220 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: We strongly recommend that you not use compatibility mode. Enter)*CS* 0 to return to full APL2 capabilities.

AP2...230 *VALUE ERROR*

Explanation: An array value was required while executing an APL2 statement, but no value is currently available.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: □*ET* is set to 3 *n*, where *n* identifies the error more specifically. A secondary message is queued which describes the problem. The point of interruption is indicated, and the user is prompted (by 6 blanks) to enter some input statement.

User Response: Display □*ET* or enter)*MORE* if you do not understand the nature of the problem. Specific discussions of the types of errors and possible corrections to each are provided for the corresponding secondary messages. Those messages can be found here as numbers 231 and 232.

AP2...231 NAME WITH NO VALUE

Explanation: A name is used in an APL2 statement, but has not been defined in the active workspace or has no current definition.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *VALUE ERROR* is signalled, with □*ET* set to 3 1. Message 230 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: You may have misspelled the name or accidentally erased the object, or else a defined function or operator on the stack may have localized the

name without assigning it a value. Use)*NMS* to see if a user-defined name exists globally in the workspace. Use)*SINL* to determine which currently suspended functions have localized the name. If the error occurs on a □-*name*, □*NC* '□..' for that name will return 0 if it is defined but currently has no value, but will return ¯1 for undefined names.

AP2...232 FUNCTION WITH NO RESULT

Explanation: A defined function has been invoked in a context which requires that it return a result, but it either has been defined not to return a result or it failed to assign a value to its result variable.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: *VALUE ERROR* is signalled, with □*ET* set to 3 2. Message 230 will have been produced; the present message is normally queued for display with)*MORE*.

User Response: The fault lies either in the way the function was defined or in the way you are attempting to use it. Some functions are designed to display their output directly rather than returning it to their callers. This design restricts the ways in which the function can be used. If you are dealing with such a function, you will have to invoke it as the principal function (leftmost, and not in parentheses) in any statement.

AP2...240 □*name ERROR*

Explanation: □*name* is a system variable whose value was required as an an implicit argument. The system variable either has no value or an invalid value at the moment. Each of the possible system variable names is listed below together with the □*ET* value assigned, further explanation, and suggestions for user responses. See "APL2 Programming: Language Reference" for definitions of the system variables.

System Action: □*ET* is set to 4 *n*, where *n* identifies the system variable as indicated below. The point of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement.

□*CT ERROR*   □*ET*←4 3

Explanation 2: An attempt has been made to execute a primitive function that uses □*CT* as an implicit argument when □*CT* has an invalid value or no value.

User Response: Check the value of □*CT* and assign it a valid value. If no particular value is required, assign □*CT* the value 1*E*¯13, which is the system default.

□*FC ERROR*   □*ET*←4 4

Explanation 3: An attempt has been made to do the following.

- Execute a primitive function that uses $\Box FC$ as an implicit argument, but $\Box FC$ has an invalid value or no value.

- Display a negative number (with $L\bar{\Phi}R$) when $\Box FC[6]$ has an inappropriate value or no value.

User Response: $\Box FC$ can be reset to the system default by executing $\Box FC \leftarrow ''$.

$\Box IO\ ERROR$ $\Box ET \leftarrow 4\ 2$

Explanation 4: An attempt has been made to execute a primitive function that uses $\Box IO$ as an implicit argument when $\Box IO$ has an invalid value or no value.

User Response: Respecify the value of $\Box IO$ to 0 or 1.

$\Box PP\ ERROR$ $\Box ET \leftarrow 4\ 1$

Explanation 5: An attempt has been made to do either of the following.

- Execute a primitive function that uses $\Box PP$ as an implicit argument when $\Box PP$ has an invalid value or no value.

- Display an array when $\Box PP$ has an invalid value or no value.

User Response: Check the value of $\Box PP$. If no particular value is required, assign $\Box PP$ the value 10, which is the system default.

$\Box PR\ ERROR$ $\Box ET \leftarrow 4\ 7$

Explanation 6: An attempt has been made to use the $\Box$ system variable to create a character prompt immediately followed by a request for character input. However, $\Box PR$ is invalid or has no value.

User Response: Check the value of $\Box PR$. If the situation requires no particular value, assign $\Box PR$ the value $\Box PR \leftarrow '\ '$, the system default.

$\Box RL\ ERROR$ $\Box ET \leftarrow 4\ 5$

Explanation 7: An attempt has been made to execute a primitive function that uses $\Box RL$ as an implicit argument when it is invalid or has no value.

User Response: Check the value of $\Box RL$. If the situation requires no particular value, assign $\Box RL$ the value 16807, the system default.

AP2...249 $WS\ DUMPED,\ ID:\ \ DUMPnumber$

Explanation: The *number* is a four-digit sequential dump number. This message may follow a $SYSTEM$ $ERROR$. It indicates that a copy of the workspace has been stored exactly as it existed at the time of the $SYSTEM\ ERROR.$

System Action: Recovery continues by displaying a register dump and providing a $CLEAR\ WS$.

User Response: Depending on the extent of damage, you may be able to retrieve objects from the dumped workspace by using
$)COPY\ DUMPnumber$. You should preserve the dumped workspace for diagnostic purposes if you report the error. When you no longer need the damaged copy, delete it with the $)DROP$ command.

AP2...250 (*timestamp or $\Box ES$ application-defined content*)

Explanation: This is the normal response to $)DROP$ or to $)SAVE$ when a new workspace name is specified on the command. This message may also be produced by dyadic $\Box ES$.

System Action: Processing continues.

User Response: None; informational message only.

AP2...251 $VALENCE\ ERROR$

Explanation: An attempt has been made to specify a left argument for a monadic function, or to specify a single argument for a dyadic function, or to execute a function declared through the use of dyadic $\Box NA$ whose definition cannot be activated.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $\Box ET$ is set to $5\ 1$, the point of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement.

User Response: Clear the suspension, then reinvoke the function with the correct number of arguments.

AP2...252 $RANK\ ERROR$

Explanation: An array specified as the argument of a function or operand of an operator has a rank that is incompatible with another argument or operand. If the array is nested, the incompatibility may exist below the top level of structure.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $\Box ET$ is set to $5\ 2$, the point of of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement.

User Response: For the rules governing the valid array structures for the particular function or operator, see *APL2 Programming: Language Reference*.

AP2...253 *LENGTH ERROR*

Explanation: An argument of a primitive function or operand of a primitive operator has an axis whose length is incompatible in relation to that of another argument or operand.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $\Box ET$ is set to 5 3, the point of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement.

User Response: For the particular primitive function or operator, see the appropriate arguments and operands in *APL2 Programming: Language Reference*.

AP2...254 *DOMAIN ERROR*

System Action: $\Box ET$ is set to 5 4, the point of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement.

One of the following explanations applies.

Explanation: The data type, degree of nesting, or number of arguments specified for a primitive function is invalid.

User Response: For a description of the number, type, and structure of arguments required for a particular primitive function, see *APL2 Programming: Language Reference*.

Explanation: An operand or argument is incompatible with an operator.

User Response: For a description of the operands and arguments required for a particular primitive operator, see *APL2 Programming: Language Reference*.

Explanation: A calculation requires or has produced data that is beyond the range of the system implementation and does not fit any of the categories of *SYSTEM LIMIT*. This can occur with some mathematical functions.

User Response: Reduce the size of the calculation. For a description of specific system limits, see *APL2 Programming: Language Reference*.

Explanation: A defined function or operator is defined to convert any nonresource error to a *DOMAIN ERROR* because 4⊃3 $\Box AT$ *FN* is 1.

User Response:

1. Correct the argument and reexecute the defined function or operator.

2. Check the attributes (3 $\Box AT$ *FN*) of the object causing the error. Also, check the $\Box LC$ system variable to identify the last line executed.

If the function or operator can be displayed, you can also use the following expression to change the conversion attribute, so that the actual error message is returned:

    0 0 0 0 □FX □CR 'FN'

The following expression resets the conversion attribute:

    0 0 0 1 □FX □CR 'FN'

Explanation: A derived function from the slash operator or inner product was presented with an empty argument, but no identity function existed for the function operand, **or** a derived function from the each operator or inner product was presented with an empty argument, but no fill function existed for the function operand.

User Response: Provide a cover function or operator to search for an empty argument before using each, slash, or inner product with a defined function as the operand, if the possibility of receiving an empty argument exists.

Explanation: An external processor, or defined function or operator, has signalled the error.

User Response: Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

AP2...255 *INDEX ERROR*

Explanation: The index specified in L for bracket indexing ($R[L]$) or pick ($L⊃R$) is invalid in relation to the array given as the argument R.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $\Box ET$ is set to 5 5, the point of interruption is indicated, and the user is prompted (by 6 blanks) to enter an input statement.

User Response: Be sure that $\Box IO$ is set as the statement requires. Check the value of $\Box L$ against the shape of $\Box R$. Note that $\Box L$ may be a nested array. The *DISPLAY* function in workspace 1 *DISPLAY* will probably be helpful. If the error occurs in ⊃ you may need to look at the internal structure of $\Box R$. If it is reasonably small, the simplest approach is *DISPLAY* $\Box R$. For large arrays, check ρ(i↑$\Box L$)⊃$\Box R$, varying i.

AP2...256 *AXIS ERROR*

Explanation: One of the following problems exists.

- Axis specification cannot be applied to the given operation.

- The axis specification contains semicolons.

- The shape of the axis specification is incompatible with the function or operator it is being applied to.

- The value of the axis specification is incompatible with the shape of the given arguments.

Note: This error may also be signalled by an external processor, or by a defined function or operator. Consult documentation for the application, external processor, or processor 11 function routine which signalled the error.

System Action: $\Box ET$ is set to 5 6, the point of interruption is indicated, and the user is prompted (by 6 blanks) to enter some input statement.

User Response: Check the function or operator which immediately precedes the axis left bracket. Only $\phi\Theta\uparrow\downarrow\subset\supset$,$/\!\!\!\backslash\!\!\!\backslash$ ⍀ are valid. Check that the expression within the brackets does not contain a semicolon. If it does, it may have been intended for indexing rather than axis.

The value of the axis expression is not directly available through $\Box L$ or $\Box R$, but in most cases it is a constant or easily recalculated. For $/\!\!\!\backslash\!\!\!\backslash\phi\Theta$ and monadic "," the value must be a scalar or 1-element vector. In most cases it must be a non-negative integer which does not exceed the rank of the right argument. Check $\Box IO$ to see whether 0 or the rank value is permitted. If the axis is being applied to "," or "⊃" additional values are permitted. See *APL2 Programming: Language Reference* for details of these.

Since axis values are not available in a system variable, the interrupted primitive cannot be resumed after correcting an axis specification. On occasion you may be able to modify $\Box R$ or $\Box IO$ and continue with $\rightarrow\iota 0$ but you will often need to edit the statement or change the axis value. In these cases you will need to restart at the beginning of the statement using $\rightarrow\Box LC$.

AP2...260 COMMAND NUMBER *nn* **IS NOT SUPPORTED**

Explanation: This problem may occur if you issue a system command that is defined in a national language table, but that table specifies an incorrect command identification number *nn* for the command name you have just used.

System Action: The command is ignored, and APL2 waits for your next input.

User Response: Type $\Box NLT$ to determine the name of the national language currently in use. Report that name, the command name you used, and *nn* from this message, to your system administrator.

System Programmer Response: Look in section 3 of the APL2LANG file whose name was returned by $\Box NLT$ for the command name used on the system command that caused the problem. Check the command number given there against the list of valid command numbers in the AP2CMDC macro.

AP2...261 *timestamp*

Explanation: This is the normal response to $)TIME$.

System Action: Processing continues.

User Response: None; informational only.

AP2...262 $SENT$

Explanation: This is the normal response to $)MSG$, $)MSGN$, $)OPR$, or $)OPRN$.

System Action: Processing continues.

User Response: None; informational only.

AP2...263 $NOT\ SENT$

Explanation: This is the error response to $)MSG$, $)MSGN$, $)OPR$, or $)OPRN$. The system was unable to send a message to the requested recipient. This may be because the recipient is not receiving messages, is not signed on, or does not exist.

System Action: The request is ignored.

User Response: Be sure that you have provided the correct user ID if the command was $)MSG$ or $)MSGN$.

AP2...264 WAS *wsid*

Explanation: Normal response to $)WSID$ when a workspace name is specified.

System Action: Processing continues.

User Response: None; informational only.

AP2...265 IS *status*

Explanation: Normal response to the query form of any of the following commands: $)CS$, $)EDITOR$, $)HOST$, $)PBS$, $)SYMBOLS$, or $)WSID$. *status* shows the value currently associated with the system command.

Note: For $)PBS$, the underbar character is shown if $)PBS\ ON$ is in effect. If $)PBS\ OFF$ is in effect, message 266 is issued instead.

System Action: Processing continues.

User Response: None; informational only.

AP2...266 $IS\ OFF$

Explanation: Normal response to the query form of $)PBS$ if $)PBS\ OFF$ is in effect.

System Action: Processing continues.

User Response: None; informational only.

AP2...267 *WS CONVERTED*, *RESAVE*

Explanation: A)*LOAD* command names a work-space that was saved using an earlier release of APL2, and the saved form is not internally compatible with the current release of APL2.

System Action: The active copy of the workspace has been converted to the format used by the current release. Processing continues.

User Response: To save processing time and avoid the message in the future you may want to issue)*SAVE* for the converted workspace. Be aware, however, that the newly saved version cannot then be processed by the earlier release of APL2.

AP2...269 *WS CANNOT BE CONVERTED*

Explanation: A)*LOAD* command names a work-space that was saved using an earlier release of APL2, and the saved form is not internally compatible with the current release of APL2. An attempt has been made to convert the source workspace to the current format, but the attempt has failed because the active workspace size is too small.

System Action: The workspace is replaced by CLEAR WS.

User Response: Provide a larger active workspace size. If you had specified a value on the)*LOAD* command, try increasing it. Otherwise you must rein-voke APL2 with a larger WSSIZE and retry the command.

AP2...270 *timestamp wsid*

Explanation: Normal response to)*SAVE* when a new workspace name is not specified on the command.

System Action: Processing continues.

User Response: None; informational only.

AP2...271 *LIB size1 FREE size2*

Explanation: This is the first line of the normal response to)*QUOTA*. Under CMS, it gives the size of the minidisk containing your default library (as defined by *LIBTAB APL2*) and the space currently free on it. Under TSO, both numbers are shown as zero.

System Action: Processing continues.

User Response: None; informational only.

AP2...272 *WS size1 MAX size2*

Explanation: This is the second line of the normal response to)*QUOTA*. It gives the size of your active workspace and the largest workspace size available to you during the current APL2 session.

System Action: Processing continues.

User Response: You affect the *WS* value by speci-fying a size on)*LOAD* or)*CLEAR*. The *MAX* value is set from WSSIZE or its default at APL2 invocation.

AP2...273 *SV number SIZE size*

Explanation: This is the final line of the normal response to)*QUOTA*. It gives both the maximum number of variables you can share concurrently, and the size of shared memory. Shared memory is used to contain information on shared variables and shared var-iable processors, as well as any shared variable values which have not yet been seen by both partners.

System Action: Processing continues.

User Response: You may set both of these values at APL2 invocation. The *SV* value is set from SVMAX, and *SIZE* value is set from SHRSIZE.

AP2...274 *SAVED timestamp*

Explanation: This is the normal response to)*COPY*,)*PCOPY*, or)*MCOPY*, and to)*LOAD* when the size of the saved workspace is the same as specified or defaulted.

System Action: Processing continues.

User Response: None; informational only.

AP2...277 *CONNECTED hrs.min.secs*

Explanation: Displayed during APL2 termination.

System Action: Termination continues.

User Response: None, unless you want to record your usage of APL2.

AP2...278 *CPU TIME hrs.min.secs*

Explanation: Displayed during APL2 termination.

System Action: Termination continues.

User Response: None, unless you want to record your usage of APL2.

AP2...280 *WS NOT FOUND*

Explanation: The workspace specified with a)*COPY*,)*PCOPY*,)*MCOPY*,)*DROP*, or)*LOAD* command does not exist, or cannot be located using your current library structure.

System Action: The previous contents of the active workspace are retained.

User Response: Be sure you have specified the correct library number and workspace name. Try a)*LIB* command for that library to see what it does contain. If you know that the workspace is supposed to exist, and it is not displayed by the)*LIB* command, you will have to change the view of libraries made available to APL2.

For CMS the library view is controlled by the LIBTAB APL2 file and the particular set of minidisks you currently have accessed. You may have your own LIBTAB APL2 file, or you may be using one provided by the system or another user. If your library view has changed recently you must be either accessing a different set of minidisks, or using a LIBTAB APL2 file which has been recently updated. LIBTAB APL2 is described in *APL2 Programming: System Services Reference*. See the chapter on APL2 Libraries, Workspaces, and Data Files.

For TSO the library view is controlled by ALLOCATE FILE(Wnnnn) statements and by constants specified when APL2 is installed. The installation can also affect the view by replacing MVS catalogs. The only part that users can control is the ALLOC statements for FILE(Wnnnn). These are described in *APL2 Programming: System Services Reference*. See the discussion of accessing VSAM libraries in the chapter on APL2 libraries, workspaces, and data files.

AP2...281 *WS LOCKED*

Explanation: The password specified with a)*COPY*,)*MCOPY*,)*PCOPY*, or)*LOAD* command differs from that for the library workspace.

User Response: Reissue the command, specifying the correct password.

AP2...283 *SI WARNING*

Explanation: Currently incomplete work may not be able to proceed as originally intended. There are three cases:

Case 1   You have replaced a currently suspended or pendent defined function or operator, and the new copy has changes on or before the current statement line. The message will occur during)*COPY* or at completion of ∇-edit, and will recur if you attempt to resume at the point of interruption within the line of the function by branching with an empty right argument.

Case 2   You have replaced a currently suspended or pendent defined function or operator, and the new copy has changes in the header line or adds or removes labels. The message will occur during)*COPY* or at completion of ∇-edit, and will recur if you attempt to resume execution of the function by branching anywhere in it.

Case 3   An error is detected during execution, but the workspace is so nearly full that the currently executing statement cannot be added to the execution stack.

System Action: This varies for the three cases listed above.

Case 1   The replacement is successful, but the stack entry is marked to prevent resumption at the point of interruption within the current statement.)*SI*,)*SINL*, and)*SIS* will report a negative line number. Attempts to branch with an empty right argument cause execution to remain suspended at the same statement in the modified function or operator.

Case 2   The replacement is successful, but the stack entry is marked to prevent restart of the function or operator. The corresponding element of □*LC* is set to zero.)*SI*,)*SINL*, and)*SIS* will report no line number, i.e. name[]. Attempts to branch anywhere in the function or operator cause execution to remain suspended with the same entry at the top of the stack.

Case 3   Execution is suspended at the call to the current function or operator.

User Response: Issue one or more of the system commands)*SI*,)*SINL*, and)*SIS* to check the line, where the altered or replaced function or operator is suspended or pendent.

Lines reported by these system commands for altered operations show:

- Negative line numbers

- No line numbers

If the operation is identified with no line number, it cannot be restarted. You may take either of these actions:

- Exit by branching to a line number outside the function or operator. (→□*LC* will accomplish this, since the first element of □*LC* has been set to zero.)

- Use the)RESET command or → with no arguments to clear all or part of the stack.

AP2...284 *SAVED timestamp size1K(size2K)*

Explanation: This is the normal response to)*LOAD* when the size of the saved workspace is different from that specified or defaulted on the)*LOAD* command. The first number shows the size of the active workspace, as just loaded. The number in parentheses is the size of that workspace at the time it was last saved.

System Action: Processing continues with the workspace now a different size.

User Response: None, unless you think that the first number is too small for the work that you need to accomplish with the workspace. If so, you will probably need to exit from APL2, using)*OFF*, and then reinvoke it with a larger *WSSIZE*.

AP2...285 *NOT SAVED, LIBRARY FULL*

Explanation: The space allotted for saving workspaces is full, or the remaining space is not large enough to save the workspace.

System Action: The active workspace remains available. If a saved copy of the workspace previously existed in the library, that copy will have been destroyed if you are on a TSO system. On CMS any saved copy is still valid.

User Response: Do one of the following steps:

- Issue the)*DROP* command to drop workspaces no longer needed.

- Use the host system facilities to remove non-APL2 files no longer needed.

- Request additional storage space from system administration.

- Issue the)*OUT* command to write the objects in the workspace to a transfer file. A transfer file may require less storage space than a workspace and is not stored in an APL2 library.

AP2...286 *NOT SAVED, THIS WS IS wsid*

Explanation: You have issued a)*SAVE* command specifying a workspace library and/or name different from that of the active workspace. A workspace already exists with the specified name in that library. The system is trying to keep you from inadvertently overlaying the earlier workspace.

System Action: The)*SAVE* is not performed. The earlier workspace is retained.

User Response: If you did not intend to overlay the earlier workspace, reissue the)*SAVE* using a different name or library number. If you do want to replace the workspace, first use)*WSID* to make the active workspace name match that of the library workspace. You may then use)*SAVE* with or without specifying the library and name.

AP2...287 *NOT SAVED, THIS WS IS CLEAR WS*

Explanation: You have issued a)*SAVE* command, either with or without a workspace library and/or name specified, and the active workspace is unnamed. If you had specified a name, see message 286 above rather than this one. If you issued)*SAVE* without a name, the system does not know where to save the workspace.

System Action: No action is taken.

User Response: Reissue the)*SAVE* command, specifying a desired workspace name and optionally a library. This will create the stored workspace and set the active workspace name.

AP2...288 *NOT FOUND: object-names*

Explanation: You have issued a)*COPY*,)*IN*,)*MCOPY*, or)*PCOPY* command, and requested objects that could not be found. "Object-names" will normally include names you listed on the command, and/or names found in an indirect copy list. If you issued)*IN*, "object-names" may instead be the file or dataset name from which the objects are to be copied. This indicates that the file could not be found.

System Action: Any requested objects that could be found are placed in the active workspace, unless they are listed in some other message.

User Response: If)*IN* indicates that the file is not found, you may need to use operating system facilities to determine the correct file name. These can be accessed through)*HOST*. You might use)*HOST LISTFILE* under CMS, or)*HOST LISTCAT* under TSO. Your installation may have made other tools available to you.

If specific objects are not found, you may need to)*SAVE* the active workspace and browse for the needed objects in a number of workspaces or transfer files.

Explanation 2: You have issued an)*OUT* command specifying a member of an MVS partitioned dataset. The dataset could not be found.

System Action: This message is displayed with no *object-names* list, and the command is terminated without taking any action.

User Response: Correct the partitioned dataset name, or create a PDS with the proper name. (ISPF option 3.2 can be used for this purpose.)

AP2...289 *NOT ERASED: object-names*

Explanation: The listed objects were not erased by the)*ERASE* command because they do not exist as global objects in the active workspace.

System Action: All requested names that are not listed have been erased.

User Response: Issue an)*NMS* command to check that the correct object names were specified with the)*ERASE* command. No action is necessary if the objects do not exist in the active workspace. If they do, reissue the)*ERASE* command, specifying the correct names.

If the objects are local to a pendent or suspended function, they will not be displayed by)*NMS* and cannot be

removed by)*ERASE*. Use □*EX* instead to erase the current referent.

AP2...290 *NOT COPIED*: *object-names*

Explanation: The listed objects were not copied by the)*PCOPY* system command because the objects already exist in the active workspace.

User Response: None. The message is informational only, unless you want to replace an object in the active workspace. In that case, issue the)*COPY*, command specifying that object.

Explanation 2: The listed objects were not copied by the)*COPY*,)*MCOPY*,)*PCOPY*, or)*IN* system command because the objects will not fit in the active workspace.

User Response: Issue the)*ERASE* system command to erase unnecessary objects from the active workspace, or restart APL2, increasing the WSSIZE option to obtain more room for the active workspace. See *APL2 Programming: System Services Reference*.

Explanation 3: The listed objects specified with the)*IN* system command do not have valid transfer forms in the file specified.

User Response: Check the name of the file to be sure you have specified the correct file. If the transfer file name is correct, the objects do not exist in valid transfer form in the file.

Explanation 4: The listed objects specified with the)*OUT* system command were not written to a transfer file because the objects do not exist in the active workspace or cannot be transferred.

User Response: Use □*NL* to obtain a list of objects currently active in the workspace, and ensure that only those objects are specified in the)*OUT* command.

If the objects named are shared variables, check the degree of coupling (□*SVO* '*name*') and retract them, if necessary, before issuing the)*OUT* command. Shared variables cannot be written to a transfer file. If global objects are named that have been shadowed, only the local referent can be processed by)*OUT*. To transfer the global object, issue)*RESET* first.

AP2...291 *WS INVALID*

Explanation: A Load or Copy type system command has been issued against a file which is not in a supported workspace format. This includes (but is not limited to) the following special cases:

-)*LOAD*,)*COPY*, or)*PCOPY* from a VS APL workspace.

-)*LOAD*,)*COPY*, or)*PCOPY* from an APL2 workspace saved by a later release of the product, and being retrieved by an earlier release.

-)*MCOPY* from an APL2 workspace.

-)*LOAD* from a *DUMPnnnn* workspace.

This message may also be displayed by the *PACKAGE* external function if its right argument is the name of a file that would be invalid for)*LOAD* or)*COPY*.

System Action: The command is ignored.

User Response: One or more of the following responses may be appropriate.

- Use)*COPY* rather than)*LOAD* to access data in a *DUMPnnnn* workspace.

- Use)*COPY* rather than)*MCOPY* to access an APL2 workspace.

- Use)*MCOPY* rather than)*LOAD* or)*COPY* to access a VS APL workspace.

- To migrate a workspace from a later release of APL2 to an earlier one, first)*LOAD* or)*COPY* the workspace while executing under the later release. Then use)*OUT* to create a transfer file. Finally, use)*IN* to retrieve the workspace while executing under the earlier release.

AP2...292 *LIBRARY NOT AVAILABLE*

Explanation: The)*CONTINUE*,)*COPY*,)*DROP*,)*LOAD*,)*PCOPY*,)*MCOPY*, or)*SAVE* operation cannot be successfully completed because APL2 is unable to obtain the necessary read or write access to the library. The data may be off-line, on a read-only volume, or in use by others.

System Action: For)*CONTINUE*, APL2 termination completes without saving the active workspace. For the remaining commands, no action is taken. The active workspace is unchanged.

User Response: Try to issue the command later, when the library is released by the other user(s), or determine who is using the library and request that the library be released, so that you can complete the requested operation. If the condition persists, contact your system administrator.

AP2...293 *LIBRARY I/O ERROR*

Explanation: One of the following has been detected while executing a system command that accesses APL2 libraries.

- A hardware error occurred doing disk I/O.

- A logic error occurred either in the operating system or in the APL interface with it.

- APL2 was unable to understand the record structure or file structure.

- Any of a number of miscellaneous problems prevented access. Additional APL or system messages have been queued or displayed.

System Action: For)*SAVE* or)*CONTINUE* under TSO any previous copy of the same workspace may

have been destroyed. For)*CONTINUE* on any system the APL2 session is terminated without saving the active workspace. For)*LOAD*, the active workspace may be replaced by *CLEAR WS*. In all other cases the library and active workspace are unchanged.

User Response: Record any associated messages. They may be needed for problem diagnosis. Issue)*MORE* to display any queued messages, and record them. You will probably want to contact your system administrator or support representative.

If you are issuing)*SAVE*, try any of the following:

- Issue)*DROP* for the same workspace and then repeat the)*SAVE*.

- Try to save the workspace in a different library.

- Use)*OUT* to save the workspace in transfer form.

If you are trying to load or copy from a workspace under CMS, it may be that someone else has replaced the workspace since you linked to the disk.)*HOST Q SEARCH* will return a list of disks you are linked to. Try "*)HOST ACCESS addr mode*" for each "R/O" disk.

In some cases the problem may be circumvented by exiting from APL2 and reinvoking it, or logging off from the operating system and logging back on.

AP2...294 *INCORRECT COMMAND*

Explanation: The APL2 system command entered is invalid, has invalid arguments, or has been rejected by the local installation. The command may not be supported in your environment.

System Action: In some cases a secondary message is queued that can be displayed by)*MORE*. No action is taken on the incorrect command.

User Response: For a description of a particular system command and valid arguments for that command, see *APL2 Programming: Language Reference*.

AP2...295 *IMPROPER LIBRARY REFERENCE*

Explanation: The library number specified on a system command cannot be located. It either does not exist or has not been defined within the context of your APL2 session.

System Action: In some cases a secondary message is queued that can be displayed by)*MORE*. No action is taken on the command in error.

User Response: Recheck the library number. If it appears to be correct, one of the following responses applies.

- Under CMS, check the LIBTAB APL2 file. You may be using the wrong file, or it may have been modified recently.

- Under TSO, issue)*HOST)LIBS*. Only the numbers in the returned list are valid library numbers. If you were expecting others, it may be that they are VSAM libraries which must be allocated within your TSO session: ALLOC F(W*nnn*) DA(*vsam.dataset.name*).

AP2...296 *ENTRY ERROR*

Explanation: During terminal input, either an invalid character has been received or a terminal buffer has overflowed. Since all standard code points are valid as input, invalid characters can only be created by using invalid overstrike sequences on line-mode terminals.

System Action: The input up to the point of the first error is returned to the terminal as a prompt, and the system waits for you to reenter the remainder of the input.

User Response: If you have entered an invalid overstrike sequence, correct your input from the point to which you are prompted. Otherwise you have encountered a buffer overflow, and must enter your data in smaller segments. You may be able to use catenation (**,**) to combine several input lines into one. In some cases the problem is due to very long prompts created by ⍞ output followed by ⍞ input.

Buffer limits are dependent on terminal type. You can affect the limit by whether you use session manager or batch (TERMCODE(-1)) operation, as well as by the terminal model you are using.

Explanation 2: When the shared variable interpreter interface is in use, the interpreter responds to invalid control codes with this message.

System Action: The interpreter sends another prompt code.

User Response: Check to see that either a valid control code or a character vector was sent to the interpreter through the shared variable. The supported codes can be found in the chapter about the shared variable interpreter interface in *APL2 Programming: System Services Reference*.

AP2...297 *DEFN ERROR*

Explanation: A syntactically incorrect ∇ or ⍫ statement has been entered to begin edit mode.

User Response: Check the command format, and reenter the command. For the correct syntax, see *APL2 Programming: Language Reference*.

Explanation 2: An attempt has been made to edit an object that cannot be edited, such as a variable under Editor 1 or a locked function.

User Response: Give the object being defined a new name. A locked object cannot be displayed or edited. You can only erase and re-create it.

Explanation 3: An invalid edit command has been entered.

User Response: Check the format of the command, and reenter the command. For valid edit commands and formats, see *APL2 Programming: Language Reference*.

Explanation 4: Under Editor 1, a ∇ or ⍫ has been entered on an unnumbered line, to close a definition.

User Response: Enter the closing ∇ or ⍫ on a numbered line.

Explanation 5: The closing ∇ or ⍫ has been entered to establish an object that is invalid.

User Response: Check the definition to determine why the object is invalid. Re-create the object.

Explanation 6: Under Editor 2, an attempt to pass lines from one segment to another has failed because two lines numbered [0] have appeared in the same segment.

User Response: Erase one of the lines [0] and press ENTER or a PF key again.

Explanation 7: An attempt has been made to name an object with a name already in use in the active workspace.

User Response: You can either use a different name for the object or rename or erase the existing object.

Explanation 8: An attempt has been made to edit a statement containing non-APL characters outside quoted strings or comments.

User Response: Ensure that all non-APL characters in the statement are contained either in quoted strings or comments.

Explanation 9: With Editor 2 or a named editor, and with *DBCS(nnn)* in effect, an attempt has been made to edit an object containing shift-in or shift-out (□AF 14 and □AF 15) characters. With Editor 2 an attempt has been made to introduce a statement containing these characters into the object being edited.

User Response: Eliminate shift-in and shift-out characters from the object being edited, or set DBCS(OFF).

Explanation 10: With Editor 2 or a named editor, and with DBCS(OFF) in effect, an attempt has been made to edit an object which contains extended characters (255<□AF of the characters).

User Response: Set DBCS(*nnn*) or eliminate the extended characters from the object being edited.

Explanation 11: With Editor 2 or a named editor, and with DBCS(*nnn*) in effect, an attempt has been made to edit an object containing extended characters which do not fall into the range of extended characters defined by the DBCS character set identification (*nnn*≠256⊥(4ρ256)⊤□AF of the character).

User Response: Eliminate offending extended characters from the object being edited, or change the DBCS character set identification to match the DBCS character set ID portion of the extended characters.

AP2...299 *CLEAR WS*

Explanation: You now have an empty active workspace. This may occur for any of the following reasons:

- You have just begun an APL2 session, and no CONTINUE workspace has been loaded.
- You have issued the)*CLEAR* command.
- A *SYSTEM ERROR* has just been signalled because APL2 has detected damage in the workspace or an error in interpreter operation.

System Action: Session variables (□PW, □NLT, and □TZ) retain their last global value, but all other system variables have been reset to default values. Any shared variables have been retracted. All user-named objects have been removed from the workspace, and all external object linkages have been broken.

User Response: Normally you should now begin entering APL statements, issuing commands, or defining functions and operators. However in some cases these actions will occur automatically because of INPUT or RUN invocation options or previously stacked input. If *CLEAR WS* arises because of a *SYSTEM ERROR*, you should first respond to that condition, as described under message 212.

AP2...300 TCP/IP IS UNAVAILABLE. IUCV RETURN CODE IS *rc*

Explanation: AP119 has received a non-zero return code while initializing the interface to TCP/IP.

System Action: Processing of the current action is terminated.

User Response: Determine if TCP/IP is available and running on your system. If it is not, have the system programmer start TCP/IP and retry the action. Otherwise notify your system programmer of the error.

AP2...301 IUCV IS UNAVAILABLE. RETURN CODE IS *rc*

Explanation: AP119 has received a non-zero return code from the attempt to initialize handling of IUCV interrupts. This message should only be issued by APL2 under TSO. It indicates that the simulated IUCV subsystem used by TCP/IP for MVS has not been installed. AP119 will be unusable.

System Action: Processing of the current action is terminated.

User Response: Have the system programmer verify the installation of TCP/IP and the simulated IUCV subsystem.

**AP2...302 THE LOCAL PORT SERVER IS UNAVAIL-
ABLE. PORT:** *port*

Explanation: AP119 has received a non-zero return
code from the attempt to issue a a TCP/IP CONNECT
to the port server on your system.

System Action: The local listening port will not be
registered with the port server.

User Response: Make sure the APL2 port server has
been installed and is running with the correct listening
port number (usually 31415). If the local port server
was started with a different number, AP119 must be
started with the correct SERVPORT parameter.

**AP2...303 THE REMOTE PORT SERVER IS UNA-
VAILABLE. PORT:** *port*

Explanation: AP119 has received a non-zero return
code from the attempt to issue a TCP/IP CONNECT to
the port server associated with a remote processor.

System Action: Processing of the current action is
terminated.

User Response: Make sure the APL2 port server has
been installed and is running with the correct listening
port number (usually 31415). If the remote port server
was started with a different number, either the
:TCPIPSRV. tag in the AP2TCPIP APL2PROF must
indicate the correct port number or the AP119
command SETLPORT must be used.

AP2...304 ERROR FROM *call* **CALL. TCP/IP
RETURN CODE** *rc*

Explanation: AP119 has received a non-zero return
code from the specified socket call.

System Action: If the socket call was associated with
an open connection, the connection will be closed.
Processing of the current action is terminated.

User Response: None.

AP2...305 TCP/IP RETURN CODE: *rc*

Explanation: The interface to TCP/IP has been dis-
connected. The return code indicates the reason.

System Action: All connections are closed and all
AP119 sockets are de-allocated.

User Response: None.

AP2...306 AUTHORIZATION FAILED FOR USER
userid **AT** *address*

Explanation: A remote APL2 session attempted to
connect to the cross-system shared variable listening
port and the userid and internet address were not found
in the AP2TCPIP APL2PROF file.

System Action: The connection is closed.

User Response: Ensure that an entry for the
processor number attempting the connection exists in
the AP2TCPIP APL2PROF file with the correct
:USERID and :RADDRESS tags.

**AP2...307 INVALID DATA RECEIVED FROM
PROCESSOR** *number*

Explanation: The data received from a remote
processor number did not have the correct format.

System Action: The connection with the processor is
closed.

User Response: Re-establish the cross-system
shared variable and retry.

**AP2...308 EXTERNAL PROCESSOR NOT FOUND IN
PROFILE**

Explanation: The processor number used to share a
variable cannot be found in the AP2TCPIP APL2PROF
file.

System Action: Processing of the share is terminated.

User Response: Ensure that the processor number
you are trying to share with is identified by a :SVOID
tag in the AP2TCPIP APL2PROF file.

AP2...309 CONNECTION RESET BY PROCESSOR
number

Explanation: The TCP/IP connection with a remote
processor was closed.

System Action: The local connection is closed. All
variables shared with the specified processor are
retracted.

User Response: None.

AP2...310 INTERNAL RESOURCE LIMIT: NO MORE
resource

Explanation: AP 119 has exhausted its supply of
resource. The resource limits are currently built into the
processor.

System Action: AP 119 processing is terminated.

User Response: If *resource* is PRBS, reduce the
number of processors you are concurrently sharing
cross-system variables with. If *resource* is VRBS,
reduce the total number of cross-system shared vari-
ables, plus variables shared with AP 119 directly.

AP2...311 INSUFFICIENT STORAGE FOR DATA

Explanation: AP119 is unable to obtain space for data to be sent to a remote partner.

System Action: Processing of the current action is terminated.

User Response: See the user response for message AP2...001.

AP2...312 CONNECTION REFUSED FOR *number* **ON PORT** *port*

Explanation: AP119 received a non-zero return code from a TCP/IP connect with a remote partner.

System Action: Processing of the action is terminated.

User Response: Usually this message is the result of a remote user with a port server registration containing an invalid port number. When the remote user offers a share, the register entry will be corrected.

AP2...313 FILE I/O ERROR READING TCP/IP PROFILE

Explanation: An I/O error occurred while reading the AP2TCPIP APL2PROF file.

System Action: Processing of the action is terminated.

User Response: Under CMS, ensure that a valid profile exists with a filename of AP2TCPIP and a filetype of APL2PROF. Under TSO, ensure that a valid profile exists as a member AP2TCPIP of a PDS allocated to DDNAME APL2PROF. Retry the action.

AP2...314 TEXT NOT RECOGNIZED IN TCP/IP PROFILE

Explanation: Invalid data was found in the AP2TCPIP APL2PROF file.

System Action: Processing of the action is terminated.

User Response: Correct the file and retry.

AP2...315 INVALID DATA FOUND IN TCP/IP PROFILE *tag data*

Explanation: An invalid tag or tag value was found in the AP2TCPIP APL2PROF file.

System Action: Processing of the action is terminated.

User Response: Correct the file and retry.

AP2...340 ENTER PORT SERVER NUMBER (DEFAULT 31415)

Explanation: The system is asking for the port number to be used for the APL2 port server.

System Action: The port server will open a listening connection using the specified port number.

User Response: Enter the desired port number, or hit enter to use the default (31415).

AP2...341 ENTER SERVER PASSWORD

Explanation: The system is asking for the password to be used when passing restricted requests to the port server.

System Action: The port server will use the specified password to authorize the CLEAR, LIST, and SHUTDOWN commands.

User Response: Enter the desired password, or hit enter if no restricted commands are to be allowed.

AP2...342 REGISTRATION ERROR. USER: *user* **PORT:** *number* **CONDITION:** *rc*

Explanation: The port server was unable to process a register request.

System Action: The register request is ignored.

User Response: Contact the system administrator.

AP2...343 REQUEST: *req* **LOOKING FOR:** *user*

Explanation: The port server has received a QUERY request.

System Action: The port server will search for a registration entry for the specified user and return the listening port number if found.

User Response: None.

AP2...344 REQUEST: *req* **USER:** *user* **PORT:** *number*

Explanation: The port server has received a REGISTER request.

System Action: The port server will register the specified user and port number.

User Response: None.

AP2...345 REQUEST: *req* **USER:** *user*

Explanation: The port server has received an UNREGISTER, CLEAR, LIST, or SHUTDOWN request from the indicated user.

System Action: The port server will process the specified request.

User Response: None.

AP2...401 COMMAND REJECTED BY EXIT

Explanation: You have entered a session manager command that your installation has rejected via a command verification exit. The command verification exit is module AP2ACMDX. As supplied by IBM, AP2ACMDX does not restrict any commands. Your installation has modified AP2ACMDX to restrict you from executing the command you entered.

System Action: The command is not executed.

User Response: See your system programmer to determine which session manager commands are permitted by AP2ACMDX on your APL2 system.

AP2...410 DUPLICATE OR CONFLICTING OPER-
ANDS

Explanation: A session manager command has been entered with an option specified twice or with two options that conflict with each other.

System Action: The command is ignored.

User Response: To determine the correct use of the options for a particular session manager command, see *APL2 Programming: System Services Reference*.

AP2...411 END OF DATA REACHED

Explanation: The session manager had already reached an end or side of the session log, when a session manager PAGE, LINE, or COLUMN command was issued to scroll even farther in the direction beyond the end or side.

Data not found at the beginning of the log has been lost because the session manager maintains a certain number of lines in the log at any time, discarding lines as new lines from the session are added.

System Action: None.

User Response: Use the PAGE, LINE, or COLUMN command to scroll in a different direction, or press ENTER to scroll immediately to the new input area.

AP2...412 ENTER PASSWORD

Explanation: A password is required for the attempted $)LOAD$, $)SAVE$, $)COPY$, AP 121 file request, or shared variable processor request.

System Action: The session manager prompts you for a password on the session manager command line.

User Response: Enter the password.

AP2...413 IN-STORAGE LOG FILE IN USE

Explanation: An error occurred when APL2 tried to access a permanent log file. This may reflect a system or I/O error, or may indicate that APL2 cannot find any DASD space in which to create a session log for you. Under CMS this might be because your private library disk is full or accessed for read only. Under TSO it might be because you do not have a valid F0 file allocated, or the VSAM cluster that it points to is full.

System Action: An in-storage log file is being maintained in virtual storage. All lines in the log are lost when the APL2 session ends.

User Response: If an in-storage log is acceptable, continue your session; otherwise, correct the problem if possible and restart your APL2 session.

AP2...414 INVALID COMMAND NAME

Explanation: This error can occur because a session manager command name has been entered incorrectly or has been truncated so that it does not uniquely identify a command.

System Action: The command is ignored.

User Response: Use the session manager HELP command to determine the command names. Provide sufficient characters when truncating command names to clearly distinguish the name from other possible names.

AP2...415 INVALID, MISSING, OR EXTRA OPER-
ANDS

Explanation: The session manager command entered contains invalid, missing, or extra operands. Some of the reasons this may occur are:

* A keyword has been specified that does not match any of the keywords for the command, or is an ambiguous abbreviation of a valid keyword.

* An operand that requires a certain number of specified values has the wrong number of values.

* Extra keywords or values are specified when none were expected.

* A number with more than nine digits has been specified with a command.

System Action: The command is ignored.

User Response: For session manager commands and operand values, use the session manager HELP command or consult *APL2 Programming: System Services Reference*. Be sure to specify sufficient characters for truncated command names to identify the command.

AP2...416   ERROR WHILE COPYING: CALL
hexnum **RC** *num1* **RS** *num2*

(See AP 126)

Explanation: An error has occurred during the processing of a session manager COPY command. *hexnum* is the hexadecimal form of the GDDM RCP code. (See *Graphical Data Display Manager (GDDM*) Base Programming Reference*.) The RC and RS are the same as defined for AP 126 calls.

System Action: The command is terminated.

User Response: Consult *APL2 Programming: System Services Reference*. Try)*MORE*. If you cannot solve it, consult your system administrator.

AP2...417   INVALID PASSWORD. ENTER PASS-WORD

Explanation: The password entered was not valid. Valid passwords consist of 1 to 8 alphanumeric characters.

System Action: The session manager reenters PASSWORD mode and prompts for a password.

User Response: Enter a valid password.

AP2...418   LINE EXCEEDS DISPLAY SIZE

Explanation: The specified size of the session manager screen is too small to completely display the line that begins at the top of the display input area.

System Action: The session manager displays as much of the line as possible.

User Response: If not enough of the line is displayed, issue the session manager DISPLAY command with the SIZE option, specifying a larger display size.

AP2...419   LINE NUMBERS EXHAUSTED

Explanation: The log file contains 1,000,000 lines, which is the maximum number of lines allowed. Normally, the session manager can discard the oldest block of lines and use the numbers freed for storing new lines, but one of the lines in the oldest block is currently displayed on the screen. The session manager cannot discard lines that are currently being displayed.

System Action: APL2 is in HOLDING mode.

User Response: Scroll toward the newest line in your log file until the session manager can discard the oldest block of lines.

AP2...420   LINE NOT IN LOG FILE

Explanation: The line number specified with the LINE command does not exist.

System Action: The command is ignored.

User Response: Specify the number of a line that exists, or specify a relative number of lines.

AP2...421   LOG FILE FULL

Explanation: The log file is full. Normally, the session manager can discard the oldest block of lines and use the area freed for storing new lines, but one of the lines in the oldest block is currently displayed on the screen. The session manager cannot discard lines that are currently being displayed.

System Action: APL2 is in HOLDING mode.

User Response: Scroll toward the newest line in your log file until the session manager is able to discard the oldest block of lines.

AP2...422   NO COPY ID SPECIFIED

Explanation: The session manager COPY command issued does not specify a copy destination.

System Action: The command is ignored.

User Response: Reissue the command, specifying the ON option or range of session log line numbers, the character id, and the copy destination.

AP2...423   NOT ENOUGH FREESPACE

Explanation: There was not enough free space available to completely perform the command.

System Action: Processing of the command is terminated. The command may have already been partially processed.

User Response: See message AP2...001 for user response.

Explanation: An attempt was made through processor 12 or AP 211 to open a file. The system was not able to acquire enough storage for the internal control blocks needed to manage the file.

System Action: The file is not opened. If the request was made through processor 12, □*NA* returns a 0. If the request was made through AP 211, return code 8 is returned.

User Response: See message AP2...001 for user response.

AP2...424   PROFILE FILE NOT FOUND

Explanation: The profile file whose load was requested through a session manager PROFILE command was not found.

System Action: The command is ignored.

User Response: Correct the profile name.

AP2...425   DISPLAY CODE UNKNOWN

Explanation: The number specified with the CODE operand of the session manager DISPLAY command is not recognized.

System Action: The command is ignored.

User Response: Reissue the command, specifying a valid code. If the code is correct, see your system administrator.

AP2...426   PROFILE FILE I/O ERROR

Explanation: An I/O error occurred while the session manager PROFILE command was being used to load or save the session manager profile.

System Action: The load or save is terminated.

User Response: Retry loading and saving the session manager profile. If the problem persists, consult your system administrator.

AP2...427   PROFILE FILE ATTRIBUTES INVALID

Explanation: The session manager profile file or data set does not have attributes allowing for records up to 255 characters long but no longer.

System Action: The session manager PROFILE command is ignored.

User Response: Correct the attributes of the file or data set.

AP2...428   PROFILE FILE NOT AVAILABLE

Explanation: The host system is not allowing access to the data set containing the session manager profile. That data set is RACF or password protected.

System Action: The command is ignored.

User Response: See your system administrator for proper authorization.

AP2...429   COPY ID ALREADY EXISTS

Explanation: The file name specified as the copy destination for session manager COPY command already exists. Because of GDDM limitations, a file with a name that already exists cannot be reopened under CMS.

System Action: The COPY command is ignored.

User Response: Outside of APL2 use the GDDM print utility to print the existing file, then delete that file. Or use a different file name.

AP2...430   NOT WITH COPY ON

Explanation: The SCREEN, ID, or CODE operand of the COPY command was specified with the ON operand in effect. Because of GDDM limitations, CMS permits no other operands while continuous copy is on.

System Action: The COPY command is ignored.

User Response: Try $)MORE$. Issue a COPY OFF to turn the continuous copy off. Then retry the command with another operand.

AP2...431   LOG SIZE REDUCED

Explanation: The log size requested through the session manager LOG command has been reduced.

System Action: APL2 continues processing with a log of whatever size is available.

User Response: To determine the actual size being used, use the LOG command, specifying no operands.

AP2...432   LOG SIZE NOT AVAILABLE

Explanation: The LOG command that was issued requests that the log size be changed, but the requested size is not available. Either not enough permanent storage is available for a log of the new size, or a temporary log is in effect for your APL2 session.

System Action: The session continues with the log in use before the change was requested.

User Response: If the old log size is acceptable, continue the session. If you have an in-storage file and need a log of a different size, restart the session. (The command log, with no parameters, will return the present size. If the value returned is zero, you are running with an in-storage log.)

If you have a permanent log on a CMS system and need more space, erase files on your private library disk. (This will be your A-disk, unless you are using a LIBTAB APL2 file that specifies something different.)

If you have a permanent log on a TSO system and need more space, you will need to drop some AP 121 files or get a larger VSAM cluster for your F0 library.

If you do not presently have a permanent log, be sure that the DEFAULT VSAPLPR profile used when starting your APL2 session does not contain the command LOG SIZE 0. Then, for CMS, ensure that you have a writable disk as your private library. For TSO, ensure that an F0 filename is allocated to an appropriate VSAM library when starting your APL2 session.

AP2...433 SESSION MANAGER RESTARTING DUE TO ERROR *date timestamp (gmtoffset) (gmtoffset)*

Explanation: The session manager has detected a condition that prevents it from continuing normally. Possible conditions include:

- Insufficient storage was available. (Restarting the session enables the session manager to obtain sufficient storage to continue.)

- An application using AP 126 to access the session manager's GDDM page altered the page in such a way that the session manager could no longer use it.

date and *time* indicate when the restart occurred, formatted according to the DATEFORM invocation parameter. *gmtoffset* is the Greenwich Mean Time offset.

System Action: The session manager reinitializes and continues the session.

User Response: If using AP 126 and sharing the session manager's GDDM page, check that none of the GDDM requests alter the session manager's fields or page.

Reinvoke APL2 with a larger FREESIZE value. For user response when invoking the FREESIZE option, see message AP2...001.

AP2...434 BEGINNING OF SESSION *date timestamp (gmtoffset)*

Explanation: APL2 has been started with the session manager in control of the session. *date* and *time* indicate when this occurred, formatted according to the DATEFORM invocation parameter. *gmtoffset* is the Greenwich Mean Time offset.

System Action: The message is placed in the log following any lines from previous sessions and preceding any other input or output from the current session.

User Response: None. The message does not indicate an error.

AP2...435 PROFILE RECORD ATTRIBUTES INVALID

Explanation: The session manager is attempting to read from or write over an old profile file and cannot because the record attributes of the existing file conflict with those supported by the session manager.

Under TSO a profile is stored in an MVS data set. The data set must have DSORG of PS and RECFM of F, FB, V, or VB. In addition, if the data set is to be written to, it must have LRECL >= 255.

Under CMS, a profile is stored in a CMS file. If the file is to be read from, it must have LRECL <= 255. If the file is to be written to and it has RECFM = F, it must also have LRECL = 255.

System Action: The profile is not loaded or saved.

User Response: For a file to be read, redefine the file using supported attributes. For a file to be replaced, simply delete or erase the old file. The session manager will assign appropriate attributes when it is writing out a new file rather than replacing an existing file.

AP2...436 INTERNAL PROFILE PROCESSING ERROR

Explanation: An APL2 error occurred during an attempt to use the session manager PROFILE command for loading or saving the session manager profile.

System Action: The profile is not loaded or saved.

User Response: Try)*MORE* and then retry the load or save. If the problem persists, contact your systems programmer.

AP2...437 TEXT NOT FOUND: *characters*

Explanation: You issued the session manager FIND command, but the session manager was unable to find another occurrence of the string being searched for. The session manager searched from the line preceding the current line backward to the oldest line in the log, then searched from the newest line in the log to the line following the current line.

System Action: None.

User Response: None. Informational message only.

AP2...438 RECORD(S) MISSING FROM SESSION LOG FILE

Explanation: APL2 has detected that there are records missing from your permanent session manager log file. This is probably because your previous APL2 or VS APL session did not end normally.

System Action: APL2 will attempt to recover the data in your log. Message AP2...439 or message AP2...440 will be issued to inform you of the result of the recovery attempt.

User Response: None. Informational message only.

AP2...439 SESSION LOG FILE REPLACED

Explanation: APL2 detected that there was something wrong in your permanent session manager log file. APL2 attempted to recover the data in it, but the recovery attempt was not successful, so APL2 has deleted your old file and replaced it with a new one.

System Action: Your session continues.

User Response: None.

AP2...440   SESSION LOG FILE RESET

Explanation: APL2 detected that there was something wrong in your permanent session manager log file and attempted to recover the data in it. The recovery attempt was successful.

System Action: Your session continues.

User Response: None.

AP2...441   INVALID RECORD(S) IN SESSION LOG FILE

Explanation: APL2 has detected that there is invalid data in your permanent session manager log file. This is probably because your previous APL2 or VS APL session did not end normally.

System Action: APL2 will attempt to recover the data in your log. Message AP2...439 or message AP2...440 will be issued to inform you of the result of the recovery attempt.

User Response: None.

AP2...450   KEYWORD NOT RECOGNIZED -- *characters*

Explanation: While it was scanning a command, the APL2 executor encountered a keyword that it did not recognize. *characters* is the keyword.

System Action: The command is ignored.

User Response: Reenter the command, correcting the invalid keyword.

AP2...451   NUMERIC VALUE OUT OF PLACE -- *number*

Explanation: While it was scanning a command, the APL2 executor encountered a number that it did not expect. *number* is the number.

System Action: The command is ignored.

User Response: Reenter the command, specifying correct keywords and values.

AP2...452   CHARACTER STRING OUT OF PLACE -- *characters*

Explanation: While it was scanning a command, the APL2 executor encountered a quoted string that it did not expect. *characters* is the quoted string.

System Action: The command is ignored.

User Response: Reenter the command, specifying correct keywords and values.

AP2...453   QUALIFIED NAME OUT OF PLACE -- *characters*

Explanation: While it was scanning a command, the APL2 executor encountered a qualified name that it did not expect. *characters* is the qualified name.

System Action: The command is ignored.

User Response: Reenter the command, specifying correct keywords and values.

AP2...454   PARAMETER LIST OUT OF PLACE -- *name*

Explanation: While it was scanning a command, the APL2 executor encountered a parameter list following a keyword that does not take a parameter list.

System Action: The command is ignored.

User Response: Reenter the command, specifying correct keywords and values.

AP2...455   TEXT INDECIPHERABLE -- *characters*

Explanation: While it was scanning a command, the APL2 executor encountered text it did not understand. *characters* is the text.

System Action: The command is ignored.

User Response: Reenter the command, specifying correct keywords and values.

AP2...456   IDENTIFIER TOO LONG -- *characters*

Explanation: While it was scanning a command, the APL2 executor encountered an identifier longer than eight characters. *characters* is the identifier.

System Action: The command is ignored.

User Response: Reenter the command, specifying correct keywords and values.

AP2...457   KEYWORD ABBREVIATION AMBIGUOUS -- *characters*

Explanation: While it was scanning a command, the APL2 executor encountered an abbreviated keyword. The abbreviation matches more than one of the keywords APL2 considers valid at that point. *characters* is the ambiguous abbreviation.

System Action: The command is ignored.

User Response: Reenter the command, specifying sufficient characters for the keyword so that it is clear which keyword is intended.

AP2...458 NUMBER OUT OF RANGE -- *number*

Explanation: While it was scanning a command, the APL2 executor encountered a number at a point where a number is valid, but the number was outside the range of acceptable values.

System Action: The command is ignored.

User Response: Reenter the command, specifying a number within the range of numbers expected by APL2 for the given purpose. For number ranges, refer to *APL2 Programming: Language Reference*.

AP2...459 REQUIRED PARAMETER NOT SUPPLIED AFTER *characters*

Explanation: While it was scanning a command, the APL2 executor expected to find another parameter or subparameter following *characters*, but there was none.

System Action: The command is ignored.

User Response: Reenter the command, providing the additional parameter.

Explanation 2: While processing an APL2LANG file in response to a □*NLT* assignment, a line was found with a required token missing.

System Action: The previous value of □*NLT* is restored.

User Response: Correct the APL2LANG file before attempting to use the national language definition. See message AP2...606, which should accompany this one, for more information.

AP2...460 INCORRECT PARAMETER VALUE GIVEN -- *characters*

Explanation: While it was scanning a command or invocation option, the APL2 executor detected an incorrect value for parameter *characters*

System Action: If a command was being processed, it is ignored. If an invocation option is in error, the APL2 session is terminated.

User Response: Reenter the APL command or the APL2 invocation command, specifying a valid value for *characters*.

AP2...461 ERROR IN DEFAULT PARAMETER LIST

Explanation: While it was scanning the default parameter list in the APL2 installation options module (AP2VIOPT for CMS; AP2TIOPT for TSO), the APL2 executor detected an error. Previous message(s) identified the error.

System Action: The APL2 command is ignored, and APL2 returns to its caller.

User Response: Have your system programmer correct the error in the default parameter list in the installation options module, reassemble the module, and relink APL2. Consult the descriptions of the installation options modules in the APL2 installation guides for more details.

AP2...462 ERROR IN OVERRIDE PARAMETER LIST

Explanation: While it was scanning the override parameter list in the APL2 installation options module (AP2VIOPT for CMS; AP2TIOPT for TSO), the APL2 executor detected an error. Previous message(s) identified the error.

System Action: The APL2 command is ignored, and APL2 returns to its caller.

User Response: Have your system programmer correct the error in the override parameter list in the installation options module, reassemble the module, and relink APL2. Consult the descriptions of the installation options modules in the APL2 installation guides for more details.

AP2...463 PERCENTAGE OUT OF PLACE -- *percentage*

Explanation: While it was scanning a command, the APL2 executor encountered a percentage that it did not expect. *percentage* is the percentage.

System Action: The command is ignored.

User Response: Reenter the command, specifying correct keywords and values.

AP2...464 IDENTIFIER CONTAINS INVALID CHARACTERS -- *characters*

Explanation: While it was scanning a command, the APL2 executor encountered an identifier with invalid characters.

System Action: The command is ignored.

User Response: Reenter the command, specifying valid characters for the keywords and values. For valid characters, see *APL2 Programming: Language Reference*.

AP2...550 MAIN STORAGE SERVICES RC=*hexnum* FROM REQUEST FOR *number* BYTES

Explanation: There is not enough virtual storage to initialize VSAM file services.

System Action: APL2 invocation continues, but AP 121, the session manager log, and VSAM workspace libraries are unavailable.

User Response: See the user response for message AP...001.

AP2...551 LRECL OF *number* **NOT IN VALID RANGE FOR APLTRACE**

Explanation: The APLTRACE file must have a record length of no less than 76 bytes and no more than 255 bytes. *number* is the actual record length, either from the existing file or as specified in a DD statement or TSO ATTRIB command.

System Action: The trace file is not opened. Any trace output is directed to your terminal.

User Response: If no LRECL was specified when APLTRACE was allocated, use a data set having an acceptable record length, or override the data set record length. If LRECL was specified, change the value to something acceptable or let it default.

AP2...552 ERROR RECOVERY: STAND BY FOR DUMP TO *destination*

Explanation: An internal APL2 error has occurred, and a storage dump is being taken. *destination* may be either "VMDUMP" or "APLDUMP". "VMDUMP" will be used under CMS if DEBUG(4) is in effect. "APLDUMP" will be used if an APLDUMP DDname exists.

System Action: An unformatted (VMDUMP) or formatted (APLDUMP) dump will be created, and then the session will be resumed if possible. Depending on the severity and location of the problem, the dump may be followed by SYSTEM ERROR (possibly CLEAR WS), or even termination of the APL2 session.

User Response: Report the circumstances to your system programmer. If a VMDUMP was taken, it will have been sent to the VM DUMP machine. Otherwise, you may have to take action to get the dump printed or transmitted to the system programmer.

AP2...553 INTERPRETER PROGRAM CHECK LOOP - CODE(*code***)**

Explanation: During processing of a program check, a second program check occurred. This message indicates a serious internal failure within APL.

System Action: Execution is terminated with an abend.

User Response: Report the problem to the system administrator.

AP2...554 AP2XMAIN GET FAILED, RC *code*, **SIZE** *nnn*

Explanation: An error occurred allocating storage during APL2 invocation.

System Action: The APL2 session is terminated.

User Response: See the user response for message AP2...001.

AP2...555 AP2XMAIN FREE FAILED, RC *code*, **SIZE** *nnn*

Explanation: An error occurred allocating storage during APL2 invocation.

System Action: The APL2 session is terminated.

User Response: This indicates an internal system problem. Report the problem to IBM service personnel.

AP2...558 PROCESSOR 126 ERROR *num1* **(HIGHEST)-** *num2*   *num3* **(FIRST)**

Explanation:

num1 is the AP 126 return code.

num2 is the severity code.

num3 is the number of the *FIRST* GDDM error.

System Action: GDDM error occurred during Editor 2 processing.

User Response: For more information, see *APL2 Programming: System Services Reference* and *Graphical Data Display Manager (GDDM) Base Programming Reference*.

AP2...559 NON-DISPLAY EDIT MODE...*name***.***type*

Explanation: The editor has gone into nondisplay mode because of a GDDM error or because the user's terminal may not be supported by GDDM. *name* is the name of the current object being edited. *type* is the type of object being edited (2=variable, 3=function, 4=operation).

System Action: A warning message is printed by Editor 2.

User Response: For information about terminating the Editor 2 session, see *APL2 Programming: Language Reference*.

AP2...561 UNEXPECTED APL FILE ERROR *number* **LOADING** *wsname*

Explanation: During a)*LOAD* or)*COPY* request, the APL file system had an unexpected error while reading the requested workspace from a library. The error code shown is one of those defined for AP 121.

System Action: The)*LOAD* or)*COPY* fails.

User Response: Enter)*HOST APL WSNAME wsid* to be sure that you have used a valid library reference. If you have, contact your system administrator. This indicates either an error in defining the library or an APL2 system error.

AP2...562   FILE SERVICES RETURN CODE *number*.
(SEE AP 121)

Explanation: This message is not normally displayed except in response to $)MORE$. It may or may not reflect an error. The message is used to record all exceptional conditions which arise while AP 121, Processor 12 for APL files, the session manager, or workspace library support is using the APL file system. Many exceptional conditions (such as end of file) are handled normally and do not often lead to errors.

System Action: The return code is passed to the processor that requested the file service. It will either recover or provide an error message.

User Response: Ignore the message unless you are using $)MORE$ to try to further understand an earlier error message. If you have specified DEBUG(1) you may see the message frequently, and you may want to reconsider that usage.

AP2...563 *characters* **TOO LARGE - NEEDED:** *num1*
ACTUAL: *num2*

Explanation: *characters* indicate either *ws* or *size* when you are attempting to load a workspace. It will be *ws* if you are attempting to load a workspace which is larger than the current workspace size. It will be *size* if the size parameter specified on load is larger than the invocation workspace size.

num1 indicates the storage you need and *num2* reflects the storage you actually have.

System Action: The workspace is not loaded.

User Response: Either specify a reduced size and try again, or see user response to message AP2...001 for WSSIZE invocation option.

AP2...570   MSP: *parameter block*

Explanation: An internal interface error occurred while some APL2 component was attempting to queue or display a message. One of the messages, AP2...571, 572, or 573 will accompany this message. This message shows the parameter block that caused the other message.

System Action: Processing continues, ignoring the original message.

User Response: Give the content of this message to your system administrator.

AP2...571   INVALID MESSAGE REQUEST FOR NR
decnum **FROM MODULE** *characters+*
hexnum

Explanation: An auxiliary processor or component of APL2 was trying to queue or display a message, but an internal interface error occurred. If this message is not associated with AP2...572 or AP2...573, message number *decnum* is unrecognized, or no message return area was provided.

characters+hexnum is an attempt to determine the entry point and hexadecimal offset from that entry point of the routine requesting the message. (In some cases, the entry point and offset cannot be determined accurately.) An offset of $'0'$ indicates that the request was from somewhere in the indicated CSECT.

System Action: Message AP2...570 is queued for diagnostic assistance. The original message request is ignored (except with AP2...572), and processing continues.

User Response: This is an installation error or an error in the named routine. Give the information to your system administrator. Your installation should verify that the message module (AP2TMSGT or AP2VMSGT) and the module containing the named entry point are at the same level.

AP2...572   MESSAGE TRUNCATED

Explanation: An auxiliary processor or a component of APL2 was trying to queue or display a message that should not be truncated. The message would have exceeded 255 characters after substitution, but the message request indicated that the message was not to be truncated.

System Action: As much of the message as possible is displayed, and processing continues.

User Response: Enter $)MORE$ to display the associated message (AP2...571), and give that information to your system administrator.

AP2...573   MESSAGE SERVICE ABEND

Explanation: A program check or system abend occurred while an auxiliary processor or a component of APL2 was trying to queue or display a message.

System Action: APL2 recovers (if possible). The original message request is ignored, and processing continues.

User Response: Enter $)MORE$ to display the associated message (AP2...571), and give that information to your system administrator.

AP2...574   NO MORE INFORMATION

Explanation: You entered)*MORE*, but there were no queued messages to display. Queued messages are cleared by any system command except)*MORE*.

System Action: The)*MORE* command completes normally.

User Response: None.

AP2...575   *type* **FILE NOT FOUND:** *name*

Explanation: APL2 was unable to locate a file that it needed internally for processing it was asked to perform. *type* is a file type on CMS, or a DDname on TSO. *name* is a file name on CMS, or a member name on TSO.

Typically *type* will be **APL2LANG**, **APL2HELP**, or the name of some application help library. *name* will normally correspond to a □*NLT* value.

System Action: If a value was being assigned to □*NLT*, the assignment will be ignored and the previous value retained. If the *HELP* external function was being used it will fail with return code 4.

User Response: Determine if the file does exist some-where, or if the name was incorrectly specified. If it does exist, make it available to APL by using ALLO-CATE under TSO or ACCESS under CMS.

AP2...580   DSOPEN FAILED, GDDM RETURN CODE WAS *number*

Explanation: Because the DSOPEN parameter was used when APL2 was invoked, APL2 issued a DSOPEN call to GDDM with the device-token supplied. GDDM returned with return code *number*.

System Action: APL2 ignores the return code and continues.

User Response: Consult *Graphical Data Display Manager (GDDM): Messages* to determine the meaning of the return code.

Each nonzero GDDM return code corresponds to a GDDM message, whose identification consists of the three characters "ADM" followed by the return code for-matted as four digits, including leading zeros as needed. For example, if the return code is 760, examine the description of message ADM0760.

AP2...582   *paramcode* **FROM** *modname + offset* **SEV-RC IS** *sev/ret*

Explanation: Control has returned from GDDM to APL2. *paramcode* is the hexadecimal form of the GDDM Request Control Parameter (RCP) Code for the request APL2 wanted performed, documented in *Graph-ical Data Display Manager (GDDM) Base Programming Reference*. For example, 0C080603 would indicate that APL2 requested that GDDM perform an ASCPUT.

sev/ret is the hexadecimal form of the severity and return code passed back from GDDM for the call. If it is nonzero, the first two hexadecimal digits are the severity, and the next four hexadecimal digits are the return code. For example, 080054 indicates severity 8, return code decimal 84. If both the severity and return are zero, 00 will be returned. The decimal form of each nonzero GDDM return code corresponds to a GDDM message, whose identification consists of the three characters, "ADM", followed by the return code for-matted as four digits, including leading zeros as needed. For example, if the return code is 84, examine the description of message ADM0084.

This message only occurs if TRACE(2) has been requested.

System Action: Execution continues.

User Response: None. Informational message only.

AP2...583   UNSUPPORTED RELEASE OF GDDM

Explanation: If GDDM is to be used with APL2, GDDM Release 3 or later is required.

System Action: Your session continues, but you will not be able to use GDDM, and thus be unable to use the session manager or AP 126.

User Response: If you require GDDM, install GDDM Release 3 or later and consult the appropriate APL2 installation manual to determine how to give APL2 access to that release of GDDM.

AP2...584   UNSUPPORTED RELEASE OF INTER-PRETER

Explanation: Installation error or the concatenation of the libraries JOBLIB, STEPLIB, and LOADLIB is invalid.

System Action: APL session is terminated.

User Response: Consult your system programmer.

AP2...591   ABEND 1501 FORCED FOR PROCESSOR *number*

Explanation: An auxiliary processor has made an invalid service request to APL2.

System Action: Messages AP2...592 and AP2...593 are also queued or displayed; then user abend 1501 is issued. This will terminate AP *number*, unless it takes recovery action, but will not end the APL2 session.

User Response: Enter)*MORE* to display the other messages, if you need additional information about the auxiliary processor error. Inform your system adminis-trator or the person responsible for the auxiliary processor.

AP2...592 SERVICE CODE *sc* FROM R*reg+offset* IN PROCESSOR *characters*

Explanation: An auxiliary processor has made an invalid service request to APL2. This message provides background information for message AP2...591. The information in this message will be useful to the person responsible for the auxiliary processor. R*reg+offset* is a base register and a hexadecimal offset, which will probably identify the location in the AP from which the call was made.

System Action: See message AP2...591.

User Response: Inform your system administrator, or the person responsible for the auxiliary processor.

AP2...593 PROCESSOR *num1* INTERFACE ERROR DETECTED BY *modname* + *offset*

Explanation: An auxiliary processor has made an invalid service request to APL2. This message provides background information for message AP2...591. The information in this message may be useful to APL2 diagnosticians.

System Action: See message AP2...591.

User Response: Give the information to your system administrator.

AP2...594 *type* INTEGRITY COMPROMISED

Explanation: An external processor has improperly modified the APL2 recovery environment. *type* will be SPIE to indicate program check recovery, or STAE to indicate general abend recovery. The terms SPIE and STAE should be interpreted generically rather than as references to particular system macros.

System Action: Processing continues, but APL2 will probably not be able to respond properly to subsequent recovery requirements.

User Response: Inform your system administrator. Be aware of any unusual results.

AP2...595 ERROR: PARAMETER *nn* INVALID; SERVICE REQUEST *cc* FROM *id*

Explanation: One of the processors services defined in *APL2 Programming: Processor Interface Reference* has been used incorrectly. *cc* is the two-letter service code, and *nn* is the parameter number where the error was detected. *id* is the name of the processor requesting the service.

System Action: APL2 issues an abnormal termination request for the processor, though the processor may recover from the abend.

User Response: If the identified processor is part of the APL2 product, the problem should be reported to IBM. If the processor was written by the user, or provided by the installation or an APL-based application, the author of that processor needs to analyze the problem.

AP2...596 ERROR: *limit* PARAMETERS PROVIDED; SERVICE REQUEST *cc* FROM *id*

Explanation: Either too many or too few parameters have been given to one of the processors services defined in *APL2 Programming: Processor Interface Reference*. *limit* will be <n if fewer than the minimum number (*n*) of parameters were provided, or >n if more than the maximum *n* were given. *cc* is the two-letter service code, and *id* is the name of the processor requesting the service.

System Action: APL2 issues an abnormal termination request for the processor, though the processor may recover from the abend.

User Response: If the identified processor is part of the APL2 product, the problem should be reported to IBM. If the processor was written by the user, or provided by the installation or an APL-based application, the author of that processor needs to analyze the problem. Typical problems are that the high order bit was not set in the last word of the parameter list, or that a high order bit was set inadvertently before the end of the list. The latter case can arise due to garbage left in words which are omitted parameters for the present call, or due to use of addresses with the high order bit set, such as those that often appear in ECVs and CDRs. See *Processor Interface Reference* for more information.

AP2...599 DEBUG=*num1* SYSDEBUG=*num2* TRACE=*num3* CASE=*num4*

Explanation: This is an informational message in response to)*CHECK SYSTEM*. It reports the current setting of a number of APL2 diagnostic switches. For details, see *APL2 Diagnosis*.

System Action: The command completes normally.

User Response: None.

AP2...600 INVALID NUMBER *number*

Explanation: You specified the ID invocation option, but supplied a number larger than 9999999.

System Action: The APL2 session is terminated.

User Response: Reinvoke APL2 with a smaller value for ID.

AP2...601   NUMBER IN USE

Explanation: You are attempting to invoke APL2, but it is already active in your virtual machine.

System Action: The new APL2 session is not started and return code 2 is generated for the command.

User Response: In order to reinvoke APL2, alter the failing procedure. You can determine if APL2 is active by issuing the CMS command "NUCEXT AP2DISP" which will return a 0 if APL2 is active.

AP2...602   INSUFFICIENT STORAGE - *purpose* **(***size***)**

Explanation: APL2 invocation could not obtain enough storage for the indicated purpose.

- *size* is the amount of storage being requested when the problem was detected. It is not the total storage for that purpose unless so indicated below.

- *purpose* is one of the following:

   AP STARTUP

   Storage is needed to start each auxiliary processor that will be available during the session.

   FREESIZE

   Many miscellaneous storage areas are used by APL2, its processors, and other programs (such as GDDM) called during APL2 invocation.

   INSTALLATION or SERVICE

   Your virtual machine size is not large enough. A VMSIZE of at least *size* (as noted in the message) is required.

   SHARED MEMORY

   This is a single storage area whose size is controlled by the SHRSIZE invocation option.

   WORKSPACE

   This is a single storage area whose size is controlled by the SIZE invocation option.

System Action: The APL2 session is terminated.

User Response: See message AP2...001 for user response.

AP2...603   SVP INITIALIZATION FAILED, RC=*number*

Explanation: During invocation of APL2, or the AP2SVP command in a service machine, the shared variable processor had an unrecoverable problem. *number* is one of following:

-1   The SVP was already initialized.

2   Shared storage is too small.

3   Attempt to limit to fewer than five processors.

The AP2SVP command is used in starting the Global Shared Variable Processor. For the AP2SVP command, return code -1 means that another service machine is already providing cross-user SVP services. Return code 2 should not occur (unless you specified an extremely large number of processors); return code 3 means you must increase the number of processors on the AP2SVP START *proc#* command.

System Action: For APL2 invocation, the APL2 session is terminated.

For error 2 or 3 from AP2SVP, cross-user shared variable services will not be available to any CMS users on the system.

User Response: If error 2 occurs while invoking APL2, try again with a larger SHRSIZE. See message AP2...001 for user response for the SHRSIZE option.

Any other errors while invoking APL2 should be reported to your system administrator.

AP2...604   LIBRARY TABLE NOT FOUND OR NO LIBRARY ENTRIES - *fn ft*

Explanation: This message occurs when you try to reference a library other than your default library, and there are no library definitions at all from the library table file. Either the file was not found, or it did not contain any valid entries.

System Action: The message '*IMPROPER LIBRARY REFERENCE*' is issued, your attempt to reference the library is ended, and APL2 continues execution.

User Response: If access to libraries other than the default library on your A-disk is required, do the following steps.

1. Terminate the APL2 session.

2. Use CMS and CP commands to access the disk that contains the library table file, or use one of the CMS editors to create the library file. A description of APL2 library files can be found in *APL2 Installation and Customization under CMS*. A description of VS APL library files (used for)*MCOPY*) can be found in *VS APL for CMS: Terminal User's Guide*.

3. Restart APL2.

AP2...605   IGNORED CARD(*number*) *card image*

Explanation: This message is additional material related to AP2...606. Its purpose is to display the library table file card containing a syntax error.

System Action: Execution continues.

User Response: See AP2...606.

AP2...606   ERROR ON LIBRARY (*name type***)
        RECORD(***number***) COLUMN (***col***)**

Explanation: A syntax error exists in the file identified by *name type*.

- If you are using APL2 under TSO, *type* is a DDname which points to a partitioned data set or concatenated set of partitioned data sets, and *name* is a member that was being processed.

- If you are using APL2 under CMS, *name type* is the file name and file type of a CMS file located on one of your accessed minidisks. Standard CMS search order is used to locate the file.

RECORD and COLUMN indicate where the syntax error was detected, but may not point precisely to the source of the problem.

System Action: Depending on the type of processing, either the file will be rejected or the record in error will be ignored. If only the record is ignored, message AP2...605 accompanies this one, and displays the record in error.

User Response: If you created the file in error, determine what the syntax problem is and correct it. The syntax for LIBTAB APL2 files is defined in *APL2 Programming: System Services Reference*. For a definition of the syntax for APL2LANG files, see either *APL2 Installation and Customization under CMS* or *APL2 Installation and Customization under TSO*.

If you did not create the file, report the problem to the responsible person or your system administrator.

AP2...608   *type* (*name***) - *tool* FAILED**

Explanation: An error occurred while APL2 was trying to access a saved segment. *type* is the type of the saved segment, either SEGMENT for CP/ESA and CP/XA machines, or DCSS for CP/HPO or CP/SP machines. *name* is the name of the saved segment. *tool* is the means used to access the saved segment, either the SEGMENT MACRO for CP/ESA and CP/XA machines, or DIAGNOSE 64 for CP/HPO or CP/SP machines.

System Action: The command is terminated with the return code from *tool*.

User Response: Report the text of the message and the return code to your system administrator.

AP2...609   SERVICE *characters* ACTIVE

Explanation: The service *characters* is currently active in your virtual machine.

System Action:

1. If the message is in response to a query, execution continues.

2. If the message occurs while attempting to start

either an APL2 session or the global SVP service machine, it indicates that *characters* is already active and the new request is aborted with return code 2.

User Response:

1. If attempting to start an APL2 session, this indicates an internal error and should be reported to your system administrator. "NUCXDROP *" and/or IPL of CMS may be required to continue.

2. If attempting to start the global SVP service machine, then you must first issue "AP2VCSVP STOP" *characters* to deactivate the currently active saved segment.

AP2...610   SERVICE *characters* UNAVAILABLE

Explanation: The system service *characters* is required for execution but is no longer available.

System Action: The current command is terminated.

User Response: This message indicates an internal failure. Report the problem to your system administrator.

**AP2...616   SESSION MANAGER INITIALIZATION
        FAILED. RC=***num1***, RS=***num2***

Explanation: The session manager cannot be initialized. The meanings of the error return and reason codes are as follows:

RC=3   The session manager installation exit will not permit you to use the session manager. RS *num2* contains the R15 return code from that module.

RC=5   Session manager task did not start. RS *num2* is undefined.

RC=6   An abend occurred. RS *num2* contains the abend code.

RC=7   Screen does not have enough columns. Session manager requires at least 40 columns. RS *num2* is the number of columns on the terminal screen as reported by GDDM.

RC=9   Screen does not have enough rows. Session manager requires at least 3 rows. RS *num2* is the number of rows on the terminal screen as reported by GDDM.

RC=10   There is not enough free storage. RS *num2* is undefined.

RC=other RC *num1* and RS *num2* are return and reason codes from APL2 or the GDDM program product. See *Graphical Data Display Manager (GDDM) Base Programming Reference*.

System Action: If SMAPL(ON) was specified, the APL2 session is terminated. If SMAPL(TRY) was the default, invocation continues without the session manager.

User Response: In most cases you should merely pass the information on to your system programmer and use SMAPL(OFF) until a solution is provided.

If RC=10, increase FREESIZE or decrease WSSIZE and reinvoke APL2. See message AP2...001 for user response for FREESIZE and WSSIZE options.

If RC=3, ask your system programmer to authorize you to use the session manager.

If RC=1 and RS=60, check with your system administrator to determine if GDDM is available to APL2 at your installation. If not, you cannot use the session manager. If so, you may need to get access to the virtual disk containing the GDDM text libraries and issue the CMS command GLOBAL TXTLIB ADMGPLIB.

You may also need to specify a larger FREESIZE or use a larger virtual machine when invoking APL2. See user response for message AP...001 for specifying a larger FREESIZE.

**AP2...617   TERMINAL APL FEATURE UNAVAIL-
ABLE - TERMCODE(*number*)**

Explanation: APL initialization cannot determine that the terminal supports APL characters.

System Action: Execution continues, but results are unpredictable, if the system or application attempts to output APL characters.

User Response: If your terminal is capable of handling APL characters, then either the AP2EXIT EXEC did not issue the "CP TERMINAL APL ON" command, or your VM system does not recognize your terminal as valid for use with APL characters. In the former case, you can revise the AP2EXIT EXEC, but the latter will require intervention from your system administrator.

**AP2...620   PROCESSOR (*characters*) ABENDED
WITH CODE DEC(*decnum*) HEX(*hexnum*)**

Explanation: Processor *characters* has failed with the indicated code and cannot recover. Additional messages may be displayed if the DEBUG(1) option is specified when you issue the APL2 command.

System Action: Processor *characters* is made unavailable for the remainder of the APL2 session. If this processor is an auxiliary processor, the state of any variables shared with it is unpredictable.

User Response: Note the values of *characters* and *hexnum*. Make an attempt to record what you were doing just prior to the failure. Report this information to your system administrator. Restart the session. If necessary, the session can be continued as long as no

requests are made to that auxiliary processor. However, restarting the session is recommended.

AP2...621   *characters*

Explanation: *characters* represents additional diagnostic text related to some previous message. For example, this message may contain the text of a CMS command that failed in an unexpected manner.

System Action: Execution continues. This message will be shown only in response to a)*MORE* request.

User Response: If you are trying to report an error, report the text of this message along with the previous messages. You may use)*MORE* to check for the existence of further messages.

**AP2...622   COMMAND FAILED BY INSTALLATION
EXIT - CODE(*number*)**

Explanation: An APL2 system command has been failed by your installation with a return code of *number*.

System Action: Execution continues with the response *INCORRECT COMMAND* from APL2.

User Response: Your installation has determined that you are not to issue some forms of system commands. If you need to issue such commands, contact your system administrator.

AP2...623   CP/CMS FAILED, RC=*number* (*characters*)

Explanation: A CP or CMS command has failed unexpectedly with return code *number*. The text within parentheses is some or all of the text of the command that failed.

System Action: Execution continues, but other errors may result from the function that tried to execute the command.

User Response: This message may indicate a user error. For example, if you incorrectly specify a USERID or disk address in LIBTAB APL2, a subsequent)*LOAD* may fail with *LIBRARY UNAVAILABLE*. A)*MORE* would then elicit this message showing the text of the failing CP LINK command. The return code and text would help you identify the error.

If you think you have not caused the error, report the problem to your system administrator.

**AP2...624   *nnn* BYTES UNAVAILABLE FROM
FREESIZE FOR *purpose***

Explanation: *nnn* bytes of storage were needed dynamically during processing, and an attempt was made to get them from the FREESIZE pool, but that storage (which is maintained by the operating system) was either too fragmented or too nearly exhausted. *purpose* may be the name of a Processor 12 variable, the name of a national language being assigned to

$\Box NLT$, or the name of a control block being obtained by a specified part of the APL2 product.

System Action: The operation in progress terminates unsuccessfully.

User Response: See the user response for message AP2...001.

AP2...631   SESSION (*number***) ABENDED - CODE (***code***)   AT (***address***)**

Explanation: A serious failure occurred in the APL2 executor.

System Action: A dump is produced, and an attempt is made to obtain a clear workspace and continue execution. If another failure occurs before a line is entered from the terminal, this message will be repeated and the session terminated.

User Response: Report the problem to your system administrator.

AP2...644   APL HAS ABENDED - CODE(*number***)**

Explanation: APL2 has encountered an unrecoverable failure indicated by *code*. A prior message should indicate the reason for the failure and produce a dump.

System Action: An attempt is made to free storage and return to CMS. The AP2EXIT EXEC will be invoked with $ERROR$ as its first token.

User Response: Report the problem to the system administrator with any preceding messages and dumps. It is probable that a re-IPL of CMS is required.

AP2...651   FILE ERROR, RC=*number* (***characters***)**

Explanation: This message is normally generated as additional diagnostic information for a preceding message. An unexpected error has occurred while APL2 was using the operating system's services. *number* is the unexpected return code and the text in parentheses is the file service name and the file identifier associated with the request. This message may reveal a user error. For example, the command $WRBUF$ with RC=12 indicates an attempt to write on a disk that is accessed read-only. The command $FSWRITE$ with RC=12 may indicate an attempt to $)SAVE$ a workspace which does not exist on the minidisk for the workspaces library number, but which does exist on a read-only extension of that minidisk. These user errors can result from an incorrect LIBTAB APL2 on CMS.

System Action: If the error occurred while processing a system command, execution continues, but other errors may result from the function that tried to use the CMS file system.

If the error occurred while using processor 12 or AP 211, the file associated with the request is closed (if it was open).

User Response: Examine the message, and, if you cannot diagnose the problem, report it to your system administrator.

AP2...655   FILE (*fn ft fm***) NOT A WS - ***characters***

Explanation: This message is queued after a WS $INVALID$ error during $)LOAD$.

During $)LOAD$, the file indicated in parentheses was determined not to be an APL2 workspace for reasons indicated by *characters*:

RECFM OR LRECL

Workspaces must be a fixed record format with either 80 or 800 byte records. The file probably only conforms to APL naming conventions and may never have been a workspace.

WRONG SIZE DATA

The file does not contain sufficient records to hold the workspace. This error can occur if records are lost from a file during transfer from disk or tape, or across a telecommunication network.

WRONG CHECKWORD

An internal verification word used only by APL is invalid. The file may have been damaged in transfer or was never a workspace. It may have been a workspace on an APL system of a more advanced level than your system.

System Action: Execution continues, but the error may have caused the active workspace to be lost.

User Response:

1. If the file is not supposed to be an APL workspace you may want to rename it to avoid confusion.
2. If the file was damaged in transfer, retrieve another copy and try again.
3. If the workspace came from a more advanced level of APL, then use the APL system commands $)OUT$ and $)IN$ to move the contents. You may also want to contact the original author to determine if it will execute on your level of APL.

AP2...656   FILE (*characters***) EXISTS - USE RENAME OR)DROP**

Explanation: An ERROR message will be queued after $LIBRARY\ I/O\ ERROR$ during $)SAVE$, if the intermediate file named in parentheses was found to already exist. When APL saves a workspace, it always first creates such an intermediate file and then renames it when the operation has completed.

This intermediate file may contain a valid workspace or a valuable fragment of a workspace from a previous APL session that terminated abruptly.

System Action: Execution continues.

User Response: You may use the CMS command RENAME to change the name of the file for further examination or use the APL system command)$DROP$ to erase it.

AP2...660 SERVICE *service* **UNUSABLE (***reason***)**

Explanation: Some service or facility, identified by *service*, required by APL2 is not usable. The *reason* provides additional information about why the *service* is unusable.

System Action: APL2 processing terminates.

User Response: Correct the condition that makes *service* unusable, and then retry APL2.

AP2...661 *program (index)*

Explanation: This message provides additional information about a previous message issued by some *program*. It identifies the *index* into the *program* that generated the message. If the *program* is a CMS EXEC, the *index* is the line number of the EXEC that generated the message. This message is usually optional, and is produced only when certain trace flags or options have been requested.

System Action: Execution continues as it otherwise does after the previous message.

User Response: Use *index* to examine *program* if the meaning of the previous message is not clear.

AP2...662 *message*

Explanation: This *message* is generated when a :MSG. tag is encountered while a NAMES files is being processed. The *message* is the value found for the :MSG. tag.

System Action: Execution continues.

User Response: Compare *message* to the values of the :MSG. tags in available NAMES files to identify what :NICK. in the NAMES files is being processed.

AP2...663 *help*

Explanation: This message is produced by various APL2 EXECs or modules when they are invoked without arguments or with a question mark for an argument. Above, *help* is online information that explains the use of the EXEC or module.

System Action: The EXEC or module terminates.

User Response: Review the information for instructions on the purpose and use of the EXEC or module.

AP2...664 UNKNOWN FUNCTION (*parm***)**

Explanation: An APL2 EXEC or module has been invoked, but it does not recognize the function it has been invoked to perform. Above, *parm* is the parameter it does not recognize.

System Action: The EXEC or module terminates.

User Response: Reinvoke the EXEC or module with a valid parameter.

AP2...665 *command*

Explanation: This message is displayed before selected commands are invoked. The *command* parameter is the text of the command. This message is displayed to help you see that execution is continuing and to aid in error diagnosis if the command fails.

System Action: Execution continues.

User Response: None; this message is informational only.

AP2...666 SERVICE *service* **ALREADY AVAILABLE**

Explanation: Something to be installed, identified by *service*, is already installed.

System Action: Installation of *service* terminates. The existing version is not changed.

User Response: If the existing *service* should be replaced, delete the old version first, and then rerun the installation process.

AP2...667 *process* **STARTING**

Explanation: Above, *process* identifies some APL2 process that is starting.

System Action: Execution continues.

User Response: None.

AP2...668 USE THE 'CLEAR' KEY TO CLEAR YOUR DISPLAY SCREEN

Explanation: The CLEAR key, not the PA2 key, should be used to clear the display screen when running under APL2. See AP2...669.

System Action: Message AP2...669 is then issued.

User Response: None.

AP2...669 THE 'PA2' KEY MEANS SUPPRESS OUTPUT TO APL2

Explanation: The CLEAR key and not the PA2 key should be used to clear the display screen when running under APL2. See AP2...668.

System Action: Execution continues.

User Response: None.

AP2...670 *process* **SUCCESSFUL**

Explanation: Above, *process* identifies some APL2 process that has completed successfully.

System Action: The *process* returns to its caller.

User Response: None.

AP2...671 *process* **FAILED, RC =** *rc*

Explanation: Above, *process* identifies some APL2 process or operating system command that has failed, with *rc* being the return code.

System Action: The process returns to its caller. If the message was generated by the AP2MP11L or AP2MP11M exec, the exec has halted, perhaps leaving temporary files unerased.

User Response: Examine the return code and any earlier error messages to determine why the process failed.

AP2...672 **IF YOU HAVE ORDERED GDDM AND/OR SQL/DS AS WELL AS APL2, YOU SHOULD INSTALL THOSE PRODUCTS BEFORE INSTALLING APL2.**

Explanation: If you do not install GDDM and/or SQL/DS before installing APL2, certain parts of the APL2 installation process may have to be rerun after GDDM or SQL/DS is installed. To minimize work, install GDDM and SQL/DS first, if possible.

System Action: Message AP2...673 is then issued.

User Response: None.

AP2...673 **DO YOU WISH TO PROCEED WITH INSTALLATION OF APL2 (Y OR YES TO INSTALL APL2, N OR NO TO SKIP TO THE NEXT PRODUCT)?**

Explanation: The APL2 installation EXEC is giving you the opportunity to skip the installation of APL2 at this time.

System Action: The system waits for your response.

User Response: Respond with the character Y or N as desired.

AP2...674 **TAPE FILE** *processed, contents*

Explanation: The APL2 installation or service EXECs have processed a file from an installation or service tape. The above variables will be replaced as follows:

processed will be replaced by either LOADED or SKIPPED, indicating what was done to the file.

contents describes what was in the file on the tape.

System Action: Execution continues.

User Response: None.

AP2...675 **APL2 SYSTEM GENERATION FOR:**

Explanation: This messages precedes and identifies a special AP2...662 message. That message contains the current settings of various options for the way APL2 is to be generated.

System Action: Execution continues.

User Response: None.

AP2...676 **PLEASE ENTER THE WRITE PASSWORD FOR** *userid cuu*

Explanation: APL2 has tried to link to some minidisk, specified above by *cuu* identified above by *userid* by a write password or some other protection mechanism.

System Action: APL2 waits for you to enter the password.

User Response: Enter the password if you want to allow the link. If you enter nothing or if you enter the wrong password three times, APL2 will abandon the attempt. Contact your system administrator for authorization to access the disk.

AP2...677 **REQUIRED PART NOT FOUND -** *partid*

Explanation: A part of APL2 required for the installation or operation of APL2 (identified above by *partid*) is not available.

System Action: Installation or operation of APL2 is terminated.

User Response: Locate the missing part, then try again.

AP2...678 **OPTIONAL PART FOUND -** *partid*

Explanation: An optional part, specified above as *partid*, of APL2 is present. This message is issued so the information is available in your console log, should any problems occur at a later time.

System Action: Execution continues.

User Response: None.

AP2...679 RESPOND 'QUIT' AT ANY TIME TO TER-MINATE PROCEDURE. PRESS ENTER (OR RETURN) KEY WITH NO DATA TO TAKE DEFAULT.

Explanation: During the servicing of APL2, you may be prompted for information. Any time you are prompted, you may respond with the word QUIT. If you do, the servicing of APL2 is terminated. If you respond by simply pressing the ENTER or RETURN key, the default value or action is selected.

System Action: Execution continues.

User Response: None.

AP2...680 INSTALLING SERVICE FOR APL2 RELEASE *release*

Explanation: The APL2 service EXEC is ready to install service for APL2 at the level you requested, represented above by *release* It is prompting you to confirm that the servicing should take place.

System Action: The service EXEC waits for your response.

User Response: Simply press ENTER or the RETURN key to begin the servicing. Type QUIT and press ENTER or RETURN to skip the servicing.

AP2...681 *process* **ENDED AT YOUR REQUEST**

Explanation: Some APL2 *process* has been terminated because of input from you.

System Action: The APL2 process returns to its caller.

User Response: None.

AP2...682 *name* **EXEC MUST BE RUN FROM YOUR C-DISK**

Explanation: The *name* EXEC resides on the A-disk or B-disk, or it is not on a disk accessed as filemode C.

System Action: The *name* EXEC terminates.

User Response: Access the disk containing *name* EXEC as your C-disk, make sure there is not a copy of the EXEC on an A-disk or B-disk, then reinvoke *name*.

AP2...683 ENTER APL2 SERVICE STAGING AREA ADDRESS (DEFAULT *cuu* **)**

Explanation: The APL2 service EXEC is ready to load files from the service tape, and needs the minidisk address of the disk on which to load them. The address will default to the address represented above by *cuu* if you press ENTER.

System Action: The service EXEC waits for you to respond.

User Response: Press ENTER to use address *cuu*. Enter another address if desired. Enter QUIT to terminate servicing.

AP2...684 TOO MANY FAILURES

Explanation: You have failed to respond to a prompt with a usable answer three times.

System Action: Processing terminates.

User Response: Determine a usable answer, and then reinvoke the APL2 process.

AP2...685 AP2MAINT FAILED, RC = *rc* **NICK =** *nick*

Explanation: The AP2MAINT EXEC has returned a nonzero return code after being called to process some :NICK tag. The return code is represented above by *rc*, and the :NICK tag is represented by *nick*. This particular :NICK must process successfully if APL2 is to run at all.

System Action: Processing terminates.

User Response: Examine the return code and any additional error messages to determine what went wrong. See *APL2 Installation and Customization under CMS* for more information on AP2MAINT EXEC and NAMES file.

AP2...686 YOU ARE NOT PRIVILEGED TO ISSUE *command*

Explanation: Some part of APL2 needs to issue a *command* during either APL2 installation or operation. Your VM USERID is not authorized to issue this particular *command*.

System Action: Processing continues, but some parts of APL2 installation or operation may fail.

User Response: Determine the required privilege class by examining the VM/CMS publications listed in the front of this manual and by consulting your VM system administrator. Either have your VM system administrator assign you the required privilege class, or run the installation from a different USERID that has the required privilege class.

AP2...687 AP2MAINT MAY HAVE FAILED, RC = *rc* **NICK =** *nick*

Explanation: The AP2MAINT EXEC has returned a nonzero return code, represented above by *rc*, after being called to process some :NICK tag, represented above by *nick*. This particular :NICK is associated with an optional feature of APL2. Some or all of APL2 may run successfully even though this :NICK was not processed successfully.

System Action: Processing continues.

User Response: Examine the return code and any additional error messages to determine what, if any-

thing, went wrong. See *APL2 Installation and Customization under CMS* for more information on AP2MAINT EXEC and NAMES file.

AP2...688 *fileid* **ALREADY EXISTS - CHECK FOR UPDATED SAMPLE**

Explanation: IBM ships samples of several files as part of APL2. These sample files all have file type AP2MSAMP as shipped. As part of the installation and service processes, APL2 checks if you have a working copy of these files installed.

One of these processes has detected that you already have one of these files, in this case with a file name and file type represented above by *fileid*.

System Action: Processing continues.

User Response: Compare the sample file shipped by IBM with your file to determine if any changes should be made to your file.

AP2...689 *fileid* **CREATED FROM SAMPLE FILE**

Explanation: IBM ships samples of several files as part of APL2. These sample files all have file type AP2MSAMP as shipped. As part of the installation and service processes, APL2 checks if you have a working copy of these files installed. One of these processes has detected that you do not have one of these files, in this case with a file name and file type represented above by *fileid* APL2 has created such a file for you by making a copy of the sample file shipped with APL2.

System Action: Processing continues.

User Response: Examine the file to determine if any local changes must be made.

AP2...690 FOR ADDITIONAL INSTRUCTIONS, REFER TO *document*

Explanation: Additional information about this and preceding messages can be found by referring to the specified *document*.

System Action: Processing continues.

User Response: If anything is not clear, refer to the document represented above by *document*.

AP2...691 CUU *vdev* **WILL BE A-DISK FOR APL2 SERVICE**

Explanation: The virtual device at address *vdev* will be accessed by the APL2 service EXEC as the A-disk while the EXEC applies APL2 service to your system.

System Action: *vdev* is accessed as the A-disk.

User Response: None.

AP2...692 *file* **ALREADY EXISTS**

Explanation: The AP2MREN maintenance EXEC is trying to replace "file" but a copy of the file already exists on the target maintenance disk.

System Action: AP2MREN will not replace the file, because it may have been locally modified. The replacement file will be retained, but with a different name. This name will be displayed with message AP2...693.

User Response: Verify whether the file named has been locally modified. If not, replace it with the new file. If it has, the local modifications may need to be made to the new file before replacing the old one.

AP2...693 CHECK FOR LOCAL MODIFICATIONS BEFORE REPLACING WITH *file*

Explanation: This message provides further clarification for message AP2...692.

AP2...700 NOT A VALID NAME

Explanation: The input specified is not a valid name for the purpose requested. Some of the reasons this might occur are:

- The name contains invalid characters
- The name is too long
- The name is not one of the names which are expected
- More than one name was supplied, and only one is expected.

System Action: The name is ignored and processing continues.

User Response: None.

AP2...800 LOADLIB WILL BE IGNORED BECAUSE OF ERROR

Explanation: The LOADLIB invocation option was specified, but a previously reported error had been found in the specification.

System Action: Invocation continues without the LOADLIB.

User Response: Correct the earlier reported error and reinvoke APL2.

AP2...801 ERROR CODE *number* **FROM SVP INITIALIZATION. SESSION TERMINATED**

Explanation: An error has occurred during initialization of shared variable processing with an auxiliary processor. The error code *number* is in decimal form.

System Action: The APL2 session is terminated.

User Response: Notify the system administrator.

AP2...803   YOU ARE NOT AUTHORIZED TO USE APL2 FOR TSO

Explanation: Your installation has prohibited user access to APL2 under TSO.

System Action: APL2 terminates.

User Response: If necessary, obtain authorization to use APL2 under TSO.

AP2...804   USING THE APL CHARACTER SET PRESS THE SHIFT 6 AND ENTER KEYS

Explanation: Different terminals require different character translations. APL2 under TSO issues this message in order to determine which translation to use for your terminal type.

The message appears only if TERMCODE has not been specified as an invocation keyword.

System Action: This message is followed by AP2...805 and an input prompt. Initialization continues after the proper response is received. If an improper response is received, message AP2...873 is displayed and then these messages are reissued. APL2 TSO initialization can proceed only after you respond to these messages.

User Response: Respond with the following procedure:

1. Change the terminal from its normal mode to APL mode.

2. Type the shift-6 character ≥, which is done by holding the shift key while pressing the 6 key.

If the terminal does not have an APL feature, APL2 initialization will continue, but some characters unique to APL will not be entered or displayed.

AP2...805   OR JUST PRESS ENTER FOR APL USE WITHOUT THE FULL SET OF APL CHARACTERS

Explanation: This is a continuation of AP2...804. See the the explanation for AP2...804 immediately preceding this message.

System Action: See AP2...804 for system action.

User Response: See AP2...804 for user response.

AP2...806   IMPROPER TERMINAL CODE

Explanation: Either the value of the TERMCODE option or the response to the AP2...804 message is invalid. See AP2...804 for more information.

System Action: APL2 displays messages AP2...804 and AP2...805 and waits for a valid response.

User Response: Respond to the AP2...804 message.

Check the list of valid TERMCODE values in the *APL2 Programming: System Services Reference*.

AP2...807   TERMINAL AND/OR ACCESS METHOD NOT SUPPORTED BY APL2 UNDER TSO

Explanation: APL2 does not support the combination of the terminal access method used by your installation with the type of terminal you are using. The most likely cause is that you are connected to TSO using TCAM.

System Action: The APL2 session is terminated unless you are using a 3270-type terminal under TCAM. In that case, invocation will continue, but IBM will not provide service for any APL problems associated with the configuration.

User Response: Find out from your system programmer whether there are VTAM* ports you can use, if your terminal has been correctly defined to the system, or if there is a properly connected terminal you can use.

AP2...812   ALLOCATION ERROR - *data set name*

Explanation: An allocation error occurred for a data set being accessed through processor 12 or AP 211, or for the data set specified in the LOADLIB invocation option.

System Action: Processor 12 returns a `0`, AP 211 returns a `10`, or invocation continues without the LOADLIB. The message will be followed by message AP2...832.

User Response: Examine message AP2...832, and, if you cannot diagnose the problem, report it to your system administrator.

AP2...813   CONCATENATION FAILURE

Explanation: The LOADLIB invocation option was specified, but a concatenation error occurred for the requested data sets.

System Action: Invocation continues without the LOADLIB. The message will be followed by message AP2...833, which indicates the error.

User Response: Determine what the problem is from message AP2...833.

AP2...814   OPEN FAILURE - LOADLIB

Explanation: The LOADLIB invocation option was specified, or a LOADLIB DD was provided, but APL2 was unable to open the file.

System Action: Invocation continues without the LOADLIB.

User Response: MVS OPEN will have produced a message indicating why the OPEN failed. Record the number and text of that message and consult the MVS messages manual, or give it to your system programmer.

AP2...828   VIRTUAL STORAGE ALLOCATION ERROR

Explanation: Inadequate virtual storage exists during invocation to allocate the FREESIZE area, the workspace, shared storage, or the common AP buffer.

System Action: The APL2 session terminates.

User Response: See message AP2...001 for user response.

AP2...829   CONTINUE WS EXISTS BUT WILL NOT BE LOADED

Explanation: This message informs you that the *CONTINUE* workspace **is not** loaded when the INPUT option is specified with the APL2 command.

System Action: APL2 invocation continues. If an abnormal termination occurs later, APL2 will attempt to save the active workspace as *CONTINUE*. This will destroy the present contents of the *CONTINUE* workspace.

User Response: The *CONTINUE* workspace may contain valuable work. To retain it, issue:

1.)*LOAD CONTINUE*

2.)*WSID newname*

3.)*SAVE*

AP2...830   AP 101 ENCOUNTERED ERROR TRYING TO STACK INPUT DATA

Explanation: An APL2 command with the INPUT option will not fit on the alternative input stack. Insufficient space exists for the character strings specified with the INPUT option.

System Action: The specified character strings are ignored.

User Response: Reissue the APL2 command, specifying a larger AISIZE option or less data for the INPUT option.

AP2...831   YOU MAY NOT DROP OWNERSHIP OF A PUBLIC OR PRIVATE LIBRARY

Explanation:)*DROP number OWNERSHIP* is permitted only if *number* is the number of a project library.

System Action: Nothing is done.

User Response: Use a project library number.

AP2...832   RC=*num1* **DARC=***num2* **CTRC=***num3*

Explanation: This message accompanies message AP2...812, and displays return codes from DAIR. RC= gives the register 15 return code, while the other two values come from DA08DARC and DA08CTRC.

System Action: Processor 12 returns a 0, AP 211 returns a 10, or invocation continues without the LOADLIB.

User Response: Record the return codes and consult *OS/VS2 TSO Guide to Writing a Terminal Monitor Program or a Command Processor*, or give them to your system programmer.

AP2...833   RC=*num1* **DARC=***num2*

Explanation: This message accompanies message AP2...813, and displays return codes from DAIR. RC= gives the register 15 return code, while DARC= comes from DA0CDARC.

System Action: Invocation continues without the LOADLIB.

User Response: Record the return codes and consult *OS/VS2 TSO Guide to Writing a Command Processor or Terminal Monitor Program*, or give them to your system programmer.

AP2...834   SESSION MANAGER INITIALIZATION FAILED. RC=*num1* **RS=***num2*

Explanation: The session manager cannot be initialized. The meanings of the error return and reason codes are as follows:

RC=1   APL2 has detected a problem associated with the use of the GDDM product. RS *num2* is as defined for AP 126, in *APL2 Programming: System Services Reference*.

RC=3   The session manager installation exit will not permit you to use the session manager. RS *num2* contains the register 15 return code from that module.

RC=5   Session manager task did not start. (RS *num2* is undefined.)

RC=6   An abend occurred. RS *num2* contains the abend code.

RC=10   There is not enough free storage. (RS *num2* is undefined.)

RC=other RC *num1* and RS *num2* are return and reason codes from the GDDM program product. See *Graphical Data Display Manager (GDDM) Base Programming Reference*.

System Action: If SMAPL(ON) was specified, the APL2 session is terminated. If SMAPL was defaulted, invocation continues without the session manager.

User Response: In most cases you should pass the information on to your system programmer and use SMAPL(OFF) until a solution is provided.

If RC=10, increase FREESIZE or decrease WSSIZE and reinvoke APL2. See message AP2...001 for user response for FREESIZE and WSSIZE options.

If RC=3, ask your system programmer to authorize you to use the session manager.

AP2...835 *hexnum, devicedata*

Explanation: The MVS Sequential Access Method has detected an error and provided the information in this message. See *Message Buffer Format*, under SYNADAF in *OS/VS2 MVS Data Management Macro Instructions* or *MVS/Extended Architecture Data Administration: Macro Instruction Reference*. The *hexnum* field may contain an input buffer address and number of bytes read, both in hexadecimal. *devicedata* begins with the job name field at offset 50 in the SYNADAF message buffer.

System Action: The system command being processed terminates with LIBRARY I/O ERROR.

User Response: Give the information in this message to your system programmer.

AP2...836 *operation* **IS APPROXIMATELY** *number%* **COMPLETE**

Explanation: This message is queued approximately once every 10 seconds during long VSAM library operations if DEBUG(8) is in effect. The variable *operation* is either *READ*, *WRITE*, or *CLEANUP*. *CLEANUP* appears when a file or workspace is being dropped or following *WRITE* when a large record or workspace is being replaced by a much smaller one.

System Action: The VSAM operation continues.

User Response: This message is primarily informational. You may decide, based on the estimates, that you want to terminate the operation. Slowly press the interrupt key twice to accomplish this.

AP2...837 SOMEONE APPEARS TO BE USING THE FILESERV WORKSPACE TO IMPORT THE FILE RIGHT NOW

Explanation: This message is provided in response to the user issuing)*MORE* after file service return code 28. (For further explanation, see message AP2...562.)

System Action: None

User Response: Retry the)*MORE* request later.

AP2...838 SOME PREVIOUS ACCESS TO THE FILE DURING YOUR APL2 SESSION HAS NOT COMPLETED

Explanation: This message is provided in response to the user issuing)*MORE* after file service return code 28. (For further explanation, see message AP2...562).

System Action: None

User Response: If you are accessing the file via AP 121, check for other AP 121 shared variables accessing the same file. If you are accessing a workspace, a previous access to that workspace has terminated abnormally. You will have to sign off from APL2 and reinvoke it to regain use of the workspace.

AP2...839 SCRATCH FAILURE - VOLUME *volser*, **CODE 08-**ated*number*

Explanation: During a)*DROP* command, APL2 issued an MVS SCRATCH SVC. The SCRATCH function has failed on the indicated volume and returned the listed error code, displayed in hexadecimal.

System Action:)*DROP* processing is terminated.

User Response: Retry the)*DROP*. If the error recurs, notify your system administrator.

AP2...840 TASK *name* **ABEND, SYSTEM X'** *num1***' OR USER U***num2*

Explanation: APL2 is abending with the displayed abend code and user code.

System Action: APL2 execution terminates.

User Response: Notify your system administrator of the abend.

AP2...841 ABORTING CURRENT ACTION DUE TO ATTENTION SIGNALS

Explanation: You have entered a number of attention signals, but APL2 does not seem to be responding normally to the signals.

System Action: The current activity will be terminated abnormally. This will probably cause a dump if an APLDUMP file exists. See AP2...552.

User Response: Report the circumstances to your system programmer.

AP2...844 DELETION OF CATALOGED LIBRARY POINTER FAILURE - CATLG RC *num1*, **SECONDARY RC** *num2*

Explanation: During an attempt to delete a library pointer in the system catalog, the system's catalog management function has failed with error code *num1* and secondary error code *num2*.

System Action: Processing of the $)DROP$ system command continues.

User Response: Notify your system administrator.

AP2...845 A SECONDARY ERROR HAS OCCURRED DURING LIBRARY DELETION PROC-ESSING

Explanation: During an attempt to delete a library pointer in the system catalog, either the system's catalog management function or TSO's dynamic allo-cation interface routine has failed. This message is accompanied by message AP2...844 or AP2...887.

System Action: Processing of the $)DROP$ system command continues.

User Response: Notify your system administrator.

AP2...846 AN I/O ERROR HAS OCCURRED WHILE READING THE WORKSPACE

Explanation: An unexpected error was reported by the sequential access method while validating workspace ownership or loading a workspace. The condition can arise during $)LOAD$, $)COPY$, $)MCOPY$, $)SAVE$, or $)DROP$.

System Action: The system command is terminated with a $LIBRARY\ I/O\ ERROR$ message. For a $)LOAD$ command, a $CLEAR\ WS$ is provided.

User Response: Enter $)MORE$ to retrieve message AP2...851, if it has not already been displayed. Report the content of that message to your system pro-grammer.

AP2...847 AN I/O ERROR HAS OCCURRED WHILE WRITING THE WORKSPACE

Explanation: An unexpected error was reported by the sequential access method during a $)SAVE$ command.

System Action: The command is terminated.

User Response: Enter $)MORE$ to retrieve message AP2...851, if it has not already been displayed. Report the content of that message to your system pro-grammer.

AP2...848 DYNAMIC ALLOCATION FAILURE - NO DDNAMES FREE

Explanation: An attempt to execute the $)LIB$ command has failed because all the available *ddnames* have been allocated to data sets.

System Action: The $)LIB$ command is terminated with no output.

User Response: Use the TSO LISTALC STATUS command to determine which *ddnames* are allocated and then use the TSO FREE command to release data set allocations no longer needed.

If there are none you can free, notify your system administrator that the application requires more dynamic *ddnames*.

AP2...849 WAS *wsname*

Explanation: This is an informational message issued when the $CONTINUE$ workspace is loaded. It gives the library number and workspace name of the active workspace from which the $CONTINUE$ workspace was created.

System Action: The workspace is loaded, and retains the name $CONTINUE$.

User Response: To update the original workspace, use $)WSID\ wsname$, followed by $)SAVE$.

AP2...850 *modname startloc endloc*

Explanation: This is an informational message in response to the $)CHECK\ SYSTEM\ CSECT$ command. *APL2 Diagnosis* describes the $)CHECK$ command.

System Action: Execution continues.

User Response: None.

AP2...851 ERROR DATA - CCHHR *cchhrr*, **CCW** *ccaaffnn*, **CSWSTAT/COUNT** *uuccnn*, **SENSE** *xxyy*

Explanation: The system's hardware or software has detected invalid or inconsistent data while transferring data between main storage and direct access storage using $)LOAD$, $)SAVE$, $)COPY$, $)PCOPY$, $)MC$ or $)DROP$.

CCHHR *cchhrr*
is the absolute disk address being read or written at the time of error. *cc* is the cylinder address; *hh* is the DASD head (track) address; *rr* is the record address.

CCW *ccaaffnn*
is the channel command word doing the reading or writing. *cc* is the command code; *aa* is the virtual storage address; *ff* is the flag bytes; *nn* is the byte count for the opera-tion.

CSWSTAT/COUNT *uuccnn*
are the status flags and residual count asso-ciated with the termination of the read or write operation. *uu* is the control unit status; *cc* is the channel status; *nn* is the residual byte count for the operation.

SENSE *xxyy*
are the first two sense bytes returned from the direct access device concerning the error, where appropriate. *xx* is the sense byte 0; *yy* is the sense byte 1.

Note: All data is displayed in hexadecimal.

System Action: The command is terminated. If your active workspace has been overwritten in main storage, a $CLEAR$ workspace results from the error.

User Response: Retry the command, if the error is recurrent, then notify your system administrator.

AP2...853 LIBRARY NOT FOUND

Explanation: This may occur on $)LOAD$, $)SAVE$, $)DROP$, $)COPY$, $)MCOPY$, $)PCOPY$, or $)LIB$. The specified library does not currently exist in the system. (The system catalog entry for the library was not found.)

System Action: Processing of the system command is ended.

User Response: Reissue the command with a correct library number, or contact your system administrator.

AP2...854 DDNAME CONTINUE REQUIRED BUT NOT ALLOCATED

Explanation: During $)CONTINUE$ or an abnormal termination, no place could be found to write a $CON-$ $TINUE$ workspace. You do not have a ddname allocated for your default library, and either your TSO PROFILE PREFIX does not match your TSO USERID, or APL2 is being used in a batch TSO job. In these cases, a CONTINUE DD (or ALLOCATE) is required.

Or this message can occur on a $)LOAD\ CONTINUE$ or a $)DROP\ CONTINUE$ under the same conditions.

It can also occur during an attempted autoload of the $CONTINUE$ workspace.

System Action: For $)CONTINUE$, the command is terminated. For abnormal termination, APL2 ends without saving your active workspace.

User Response: For abnormal termination, your current work has been lost. Change your procedures (or have your system administrator change them) so that you have a $CONTINUE$ data set allocated when running in this mode.

For $)CONTINUE$, you will want to enter one of the following:

$)HOST\ ALLOC\ F($ CONTINUE $)$
$\quad DA$ (*some.dataset.name*)
$\quad)CONTINUE$

or

$\quad)HOST\ PROFILE\ PREFIX$ (*userid*)
$\quad)CONTINUE$

or

$)SAVE$ *workspaceid*
$)OFF$

AP2...855 MORE WORKSPACE DATA SETS EXIST IN THIS LIBRARY THAN CAN BE LISTED

Explanation: The library contains more than 1000 workspaces. However, $)LOAD$, $)SAVE$, $)COPY$, and $)DROP$ commands can still be issued.

System Action: Only the first 1000 workspaces (in alphabetic order) in the library are listed.

User Response: To produce a full listing of the library, use the TSO "listcat" command. You should obtain an explanation of the data set naming conventions of APL2 workspaces from your installation.

AP2...856 SYSTEM CATALOG SEARCH ERROR - DISPLAY ABORTED

Explanation: A failure has occurred in the system's catalog search function.

System Action: Execution of the $)LIB$ command is terminated.

User Response: Reissue $)LIB$. If the error recurs, notify your system administrator.

AP2...857 THE WORKSPACE DATA SET DOES NOT CONTAIN A VALID APL WORKSPACE

Explanation: An OS data set had an APL2 workspace type data set name, but does not actually contain an APL2 workspace. The contents of the data set have not been created by the $)SAVE$ command.

System Action: The $)LOAD$, $)COPY$, or $)PCOPY$ function is not executed.

User Response: Avoid naming non-APL2 workspace data sets with APL2 type data set names. To determine the form of these data set names, issue the command $)HOST\ APL\ WSNAME$ *wsid*.

AP2...858 WORKSPACE SAVED DURING ABEND, USE)COPY

Explanation: APL2 abends may cause the state indicator in the active workspace to be corrupted. The $)COPY$ command will clear the state indicator, and reset all functions that were in execution at the time of the failure.

System Action: The $)LOAD$ command is not executed. Your active workspace is undisturbed.

User Response: Issue $)COPY$ instead of $)LOAD$.

AP2...859 WORKSPACE DATA SET OPEN FAILURE

Explanation: The system's data set open function has failed for the workspace data set.

System Action: The $)LOAD$, $)SAVE$, $)COPY$, or $)PCOPY$ command is not executed.

User Response: Retry the command. If the error recurs, then notify your system administrator.

AP2...860 THE WORKSPACE WAS SAVED BY
userid

Explanation: APL2 for TSO records the identification of the TSO user who issues a $)SAVE$ command for a workspace. If the data set containing the workspace is named with your TSO prefix, you are assumed to be the owner of the library. If you access a workspace in that library using $)LOAD$, $)SAVE$, or $)PCOPY$, this message informs you that you are not the user who originally issued a $)SAVE$ for the workspace.

System Action: The $)LOAD$, $)SAVE$, $)COPY$, or $)PCOPY$ function is executed.

User Response: Verify with the owning user that the workspace belongs in your library. If you issued the $)SAVE$ command, the workspace now contains your TSO identifier as the owning user.

AP2...861 THE WORKSPACE SIZE IS LARGER THAN AVAILABLE SPACE IN YOUR REGION

Explanation: You specified a size operand on the $)LOAD$ command that is larger than your MAX WS value set by the WSSIZE operand of the APL invocation command.

System Action: The $)LOAD$ command is not executed.

User Response: Issue the $)LOAD$ command without the size operand. TSO automatically adjusts the size of the active workspace to be the largest available in your region. If you get the message $WS\ TOO\ LARGE$, you must revert to the TSO environment and invoke APL with an appropriate WSSIZE operand. See user response to message AP2...001 for the WSSIZE operand.

If you are unable to obtain a large enough workspace in your TSO region, you will have to issue the TSO LOGON command, specifying the SIZE operand to increase available TSO storage.

AP2...862 THE WORKSPACE SIZE TO BE LOADED IS SMALLER THAN THE SYSTEM DEFINED MINIMUM

Explanation: A size specified on $)LOAD$ is too small to be used as a workspace.

System Action: The command is not executed.

User Response: Either omit the size parameter on $)LOAD$, or use a larger size.

AP2...863 THE USED PORTION OF THE WORKSPACE TO BE LOADED IS TOO LARGE

Explanation: The storage occupied by tables, variables, and functions in the workspace to be loaded exceeds the available workspace size.

System Action: The $)LOAD$, $)COPY$, or $)PCOPY$ command is not done.

User Response: Revert to TSO, and reinvoke APL2 using the WSSIZE operand to increase available workspace size. (See user response to message AP2...001 for WSSIZE invocation option.) Alternatively, revert to TSO, and issue the LOGON command with the SIZE operand to increase your TSO region size.

AP2...864 THE WORKSPACE DATA SET IS EMPTY

Explanation: Either an OS data set that follows APL2 workspace naming conventions is empty, or a workspace data set has been emptied via non-APL2 means.

System Action: The $)LOAD$, $)COPY$, $)MCOPY$, or $)PCOPY$ command is not executed.

User Response: Restore the workspace, if you have a backup copy. See your system administrator if additional help is needed.

AP2...865 THE WORKSPACE SIZE IS LARGER THAN THE WORKSPACE DATA SET

Explanation: The workspace data set contains an incomplete, and therefore unusable, copy of the workspace being loaded.

System Action: If the error is detected before the currently loaded workspace is overwritten, then the $)LOAD$, $)COPY$, or $)PCOPY$ command is not executed. Otherwise, a $CLEAR$ workspace is loaded.

User Response: Restore the workspace, if you have a backup copy. See your system administrator if additional help is needed.

AP2...866   YOU ARE NOT AUTHORIZED TO SAVE OR DROP WORKSPACES IN THIS LIBRARY

Explanation: Your installation prohibits users from allocating this public or project library to themselves. This library allows only loading of workspaces from it—not saving of workspaces into it or dropping of workspaces from it.

System Action: The $)SAVE$ or $)DROP$ command is not executed.

User Response: Use your private library or a project library to which you have access. Also ask your system administrator what library numbers have been authorized for you to use.

AP2...867   THE WORKSPACE MUST BE NAMED BEFORE IT CAN BE SAVED

Explanation: An attempt has been made to save a workspace named $CLEAR\ WS$.

System Action: The $)SAVE$ command is not executed.

User Response: Either use the $)WSID$ command to name the workspace before saving it, or issue the $)SAVE$ command, specifying the workspace name.

AP2...868   BLDL RETURN CODE 08, REASON CODE *hexnum*

Explanation: While executing a $)MCOPY$ command, APL2 attempted to locate module APLYUOPT, which defines the data set naming conventions your installation uses for VS APL. The MVS BLDL SVC returned the indicated error looking for the module. The reason code is displayed in hexadecimal.

System Action: The $)MCOPY$ command is terminated.

User Response: For more information, refer to *OS/VS2 MVS Data Management Macro Instructions*, or see your system administrator.

AP2...869   YOU ARE CHANGING THE NAME OF YOUR WORKSPACE TO AN EXISTING NAME

Explanation: You have specified a workspace name on the $SAVE$ command that is different from the name of the currently loaded workspace. This is valid only if the new name does not already exist on the library.

System Action: The $)SAVE$ command is not executed.

User Response: Take one of the following corrective actions:

- If you intended to update the library workspace that you loaded into the active workspace, issue the

$)SAVE$ command without specifying a workspace name.

- If you intended to create a new library workspace, issue the $)SAVE$ command, specifying a workspace name that does not exist.

- If you intended to replace an existing library workspace with the contents of the active workspace, use the following procedure:

  1. Issue the $)WSID$ command, specifying the name of the library workspace into which you want to save.

  2. Issue the $)SAVE$ command, without specifying a workspace name.

AP2...870   IF THIS IS REALLY WHAT YOU WANT TO DO, THEN USE)WSID TO CHANGE THE WORKSPACE NAME BEFORE ATTEMPTING)SAVE.

Explanation: See message AP2...869, which is associated with this message.

AP2...871   A *type* **LIBRARY CONTINUE WORKSPACE HAS JUST BEEN SAVED**

Explanation: This message is displayed during $)CONTINUE$ or forced termination. The *type* will be either *SAM* or *VSAM*.

System Action: The termination continues.

User Response: This message is primarily informational. If you reinvoke APL2 later using the same allocations, the workspace should be loaded automatically.

AP2...872   TO EXIT APL2, TYPE   )OFF

Explanation: This message is informational. It reminds you of characters for exiting the APL2 environment. For instance, you may provide a wrong response to the prompt for entering the APL shift-6 character during invocation. You may lose the ability to communicate with APL and be unable to return to TSO.

System Action: Informational message only. APL2 invocation continues.

User Response: No action is necessary. When the need to exit the APL2 environment does arise, enter the characters as displayed.

AP2...873   UNRECOGNIZED CHARACTER OR TOO MANY CHARACTERS ENTERED

Explanation: You have provided an incorrect response to AP2...804/805.

System Action: APL2 continues to prompt you for the shift-6 character until you enter a correct response.

User Response: Enter a shift-6 character by holding

the shift key while pressing the 6 key. To escape from the prompt, enter a null line or press the attention key.

AP2...874 MAXIMUM LIBRARY NUMBER EXCEEDED

Explanation: The specified library number is too large. OS data set names are limited to eight characters per index level. The APL2 workspace name prefix is selected at APL2 system generation by the installation. If your installation selects the character prefix V, which is the default supplied by IBM, then the maximum library number is seven digits.

System Action: The command is not executed.

User Response: Choose a smaller library number, or try to get your system administrator to change the work-space prefix.

Warning: Changing the prefix name requires all users to rename their workspace data sets to be recognized by APL2.

AP2...875 MULTIPLE OWNERSHIPS OF THIS LIBRARY EXIST

Explanation: Multiple library pointers for this library exist in the system catalog.

System Action: The command proceeds alphabet-ically, using the first pointer found. Message AP2...876 lists the library pointer actually used. Libraries associ-ated with the remaining pointers are inaccessible to APL2.

User Response: Notify your system administrator.

AP2...876 ACCESS IS RESTRICTED TO THE FIRST, AND THAT IS CATALOGED UNDER *characters*

Explanation: See message AP2...875, which accom-panies this message.

AP2...877 A SECONDARY ERROR HAS OCCURRED DURING LIBRARY CREATION PROC-ESSING

Explanation: Either the system's catalog management function or TSO's dynamic allocation interface routine has failed during an attempt to add a library pointer in the system catalog. Message AP2...878 or AP2...887 accompanies this message.

System Action: Execution of the $)SAVE$ or $)DROP$ command continues.

User Response: Notify your system administrator.

AP2...878 CREATION OF CATALOGED LIBRARY POINTER FAILURE - CATLG RC *num1* SECONDARY RC *num2*

Explanation: The system's catalog management func-tion has failed with the indicated error codes during an attempt to add a library pointer in the system catalog.

Note: Error codes are displayed in hexadecimal.

System Action: Execution of the $)SAVE$ command continues.

User Response: Notify your system administrator.

AP2...879 FILE NOT FOUND - *file identifier*

Explanation: The listed file was not found. The file is assumed not to exist.

System Action: If the message occurs due to exe-cution of a processor 12 or AP 211 request, a return code is returned. The file is not associated or opened. If the message occurs after entry of a system command, the system command is not executed.

If the message was issued by the AP2MP11L or AP2MP11M exec, the exec halted when it could not find the specified file. The exec may have left temporary files on your A-disk.

User Response: For processor 12 and AP 211, verify that you have supplied the correct file identifier or data set name. For system commands, verify that you have supplied the correct library number and workspace name.

If the message was issued by the AP2MP11L or AP2MP11M exec, locate the correct file, erase any tem-porary files left by the exec, and reinvoke the exec.

AP2...880 DATA SET NAMING CONFLICT

Explanation: The system catalog contains a con-flicting data set name structure, thus making the cata-loging of the name of the requested workspace impossible.

System Action: The system command is not exe-cuted.

User Response: If appropriate, choose a different workspace name and/or library number.

Use the TSO LISTC command to list the index structure for the data set name you are trying to save.

You may issue $)HOST\ APL\ WSNAME$ *wsid* to deter-mine the name that the system was attempting to catalog.

AP2...881 INVALID WORKSPACE NAME. WSID MUST CONTAIN NO MORE THAN 8 CHARACTERS

Explanation: Because of restrictions on the MVS catalog structure, under TSO, APL2 workspace names cannot contain more than eight alphameric characters.

System Action: The system command is not executed.

User Response: Enter a valid workspace name.

AP2...882 INVALID WORKSPACE NAME. WSID CONTAINS INVALID CHARACTERS

Explanation: The first character of the workspace name must be one of the 26 alphabetic characters A through Z. Subsequent characters may be either alphabetic A through Z or numeric 0 through 9.

System Action: The command is not executed.

User Response: Choose a valid workspace name.

AP2...883 THE WORKSPACE DATA SET IS IN USE BY SOMEONE ELSE. TRY AGAIN LATER

Explanation: Either another TSO user or a batch program is currently using the workspace data set. You must wait until the data set is released by that user or batch program before you will be allowed access to it.

System Action: The system command is not executed.

User Response: Wait, and then try the command again. If the data set continues to be unavailable, your system operator may be able to tell you who is using it. Issue)*HOST APL WSNAME* [lib] *wsid* to find the workspace data set name.

AP2...884 THE WORKSPACE DATA SET RESIDES ON A CURRENTLY UNAVAILABLE VOLUME

Explanation: The direct access storage volume containing the required workspace data set is not currently available to the system.

System Action: The system command is not executed.

User Response: Arrange with your installation's operations staff for the volume to be mounted, and then reissue the command.

AP2...885 THERE IS INSUFFICIENT DIRECT ACCESS STORAGE SPACE TO SAVE THIS WORKSPACE

Explanation: All direct access volumes onto which the workspace could be allocated are already filled to capacity with other data sets.

System Action: The system command is not executed.

User Response: Notify your system administrator so that steps can be taken to correct the situation.

If this is a new workspace, you can often retain your work in progress by saving into a large existing workspace already allocated to disk. To do this:

1.)*PCOPY* the existing workspace into the active workspace.

2. If you get a *NOT COPIED* message, you may not want to proceed.

3. Issue)*WSID* to change the name to that of the existing workspace.

4. Issue)*SAVE*.

Another approach is to issue)*OUT*, specifying a preallocated data set on a different volume.

Note: Issuing the)*DROP* command for one of your workspaces may also help, but it will not guarantee that enough space will be available.

AP2...886 A DUMPED WORKSPACE CANNOT BE LOADED

Explanation: If an error occurs internal to APL2, the active workspace will probably be saved with a workspace name of DUMP*number* (for example, DUMP0003). If you have any such workspaces, they will be displayed by)*LIB*, but)*LOAD* is not permitted, because the workspace is damaged.

System Action: The)*LOAD* command is terminated.

User Response: Inform your system administrator of the workspace name. It may be needed for diagnostic purposes. You may be able to recover functions or data from the workspace by using)*COPY*.

AP2...887 DYNAMIC ALLOCATION FAILURE - FUNCTION CODE *cc*, DAIR RC *dd*, CATLG RC *ee*, DYNAM RC *ff*

Explanation: Workspace data sets are made accessible to APL2 via the system's dynamic allocation routines. These routines may fail in any particular instance for any of a number of reasons. The more common reasons for failure are documented by other more descriptive messages. This message is used for all of the other less common causes.

cc   indicates the DAIR function request code that APL2 has issued. See "Dynamic Allocation Interface Routine," in *OS/VS2 TSO Guide to Writing a Command Processor or Terminal Monitor Program*.

dd   is the error code returned from DAIR.

ee   is the error code returned to DAIR from the system's catalog management routines.

ff   is the error code returned to DAIR from the system's dynamic allocation routines.

> **Note:** All error codes are displayed in hexadecimal.

System Action: The system command being processed is terminated.

User Response: If appropriate, bring the error to the attention of your system administrator.

AP2...888   VSAM CLOSE: R15=_number_
                OFLGS/ERFLG/INFLG=X'_hexnum1_**'**
                MACRF=X'_hexnum2_**' DDNAME=**_ddname_

Explanation: APL2 was terminating its use of a VSAM cluster, but got a nonzero return code from the access method. APL2 uses VSAM clusters for workspaces, AP 121 files, session manager log files, and work files during $)COPY$.

num1 is a decimal return code, as documented in VSAM publications. The *ddname* will be "L*number*", where *number* is an APL library number. The other information is the contents of selected fields in the VSAM ACB.

System Action: Processing continues.

User Response: None required. It might be prudent to attempt another access to the resource. If that fails, give your system administrator the content of this message, with the reaccess error symptoms.

AP2...889   A DUMP WAS NOT TAKEN

Explanation: Because of an earlier error, APL2 is attempting to save a copy of your active workspace as DUMP*nnn*. That attempt has also failed.

System Action: APL2 continues with $SYSTEM$ $ERROR$ action, and provides a $CLEAR\ WS$. Your previous active workspace is lost.

User Response: Type $)MORE$ to get an associated message with more detail.

AP2...890   SCRATCH FAILURE - CODE *number*

Explanation: The system's data set scratch function has failed with the indicated error code, displayed in hexadecimal.

System Action: Execution of the $)DROP$ command terminates.

User Response: Reissue the $)DROP$ command. If errors recur, notify your system administrator.

AP2...891   SCRATCH FAILURE - VOLUME
                *characters***, INCORRECT PASSWORD**

Explanation: You were trying to $)DROP$ a password protected workspace, but you did not supply the correct password.

System Action: $)DROP$ processing is terminated.

User Response: Reissue the $)DROP$ with the correct password.

AP2...892   UNCATALOG FAILURE - CODE *number*

Explanation: During $)DROP$ processing, the system's UNCATALOG function has failed with the indicated hexadecimal error code.

System Action: Because the workspace has by this time already been scratched, the $)DROP$ command is executed as if no error had occurred.

User Response: If the error recurs, notify your system administrator.

AP2...893   LIBRARY NOT EMPTY

Explanation: With APL2 under TSO, you can relinquish control of an empty project library by issuing $)DROP\ libno\ OWNERSHIP$. You are attempting to relinquish ownership, but the library is not empty.

System Action: The $)DROP$ command has no effect.

User Response: Issue the $)LIB$ command to determine which workspaces remain. If you really want to discard them, use the $)DROP$ command to drop each of the listed workspaces before you attempt to relinquish ownership.

AP2...894   YOUR PREFIX IS NOT DEFINED.
                DSNAMES REQUIRING IT CANNOT BE
                CONSTRUCTED

Explanation: You have not issued the PROFILE PREFIX command in batch. The TSO prefix is required to access your private library workspace data sets.

System Action: All library commands that access the private library data sets will fail. A $CONTINUE$ workspace may be saved if you have allocated the ddname $CONTINUE$

User Response: Change your batch job setup to issue the TSO PROFILE PREFIX command before invoking APL2.

AP2...895 ISSUE THE PROFILE PREFIX COMMAND PRIOR TO THE APL2 COMMAND

Explanation: This explanatory message accompanies AP2...894.

AP2...896 THE APLYUOPT MODULE CANNOT BE LOCATED

Explanation: In order to process a $)MCOPY$ command, APL2 needed to find the VS APL APLYUOPT module. Its search for that module has failed.

System Action: APL2 produces message AP2...868 and terminates the $)MCOPY$.

User Response: Enter $)MORE$ if message AP2...868 has not been displayed. Inform your system administrator of the reason for the failure.

AP2...897 PASSWORD PROTECTION NOT AVAILABLE ON THIS SYSTEM

Explanation: During a $)SAVE$, you appear to be trying to specify a new password for the saved workspace. Under TSO, APL2 does not add or change workspace passwords.

System Action: Execution of the $)SAVE$ command continues.

User Response: To add or change a password, you must use the TSO PROTECT command.

AP2...898 VSAM ERROR: REQ=num1** R15=**num2**
          FDBK=X'**num3**' OPTCD=X'**num4**'**

Explanation: APL2 was using a VSAM cluster and got a nonzero return code from the access method. APL2 uses VSAM clusters for workspaces, AP 121 files, session manager log files, and work files during $)COPY$.

Note: If you are using DEBUG(1), this message often does not mean there is an error. In many cases,

nonzero VSAM return codes are anticipated by APL2, and handled appropriately. For example, $)LIB$ always continues until it gets an end-of-file return code from VSAM.

The information returned (except R15) is from the VSAM RPL. Most of it will probably be of interest only to system programmers.

System Action: This message is queued, and execution continues. APL2 will either handle the return code or produce a separate error message.

User Response: Ignore the message if you are using DEBUG(1) and no other messages are associated with it. If there is an associated error you do not understand, give the text of this message to your system administrator along with other messages.

AP2...899 VSAM OPEN: R15=num1**
          OFLGS/ERFLG/INFLG=X'**num2**'
          MACRF=X'**num3**' DDNAME=**characters**
          MACRF=X'**num3**' DDNAME=**characters**

Explanation: APL2 was attempting to access a VSAM cluster and got a nonzero return code from the OPEN SVC. APL2 uses VSAM clusters for workspaces, AP 121 files, session manager log files, and work files during $)COPY$.

Note: If you are using DEBUG(1), this message may *not* mean there is an error. In some cases nonzero VSAM return codes are anticipated by APL2, and handled appropriately. For example, $)COPY$ attempts to use a VSAM work file, but if that fails it may be able to continue with a SAM work file.

The information returned (except R15) is from the VSAM ACB. Most of it will probably be of interest only to system programmers. The *ddname* will be L*number*, where *number* is an APL library number.

System Action: This message is queued, and execution continues. APL2 will either handle the return code or produce a separate error message.

User Response: Ignore the message if you are using DEBUG(1) and no other messages are associated with it. If there is an associated error you do not understand, give the text of this message to your system administrator along with other messages.

Chapter 3. APL2 Workspace Messages

This chapter lists all messages issued by APL2 workspaces. These messages are organized alphabetically by workspace, and underneath each workspace name, the messages are organized numerically.

Note, however, that the *OFFER* messages, which are issued by the *CMSIVP* and *TSOIVP* namespaces, are listed in alphabetical order with the workspace messages. A sample message is shown below:

```
NAME ASSOCIATION WITH PROCESSOR 11 NOT ACTIVE
```

The message code is not included with the above text; you must enter *MORE* to view the code. *MORE* may also contain additional diagnostic information.

For example, if you enter *MORE* following the above message, you would see the following message code:

```
AP2WPRINTWS3
```

where:

AP2   identifies the message as being issued from APL2.

W   means that the error was issued by an IBM-supplied workspace.

PRINTWS is the name of the workspace issuing the message.

3   is the message number of the error that you received.

If you needed more information, you would turn to this chapter in *APL2 Messages and Codes*, find messages listed under the subheading *PRINTWS*, and go to the explanation listed under the specific message you want, AP2WPRINTWS3.

APLDATA Workspace Messages

AP2WAP1211   **RECORD NOT FOUND**

Explanation: Auxiliary processor 121 has returned an error with an error code of 1.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP1212   **EXCEEDED ALLOWABLE NUMBER OF RECORDS**

Explanation: Auxiliary processor 121 has returned an error with an error code of 2.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP1213   **ATTEMPT TO WRITE A NON-VSAPL OBJECT WITH 'SW' OR 'DU'**

Explanation: Auxiliary processor 121 has returned an error with an error code of 3.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP1214   **INVALID APL OBJECT IN FILE**

Explanation: Auxiliary processor 121 has returned an error with an error code of 4.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP1215   **END OF FILE**

Explanation: Auxiliary processor 121 has returned an error with an error code of 5.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP1216   **DIRECT READ OR UPDATE ATTEMPTED ON SEQUENTIAL FILE**

Explanation: Auxiliary processor 121 has returned an error. The error code is 6.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP1217   **RECORD LENGTH ERROR**

Explanation: Auxiliary processor 121 has returned an error with an error code of 7.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP1219   **SYSTEM LIBRARY FULL**

Explanation: Auxiliary processor 121 has returned an error with an error code of 9.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12111   **EXCEEDED DEFINED FILE SIZE**

Explanation: Auxiliary processor 121 has returned an error with an error code of 11.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12112   **SYNTAX ERROR IN SERVICE REQUEST**

Explanation: Auxiliary processor 121 has returned an error with an error code of 12.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12123   **OPEN MODE SHOULD BE DIRECT UPDATE**

Explanation: Auxiliary processor 121 has returned an error with an error code of 23.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12124   **IMPROPER LIBRARY REFERENCE**

Explanation: Auxiliary processor 121 has returned an error with an error code of 24.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12125   **FILE ALREADY EXISTS**

Explanation: Auxiliary processor 121 has returned an error with an error code of 25.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12126   **FILE DOES NOT EXIST OR IS NOT AVAILABLE**

Explanation: Auxiliary processor 121 has returned an error with an error code of 26.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12128   **FILE IN USE BY OTHERS**

Explanation: Auxiliary processor 121 has returned an error with an error code of 28.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12130   **FILE PROTECTED FROM THIS OPERATION**

Explanation: Auxiliary processor 121 has returned an error with an error code of 30.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12131   **MAXIMUM NUMBER OF FILES ALREADY OPEN**

Explanation: Auxiliary processor 121 has returned an error with an error code of 31.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12132   **NO SPACE FOR IO BUFFER**

Explanation: Auxiliary processor 121 has returned an error with an error code of 32.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12133   **DATA HAS NOT BEEN SPECIFIED**

Explanation: Auxiliary processor 121 has returned an error with an error code of 33.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12134   **NOT AUTHORIZED OR INCORRECT PASSWORD**

Explanation: Auxiliary processor 121 has returned an error with an error code of 34.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12135   **HARDWARE I/O ERROR OR APL2 PRODUCT ERROR**

Explanation: Auxiliary processor 121 has returned an error with an error code of 35.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12136   **NOT AN APL2 DATA FILE**

Explanation: Auxiliary processor 121 has returned an error with an error code of 36.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12137   **FILE SIZE LIMIT REDUCED; DATA MAY BE LOST**

Explanation: Auxiliary processor 121 has returned an error with an error code of 37.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12138   **INVALID RECORD LENGTH SPECIFIED**

Explanation: Auxiliary processor 121 has returned an error with an error code of 38.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12139   **REQUIRED RECORD LENGTH NOT SPECIFIED**

Explanation: Auxiliary processor 121 has returned an error with an error code of 39.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12142   **DATA HAS NOT BEEN REFERENCED**

Explanation: Auxiliary processor 121 has returned an error with an error code of 42.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP12150   **INVALID CMS/TSO REQUEST, OR FILE SYSTEM FAILURE**

Explanation: Auxiliary processor 121 has returned an error with an error code of 50.

User Response: See "AP 121 Return Codes" on page 108. Consult *APL2 Programming: System Services Reference* for more information.

AP2WAP121100   **AP***number* **OFFER FAILED FOR VARIABLE(S)** *name(s)*

Explanation: The auxiliary processor, AP *number*, cannot share the *names* that you have specified.

User Response: Reload the workspace to retract all variables, and try again. If the failure persists, consult *APL2 Diagnosis* to troubleshoot the share failure.

AP2WAPLDATA101   **THE VARIABLE NAMED** *name* **DOES NOT EXIST**

Explanation: The *name* given as an argument to the $STORE$ function is not the name of an existing variable.

User Response: Provide the correct *name* of the variable containing the data to be stored.

AP2WAPLDATA102   **VARIABLE** fn **HAS NOT BEEN SET**

Explanation: The global variable fn does not contain the name of the file to be read. The $MORE$ variable lists the name of the function which requires it.

User Response: Use USE, $create$, or AT functions to set the filename.

AP2WAPLDATA103   **END OF FILE - FILE CLOSED**

Explanation: An end-of-file condition (an empty record) has been encountered. No attempt is made to read past it.

User Response: None; this is an informational message only.

AP2WAPLDATA104   **LEFT ARGUMENT MUST BE SPECIFIED**

Explanation: A function has been invoked without a required left argument. The $MORE$ variable describes the missing argument.

User Response: Provide the appropriate argument to the function.

AP2WAPLDATA106   **THE VARIABLE NAME** *name* **IS THE SAME AS A LOCAL NAME**

Explanation: The *name* of a global variable has been specified as an argument to the $STORE$ or $RETRIEVE$ function. The *name* specified conflicts with a name that is local to the function.

User Response: Choose another *name* for the global variable.

CMS Workspace Messages

AP2WCMS1   FILE *file* **ALREADY OPEN**

Explanation: An attempt has been made to open a *file* that is already open, as indicated by the global variable *OPENFILES*.

User Response: This is an informational message. You may want to close and reopen the file, or to reset *OPENFILES* by reloading the workspace.

**AP2WCMS2   CURRENT SHARE STATUS: VARI-
ABLES -** *vars : status*

Explanation: This message is given by the *OPEN* function. Above, *vars* contains the names of the shared variables used. The *status* is the result of issuing a □*SVO* for those variables.

User Response: None; this is an informational message from the *OPEN* function. It is displayed only if the value of the global variable *msgW* is 1 or greater.

AP2WCMS3   SHARE FAILURE: *code* **FILE:** *fn*

Explanation: The *OPEN* function cannot share with the CMS file processor (AP 110). Above, *code* is the result of □*SVO*, and *fn* is the file name.

User Response: Consult *APL2 Diagnosis* to trouble-shoot the share failure.

AP2WCMS4   OPEN FAILURE: *values*

Explanation: A file cannot be opened, and *values* contains the current values of the record and control variables. The latter value is an AP 110 return code.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 110 return code.

AP2WCMS5   FILE *file* **OPENED. SHARED VARI-
ABLES:** *rec ctl*

Explanation: A *file* has been opened with the shared variables named *rec* and *ctl*.

User Response: None; this is an informational message from the *OPEN* function. It is displayed only if the value of the global variable *msgW* is 2 or greater.

AP2WCMS6   FILE STATUS: *status*

Explanation: This message follows AP2WCMS5. The *status* is the value of the control variable whose name is given by message AP2WCMS5.

User Response: None; this is an informational message from the *OPEN* function. It is displayed only if the value of the global variable *msgW* is 2 or greater.

AP2WCMS7   OPEN ERROR

Explanation: *GETFILE* failed on a call to *OPEN*.

User Response: To diagnose the cause of failure, set *msgW* to 2 and rerun. The *OPEN* function will print the diagnostic messages.

AP2WCMS8   CLOSING RETURN CODE: *code*

Explanation: The closing return code, represented above by *code*, is not an expected one. This message is printed by *GETFILE*.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 110 return code.

AP2WCMS9   GET ERROR. FILE: *file* **RETURN
CODE:** *code*

Explanation: *GET* has encountered an error while attempting to read a *file*. Above, *code* is the return code for that error.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 110 return code.

AP2WCMS10 PUT ERROR. FILE: *file* **RETURN
CODE:** *code*

Explanation: *PUT* has encountered an error while attempting to write to a file. Above, *code* is the return code for that error.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 110 return code.

**AP2WCMS11   *name* PROCEDURE HAS BEEN
CREATED.**

Explanation: The *PROC* function has created some procedure, represented above by *name*

User Response: None; this is an informational message only.

AP2WCMS12 INVALID OPEN ARGUMENTS:
filename info

Explanation: One or more of the arguments to *OPEN* are not correctly formed. Both must be character strings. The *filename* must be properly constructed.

User Response: Display the *HOW* variable for information on the arguments of *OPEN*, and correct the arguments.

AP2WCMS13   LEFT ARGUMENT IS MISSING

Explanation: The left argument $OPEN$, PUT or $PUTFILE$ has not been specified. The $MORE$ variable provides more information about the argument.

User Response: Reinvoke the function with the proper left argument.

CMSIVP and TSOIVP Namespace Messages

AP2WIVP2   **>>> ERROR RESOLVING INTERNAL POINTERS,** *reason*

Explanation: An inconsistency has been detected while APL2 tried to extract system information. This message is followed by the *reason* for this error. An interpreter error is also signalled to cause suspension in the calling function.

User Response: Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP3   **>>> UNABLE TO SHARE VARIABLE(S) WITH AP** *id*

Explanation: An attempt to share variables with the processor whose ID is represented above by *id* failed. An interpreter error is also signalled to cause suspension in the calling function.

User Response: Ensure that the processor has been properly installed.

AP2WIVP4   **>>> UNRECOGNIZED COMMAND** *command* **IN PATH**

Explanation: A *command* was not recognized while APL2 tried to extract system information. An interpreter error is also signalled to cause suspension in the calling function.

User Response: Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP5   **>>> BAD STORAGE DESCRIPTOR:** *expected* **GIVEN AS** *actual*

Explanation: An inconsistency has been detected while APL2 tried to extract system information. An interpreter error is also signalled to cause suspension in the calling function.

User Response: Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP6   **>>> BAD RETURN CODE FROM** *procid : rc*

Explanation: An error has occurred while a command to a processor, represented above by *procid*, was issued. Above, *rc* is the processor return code. An interpreter error is also signalled to cause suspension in the calling function.

User Response: Consult *APL2 System Support Services* to analyze the return code from the auxiliary processor.

AP2WIVP7   **>>> BAD DIGIT IN HEXADECIMAL NUMBER:** *number*

Explanation: A *number* is not in the correct form for conversion from hexadecimal. An interpreter error is also signalled to cause suspension in the calling function.

User Response: This error may occur while following internal storage pointers. Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP8   **>>> BAD CALL TO 'LOOKUP' FUNCTION**

Explanation: An inconsistency has been detected while APL2 tried to extract system information. An interpreter error is also signalled to cause suspension in the calling function.

User Response: This error may occur while following internal storage pointers. Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP9   **>>> THIS IS NOT THE system SUBSYSTEM**

Explanation: The installed system is not the indicated system.

User Response: Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP10   >>> **FAILURE OF LOOKUP TO FIND** *name*

Explanation: The $LOOKUP$ function cannot find some component, represented by *name*.

User Response: Ensure installation was properly done for the named component.

AP2WIVP11   >>> **INSTALLATION VERIFICATION CAN NOT CONTINUE:** *reason*

Explanation: A needed component is not available. The IVP program cannot continue. *reason* identifies the component.

User Response: Determine why the indicated component is missing.

AP2WIVP12   **VERIFYING** *id*

Explanation: The AP or Associated Processor, represented by *id* above, is being verified.

User Response: None; this is an informational message.

AP2WIVP13   *id* **VERIFIED.**

Explanation: The AP or Associated Processor, represented by *id* above, has been verified.

User Response: None; this is an informational message.

AP2WIVP14   >>> *level* **NOT VERIFIED:** *actual*

Explanation: The host system, CMS *level* has not been verified. Above, *actual* contains the actual system or level.

User Response: Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP15   >>> **RETURN CODE (***rc***) FROM** *ap*

Explanation: The auxiliary processor, *ap*, has detected an error. *rc* is the return code for that error.

User Response: Diagnose the return code for the indicated AP. Consult *APL2 Programming: System Services Reference* for an explanation of the return code.

AP2WIVP16   >>> **AP***number* **UNAVAILABLE - IVP TERMINATED.**

Explanation: The auxiliary processor, represented by *number* above, is not available. This AP is needed by the installation verification procedure.

User Response: Review the installation process for the indicated AP.

AP2WIVP17   *sysname* **SYSTEM: CP** *version.release.level* **CMS** *version.release*

Explanation: This message identifies the system (represented by *sysname*) and the *version/release/[level]* of CP and CMS being used.

User Response: None; this is an informational message.

AP2WIVP18   **APL2 SYSTEM LEVEL:** *version.release.level*

Explanation: This message identifies the *version/release/level* of APL2.

User Response: None; this is an informational message.

AP2WIVP19   **SHARED MEMORY:** *memory*

Explanation: This message indicates whether shared *memory* is LOCAL or GLOBAL.

User Response: None; this is an informational message.

AP2WIVP20   **GLOBAL SM SERVER:** *module*

Explanation: This message identifies the global shared memory server *module*. Above, *module* identifies a VM user id, not a CMS module.

User Response: None; this is an informational message.

AP2WIVP21   **GLOBAL SVP REQUESTED (SERVER** *userid***) BUT RUNNING AS LOCAL.**

Explanation: AP2VIOPT identifies *userid* as the global SVP server, but your APL2 session can access variables only in local shared memory. This may be due to any one of the following reasons:

- You may have to explicitly request access to global shared memory when APL2 is invoked, via the ID operand, and you did not do so.

- The global SVP server may not be logged on to VM and initialized.

- The global SVP server may have denied you access to global shared memory.

- Your virtual machine may overlap the global shared memory saved segments.

User Response: If you must access global shared memory, check that the global SVP user ID is logged on and initialized, ensure that your virtual machine does not overlap the global shared memory saved segments, try specifying an ID when APL2 is invoked, and check if an installation exit has been installed to authorize or deny access to shared memory.

AP2WIVP22   *component* **SAVED SEGMENTS AVAILABLE:**

Explanation: This message precedes a list of available saved segments and their storage addresses. Above, *component* identifies a component such as GDDM or SQL.

User Response: None; this is an informational message.

AP2WIVP23   *component* **SAVED SEGMENT** *segment* **NOT FOUND. ASSUMED RENAMED OR UNAVAILABLE.**

Explanation: The *segment* named cannot be found for the *component* named above.

User Response: If such a saved segment is installed on your system under a different name, check its address range. The address range should not overlap the address range of any other saved segments you may be using concurrently. If they do not overlap, move the saved segments to correct the situation .

AP2WIVP24   *feature* **IN USE**

Explanation: *feature*, which APL2 will automatically make use of if available, is available and in use.

System Action: Processing continues.

User Response: None.

AP2WIVP25   *feature* **NOT IN USE**

Explanation: *feature*, which APL2 will automatically make use of if available, is not available and not in use.

System Action: Processing continues.

User Response: None.

AP2WIVP26   *text*

Explanation: *text* is additional text from CMS or TSO which may help explain some preceding message.

System Action: Processing continues.

User Response: None.

AP2WIVP27   (*number*-**BIT ADDRESSING MODE)**

Explanation: This message identifies the addressing mode of the CPU, represented above by *number*.

User Response: None; this is an informational message.

AP2WIVP30   **AVAILABLE, NOT IN USE**

Explanation: The APL2 session manager is available, but not in use, for this session.

User Response: None; this is an informational message.

AP2WIVP31   **AVAILABLE, IN USE (***pssuse***)**

Explanation: The APL2 session manager is available and in use for this session. The *pssuse* indicates whether the Programmable Symbol Set (PSS) is in use or not.

User Response: None; this is an informational message.

AP2WIVP32   >>> **NOT IN USE, AP120 NOT VERIFIED.**

Explanation: The session manager is not in use; AP 120 is not available.

User Response: Review the installation of auxiliary processor 120.

AP2WIVP40   >>> **LEVEL** *level* **INVALID**

Explanation: An attempt to find the APL2 *level* has produced an invalid data string.

User Response: Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP41   **EXECUTING WITHIN USER ADDRESS SPACE (NON-SAVED SEGMENTS)**

Explanation: APL2 is not executing in a saved segment.

User Response: None; this is an informational message.

AP2WIVP42   **EXECUTING OUTSIDE USER ADDRESS SPACE (SAVED SEGMENT NAME** *name***)**

Explanation: APL2 is executing in a saved segments, represented above by *name*.

User Response: None; this is an informational message.

AP2WIVP43   **CPU MODEL** *model* **- SMF IDEN-TIFIER** *id*

Explanation: This message informs you of the *model* and SMF identifier for the CPU (represented by *id*).

User Response: None; this is an informational message.

AP2WIVP44   *component* **LEVEL** *level*

Explanation: The named *component* is at the *level* noted above.

User Response: None; this is an informational message.

AP2WIVP52   **>>> SQL RC**(*rc*) *text*

Explanation: An attempt to access the SQL/DS or DB2 system catalog through AP 127 has resulted in an error. *rc* is the AP 127 return code, and *text* is the corresponding message text.

User Response: Check the AP 127 message description, and take the appropriate action.

AP2WIVP53   **SYSTEM CATALOG ENTRY FOR APL2:**

Explanation: Following the message shown above, you will receive a listing of the contents of the SQL/DS or DB2 system catalog entry for AP 127.

User Response: None; this is an informational message.

AP2WIVP54   **SESSION MANAGER NOT IN USE. GDDM MAY ERASE 327X SCREEN.**

Explanation: The session manager is not in use. A subsequent GDDM call may erase the screen. This message may be given during AP 126 verification.

User Response: None; this is an informational message.

AP2WIVP59   *product* **SAVED SEGMENT(S) LISTED IN NAMES FILE:** *list*

Explanation: The saved segment(s) for product *product*, as identified in the APL2 installation NAMES files, are named in *list*. If *list* is the single value *NONE*, then the APL2 CMSIVP will assume *product* is not installed in a saved segment. The APL2 installation NAMES files and their use are described in more detail in *APL2 Installation and Customization under CMS*.

System Action: If the value of *list* is not *NONE*, the CMSIVP checks if *product*'s saved segment(s) overlap with those of APL2 or those segments the CMSIVP knows about from other products used by APL2.

User Response: None.

AP2WIVP60   **>>> SAVED SEGMENT** *name*

Explanation: Overlaps virtual machine

User Response: None; this is an informational message.

AP2WIVP61   *name - address1* **TO** *address2*

Explanation: Above, *name* is the name of a saved segment, and *address1* and *address2* are its starting and ending addresses.

User Response: None; this is an informational message.

AP2WIVP62   **>>> VSAM NOT AVAILABLE.**

Explanation: The presence of VSAM cannot be verified.

User Response: Ensure that APL2 is installed on a current version of the host system to which all fixes have been applied. Contact your local service representative for further assistance.

AP2WIVP63   **>>> SAVED SEGMENTS: OVERLAPPING SEGMENTS**

Explanation: SAVED SEGMENT segments overlap. The addresses follow.

User Response: Examine the saved segments which overlap. If you plan to use any of them concurrently, you must move them so they do not overlap.

AP2WIVP64   *product* **SAVED SEGMENT NAME(S) UNKNOWN;** *product* **WILL NOT BE TESTED**

Explanation: The saved segment name(s) for *product* have not been defined in the APL2 installation NAMES files. The APL2 installation NAMES files and their use are described in more detail in *APL2 Installation and Customization under CMS*.

System Action: The CMSIVP avoids testing parts of APL2 which use *product*, since *product* might load saved segments which overlap the APL2 product, causing APL2 to fail.

User Response: None.

AP2WIVP65   **>>> DUE TO saved segments OVERLAP,** *component* **WILL NOT BE TESTED.**

Explanation: The test of some *component* (VSAM, GDDM, SQL) is being bypassed because of saved segments overlap.

User Response: Review the components not tested.

If you plan to use them and are not going to move the overlapping saved segments, you may want to perform your own testing to ensure the components are working properly.

AP2WIVP66   *indicator* **APL2 WORKSPACE**
wsid: message

Explanation: If the APL2 workspace, *wsid*, is in place, then the *message* says 'WS IN PLACE' and *indicator* is blank. Otherwise, *indicator* is '>>>' and *message* is an AP 104 error message.

User Response: This is an informational message. If the message indicates the workspace wasn't verified, correct the condition specified in the message.

AP2WIVP67   **>>> AP104 NOT AVAILABLE. UNABLE TO VERIFY LIBRARY.**

Explanation: Auxiliary processor 104 is not available, but it is needed to verify workspace installation.

User Response: Review the auxiliary processor installation procedure.

AP2WIVP68   **NO SAVED SEGMENTS OVERLAP FOR** *component*

Explanation: There is no saved segments overlap for the listed *component*(s).

User Response: None; this is an informational message.

AP2WIVP69   **>>> FOR EXPLANATION OF SQL RETURN CODES, SEE**

Explanation: This message is printed after message AP2WIVP52. Together with message AP2WIVP70, it gives you necessary information about where to find SQL return codes.

User Response: None; this is an informational message.

AP2WIVP70   **>>>** *publication*

Explanation: This message is a continuation of message AP2WIVP69. *publication* contains the name of the appropriate database manual.

User Response: None; this is an informational message.

AP2WIVP71   **>>> ASSOCIATED PROCESSOR** *processor* **FAILED**

Explanation: An attempt to use an associated *processor* has failed.

User Response: Verify that all installation steps have been completed for the associated processor.

AP2WIVP100   **AP100 CMS COMMAND AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP101   **AP101 ALTERNATE INPUT AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP102   **AP102 STORAGE ACCESS AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP104   **AP104 (ONLY FOR INTERNAL USE BY APL2)**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP110   **AP110 CMS FILE AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP111   **AP111 QSAM AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP119   **AP119 TCP/IP AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP120   **AP120 APL2 SESSION MANAGER COMMAND AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP121   **AP121 APL DATA FILE AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP123   **AP123 VSAM AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP124   **AP124 FULLSCREEN AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP126   **AP126 GDDM AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP127   **AP127 SQL AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP210   **AP210 BDAM FILE AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WIVP211   **AP211 APL2 OBJECT FILE AUXILIARY PROCESSOR**

Explanation: This message identifies an auxiliary processor that is being verified.

User Response: None; this is an informational message.

AP2WOFFER0   **LEFT ARGUMENT MUST BE SPECIFIED**

Explanation: A left argument has not been specified in a call to the $OFFER$ function.

User Response: This is a programming logic error. Consult your local service representative.

AP2WOFFER1   **LOCAL NAME CONFLICTS WITH SHARE NAME** *name*

Explanation: An attempt has been made to share a variable whose *name* conflicts with a local variable in the $OFFER$ function.

User Response: This is a programming logic error. Consult your local service representative.

AP2WOFFER2   *name* **IS ALREADY OFFERED OR IS NOT A VALID NAME**

Explanation: An attempt has been made to share a variable, *name*, that has already been offered.

User Response: Reinvoke the $CMSIVP$ or $TSOIVP$ function.

AP2WOFFER3   **UNABLE TO COUPLE** *names*

Explanation: An attempt has been made to share two variables, *names*, that do not couple.

User Response: Reinvoke the $CMSIVP$ or $TSOIVP$ function.

FILESERV Workspace Messages

Messages from the VSAMDATA workspace (see "VSAMDATA Workspace Messages" on page 95) may also be issued from the FILESERV workspace.

AP2WFILESERV51 UNABLE TO OPEN SYSIN

Explanation: An attempt to open a file with ddname SYSIN failed.

User Response: Allocate a SYSIN file which contains *IMPORT* and/or *EXPORT* commands, or specify such a command to a *SYSIN* variable, then reinvoke *FILESERV*.

AP2WFILESERV52 UNKNOWN CONTROL STATEMENT TYPE *command*

Explanation: The *command* in SYSIN does not begin with either *EXPORT* or *IMPORT*, the continuation character is missing in the previous command, or the statement name is misspelled.

User Response: Correct the *command* in *SYSIN*, and reinvoke *FILESERV*.

AP2WFILESERV53 INVALID OPERAND *operand*

Explanation: The command issued is invalid for one of the following reasons:

1. An *operand* is misspelled or invalid for the command.

2. The command contains invalid delimiters or mismatched parentheses.

3. An operand sublist contains misspelled items or invalid characters.

User Response: Change the command so that it is properly formed.

AP2WFILESERV54 REQUIRED OPERAND NOT SPECIFIED

Explanation: You specified the character / in your *IMPORT* or *EXPORT* command, but you did not specify a password.

User Response: Specify the password, and issue the command again.

AP2WFILESERV56 FILE ALREADY EXISTS

Explanation: The *IMPORT* command issued implies replacement of the named file, but the *REPLACE* operand is not specified.

User Response: Delete the existing file, or specify the *REPLACE* operand.

AP2WFILESERV57 FILE DOES NOT EXIST

Explanation: The *EXPORT* command issued specifies a file that could not be found in the input APL library (or in the dataset specified in the *FROM* operand), or the wrong input APL library is allocated. Either the library or file name is incorrect.

User Response: Make sure that the file name and the library number are correct.

AP2WFILESERV59 LIBRARY *number* **IS FULL**

Explanation: The numbered APL library, *number*, into which you are attempting to import an AP 121 data file, is full.

User Response: Provide sufficient space in the APL library and reexecute the command.

AP2WFILESERV60 UNABLE TO OPEN DDNAME *ddname*. **RS:** *reasoncode*; **RC:** *rc*

Explanation: An attempt to open the file designated by *ddname* failed. The reason and return code for the failure are represented above by the variables *reasoncode* and *rc*.

User Response: Do the following:

1. Correct the error indicated by the associated error message and reexecute the command.

2. Determine the meaning of the return codes for the specified auxiliary processor (see *reasoncode*), correct the error, and reexecute the command.

3. Correct any other errors for the same *ddname* and reexecute the command.

AP2WFILESERV61 FILE IS EMPTY

Explanation:

One of the following is the cause of the problem:

- The sequential file that you are attempting to *IMPORT* into an AP 121 data file contains no data records.

- The AP 121 data file that you are attempting to *EXPORT* to a sequential file contains no data records.

User Response: Specify an appropriate input file and reexecute the command.

AP2WFILESERV62   I/O ERROR IN *function*; **RS:**
                      reasoncode; **RC:** *rc*

Explanation:

1. An I/O error was detected in the function *function*. The return codes, represented by the variable *rc*, are given by AP 111.

2. An I/O error was detected in the APL library indicated by *function*; the return codes are given by AP 123.

User Response: Determine the meaning of the return codes provided, correct the error, and reexecute the command. Consult *APL2 Programming: System Services Reference* for more information.

AP2WFILESERV63   VARIABLE NOT SHARED. RS:
                      reasoncode; **RC:** *rc*

Explanation: An attempt to share a variable with the auxiliary processor specified *reasoncode* failed; the failure is indicated by the return code, *rc*.

User Response: Determine the meaning of the return codes provided for the specified auxiliary processor,

correct the error, and reexecute the command. Consult *APL2 Programming: System Services Reference* for more information.

AP2WFILESERV64   OPEN FAILURE. RS:
                      reasoncode; **RC:** *rc*

Explanation: The return code, *rc*, returned in the record and control variables shared with the auxiliary processor, *reasoncode*, indicate that the file was not successfully opened.

User Response: Determine the meaning of the return codes provided for the specified auxiliary processor, correct the error, and reexecute the command. Consult *APL2 Programming: System Services Reference* for more information.

AP2WFILESERV65   DD NAME *ddname* **NOT ALLO-**
                      CATED

Explanation: The *ddname* shown has not been allocated.

User Response: Allocate the ddname to the appropriate data set and reexecute the command.

FSC126 Workspace Messages

For $FSC126$ Workspace messages, see the
messages for workspace FSM on page 85.

FSM Workspace Messages

AP2WFSM0   UNABLE TO SHARE WITH AP *number;degree* **IS** □*SVO* **OF** *CTLs, DATs*

Explanation: An attempt to share with the indicated auxiliary processor has failed. 'degree' is the result of issuing the share offer.

User Response: Ensure that *CTLs* and *DATs* have been retracted and attempt to share again.

AP2WFSM1   ARGUMENT MUST BE SHAPE *shape*

Explanation: The argument to a function is the wrong shape.

User Response: Reinvoke the function with a correctly formed argument.

AP2WFSM2   ARGUMENT MUST BE RANK *rank*

Explanation: The argument to a function is the wrong rank.

User Response: Reinvoke the function with a correctly formed argument.

AP2WFSM3   LENGTH SPECIFIED IS INCONSISTENT WITH CHARACTER DATA

Explanation: The length specified in a call is not the same as the shape of the character data passed.

User Response: Reissue the call with the data and length consistent.

AP2WFSM4   LENGTH SPECIFIED IS INCONSISTENT WITH NUMERIC DATA

Explanation: The length specified in a call is not the same as the shape of the numeric data passed.

User Response: Reissue the call with the data and length consistent.

AP2WFSM5   CODE SPECIFIED IS INVALID

Explanation: An invalid code was specified in a GDDM call.

User Response: Reissue the call, ensuring that the code is valid and has the correct shape.

AP2WFSM6   NUMERIC DATA SPECIFIED MUST HAVE *number dimension*

Explanation: The numeric data specified must have the indicated number of columns or rows.

User Response: Reissue the call, ensuring that the data has the correct shape.

AP2WFSM7   INVALID FORMAT: *reason*

Explanation: An invalid format has been encountered. *reason* describes the detected error.

User Response: Take the appropriate action suggested by the *reason* noted in the message.

AP2WFSM9   SESSION MANAGER RETURN CODE: *code*

Explanation: An error has occurred in accessing AP 120. *code* is the AP 120 return code.

User Response: Diagnose the AP 120 return code. Consult *APL2 Programming: System Services Reference* for more information.

AP2WFSM11   LOST SHARE WITH AP *number*

Explanation: The shared variable link with the indicated auxiliary processor has been broken.

User Response: See your system administrator if the problem persists.

AP2WFSM12   FULLSCREEN AP RETURN CODE

Explanation: AP 126 has returned an error code. The code will follow.

User Response: Diagnose the AP 126 return code.

AP2WFSM13   RETURNED DATA MAY BE INCORRECT BECAUSE OF RETURN CODES

Explanation: Errors have been processed by the *GDDMANALYZE* function that are not serious enough to stop the program.

User Response: This is a warning message. Issue *HOW* in the *FSM* workspace for a description of how to obtain further information about the nature of the errors.

GRAPHPAK Workspace Messages

AP2WGRAPHPAK001 "AND" LENGTH ERROR

Explanation: Arguments of AND do not conform. For example, in the expression $A\ AND\ B\ VS\ C$, the three vectors must have the same length.

AP2WGRAPHPAK002 RIGHT ARGUMENT OF "AND" MUST BE A HOMO-GENEOUS GROUP

Explanation: If the left argument of AND is a vector, the right argument must consist of curves, all of the same length.

AP2WGRAPHPAK003 IF RT ARG OF "AND" IS A "VS"GRP, LEFT CANNOT BE A MATRIX

Explanation: In the structure $Y1\ AND\ Y2\ VS\ X$, $Y1$ cannot be a matrix. $Y1$ must be a vector with the same length as X.

AP2WGRAPHPAK004 0 IS INVALID ATTRIBUTE PARAMETER

Explanation: Zero cannot be used as the right argument of $COLOR$, $STYLE$, $WIDTH$, or $MODE$.

AP2WGRAPHPAK005 ATTRIBUTE LENGTH ERROR

Explanation: If two or more of $COLOR$, $STYLE$, $WIDTH$, or $MODE$ are used in combination, their right arguments must have the same length or one must be a scalar.

AP2WGRAPHPAK006 NOT ABLE TO PRODUCE FUNCTION "FITFUN"

Explanation: The function $FITFUN$ is probably suspended. Check the state indicator by entering $)SI$. A list of function names and statement numbers will be displayed, some indicated by asterisks ($*$). Enter → as many times as there are asterisks in the list.

AP2WGRAPHPAK007 CURSOR OUTSIDE OF WINDOW

Explanation: The $READ$ function was used to return the location of the cursor but the cursor was outside the graphics field.

AP2WGRAPHPAK008 SYMBOL SETS DO NOT EXIST ON AUXILIARY STORAGE

Explanation: The symbol sets have not been found. An attempt will be made by the function to load the symbol sets from the workspace. If this is done, however, the symbol sets will not be available to the printer utility and default symbol sets will be used.

AP2WGRAPHPAK009 DEVICE NOT SUPPORTED

Explanation: A call of $READ$ (graphical input) was made for an unsupported device.

AP2WGRAPHPAK010 ARGUMENTS OF "WITH" MUST BE OF OPPOSITE TYPE

Explanation: $WITH$ is used to pass numeric data and alphanumeric data to the $PIECHART$ function. Its two arguments may appear in either order, but if one is numeric the other must be literal.

AP2WGRAPHPAK011 ARGUMENT OF ¨USE¨ MUST HAVE RANK LESS THAN 3

Explanation: The function USE expects a scalar, vector, or matrix argument.

AP2WGRAPHPAK012 LEFT ARGUMENT OF ¨USING¨ MUST BE A MATRIX

Explanation: The left argument of $USING$ must be a matrix of data that can be passed to $DRAW$ or $FILL$. For example,

$DRAW\ DATA\ USING\ COLOR\ 5$

AP2WGRAPHPAK013 ¨VS¨ ERROR

Explanation: Length problem. The arguments of the function VS are not of the same length. Check the lengths and try again.

AP2WGRAPHPAK014 CANNOT DO LOG PLOT OF NON-POSITIVE NUMBERS

Explanation: All data for a logarithmic plot must be positive, and at least one of your values is negative or zero.

AP2WGRAPHPAK015   ¨B¨ ON LEFT ONLY POS-
SIBLE IN ¨STEP¨ WITH 3
COLUMNS ON RIGHT

Explanation: If the control character B is used in the
left argument of $STEP$, then the right argument must
be a three-column matrix. The first column is the vector
of x-values to begin each interval, the second column is
the vector of x-values to end each interval, and the third
column is the vector of y-values.

AP2WGRAPHPAK016   INVALID ¨PIECHART¨
ARRAY

Explanation: The right argument of $PIECHART$ must
be a numeric vector, a matrix, or a special rank 5 array
produced by the $WITH$ function.

AP2WGRAPHPAK017   SYMBOL SETS NOT FOUND
IN WORKSPACE

Explanation: Go back to an original version of
GRAPHPAK to find the symbol sets. You can use
GRAPHPAK even if the symbol sets are not found, but
the default symbol set does not have the APL charac-
ters.

AP2WGRAPHPAK018   WARNING - IMAGE
SYMBOL SET IN USE

Explanation: A call to $PRINTFILE$ was made to
produce a PSEG from an image which used image
symbol sets. $PRINTFILE$ produces a file suitable for
printing on Family 4 devices. No APL image symbol set
supporting these devices is shipped with APL2. The
text portions of the image will probably appear too small
when printed. Set $dd[9]\leftarrow2$ to force $GRAPHPAK$ to
use a vector symbol set (which is device-independent),
reconstruct the image, and recall $PRINTFILE$.

AP2WGRAPHPAK021   GDDM AP RETURN CODE
ERROR: number

Explanation: The GDDM auxiliary processor (AP 126)
detected an error and did not pass the information on to
GDDM. The *number* following the message is the
return code from the auxiliary processor. An explana-
tion of the return code is given in *APL2 Programming:
System Services Reference*.

AP2WGRAPHPAK022   GDDM RETURN CODE
ERROR: number

Explanation: When this message occurs, it is followed
on the next line by a message from GDDM about the
last error it detected. If more than one error was
detected by GDDM, previous errors are identified only
by the GDDM error number; information about these
can be found in the *GDDM User's Guide*.

If one passage of information from AP 126 to GDDM
results in several errors, the order in which these errors
are listed is not necessarily the order in which the corre-
sponding GDDM calls were listed in AP 126.

System Action: APL2 processing is stopped and
control is returned to the APL2 interactive session
through a branch out (→).

AP2WGRAPHPAK023   GDDM RETURN CODE
WARNING: number

Explanation: See message AP2WGRAPHPAK022.

System Action: APL2 processing continues.

AP2WGRAPHPAK024   SESSION MANAGER AP
RETURN CODE ERROR:
number

Explanation: The values passed to the APL2 session
manager are invalid. The field defined must be at least
3 lines high and 40 characters wide and cannot exceed
the limits of the screen.

AP2WGRAPHPAK025   GDDM AP NOT SHARING

Explanation: The GDDM auxiliary processor (AP 126)
is not active; it must be active for you to produce a
graphic display. (Probably APL2 was not installed cor-
rectly.)

AP2WGRAPHPAK026   SYSTEM ABEND
DETECTED IN FSQERR.
PROBABLY INSUFFICIENT
SPACE

Explanation: The amount of storage is not large
enough to complete processing. Contact your system
administrator.

AP2WGRAPHPAK027   DEVICE name NOT SUP-
PORTED

Explanation: The specified device is not supported by
GRAPHPAK. Specify a supported device or contact
your system administrator.

AP2WGRAPHPAK028   MUST BE MORE THAN
number DATA POINTS

Explanation: You have specified an insufficient
number of data points. Reissue the command with
more data points than *number*

MEDIT Workspace Messages

AP2WMEDIT0   **NO REPLACEMENT PAST COLUMN** *number*

Explanation: This message is from the $CHANGE$ function. No text may be replaced beyond the column *number* specified.

User Response: None; this is an informational message. You may need to rephrase your $CHANGE$ request.

PRINTWS Workspace Messages

AP2WPRINTWS1 **PRINTWS IS NOT IMPLE-MENTED FOR HOST SYSTEM** *host*

Explanation: The $HOST$ function has returned a value that is neither CMS nor TSO.

User Response: Ensure that APL2 is installed on a supported version of the host system.

AP2WPRINTWS2 **AP***number* **SHARE FAILURE**

Explanation: An attempt to share a variable with the listed auxiliary processor has failed.

User Response: Consult *APL2 Diagnosis* to troubleshoot the share failure.

AP2WPRINTWS3 **NAME ASSOCIATION WITH PROCESSOR 11 NOT ACTIVE**

Explanation: The $LIST$ function requires associated processor 11, and it is not available.

User Response: Verify that processor 11 has been properly installed and that a NAMES file exists for it.

AP2WPRINTWS4 **AP111 READ ERROR. RETURN CODE:** *rc*

Explanation: An attempt to read a file with AP 111 has failed. *rc* is the auxiliary processor return code.

User Response: Diagnose the auxiliary processor return code, and take the appropriate action. Consult *APL2 Programming: System Services Reference* for more information.

AP2WPRINTWS5 **AP111 WRITE ERROR. RETURN CODE:** *rc*

Explanation: An attempt to write a file with AP 111 has failed. *rc* is the auxiliary processor return code.

User Response: Diagnose the auxiliary processor return code, and take the appropriate action. Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WPRINTWS6 **AP111 OPEN ERROR. JCL FILE:** *rc*

Explanation: An attempt to open an AP 111 JCL file has failed. *rc* is the return code.

User Response: Diagnose the auxiliary processor return code, and take the appropriate action. Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WPRINTWS7 **AP111 OPEN ERROR. RETURN CODE:** *rc*

Explanation: An attempt to open an AP 111 file has failed. *rc* is the return code.

User Response: Diagnose the auxiliary processor return code, and take the appropriate action. Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WPRINTWS8 **ARGUMENT NOT LITERAL**

Explanation: The argument to the $LIST$ function is not a character vector or matrix.

User Response: Reinvoke the function with the correct argument.

AP2WPRINTWS9 **AP***number* **ERROR. RETURN CODE:** *rc*

Explanation: An auxiliary processor (AP), represented above by *number*, has an error with a return code *rc*.

User Response: Diagnose the auxiliary processor return code, and take the appropriate action. Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WPRINTWS10 **FUNCTION** *function* **NOT MEANT FOR HOST SYSTEM** *host*

Explanation: A *function* has been invoked that is intended for a different *host* system.

User Response: Invoke the function under the intended host system only.

SQL Workspace Messages

AP2WSQL1   ERROR DURING OFFER

Explanation: An offer to share $'DAT'$ with AP 127 returned a 0.

AP2WSQL2   UNABLE TO OFFER

Explanation: An offer to share $'DAT'$ was not coupled after successive tries.

AP2WSQL3   'DAT' IS NOT SHARED WITH AP 127.

Explanation: A function was called that requires a prior coupling of the variable $'DAT'$ but $'DAT'$ is not coupled.

AP2WSQL4   ITEM IS EMPTY, ALL ZERO, OR ALL BLANK

Explanation: An argument to a workspace function is empty, all zeros, or all blank, but the function requires an argument with values.

AP2WSQL5   ARGUMENT MUST BE A VECTOR

Explanation: The workspace function requires a vector argument.

AP2WSQL6   RIGHT ARGUMENT LENGTH ERROR

Explanation: The right argument to a workspace function ($SHOW$, SQL, QUE) contains the wrong number of items.

AP2WSQL7   STATEMENT ITEM MUST BE SIMPLE

Explanation: The SQL statement passed to the SQL function contains an item that is not a simple scalar, vector, or matrix.

AP2WSQL8   ARGUMENT MUST BE SCALAR OR VECTOR

Explanation: The right argument of an access function ($OPEN$ or $CALL$, for example) must be a simple vector, an enclosed simple vector, or a depth 2 vector.

AP2WSQL9   ARGUMENT MUST NOT BE EMPTY

Explanation: The workspace function requires a non-empty argument.

AP2WSQL10   ARGUMENT MUST NOT BE SIMPLE

Explanation: The right argument of the PREP function must not be simple.

AP2WSQL11   FIRST ITEM IS NOT A CHARACTER STRING

Explanation: The SQL function requires a character string as the first item of its argument.

AP2WSQL12   CHARACTER STRING LENGTH IS ONE

Explanation: The workspace function SQL requires an SQL statement; it must contain more than one character.

AP2WSQL13   WRONG NUMBER OF ARGUMENTS FOR THIS STATEMENT

Explanation: The SQL function does not allow a value list with an EXEC immediate statement.

AP2WSQL14   MESSAGE CODE MUST BE A SIMPLE LENGTH 5 VECTOR

Explanation: The $MESSAGE$ function requires a simple vector of length 5 as an argument.

AP2WSQL15   STACK MUST BE A MATRIX

Explanation: The stack passed by $RESUME$ to QUE must be a one-column matrix.

AP2WSQL16   CHART CALL HAS FAILED. RETURN CODE IS rc

Explanation: A call to the Interactive Chart Utility has failed. rc is the return code from GDDM.

TRANSFER Workspace Messages

AP2WTRANSFER1   ERROR BUILDING FILE MCOPY, *reason*

Explanation: This message gives you the *reason* that an attempt to build a $MASSMCOPY\_$ file has failed.

User Response: Use the reason indication to analyze the failure.

AP2WTRANSFER4   AP101 NOT SHARING

Explanation: An attempt to share a variable with AP 101 has failed.

User Response: Reload the workspace to retract all variables, and try again. If the failure persists, consult *APL2 Diagnosis* to troubleshoot the share failure.

AP2WTRANSFER5   AP101 STACK BUILDING FAILED WITH CODE *rc*

Explanation: An attempt to add an item to the input stack has failed. *rc* is the AP 101 return code caused by this error.

User Response: Analyze the AP 101 return code to determine the reason for failure. Consult *APL2 Programming: System Services Reference* and Chapter 6, "APL2 Auxiliary Processor Codes" on page 101 for more information.

AP2WTRANSFER6   AP110 NOT SHARING

Explanation: An attempt to share a variable with AP 110 has failed.

User Response: Reload the workspace to retract all variables, and try again. If the failure persists, consult *APL2 Diagnosis* to troubleshoot the share failure.

AP2WTRANSFER7   AP110 READ ERROR

Explanation: An attempt to read an AP 110 file has failed.

User Response: Ensure that the file contains valid data.

AP2WTRANSFER8   AP121 NOT SHARING

Explanation: An attempt to share a variable with AP 121 has failed.

User Response: Reload the workspace to retract all variables, and try again. If the failure persists, consult *APL2 Diagnosis* to troubleshoot the share failure.

AP2WTRANSFER9   READ ERROR ON FILE MCOPY, *reason*

Explanation: This message gives you the *reason* that an attempt to read a $MASSMCOPY\_$ file has failed.

User Response: Use the reason indication to analyze the failure.

AP2WTRANSFER10   UPDATE ERROR ON FILE MCOPY, *reason*

Explanation: This message gives you the *reason* that an attempt to update a $MASSMCOPY\_$ file has failed.

User Response: Use the reason indication to analyze the failure.

AP2WTRANSFER11   ERASE ERROR ON FILE MCOPY

Explanation: An attempt to delete the $MASSMCOPY\_$ file has failed because the file does not exist.

User Response: None; this is an informational message.

AP2WTRANSFER12   AP111 NOT SHARING

Explanation: An attempt to share a variable with AP 111 has failed.

User Response: Reload the workspace to retract all variables, and try again. If the failure persists, consult *APL2 Diagnosis* to troubleshoot the share failure.

AP2WTRANSFER13   AP111 ERROR, *rc*

Explanation: An error has occurred using AP 111. *rc* is the return code for this error.

User Response: Analyze the AP 111 return code to determine the reason for failure. Consult *APL2 Programming: System Services Reference* and Chapter 6, "APL2 Auxiliary Processor Codes" on page 101 for more information.

AP2WTRANSFER14   ALLOCATION FAILED FOR FILE *filename*

Explanation: An attempt to allocate the file, represented above by the variable *filename*, has failed.

User Response: Under CMS, the *filename* you specified is invalid. Use a file name that is valid for CMS.

For TSO, this message follows a TSO message describing the problem. For more information, see *OS/VS Message Library: VS2 TSO Terminal Messages*.

TSO Workspace Messages

AP2WTSO0   FILE *filename* **ALREADY OPEN.**

Explanation: An attempt has been made to open a file, called *filename*, that is already open.

User Response: Use the $CLOSE$ function to close the file, and then rerun.

AP2WTSO1   ALLOCATE ERROR *options*

Explanation: A file cannot be allocated. *options* is a list of allocation options.

User Response: Use the TSO LISTDS command to verify the dataset attributes, and ensure that the allocate options used by the $OPEN$ function are valid.

AP2WTSO2   SHARE FAILURE: *code*

Explanation: The $OPEN$ function cannot share with the QSAM file processor (AP 111). *code* is the result of $\Box SVO$.

User Response: Consult *APL2 Diagnosis* to troubleshoot the share failure.

AP2WTSO3   OPEN FAILURE: *values*

Explanation: A file cannot be opened. *values* contains the current values of the record and control variables. The latter value is an AP 111 return code.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WTSO4   READ ERROR. RETURN CODE: *rc*

Explanation: An error has occurred while attempting a file read. *rc* is the AP 111 return code.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WTSO5   GET ERROR. FILE *file* **RETURN CODE:** *rc*

Explanation: An error has occurred in the *GET* function. *file* is the file name, and *rc* is the AP 111 return code.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WTSO6   ALLOCATE FAILURE. CODE = *code*

Explanation: A file cannot be allocated by the $PUTFILE$ function. *code* is the return code from the TSO ALLOC command.

User Response: Use the TSO LISTDS command to verify the dataset attributes, and ensure that the allocate options passed to the $PUTFILE$ function are valid.

AP2WTSO7   WRITE ERROR: *rc*

Explanation: An error has occurred while attempting a file read. *rc* is the AP 111 return code.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WTSO8   PUT ERROR: *filecode*

Explanation: An error has occurred in the GET function. *filecode* is the file name followed by the AP 111 return code.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WTSO9   TSO SHARE OFFER FAILURE

Explanation: An attempt to share a variable with the TSO command processor (AP 100) has failed.

User Response: Consult *APL2 Diagnosis* to troubleshoot the share failure.

AP2WTSO10   TSO RETURN CODE: *rc*

Explanation: A nonzero return code has been returned by the TSO command processor, and *rc* is the AP 111 return code.

User Response: Consult *APL2 Programming: System Services Reference* for an explanation of the AP 111 return code.

AP2WTSO11   TSO ABEND CODE: *rc*

Explanation: An abend has occurred in TSO while using the TSO command processor. *rc* is the TSO abend code.

User Response: A TSO abend message and system abend should have been displayed on your terminal. Consult *TSO Terminal User's Guide*.

AP2WTSO12   COMMAND *command*

Explanation: An error has occurred while using the TSO command processor. *command* is the command issued. This message follows AP2WTSO10 or AP2WTSO11.

User Response: Pursue the action appropriate to the previous message displayed.

**AP2WTSO13   SHARE FAILURE - ALTERNATE
                       INPUT PROCESSOR:** *code*

Explanation: An attempt to share a variable with the alternate input processor (AP 101) has failed.

User Response: Consult *APL2 Diagnosis* to trouble-shoot the share failure.

**AP2WTSO14   NONZERO RETURN CODE - ALTER-
                       NATE INPUT PROCESSOR:** *rc*

Explanation: An error has occurred while using the alternate input processor. *rc* is the return code.

User Response: Consult Chapter 6, "APL2 Auxiliary Processor Codes" on page 101 and *APL2 Program-ming: System Services Reference* for an explanation of the AP 101 return code.

AP2WTSO15   *name* **PROCEDURE HAS BEEN
                       CREATED.**

Explanation: The $PROC$ function has created some procedure called *name*

User Response: None; this is an informational message.

VAPLFILE Workspace Messages

See also messages AP2WAP1211 through AP2WAP121100, page 71, for messages associated with auxiliary processor 121.

AP2WVAPLFILE101   FILE INDEX ERROR ON *function*

Explanation: One of the *functions* $ERASE$, GET, or SET encountered a file index error.

User Response: Reinvoke the function with a correct file index.

AP2WVAPLFILE102   *fileid* **FILE NOT IN USE**

Explanation: An attempt was made to get information from *fileid*, which is not in use.

User Response: Use the USE or AT commands to specify the appropriate values for *fileid*.

AP2WVAPLFILE103   *fileid* **FILE FULL**

Explanation: *fileid* does not contain enough free blocks to satisfy a request.

User Response: Specify a new file or increase the size of the current file.

AP2WVAPLFILE104   FILE VALUE ERROR

Explanation: There is no data on the file at the index specified.

User Response: Specify a correct index for the file.

AP2WVAPLFILE105   ARGUMENT ERROR ON *function*

Explanation: Some *function* has been given an invalid argument.

User Response: Specify a correct argument for the function.

AP2WVAPLFILE106   LEFT ARGUMENT MUST BE SPECIFIED

Explanation: The left argument to a function must be specified.

User Response: The $MORE$ variable contains an indication of what the argument must be. Specify the appropriate left argument to the function.

VSAMDATA Workspace Messages

AP2WVSAMDATA1   VSAM ERROR RETURN *rc*;
                     REASON *reasoncode*

Explanation: AP 123 has returned an error code. *rc*
and *reasoncode* are the two elements of the error code.

User Response: See "AP 123 Return Codes" on
page 111 to diagnose this error.

AP2WVSAMDATA2   SHARE OFFER WITH 123
                     FAILED. CODE *degrees*

Explanation: An attempt to share variables with AP
123 has failed. *degrees* is the result from $\Box SVO$.

User Response: Reload the workspace or retract all
variables and try again.

AP2WVSAMDATA3   FILE NOT CLEARED.
                     ALREADY OPEN CLEAR

Explanation: The $VCLEAR$ function has tried to clear
a file that is already $OPEN$ and $CLEAR$.

User Response: This is a warning message.

AP2WVSAMDATA4   KEY FEEDBACK AVAILABLE
                     ONLY AFTER READ, WRITE,
                     OR UPDATE

Explanation: A call to VKF has been issued right
after $POSITION$ or $OPEN$.

User Response: Issue this request only after a read,
write, or update has taken place.

AP2WVSAMDATA12   COMMAND SYNTAX ERROR.
                      CODE 12

Explanation: Syntax error in the value specified in the
control variable.

User Response: Probable workspace error. Report to
your system administrator unless you have modified
workspace functions.

AP2WVSAMDATA13   VSAM FILE ALREADY OPEN.
                      CODE 13

Explanation: You have issued a service request to
open the file. The file is already open.

User Response: Probable workspace error. Report to
your system administrator unless you have modified
workspace functions.

AP2WVSAMDATA15   INAPPROPRIATE VSAM
                      OPEN MODE. CODE 15

Explanation: The RU, W, or E service request speci-
fied is not permitted by the current open mode. RU and
E require an OU open. W requires OW, OU, or OC.

User Response: Probable workspace error. Report to
your system administrator unless you have modified
workspace functions.

AP2WVSAMDATA16   REQUEST INVALID FOR FILE
                      TYPE. CODE 16

Explanation: The service request is not valid for the
type of VSAM file being processed, or ERASE was
specified for an entry-sequenced data set.

User Response: Allocate a VSAM cluster which is not
entry sequenced.

AP2WVSAMDATA17   KEY LENGTH ERROR. CODE
                      17

Explanation: Length of record key specified in a
service request was too long, or it was too short on a
RU key request. Note that blanks are included in the
count of the key length.

User Response: Be sure you are accessing the
correct VSAM cluster. If you are, use a shorter key in
your request.

AP2WVSAMDATA18   VSAM MODCB ERROR.
                      CODE 18

Explanation: VSAM MODCB error.

User Response: VSAM or APL2 product error.
Report this to your system administrator.

AP2WVSAMDATA19   VSAM SHOWCB ERROR.
                      CODE 19

Explanation: VSAM SHOWCB error.

User Response: VSAM or APL2 product error.
Report this to your system administrator.

AP2WVSAMDATA20   NOT CHARACTER VECTOR
                      TO VSAM AP. CODE 20

Explanation: Something other than a character vector
was specified in the data variable.

User Response: Reformat your data as a character
vector.

AP2WVSAMDATA21   **RECORD LENGTH ERROR. CODE 21**

Explanation: Data variable was specified with an incorrect record length.

User Response: Be sure you are accessing the correct VSAM cluster. If you are, use a shorter key in your request.

AP2WVSAMDATA22   **FILE IS NOT OPEN. CODE 22**

Explanation: You have issued a service request to process a file that is not currently open.

User Response: Probable workspace error. Report this to your system administrator unless you have modified workspace functions.

AP2WVSAMDATA27   **VSAM TESTCB ERROR. CODE 27**

Explanation: VSAM TESTCB error.

User Response: VSAM or APL2 product error. Report this to your system administrator.

AP2WVSAMDATA32   **NO FREE SPACE FOR I/O BUFFER. CODE 32**

Explanation: Insufficient free storage for input/output buffers.

User Response: Refer to message AP...001.

AP2WVSAMDATA33   **DATA HAS NOT BEEN SPEC-IFIED. CODE 33**

Explanation: No value has been assigned to the data variable for a write request.

User Response: Probable workspace error. Report this to your system administrator unless you have modified workspace functions.

AP2WVSAMDATA42   **DATA HAS NOT BEEN REF-ERENCED. CODE 42**

Explanation: You have set access control on the data variable and you have not yet referenced it for an earlier request. (The first read request resulted in a return code of $0 \quad 0$.)

User Response: Probable workspace error. Report this to your system administrator unless you have modified workspace functions.

AP2WVSAMDATA45   **VSAM GENCB ERROR. CODE 45**

Explanation: VSAM GENCB error.

User Response: VSAM or APL2 product error. Report this to your system administrator.

AP2WVSAMDATA48   **INVALID COMMAND SEQUENCE. CODE 48**

Explanation: Invalid sequence of service requests.

- A key feedback (KF) request followed an open request or an error.
- A relative-record write without a key (W) was requested without a preceding read-for-update request (RU).

User Response: Probable workspace error. Report this to your system administrator unless you have modified workspace functions.

AP2WVSAMDATA50   **UNRECOGNIZED ERROR. CODE 50**

User Response: Probable workspace error. Report this to your system administrator unless you have modified workspace functions.

Chapter 4. Abend Codes

APL2 issues the abend codes listed in the figure below. The codes are identified in the Abend Code column, first by decimal value and then in hexadecimal form. The figure also lists the module(s) that issued the abend codes. Under the remarks column of the figure:

BAD　　　　means unexpected.

ERROR　　　means nonzero, unless qualified.

ANY CALL　means there are several calls in the module.

Figure 7 (Page 1 of 3). APL2 Abend Codes

Abend Code	Issuing Modules	Remarks
1 X'001'	AP2TTIO	Unsupported request or unexpected error.
1 X'001'	AP2T100	No space, already on, or logic error. See)*MORE* msg.
1 X'001'	AP2T101 AP2T111 AP2T210	Another AP has already signed on with same number.
3 X'003'	AP2T101 AP2T111 AP2T123 AP2T210 AP2V123	Sign off (ASVPSOF) failed.
17 X'011'	AP2TCSVS	Global SVP shutdown, dump requested.
1000 X'3E8'	AP2TYMSC	)*CHECK SYSTEM ABEND* intentional abend.
1002 X'3EA'	AP2ACPRO	Bad return code from AP2ACSF (any call).
1004 X'3EC'	AP2ACPRO	Error return code from AP2ACQUE (any call).
1010 X'3F2'	AP2ACDSL	Error (not warning) from GDDM (any call).
1020 X'3FC'	AP2ADISP	Bad return code from GDDM (any call).
1022 X'3FE'	AP2ACOPY AP2ACXCM AP2AFIND AP2ALINE	Bad return code from AP2ACSF (any call).
1030 X'406'	AP2ACRDA	Error (not warning) from GDDM (any call).
1032 X'408'	AP2ACRDA	Bad return code from AP2ACSF reading from log.
1040 X'410'	AP2ACRSA	Error (not warning) from GDDM (any call).
1050 X'41A'	AP2ACNDP	Error (not warning) from GDDM (any call).

Figure 7 (Page 2 of 3). APL2 Abend Codes

Abend Code	Issuing Modules	Remarks
1055 X'41F'	AP2ACNDP	Error getting dynamic virtual storage.
1060 X'424'	AP2ADMSG	Error (not warning) from GDDM (any call).
1070 X'42E'	AP2ACPRM	Error (not warning) from GDDM (any call).
1072 X'430'	AP2ACPRM	Error return code from AP2ACSF reading from log.
1074 X'432'	AP2ACPRM	Error return code from AP2ACQUE (any call).
1080 X'438'	AP2APUT	Error (not warning) from GDDM (any call).
1101 X'44D'	AP2TMSRV	Invalid AP2GLOCK free. Not held or bad pointer.
1103 X'44F'	AP2TMSRV	Request to free terminal, but not owned.
1208 X'4B8'	AP2TACTL	Task terminating normally without AP2GLOCK free.
1211 X'4BB'	AP2TASVP	APENTRY unresolved in invoking an AP using APNAMES.
1261 X'4ED'	AP2X126	Unable to get default storage for command buffer.
1262 X'4EE'	AP2X126	Unable to get default storage for return codes.
1263 X'4EF'	AP2X126	Unable to get default storage for input data.
1264 X'4F0'	AP2X126	Unable to get default storage for return data.
1265 X'4F1'	AP2X126	Unable to get default storage for a buffer.
1266 X'4F2'	AP2X126	Unable to get minimum storage for CTL output.
1267 X'4F3'	AP2X126	Unable to get minimum storage for CTL input.
1268 X'4F4'	AP2X126	Bad return code referencing CTL variable.
1269 X'4F5'	AP2X126	Bad return code referencing DAT variable.
1270 X'4F6'	AP2X126	Bad return code specifying CTL variable.
1271 X'4F7'	AP2X126	Bad return code specifying DAT variable.
1272 X'4F8'	AP2X126	Bad return code on CTL authcheck.
1300 X'514'	AP2XMSTK	Invalid AP2XPROC prolog.

Figure 7 (Page 3 of 3). APL2 Abend Codes

Abend Code	Issuing Modules	Remarks
1301 X'515'	AP2XMSTK	Stack block too small.
1302 X'516'	AP2XMSTK	Unable to get stack extension storage.
1501 X'5DD'	AP2XMSRV AP2XASRV AP2TNCAL AP2VNCAL	Invalid service request code.
1502 X'5DE'	AP2XMSRV AP2XASRV AP2TNCAL AP2VNCAL	Invalid number of parameters.
1600 X'640'	AP2VMSRV	APL2 dispatch has abended.
1610 X'64A'	AP2VMDSP	Error while all tasks were waiting.
1611 X'64B'	AP2VMDSP	Wait request issued from the wrong SVC nesting level.
2001 X'7D1'	AP2XACAP	Error during AP signon, not enough storage or rejected by SVP.
2011 X'7DB'	AP2X126	GDDX has forced an abend.
2103 X'837'	AP2X127M AP2X127F	Insufficient storage allocated by processor.
2104 X'838'	AP2X1270	Unrecognizable data format from APL.
2106 X'83A'	AP2V127I AP2T127I AP2T127C	Invalid internal code.
2148 X'864'	AP2X127S	Insufficient storage to allocate data buffer.
2211 X'8A3'	AP2X211	Unexpected return code from processor services.
3001 X'BB9'	AP2TYYR	Four ATTN signals received, being ignored by active process.
3140 X'C44'	AP2TINIT	APL interlocked, not receiving ATTN signals.
4041 X'FC9'	AP2TINC	No subtask start requests found.

Chapter 5. Dump Codes

APL2 issues dumps with the codes which are listed below. Next to each code is the module or modules which issued it.

ACTL	AP2TACTL AP2VACTL
APL2	AP2TYYR AP2VYYR
ASVP	AP2TASVP AP2VASVP
A104	AP2X104
A126	AP2X126
CAPS	AP2XACAP
CPRO	AP2ACPRO
FILE	AP2TFILE
LWSS	AP2TLWSS
NCAL	TNCAL VNCAL
SSM	AP2XACAP
XGDA	AP2XGCOM
XMSG	AP2XMSG
119X	AP2X119
120X	AP2X120
121X	AP2X121
127X	AP2X127M
211X	AP2X211

Chapter 6. APL2 Auxiliary Processor Codes

For more information on the APL2 Auxiliary Processors, see *APL2 Programming: System Services Reference*.

AP 100 Return Codes

Figure 8. Return Codes Issued by AP 100 under CMS

Code	Description
‾100	CMS command has abended. A CMS abend code accompanies this return code. The failing command may have damaged the integrity of CMS or APL2.
0	No error exists; the initial value was accepted or the command completed successfully.
1	Unknown CP command or improper initial value.
444	Invalid value assigned to the shared variable. It is the wrong shape, size, or data type.

Figure 9. Return Codes Issued by AP 100 under TSO

Code	Description
0	The TSO command completed successfully.
1	Invalid TSO command or unauthorized use of the command.
2	You terminated the command with an attention interrupt.
	If PROFILE PROMPT is set, TSO may have prompted you to supply a missing or invalid operand. If you were attempting to allocate a data set, the effect is the same as if PROFILE NOPROMPT was set.
3	Insufficient virtual storage was available to execute the command.
	Action: Reinvoke APL2, increasing the value of the FREESIZE option.
4	Insufficient shared storage to return the result. This may happen if you have specified one of the AP 100 built-in commands that return a large result value.
	Action: Reference other shared variables or reinvoke APL2 with a larger SHRSIZE value.
5	PDSI output is too large for the available buffer. This will occur only for small values of shared memory size (less than 64K) or for Partitioned Data Sets (PDSs) with more than about 4K members.
8	The specified data set was not found. This return code applies only to AP 100 built-in commands DDI and DSI.
444	An invalid data type was set into the shared variable. It is the wrong size, shape, or type.

AP 101 Return Codes

Figure 10. Return Codes Issued by AP 101

Code	Description
0	Success. The initial value, or the last user specification of the shared variable, was successful.
1	Invalid initial value. Subsequent specifications of the variable are interpreted as attempts to assign a valid initial value. **Action:** Specify a valid initial value. Input is not stacked until a return code of 0 is returned from the initial value.
12	**TSO Only:** Stack overflow. You have specified a data value that was too large to fit into the remaining space on the stack. The entire stack is purged. **Action:** If there is no program logic error on your part, reinvoke APL2 with a larger value specified in the AISIZE option. For more information on the AISIZE operand and the default stack size for AP 101, see *APL2 Programming: System Services Reference*.
444	Invalid value specified in the shared variable. It is the wrong size, shape, or data type.

AP 102 Return Codes

Figure 11. Return Codes Issued by AP 102

Code	Description
0	Successful completion of requested function.
2	The request specified in the control variable cannot be satisfied because you have not offered a data variable. The AP has offered the data variable to you, but you have not matched that offer. **Action:** Offer the data variable, ensure the degree of coupling is 2, and respecify the request.
3	Invalid service request. The first item specified in the control variable was not a 0 or 1.
4	Invalid number of parameters for specified service request.
5	Invalid APL2 data specified in the control variable. It is the wrong size, shape, or data type.
6	Invalid address specified in the control variable. The value is negative or represents an address that is outside the range of accessible storage.
7	Invalid length specified in the control variable. The value is negative or zero.
8	Insufficient storage available for AP 102 to return the contents of storage in the data variable. **Action:** Either specify a smaller length in the control variable, or reinvoke APL2 with a larger size specified in either the SHRSIZE or FREESIZE option.

AP 110 Return Codes

Figure 12. Return Codes Issued by AP 110 under CMS Only

Code	Description
0	No error exists.
1	Attempt to read a nonexistent file, or improper initial value.
3	Permanent I/O error.
4	First character of file mode is invalid.
5	Attempt to read more records than the maximum allowed by CMS.
6	Attempt to write too many records in a CMS file.
7	Attempt to write past the end of a variable-length file.
8	Attempt to read a record with incorrect record length.
10	Attempt to create a file when you already have the maximum allowed by CMS.
12	End-of-file read or attempt to write on a read-only disk.
13	Attempt to write on a full disk.
14	Attempt to write on an unformatted disk.
15	Attempt to write a record with incorrect length into a file with fixed format.
17	Attempt to write a record that is too large into a variable length file.
19	Attempt to write in a file already containing as many data blocks as CMS will allow.
22	Virtual storage capacity exceeded.
25	Insufficient storage for CMS file system.
26	Item number is invalid.
27	Attempt to replace a variable-length record with one of a different length.
443	Insufficient free storage for input/output buffers. Note that the amount of free storage allocated is based on the specified shared storage size. **Action:** Reinvoke APL2, increasing the value of the SHRSIZE invocation option. If the resultant workspace is too small, restart CMS with more virtual storage.
444	The value assigned to the shared variable is invalid. It is the wrong shape, size, or data type.
445	Attempt to read a record larger than shared storage. **Action:** Reinvoke APL2 increasing the value of the SHRSIZE invocation option. If the resultant workspace is too small, restart CMS with more virtual storage.

AP 111 Return Codes

Figure 13 (Page 1 of 2). Return Codes Issued by AP 111

Code	Description
0	The initial value in the shared variable offer was accepted; or successful completion of the requested I/O operation.
1	The control variable was specified and no matching data variable could be found or one of the two variables contains an invalid initial value: • No initial value was specified. • The value is not a character vector. • The syntax is incorrect. **Action:** Retract the variables offered, specify a valid syntax for the initial values, and reoffer the variables.
12	End of file. The value in the data variable is empty. The last record from the data set was passed in the previous read request. **Action:** Close the data set and retract the data variable.
15	Wrong length record on fixed-length output. When RECFM=F or FB, all records must be exactly equal to the record length specified in the FILEDEF or ALLOCATE command. **Exception:** With the CDR or VAR conversion option, this error occurs only if the APL variable exceeds the logical record size of the data set.
17	Record too large for output. This error occurs when a record to be written exceeds the LRECL specified for a data set with variable or undefined record lengths (RECFM=U, V, or VB).
440	Error in open routine for output. This error occurs under the following conditions: • **CMS or TSO:** The open routine for the data set has failed. System error messages accompany this return code. It usually means you have not properly defined or allocated the data set. • **TSO Only:** The R access option was specified in the initial value of the shared variable, and you have specified the data variable, indicating output (write) processing. • **TSO Only:** Your installation has prohibited your access to the data set. Additional MVS error messages are displayed. **Action:** Verify your allocation, initial values, and authorization. Verify the data set attributes on direct access by using the CMS LISTFILE or the TSO LISTDS command. Verify the options you have selected, using the CMS FILEDEF or TSO ATTRIB command, to see if there is conflict.
441	Error in open routine for input. This error may occur in the following situations: • You have not properly allocated the data set using the appropriate options. • The data set was opened for output, and you referenced the data variable. CLOSE processing failed. • **TSO Only:** Your installation has prohibited your access to the data set with security mechanisms. **Action:** Verify your allocation, initial values, and authorization. Verify the data set attributes on direct access by using the CMS LISTFILE or TSO LISTDS command with the options you selected in the CMS FILEDEF or TSO ATTRIB command to see if there is conflict.

Figure 13 (Page 2 of 2). Return Codes Issued by AP 111

Code	Description
443	Insufficient space to process output data. **Action:** Try reinvoking APL2, increasing the value of the SHRSIZE invocation option.
444	Invalid data type or shape. There is a conflict between the conversion option you have selected and the APL value you have specified in the record variable.
445	Insufficient shared storage for input data. **Action:** Reinvoke APL2 and specify a larger value in the SHRSIZE invocation option. **Note:** If you are using the VAR or CDR conversion option and you think the SHRSIZE invocation option is set properly, verify that the record you are trying to read has a valid APL internal form. Functions are available in the $UTILITY$ workspace to read a record from your data set (specified with the $BYTE$ conversion option) to verify the APL internal format.

AP 119 Return Codes

The result passed back to the user in the shared variable is always a three item vector: AP 119 return code, subsystem (AP, TCP/IP or IUCV) return code, and data.

Code	Definition
0	Success
1	Incorrect command
2	Wrong type
3	Wrong rank
4	Wrong shape
5	Item is wrong type
6	Item is wrong rank
7	Item is wrong shape
8	Item data is wrong
10	AP119 subsystem support error. Second element of result is defined as follows: **1** No more sockets available through this variable. **2** Insufficient storage available to process the command.
11	TCP/IP error occurred. Second element of result is TCP/IP return code.
12	IUCV error occurred. Second element of result is IUCV return code.

Figure 14. AP 119 Return Codes

AP codes 2 3 4 5 6 7 8 return in the second item of the return code the index to the item being processed when the error was detected. The first item is numbered 0.

If the first item of the return code is 11, the second item will be set to one of the character values listed under *Errno* in Figure 15 The numeric *Code* is used in messages AP2...304 and AP2...305.

Figure 15 (Page 1 of 2). TCP/IP Return Codes		
Code	**Errno**	**Definition**
0		Success
1	EPERM	Not Owner
2	ENOENT	No such file or directory
3	ESRCH	No such process
4	EINTR	Interrupted system call
5	EIO	I/O error
6	ENXIO	No such device or address
7	E2BIG	Arg list too long
8	ENOEXEC	Exec format error
9	EBADF	Bad file number
10	ECHILD	No children
11	EAGAIN	No more processes
12	ENOMEM	Not enough core
13	EACCES	Permission denied
14	EFAULT	Bad address
15	ENOTBLK	Block device required
16	EBUSY	Mount device busy
17	EEXIST	File exists
18	EXDEV	Cross-device link
19	ENODEV	No such device
20	ENOTDIR	Not a directory
21	EISDIR	Is a directory
22	EINVAL	Invalid argument
23	ENFILE	File table overflow
24	EMFILE	Too many open files
25	ENOTTY	Not a typewriter
26	ETXTBSY	Text file busy
27	EFBIG	File too large
28	ENOSPC	No space left on device
29	ESPIPE	Illegal seek
30	EROFS	Read-only file system
31	EMLINK	Too many links
32	EPIPE	Broken pipe
33	EDOM	Argument too large
34	ERANGE	Result too large
35	EWOULDBLOCK	Operation would block
36	EINPROGRESS	Operation now in progress
37	EALREADY	Operation already in progress
38	ENOTSOCK	Socket operation on non-socket
39	EDESTADDRREQ	Destination address required
40	EMSGSIZE	Message too long
41	EPROTOTYPE	Protocol wrong type for socket
42	ENOPROTOOPT	Protocol not available

Figure 15 (Page 2 of 2). TCP/IP Return Codes		
Code	**Errno**	**Definition**
43	EPROTONOSUPPORT	Protocol not supported
44	ESOCKTNOSUPPORT	Socket type not supported
45	EOPNOTSUPP	Operation not supported on socket
46	EPFNOSUPPORT	Protocol family not supported
47	EAFNOSUPPORT	Address family not supported by protocol family
48	EADDRINUSE	Address already in use
49	EADDRNOTAVAIL	Can't assign requested address
50	ENETDOWN	Network is down
51	ENETUNREACH	Network is unreachable
52	ENETRESET	Network dropped connection on reset
53	ECONNABORTED	Software caused connection abort
54	ECONNRESET	Connection reset by peer
55	ENOBUFS	No buffer space available
56	EISCONN	Socket is already connected
57	ENOTCONN	Socket is not connected
58	ESHUTDOWN	Can't send after socket shutdown
59	ETOOMANYREFS	Too many references: can't splice
60	ETIMEDOUT	Connection timed out
61	ECONNREFUSED	Connection refused
62	ELOOP	Too many levels of symbolic links
63	ENAMETOOLONG	File name too long
64	EHOSTDOWN	Host is down
65	EHOSTUNREACH	No route to host
66	ENOTEMPTY	Directory not empty
67	EPROCLIM	Too many processes
68	EUSERS	Too many users
69	EDQUOT	Disc quota exceeded
70	EVDBAD	RVD related disk error

AP 120 Return Codes

Figure 16. Return Codes Issued by AP 120

Code	Description
0 0	Normal return. Command was successful.
1 11	Rank error on control variable. Rank is greater than 1.
1 13	Domain error on control variable. Control variable contains noncharacter data.
1 53	Required storage is unavailable. **Action**: Reinvoke APL2, specifying a larger value for the FREESIZE option.
1 56	Shared variable quota exceeded.
1 60	The APL2 session manager is not available.
4 *nnn*	Command was passed to the session manager and an error was encountered. *nnn* is the numeric part of the session manager error message identifier. **Action**: Examine the DAT variable; it contains the text of the error message.

AP 121 Return Codes

Figure 17 (Page 1 of 3). Return Codes Issued by AP 121

Code	Description
0	Operation successful.
1	Record not found for a direct read or update request. **Action:** Specify an existing record number, and reissue the request.
2	**CMS Only:** File on a virtual disk blocked 800 contains maximum number of records. As a result, the file is closed. Maximum file size is 64K. **Action:** Move the file to a disk that is not formatted with 800-byte blocks.
3	Attempted to write an APL2-only object with `'SW'` or `'DU'`. **Action:** Use the SWC and DUC requests if you want to write or update APL2 objects in a file, or reformat the data so that it will be compatible with VS APL.
4	Either a logic error has been detected within the auxiliary processor, or (TSO only) the VSAM file is not a valid APL2 file library. **Action:** If you are using a TSO system, check the allocations in effect for APL2 files. These have DD*names* of F*number*, where *number* is a library number or zero. You may be trying to use a workspace library, or some non-APL VSAM cluster, as an APL file library. Otherwise, report the problem to your system administrator.
5	End of file. As a result, the file is closed and, for sequential read operations, an empty vector is assigned to the control variable.
6	You have attempted a direct read or direct update on a sequential file.
7	Record length error. You have tried to write a record (on a direct file) that is longer than the record length specified on the initial sequential-write (`'SW'`) open request.
9	**CMS:** Virtual disk is full. As a result, the file is closed. **TSO:** VSAM library is full. As a result, the file is closed.

Figure 17 (Page 2 of 3). Return Codes Issued by AP 121

Code	Description
10	**TSO Only:** User interrupted the operation with an attention key. If a WRITE or DROP was in progress, the record or file may be in an unusable state. It may later be deleted implicitly by the system, or an explicit DROP request may be used.
11	Defined file size limit exceeded, or request to reduce file size limit would cause data loss. As a result, the file is closed. **Action:** Increase the file size limit (via the FS service request), reopen the file, and retry your request.
12	Invalid service request, or syntax error in request. This code is returned for several error conditions associated with a service request: • Control variable does not contain a character vector when a service request is required. • An invalid service request code was specified. • A library number or the file size was not specified as a number, or it was too large a number. • File name or password is too long or has invalid characters. • A required field for the service request was omitted. • Unexpected fields were found. • Invalid direct read or update request. It was not a two-item numeric vector with the first item being either zero or one.
15	File not open. An attempt is being made to issue read or write requests against a file which is not open.
23	You have tried to replace a record on a direct file, but the file is not opened for direct update. Close and reopen file for direct update.
24	Improper library reference. **CMS Action:** Check LIBTAB APL2 for correct library numbers. **TSO Action:** Make sure the library is allocated.
25	File already exists.
26	File does not exist.
28	File in use by others. Your shared or exclusive file request is blocked. **Action:** Retry later. Be sure you do not already have the file open through another variable.
32	No space for I/O buffer. **Action:** Reinvoke APL2 and increase your FREESIZE. If this makes your WSSIZE or SHRSIZE too small, you will need to increase your: • Virtual machine size under CMS • TSO region size under TSO
33	You have attempted to replace a record on a direct file, but you did not specify the data variable before you issued the write service request.
34	Incorrect or missing password.

Figure 17 (Page 3 of 3). Return Codes Issued by AP 121

Code	Description
35	Hardware I/O error or APL2 product failure. As a result, the file is closed and, for sequential read operations, an empty vector is assigned to the control variable. **CMS Action:** Retry request. If problem persists, notify your system administrator. **TSO Action:** Verify that the library is a VSAM cluster defined with KEYS(14 0). Retry request. If problem continues, notify your system administrator.
36	Named file is not an APL2 data file.
37	File size limit has been reduced below the previous file size limit. This code is issued as a warning only; the FS operation is successful. There is still space for all records currently on the file.
38	An invalid record length was specified on a sequential write (`'SW'`) request for a direct file. It does not equal the record length specified when the file was first loaded. **Action:** Reissue the request either with an appropriate record length or without specifying a record length.
39	Required record length is not specified in the sequential write (SW) open request to load records on a newly-created direct file.
42	Data variable has not been referenced.
50	Invalid request for CMS/TSO, or the file system is not operational. **Action:** See your system administrator.

AP 123 Return Codes

Figure 18. Return Codes Issued by AP 123

Code	Description
1 12	Syntax error in the value specified in the control variable.
1 13	You have issued a service request to open the file. The file is already open.
	No action is required. Processing continues.
1 15	The *RU*, *W*, or *E* service request specified is not permitted by the current open mode. *RU* and *E* require an *OU* open. *W* requires *OW*, *OU*, or *OC*.
1 16	The service request is not valid for the type of VSAM file being processed, or ERASE was specified for an entry-sequenced data set.
1 17	Length of record key specified in a service request was too long, or it was too short on a *RU* key request. Note that blanks are included in the count of the key length.
1 18	VSAM MODCB error.
1 19	VSAM SHOWCB error.
1 20	Something other than a character vector was specified in the data variable.
1 21	Data variable was specified with an incorrect record length.
1 22	You have issued a service request to process a file that is not currently open.
1 27	VSAM TESTCB error.
1 32	Insufficient free storage for input/output buffers.
	Action: Restart the APL2 session with a larger value in the FREESIZE invocation option. If the resultant freespace size or workspace size is too small:
	• **CMS:** In CP mode, define a larger virtual machine size; re-IPL CMS; and restart APL2.
	• **TSO:** Logoff; logon with a larger region size specified in the LOGON command; and restart APL2.
1 33	No value has been assigned to the data variable for a write request.
1 42	You have set access control on the data variable and you have not yet referenced it for an earlier request. (The first read request resulted in a return code of 0 0.)
1 45	VSAM GENCB error.
1 48	Invalid sequence of service requests.
	• A key feedback (*KF*) request followed an open request or an error.
	• A relative-record write without a key (*W*) was requested without a preceding read-for-update request (*RU*).

Figure 19 (Page 1 of 2). Selected VSAM Return Codes Applicable to AP 123

Code	Description
0 8	More than one record has the specified key. The first is returned.
4 116	The file was not properly closed after its last use.
	Action: Use the access method services VERIFY command to check the status of the file.
8 4	End of file; or specified key is greater than the highest key on the file.
8 8	Duplicate key.

Code		Description
8	12	Key sequence check found an error.
8	16	Record not found.
8	20	Record is in use by another user.
8	28	Data set is full.
8	32	Invalid relative byte address.
8	40	Insufficient virtual storage.
8	88	Sequential read requested without prior positioning.
8	96	Attempt to change key of record.
8	100	Attempt to change length of record on an entry-sequenced or relative-record data set.
8	110	Attempt to open an empty file for read or update.
8	128	Attempt to open file that has not been properly allocated.
8	136	Insufficient virtual storage.
8	152	Password error.
8	168	The data set is not available in the mode requested. It is in use by another job.
8	192	Invalid relative record number.

AP 124 Return Codes

Figure 20 (Page 1 of 2). Return Codes Issued by AP 124

Code	Description
0	Normal return—operation is successful.
11	Control variable rank error.
12	Control variable length error.
13	Control variable domain error.
14	Invalid command.
15	Request to position cursor in an undefined field.
21	Data variable rank error.
22	Data variable length error.
23	Data variable domain error.
24	Data variable not shared.
30	Invalid field number.
32	Defined field extends beyond the screen.
33	Reference outside field definition.
37	Invalid field type.
38	Invalid field color or intensity.
41	Data variable has not been specified in the correct sequence.
42	Data variable has not been referenced in the correct sequence.
52	Device is not a 3270.

Figure 20 (Page 2 of 2). Return Codes Issued by AP 124

Code	Description
53	Required shared storage unavailable -- increase the shared storage size parameter when invoking APL2.
89	Unknown Shared variable return code.
91	Physical field table overflow.
92	Physical field table error, interrupt field not found.
94	Device not available.
95	Unexpected I/O error.
96	Chained CCW string not complete.
97	Bad 3270 orders in output data.
98	Full-Screen support not available.
99	Unknown 3270 device error.

AP 126 Return Codes

Figure 21 (Page 1 of 2). Return Codes Issued by AP 126

Code		Description
0	0	Normal return.
1	6	Copy destination unsupported.
1	7	Data variable is wrong length. **Action**: Supply data of correct length in the data variable, then respecify the control variable.
1	9	Copy destination full, or print limit exceeded.
1	11	$1 < \rho\rho CTL$ **Action**: Ravel the value in the control variable, and respecify it.
1	12	Syntax error in request. **Action**: Reconstruct the value in the control variable.
1	13	*DOMAIN ERROR* on the control variable; it contains character data or a mixed or nonsimple array.
1	14	Invalid request code in the control variable.
1	21	$1 < \rho\rho DAT$ **Action**: Ravel the value in the data variable.
1	22	Copy destination disabled or closed.
1	23	*DOMAIN ERROR* on data variable; it contains numeric data or a mixed or nonsimple array.
1	26	Copy destination not defined.
1	28	Copy destination enquiry failed, or enquiry lost due to sync point.
1	30	Authorization check for copy destination failed.
1	35	Copy destination I/O error.

Figure 21 (Page 2 of 2). Return Codes Issued by AP 126

Code	Description
1 41	Data variable not specified. **Action**: Assign the required data to the data variable and respecify the control variable.
1 53	Required storage is not available. **Action**: If requests have been catenated, reduce the number of requests in one specification or try to retract other shared variable pairs to free up storage.
1 54	Copy translate table unavailable.
1 56	Insufficient space in shared variable quota. **Action**: If requests have been catenated, reduce the number of requests in one specification.
1 60	GDDM not available.
1 61	Invalid parameter for service request.
1 62	Invalid count, code, or length value on request.
1 63	Hard-copy translate table not available.
1 65	GSCOPY attempted to an APL destination. **Action**: Use GDDM destination rather than an APL destination.
1 66	Hard-copy destination not available. If this occurs on a request issued after a successful open, the destination has been closed. **Action**: Issue an FSOPEN request for the destination.
1 67	A hard-copy destination already open. **Action**: Query the hard-copy destination (request code $^-9$); (if no name is returned, issue DSQUSE instead); issue FSCLS for that destination; issue FSOPEN for the new destination.
1 96	Invalid request. **Action:** Notify your system administrator because this code indicates an APL2 or system problem.
2 *nnn*	Abend occurred in APL2 outside of AP 126 and GDDM. *nnn* is the abend code.
ss nnn	A return code greater than 2 indicates a GDDM warning or error. *ss* is the severity of the error. *nnn* is the return code or abend code from GDDM.
4 *nnn*	Warning. *nnn* is the return code from GDDM.
8 *nnn*	Error. *nnn* is the return code from GDDM.
12 *nnn*	Severe error. *nnn* is the return code from GDDM. **Note:** Issue the GDDM call FSQERR, via AP 126, to retrieve the message text for abend codes 4, 8, and 12. See *Graphical Data Display Manager: Messages* and consult GDDM request made and message ADM0*nnn*.
16 *nnn*	Abend. *nnn* is the abend code issued by the failed GDDM module. **Note:** Refer to *Graphical Display Manager: Messages*. Then, consult the GDDM request made and MSG ADMO*nnn*, and issue the GDDM call FSQERR, via AP 126, to retrieve the message text.

AP 127 Return Codes

Figure 22. Return Codes Issued by AP 127

Code	Description
0 0 0 0 0	Normal return. All operations completed. Table retrieved by a *FETCH* request is complete.
0 1 0 *n msgn*	Normal return, but a warning message has been issued. For example, cursor is beyond end of table on a *FETCH* request, or *DELETE* statement deletes nothing. *n*=**1** Indicates a warning from AP 127. *n*=**2** Indicates a warning from SQL/DS or DB2.
0 0 1 0 0	Normal return, but a result table may not have been completely retrieved.
1 1 0 *n msgn*	Transaction backout. All changes made to tables since the last *COMMIT* or *ROLLBACK* have been discarded. Application must restore processing to point of last *COMMIT* or *ROLLBACK*.
1 0 0 1 *msgn*	Error in AP 127. *msgn* is the number of the AP 127 error message.
1 0 0 2 *msgn*	Error detected in DB2 or SQL/DS. *msgn* gives the value of DB2 or SQL/DS return code.
1 0 0 3 *msgn*	Error detected in an SQL workspace function. *msgn* gives the message number.

AP 210 Return Codes

Figure 23 (Page 1 of 2). Return Codes Issued by AP 210 under TSO Only

Code	Description
^-nn	Negative number: During formatting, the file was too small to contain the number of records requested. The value returned represents the number of records *not* formatted.
0	Successful completion of requested function. If returned for the initial offer, the *syntax* of the value was correct. It does not mean that the specified ddname is correctly allocated to a file.
1	Invalid initial value. **Action:** Retract the variable, specify a valid syntax for the initial value, and reoffer the variable.
12	End of file on a sequential read request. The value in the data variable is empty. **Action:** To continue reading the file, reposition the record pointer via the control variable. To close the file, retract the record variable.
15	Wrong length record on fixed-length output. All conversion options other than VAR or CDR require that all records on the file be exactly equal to the LRECL.
17	Record too large for file. With conversion option *CDR* or *VAR*, you have specified the record variable with a value greater than the LRECL of the file.

Figure 23 (Page 2 of 2). Return Codes Issued by AP 210 under TSO Only

Code	Description
30	Record not found.
	This error occurs when you try to read or write a record beyond the range of existing record numbers in the file.
440	Error in open for output.
	This error occurs under one of the following conditions:
	• You have not properly allocated the file.
	• You have allocated a file that cannot be processed by BDAM.
	• A system error occurred while trying to find the allocation information to open the file.
	• Your installation has prohibited your access to the file.
	• You requested formatting, but have not specified a valid number of records to format.
	• You specified read-only access (the **R** access option), and you have specified a value in the record variable, indicating output processing.
	Action: Verify your allocation, authorization, and initial values.
441	Error in open for input.
	This error occurs under one of the following conditions:
	• You have not properly allocated the file.
	• You have allocated a file that cannot be processed by BDAM.
	• A system error occurred while trying to find the allocation information to open the file.
	• Your installation has prohibited your access to the file.
	• You specified write-only access (the **W** access option), and you have referenced the record variable, requesting a read operation.
	Action: Verify your allocation, authorization, and initial values.
443	Error in writing a record. The value assigned to the shared variable exceeds 32760 bytes.
444	Invalid data specified. It is the wrong size, shape, or data type.
445	Insufficient shared storage (SHRSIZE) for input data on a read request.
	Action: Reinvoke APL2 and specify a larger value for the SHRSIZE invocation option.
	Note: If you are using conversion option VAR or CDR and you think your SHRSIZE value is properly set, verify that the record you are trying to read was written by AP 210, using the VAR or CDR option. Try reading the record using the BYTE conversion option.
901	I/O error while trying to read data.
	This return code is accompanied by detailed TSO messages that describe the error. It is issued in place of the system abend code 001 which would have been issued if the AP had not isolated and detected the error.
	Normally, this error means that you are trying to process the file with DCB attributes that do not match the physical data in the file.
	Action: Verify your allocation, initial values, and authorization.

AP 211 Return Codes

Figure 24. Return Codes Issued by AP 211

Code	Description
0	Success
2	File, data set, or DDNAME not found.
4	Attempt to write when open for read-only access.
7	Unsupported RECFM.
8	Insufficient storage for file services.
10	General file I/O error.
14	Unsupported file type.
15	Unsupported DSORG.
-2	Rank error.
-3	Length error.
-4	Type error.
-7	Invalid command.
-8	Invalid block size.
-9	Not an AP 211 file.
-10	No file accessed.
-11	Name has no value.
-12	Invalid object name.
-14	Invalid file name.
-15	Invalid access mode.
-16	Invalid userid.
-17	File already exists on $CREATE$.
-20	Unsupported data format.
-25	Unable to convert IEEE data to 370 format.
-26	Insufficient storage for AP 211 work areas.
-27	Abend. AP 211 has restarted. Variables are still shared, but all files have been released.

Chapter 7. Return Codes from APL2 Modules and EXECs

APL2 Module Return Codes under TSO

Figure 25. Return Codes Issued by APL2 Module under TSO

Code	Description
0	Successful completion.
4	All sources of input for APL2 are exhausted, but no)*OFF* or)*CONTINUE* system command was encountered. APL2 simulated)*CONTINUE* and saved your active workspace as *CONTINUE*.
8	An error occurred which prevented APL2 from initializing.

APL2 Module Return Codes under CMS

Figure 26. Return Codes Issued by APL2 Module under CMS

Code	Description
0	Successful completion.
4	All sources of input for APL2 are exhausted, but no)*OFF* or)*CONTINUE* system command was encountered. APL2 simulated)*CONTINUE* and saved your active workspace as *CONTINUE*.
8	An error occurred which prevented APL2 from initializing.
16	An error occurred after initialization that forced APL2 to abnormally terminate further processing. Your active workspace may have been saved as *CONTINUE*.
100	Only help text was produced.

AP2CSVP Module Return Codes under CMS

Figure 27. Return Codes Issued by AP2CSVP Module under CMS

Code	Description
0	Successful completion.
1	A command for the global SVP other than AP2CSVP START was issued, but the global SVP is not currently active.
2	The global SVP has already been started.
24	Invalid or missing operand.
100	Only help text was produced.
others	Other return codes are from CMS commands or macros.

AP2VMCF Module Return Codes under CMS

Figure 28. Return Codes Issued by AP2VMCF Module under CMS

Code	Description
0	Successful completion.
1	Server or user is not active.
24	Invalid or missing operand.
25	Insufficient storage is available.
100	Only help text was produced.

AP2VUTIL Module Return Codes under CMS

Figure 29. Return Codes Issued by AP2VUTIL Module under CMS

Code	Description
0	Successful completion.
1	For 'AP2VUTIL DIAG74', you are not authorized to issue diagnose 74. For 'AP2VUTIL CHECK SEGMENT', saved segment overlaps your virtual machine size.
2	For 'AP2VUTIL CHECK SEGMENT', saved segment fits within your virtual machine size.
24	Invalid or missing operand.
100	Only help text was produced.
others	Other return codes are from CMS commands or macros.

AP2XLOAD Module Return Codes under CMS

For return codes from the CMS command, NUCXLOAD, see the *CMS Command Reference*.

Figure 30 (Page 1 of 2). Return Codes Issued by AP2XLOAD Module under CMS

Code	Description
0	Successful completion.
1	The specified command already exists as a nucleus extension, and the PUSH option was not specified.
4	Module is marked "Not executable."
6	Module is marked "Only loadable."
8	Module was not found.
10	Module is an overlay structure. The module is not loaded. Overlay structures may not be used as nucleus extensions, because CMS does not support more than one such program at a time.
12	Error reading the module from disk.
13	NUCEXT function failed.
20	Invalid or missing operand.
24	Invalid or missing operand.

Figure 30 (Page 2 of 2). Return Codes Issued by AP2XLOAD Module under CMS

Code	Description
28	Open or BLDL for module failed.
32	Missing FILEDEF for DDNAME.
36	Module is larger than the saved segment.
41	Insufficient free storage.
42	Saved segment definition overlaps virtual machine size.
43	Saved segment could not be defined by DMSFREE storage.
44	Saved segment named is not defined.
64	RMODE of module = 24; not usable above 16MB.
100	Only help text was produced.
177	Paging I/O errors.

I5688228 or I5688229 EXEC Return Codes under CMS

Figure 31. Return Codes Issued by I5688228 or I5688229 EXEC under CMS

Code	Description
0	Product installed and required parts verified.
12	Installation failed for one of the following reasons: • Virtual machine size too small. • Tape error: unit not available, wrong tape mounted, or positioning error. • Required elements not available—AP2VUTIL module and AP2MAINT exec. • AP2MAINT exec failed, AP2MAINT return code given. • VMFPLC2 load or scan failed. • STATE failed for APL2 LOADLIB. • LINK to APL product disk failed (System Offering only). • Format of APL product disk failed (System Offering only). • Attempt to load contents of files 1 and 2 from installation tape to APL product disk failed (System Offering only). • 19E disk not linked in read/write mode (System Offering only).
24	Invalid or missing operand.
100	Only help text was produced.
111	You responded "no" or responded invalidly three times to the "Do you want to proceed..." prompt (System Offering only).
114	Incorrect password supplied for link to APL product disk (System Offering only).
222	The product files were loaded successfully, but an error occurred in the generation of the product.
333	The product files were loaded successfully, and the generation of the product was successful, but an error occurred in the verification of the product.
999	Product installed but verification failed.

V5688228 or V5688229 EXEC Return Codes under CMS

Figure 32. Return Codes Issued by V5688228 or V5688229 EXEC under CMS

Code	Description
0	Successful completion.
100	Only help text was produced.
999	Verification failed.
other	Verification failed.

5688228A or 5688229A EXEC Return Codes under CMS

Figure 33. Return Codes Issued by 5688228A or 5688229A EXEC under CMS

Code	Description
0	All load and build functions complete.
4	Load complete. Build not complete.
8	Load not complete but tape position known and correct (following last APL2 tape file).
12	Load not complete and tape position unknown.

AP2F3800 EXEC Return Codes under CMS

Figure 34. Return Codes Issued by AP2F3800 EXEC under CMS

Code	Description
0	Successful completion.
100	Parameters are required and none were included.
others	Are from the first parameter of this EXEC executed as a command.

AP2F3803 EXEC Return Codes under CMS

Figure 35. Return Codes Issued by AP2F3803 EXEC under CMS

Code	Description
0	Successful completion.
100	Parameters are required and none were included.
others	Are from the first parameter of this EXEC executed as a command.

AP2MAINT EXEC Return Codes under CMS

Figure 36 (Page 1 of 2). Return Codes Issued by AP2MAINT EXEC under CMS

Code	Description
0	Successful completion.

Figure 36 (Page 2 of 2). Return Codes Issued by AP2MAINT EXEC under CMS

Code	Description
8	An error occurred during AP2MAINT processing: • NAMES file unusable, file name given. • Level in local NAMES file does not match level in distributed NAMES file. • NAMEFIND command failed.
12	A required part is not available: • AP2VUTIL module required, not found. • Target disk not accessed. • Target disk not accessed in read/write mode.
100	Only help text has been produced.
others	Are from APL2 EXECs called by AP2MAINT.

AP2MCOPY EXEC Return Codes under CMS

Figure 37. Return Codes Issued by AP2MCOPY EXEC under CMS

Code	Description
0	Successful completion.
8	Invalid invocation argument.
28	COPYFILE command failed.
100	Help information produced.

AP2MLNK EXEC Return Codes under CMS

Figure 38 (Page 1 of 2). Return Codes Issued by AP2MLNK EXEC under CMS

Code	Description
0	Successful completion.
8	AP2MAINT processing failed for one of the following reasons: • NAMES file unusable, file name given. • Level in local NAMES file does not match level in distributed NAMES file. • NAMEFIND command failed. • EXECIO command failed. • Target disk not accessed. • Target disk not accessed in read/write mode. • File already exists; file name, file type, and file mode given. • User lacks privilege class required for SAVESYS. • Saved segment requested to be saved is outside the user's virtual machine. • Saved segment requested to be saved overlaps the user's virtual machine. • Saved segment requested to be saved is invalid or unknown. • SAVESYS command failed.
12	STATE command failed for AP2VUTIL module.

Figure 38 (Page 2 of 2). Return Codes Issued by AP2MLNK EXEC under CMS

Code	Description
24	Second invocation argument invalid.
28	Required text file not found, file name given.
44	Saved segment requested to be saved is not defined.
100	Only help text was produced.

AP2MP11L and AP2MP11M EXEC Return Codes

These execs normally issue a zero return code. On CMS, they issue a *20* return code if an object file specified in the argument cannot be found. If a command that they have issued fails, the execs display a message stating the command name and issue the command's return code.

AP2MV1AP EXEC Return Codes under CMS

Figure 39. Return Codes Issued by AP2MV1AP EXEC under CMS

Code	Description
0	Successful completion.
8	Nonzero RC from LKED command.
12	A required part is not available: • Target disk not accessed. • Target disk not accessed in read/write mode.
28	A text deck is missing, either the AP text deck or the AP2VASVP text deck.
100	Only help text was produced.

AP2MWS EXEC Return Codes under CMS

Figure 40. Return Codes Issued by AP2MWS EXEC under CMS

Code	Description
0	Successful completion.
28	A part of file type APLIMAGE is missing.
100	Only help text was produced.

AP2M3800 EXEC Return Codes under CMS

Figure 41. Return Codes Issued by AP2M3800 EXEC under CMS

Code	Description
0	Successful completion.
12	A required part is not available: • Target disk not accessed. • Target disk not accessed in read/write mode.
24	Invalid operand, or user ID not privileged for CMS IMAGELIB command.
28	A required file is missing.
100	Only help text was produced.

Bibliography

Figure 42 (Page 1 of 2). Related Publications		
Library	**Order Number**	**Publication Name**
DB2	SC26-4377 SC26-4378 LY27-9536 GC26-4373 SC26-4379 SX26-3771	*IBM DATABASE 2 Application Programming and SQL Guide* *IBM DATABASE 2 Command and Utility Reference* *IBM DATABASE 2 Diagnosis Guide and Reference* *IBM DATABASE 2 General Information* *IBM DATABASE 2 Messages and Codes* *IBM DATABASE 2 Reference Summary*
DFP	SC26-4500 SC26-4501 SC26-4506 SC26-4510 SC26-4517 SC26-4562 SC26-4564 SC26-4569 SC26-4570	*Integrated Catalog Administration: Access Method Services for MVS/DFP* *VSAM Catalog Administration: Access Method Services for MVS/DFP* *Data Administration: Macro Instruction Reference for MVS/DFP* *Linkage Editor and Loader for MVS/DFP* *MVS/ESA VSAM Administration: Macro Instruction Reference for MVS/DFP* *MVS/DFP: Access Method Services for the Integrated Catalog Facility* *MVS/DFP: Linkage Editor and Loader* *MVS/DFP: Macro Instructions for VSAM Data Sets* *MVS/DFP: Access Method Services for VSAM Catalogs*
GDDM	SC33-0148 GC33-0323 SC33-0325 SC33-0326 LC33-0104 SC33-0327 SC33-0328 SC33-0330 GC33-0319	*GDDM Application Programming Guide* *GDDM Installation and System Management for VM* *GDDM Messages* *GDDM Diagnosis and Problem Determination Guide* *GDDM-PGF Diagnosis Guide and Reference* *GDDM Image Symbol Editor* *GDDM-PGF Interactive Chart Utility* *GDDM-PGF Vector Symbol Editor* *GDDM General Information*
MVS	GC38-0229 GC38-1008 GC38-1002 GC26-3819 GC26-3838 GC26-3841 GC26-3873 GC28-0683 GC28-1029 GC28-0706 GC26-3830 GC28-1047 GC28-0725 GC28-0674	*Operator's Library: OS/VS2 MVS System Commands* *OS/VS Message Library: VS2 System Codes* *OS/VS Message Library: VS2 System Messages* *OS/VS Virtual Storage Access Method (VSAM) Options for Advanced Applications* *OS/VS Virtual Storage Access Method (VSAM) Programming Guide* *OS/VS2 Access Method Services* *OS/VS2 MVS Data Management Macro Instructions* *OS/VS2 MVS Supervisor Services and Macro Instructions* *OS/VS2 MVS System Programming Library: Initialization and Tuning Guide* *OS/VS2 System Management Facilities (SMF)* *OS/VS2 System Programming Library: Data Management* *OS/VS2 System Programming Library: Debugging Handbook, Vol. 1-3* *OS/VS2 System Programming Library: MVS Diagnostic Techniques* *OS/VS2 System Programming Library: Service Aids*
MVS/ESA	GC28-1635 GC28-1639 GC28-1656 GC28-1657 GC28-1658 GC28-1664 GC28-1666 GC28-1667 GC28-1669 GC28-1829	*MVS/ESA Initialization and Tuning Reference* *MVS/ESA Appl Dev: Callable Services for High-Level Languages* *MVS/ESA System Messages Volume 1* *MVS/ESA System Messages Volume 2* *MVS/ESA System Messages Volume 3* *MVS/ESA System Codes* *MVS/ESA Routing and Descriptor Codes* *MVS/ESA Problem Determination Guide* *MVS/ESA Service Aids* *MVS/ESA JCL Reference*

Library	Order Number	Publication Name
MVS/XA	GC26-4019 GC26-4075 GC26-4014 LC28-1164 LY28-1199 GC28-1157 GC28-1156 GC28-1206 GC28-1154 GC26-4010 GC28-1149 GC28-1159 GC28-1153 GC28-1158 GC26-4016	*MVS/Extended Architecture Access Method Services Reference for the Integrated Catalog Facility* *MVS/Extended Architecture Access Method Services Reference for VSAM Catalogs* *MVS/Extended Architecture Data Management Macro Instructions* *MVS/Extended Architecture Debugging Handbook, Vols. 1-5* *MVS/Extended Architecture Diagnostic Techniques* *MVS/Extended Architecture Message Library: System Codes* *MVS/Extended Architecture Message Library: System Messages* *MVS/Extended Architecture Operations: System Commands* *MVS/Extended Architecture Supervisor Services and Macro Instructions* *MVS/Extended Architecture System Programming Library: Data Management* *MVS/Extended Architecture System Programming Library: Initialization and Tuning* *MVS/Extended Architecture System Programming Library: Service Aids* *MVS/Extended Architecture System Programming Library: System Management Facilities* *MVS/Extended Architecture System Programming Library: 31-Bit Addressing* *MVS/Extended Architecture VSAM Reference*
SMP/E	GC28-1108 SC28-1107	*System Modification Program Extended (SMP/E): Messages and Codes* *System Modification Program Extended (SMP/E): Reference*
SQL/DS	SH09-8086 SH09-8079 SH09-8081 SH09-8088	*SQL/Data System Application Programming* *SQL/Data System Messages and Codes* *SQL/Data System Diagnosis Guide and Reference* *SQL/Data System Database Service Utility*
TCP/IP	SC31-6084 SC31-6081 SC09-1261 SC09-1255 SC31-6087 SC31-6142 SC31-6151	*TCP/IP Version 2.0 for VM: Programmer's Reference* *TCP/IP for VM: User's Guide* *TCP/IP Version 2.0 for MVS: Programmer's Reference* *TCP/IP for MVS: User's Guide* *TCP/IP for MVS: Programmer's Reference Manual* *TCP/IP for MVS Messages and Codes* *TCP/IP for VM Messages and Codes*
TSO	SC28-1134 SC28-1136 SC28-1307 SC28-1364 SC28-1875 SC28-1881	*MVS/XA TSO/E Command Language Reference* *MVS/XA TSO/E Guide to Writing a Terminal Monitor Program or a Command Processor* *TSO/E Command Language Reference* *TSO/E Programming Services* *TSO/E Version 2 Programming Services* *TSO/E Version 2 Command Reference*
VM/ESA	SC24-5460 SC24-5466 SC24-5460 SC24-5461 SC24-5433 SC24-5437 SC24-5519 SC24-5529	*VM/ESA CMS User's Guide* *VM/ESA Procedures Language VM/REXX Reference* *VM/ESA CMS User's Guide* *VM/ESA CMS Command Reference* *VM/ESA CP General User Command Reference for 370* *VM/ESA System Messages and Codes for 370* *VM/ESA CP Command and Utility Reference* *VM/ESA System Messages and Codes Reference*
VM/XA	SC23-0356 SC23-0358 SC23-0354 SC23-0374 SC23-0376	*VM/XA System Product CMS User's Guide* *VM/XA System Product CP Command Reference* *VM/XA System Product CMS Command Reference* *VM/XA System Product System Product Interpreter Reference* *VM/XA System Product System Messages and Codes Reference*
VM/SP	LY20-0891 LY20-0892 LY20-0893 LY20-0890 SC19-6203 SC19-6204 SC19-6209 SC19-6210	*VM/SP Data Areas and Control Block Logic* *VM/SP System Logic and Problem Determination Guide Volume 1 (CP)* *VM/SP System Logic and Problem Determination Guide Volume 2 (CMS)* *VM/SP Service Routines Program Logic* *VM/SP System Programmer's Guide* *VM/SP Messages and Codes* *VM/SP CMS Command and Macro Reference* *VM/SP CMS User's Guide*

Figure 42 (Page 2 of 2). Related Publications

Index

Numerics

left argument must be specified
  (AP2WVAPLFILE106)   94
LENGTH ERROR (AP2...253)   31
length specified is inconsistent with character data
  (AP2WFSM3)   85
length specified is inconsistent with numeric data
  (AP2WFSM4)   85
level level (AP2WIVP44)   79
level level invalid (AP2WIVP40)   78
level not verified: actual (AP2WIVP14)   77
LIB size1 FREE size2 (AP2...271)   33
library continue workspace has just been saved
  (AP2...871)   65
LIBRARY I/O ERROR (AP2...293)   36
LIBRARY NOT AVAILABLE (AP2...292)   36
library not empty (AP2...893)   68
library not found (AP2...853)   63
library number is full (AP2WFILESERV59)   82
library table not found or no library entries - fn ft
  (AP2...604)   51
line exceeds display size (AP2...418)   42
line not in log file (AP2...420)   42
line numbers exhausted (AP2...419)   42
loadlib will be ignored because of error (AP2...800)   58
local name conflicts with share name name
  (AP2WOFFER1)   81
local port server is unavailable, port: port
  (AP2...302)   39
log file full (AP2...421)   42
log size not available (AP2...432)   43
log size reduced (AP2...431)   43
lost share with AP number (AP2WFSM11)   85
lrecl of number not in valid range for APLTRACE
  (AP2...551)   47

M

main storage services rc=hexnum from request for
  number bytes (AP2...550)   46
maximum library number exceeded (AP2...874)   66
maximum number of files already open
  (AP2WAP12131)   72
message code must be a simple length 5 vector
  (AP2WSQL14)   90
message service abend (AP2...573)   48
message truncated (AP2...572)   48
more workspace data sets exist in this library than can
  be listed (AP2...855)   63
msp: parameter block (AP2...570)   48
multiple ownerships of this library exist (AP2...875)   66
must be more than number data points
  (AP2WGRAPHPAK028)   87
MVS
  publications   126
MVS/ESA publications

MVS/XA
  publications   126

N

name association with processor 11 not active
  (AP2WPRINTWS3)   89
name class (AP2...223)   28
name is already offered or is not a valid name
  (AP2WOFFER2)   81
name with no value (AP2...231)   29
namespace name modified since last access
  (AP2...026)   11
no active statements exist (AP2...126)   20
no array or operand (AP2...221)   28
no copy id specified (AP2...422)   42
no free space for I/O buffer, code 32
  (AP2WVSAMDATA32)   96
no more information (AP2...574)   49
no replacement past column number
  (AP2WMEDIT0)   88
no saved segments overlap for component
  (AP2WIVP68)   80
no searches have been made
   there is nothing in the window (AP2...061)   16
no shares (AP2...215)   26
no space for IO buffer (AP2WAP12132)   72
no value-list passed for PUT   23
non-display edit mode...name type (AP2...559)   47
nonzero return code - alternate input processor: rc
  (AP2WTSO14)   93
not a recognized rows value (AP2...119)   19
not a select statement - operation cannot be completed
  (AP2...136)   20
not a valid name (AP2...700)   58
not able to produce function FITFUN
  (AP2WGRAPHPAK006)   86
not an APL2 data file (AP2WAP12136)   72
not authorized or incorrect password
  (AP2WAP12134)   72
not character vector to VSAM AP, code 20
  (AP2WVSAMDATA20)   95
NOT COPIED: object-names (AP2...290)   36
not enough freespace (AP2...423)   42
not enough virtual storage to satisfy processor number
  request (AP2...047)   15
NOT ERASED: object-names (AP2...289)   35
NOT FOUND: object-names (AP2...288)   35
not in use (AP2WIVP25)   78
not in use, AP120 not verified (AP2WIVP32)   78
not open - operation cannot be completed
  (AP2...139)   21
not prepared - operation cannot be completed
  (AP2...137)   21
NOT SAVED, LIBRARY FULL (AP2...285)   35

T

U

V

IBM®