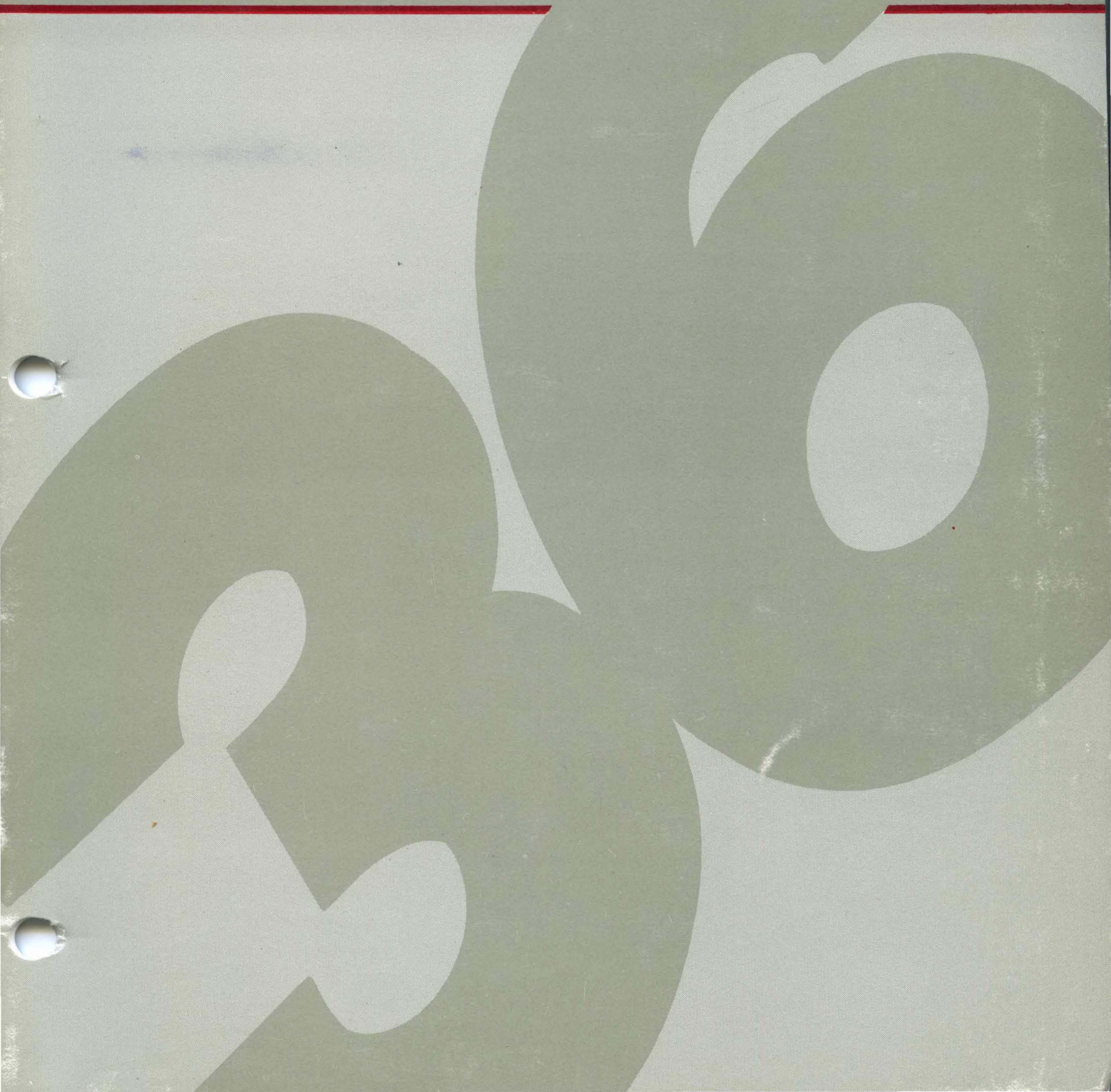


 System/36

# Service Aid Procedures



**IBM** System/36 PC

**Service Aid Procedures**

Program Number 5727-SS6

File Number  
S36-36

Order Number  
SC21-9494-1

---

**Second Edition (December 1986)**

This major revision makes obsolete SC21-9494-0. Changes or additions to the text and illustrations are indicated by a vertical line to the left of the change or addition.

Changes are periodically made to the information herein; any such changes will be reported in subsequent revisions or Technical Newsletters.

This edition applies to Release 5, Modification Level 0, of IBM System/36 System Support Program Product (Program 5727-SS6 for the 5364 System Unit), and to all subsequent releases and modifications until otherwise indicated. Also, this publication contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

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 licensed program in this publication is not intended to state or imply that only IBM's licensed program may be used. Any functionally equivalent program may be used instead.

The numbers at the bottom right of illustrations are publishing control numbers and are not part of the technical content of this manual.

Publications are not stocked at the address below. Requests for IBM publications should be made to your IBM representative or to your IBM-approved remarketer.

This publication could contain technical inaccuracies or typographical errors. A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Information Development, Department 245, Rochester, Minnesota, U.S.A. 55901. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

---

## Contents

<b>About This Manual</b> .....	<b>v</b>
<b>Chapter 1. Service Aid Procedures</b> .....	<b>1-1</b>
APAR Procedure .....	1-2
DFA Procedure .....	1-4
DUMP Procedure .....	1-5
ERAP Procedure .....	1-13
ICFDEBUG Procedure .....	1-14
PATCH Procedure .....	1-15
PROBLEM Procedure .....	1-16
PTF Procedure .....	1-17
SERVICE Procedure .....	1-26
SERVLOG Procedure .....	1-27
SETDUMP Procedure .....	1-28
TOLIBR Procedure .....	1-29
TRACE Procedure .....	1-30
<b>Chapter 2. Copy and List Library Members</b> ...	<b>2-1</b>
Copy Members from a Library .....	2-2
List Library Members and Information .....	2-3
<b>Index</b> .....	<b>X-1</b>





## | About This Manual

This manual contains information to use when your System/36 5364 System Unit requires software support, such as problem determination. This manual should be retained because it may be referenced by a software service representative.

Chapter 1 describes the service aid procedures. Chapter 2 describes two utility control statements that can be used to copy members from a library and to list library members.

You will not normally have to run these procedures; however, you may be asked to run some of these procedures to do problem determination and correction. For more information about problem determination, see the manual *IBM System Problem Determination-5364*, SC21-9375.

The procedures described in this manual apply specifically to the System/36 5364 System Unit. (Information in the *IBM System/36 System Reference*, SC21-9020, is for the general System/36 user, and therefore, some procedure parameters shown in that manual may be different.) Information about all of the system commands and parameters can be found in the *IBM System/36 System Reference*, which must be ordered separately. If this manual is not ordered, it is recommended that the optional SSP support for additional SSP help text be installed on your system. This help text can be installed by using the CNFIGSSP procedure, which is described in the manual *Changing Your System Configuration*, SC21-9052.

## About This Manual

---

## Chapter 1. Service Aid Procedures

This chapter describes the System/36 service aid procedures. You may need to run some of these procedures to do problem determination and correction. These procedures are listed in alphabetical order.

The following information is given for each procedure described in this chapter:

- The function of the procedure
- The syntax format of the procedure command that calls the procedure
- The descriptions of the parameters used in the procedure command
- Examples of how to use the procedure



## | APAR Procedure

The APAR procedure collects diagnostic information that helps software service representatives to correct programming problems that might occur in the system. The APAR procedure creates one or more diskette or tape files that contain the following:

- Control storage dump area.
- The input/output controller storage dump area.
- The system work area (if you are not running the APAR procedure during IPL after a system dump), including:
  - The system configuration
  - The disk volume table of contents (VTOC)
  - The #SYSWORK index
  - The trace work area
  - The security work area
  - The PTF work area
  - The diskette VTOC
  - The volume label
  - The IPL bootstrap
- IBM program product library and system library program temporary fix (PTF) logs.
- The system service log.
- The disk trace files. If you are not running the APAR procedure during IPL and you choose not to copy a task dump, a trace file prompt display is shown and you can select up to 16 trace files to copy.
- Microcode tables.
- Error logging (ERAP) tables.
- Task dump file (optional).
- The history file.
- The spool file (optional).
- The job queue (optional).
- The message file (optional).
- The product level data file.

Also, the APAR procedure can copy a specified load member to a diskette or tape file named APARLOAD, a specified source member to a diskette or tape file named APARSRCE, and a specified procedure member to a diskette or tape file named APARPROC. When the APAR procedure begins running, a display appears, and you can select the spool file, job queue, message file, and user file index to be copied to the diskette or tape.

Most of the data areas copied to diskette or tape by the APAR procedure can be displayed using the DUMP procedure, or the diskettes or tapes can be included with an Authorized Program Analysis Report (APAR).

The APAR procedure should be run during IPL after a system dump has been taken.

```

APAR      volume id, [load member name], [source member name],
           [procedure member name], [dump file name], , , [I1]
           0
    
```

S9020567-0

**volume id** specifies the volume ID of the one or more diskettes or tapes to receive the system data areas.

**load member name** specifies the load member containing the program that caused the program check to occur. The load member is placed in a diskette or tape file named APARLOAD.

**source member name** specifies the source member from which the program was created. The source member is placed in a diskette or tape file named APARSRCE.

**procedure member name** specifies the procedure member from which the program was called. The procedure member is placed in a diskette or tape file labeled APARPROC.

**dump file name** specifies the file created by a task dump. Zero (0) specifies that the most recent dump file is to be copied. If no file name is specified and the APAR procedure is being run from a display station, the status of all dump files is displayed, and you can select one (or none) of the files to copy. If no file name is specified and the APAR procedure is not being run from a display station, no dump file is copied to diskette or tape.

Dump files are named #DUMP.nn on disk, where nn is a number from 00 through 99.

**I1** specifies that the information collected by the APAR procedure is to be copied to diskette. If no parameter is specified, I1 is assumed.

**Example**

The following example shows how to enter the APAR procedure to copy the system data areas to one or more diskettes. A file named #DUMP.03 was created by a task dump. The diskettes have volume IDs of VOL001. The load member named PROGRAM is also to be copied.

```

APAR VOL001,PROGRAM, , , #DUMP.03
    
```

## DFA Procedure

The DFA (dump file analysis) procedure retrieves selected information from a dump file, formats the information, and either prints or displays it. The DFA procedure can also be used to format dump files copied to diskette or tape by the APAR procedure.

The DFA procedure can be run from any display station. If password security is active, the operator must have service aid authority to run the DFA procedure.

DFA	$\left[ \begin{array}{l} \text{PRINTER} \\ \text{printer id} \\ \text{CRT} \end{array} \right]$	,	$\left[ \begin{array}{l} \text{F1} \\ \text{I1} \end{array} \right]$	,	$\left[ \begin{array}{l} \text{file name} \\ 0 \end{array} \right]$
-----	---	---	--	---	---

S9020568-0

**PRINTER** specifies that the output should be printed on the session printer assigned to the display station running DFA. If no parameter is specified, PRINTER is assumed.

**printer id** specifies the work station ID of the printer that is to print the output.

**CRT** specifies that the output is to be displayed at the display station running the DFA procedure.

**F1** specifies that the input to the DFA procedure is a dump file on disk. If no parameter is specified, F1 is assumed.

**I1** specifies that the input to the DFA procedure is a dump file on diskette that was created by the APAR procedure, by a system dump, or by a task dump.

**file name** specifies the file to be processed by DFA. If F1 is specified, this file must have been created by a task dump, the DUMP procedure, or a system dump. Dump files are named #DUMP.nn on disk, where nn is a number from 00 through 99.

If I1 is specified, the file must be a diskette file created by the APAR procedure, by a system dump, or by a task dump. The DFA procedure first copies the APAR file to disk. If the file name specified already exists on the disk as a dump file, it is replaced. If the file name specified already exists on disk as a user file, a message is displayed, and you can either replace the file or cancel the procedure.

**0:** If F1 is specified, 0 (zero) indicates that DFA is to use the most recent dump file. If you place the DFA procedure on the job queue, 0 is assumed if no file name is specified.

If I1 is specified, 0 indicates that the APARFILE file should be copied from diskette to a disk file. The disk file is automatically deleted when DFA ends.

### Example

To format and print the information in the most recent disk dump file.

```
DFA , , 0
```

## DUMP Procedure

The DUMP procedure prints or displays any of the following areas from disk, or from a diskette or tape created by the APAR procedure:

- Task dump file
- System dump file
- Control storage dump file
- Input/output controller storage dump file
- Disk trace file
- Library PTF log
- Disk VTOC (volume table of contents)
- Spool file
- Task work area
- Input job queue
- Microcode tables
- System service log
- Product level data file

DUMP can also print or display selected sectors from the disk or diskette, and print selected blocks from a tape or tape cartridge.

The DUMP procedure can be run from any display station. If password security is active, some functions of the DUMP procedure are restricted to operators who have service aid authority. For more information about service aid authority, see the *IBM System/36 System Security Guide*, SC21-9042.

```

DUMP      [ MAIN          ] [ PRINTER ] , [ F1 ] , [ file name ]
          [ CONTROL      ] [ CRT      ] , [ I1 ] , [ library name ]
          [ DISK         ] [ printer id ] [ TC  ] , [ 0      ]
          [ IOC          ]
          [ JOBQ         ]
          [ MCODE        ]
          [ PLD          ]
          [ PTF          ]
          [ SERVLOG      ]
          [ SPOOL        ]
          [ STATUS       ]
          [ TAPE         ]
          [ TRACE        ]
          [ TWA          ]
          [ VTOC         ]
    
```

S9020569-0

# DUMP

---

**MAIN** specifies that selected portions of a task or system dump file are to be listed. If password security is active, the operator must have service aid authority to use **DUMP MAIN**.

If the information is displayed, the DUMP procedure displays the following:

- A storage dump summary
- The first segment of main storage
- The address of the task control block and request block for the abnormally ended task
- The last set of saved register values for the abnormally ended task

You can use the roll keys to page forward and backward through the data. You can also specify a different storage address or storage type. The possible storage types are:

- M Untranslated main storage
- V Untranslated virtual storage
- X Translated task storage
- C Control storage tables
- F Data storage controller tables

If the information is printed, the DUMP procedure first displays and prints the storage dump summary. You can then select one of the following options:

- M Untranslated main storage between limits
- V Untranslated virtual storage between limits
- X Translated task storage between limits
- C Control storage tables
- F Data storage controller tables
- N Nucleus storage dump
- P Formatted control block dump, where frequently used storage areas are printed including the control storage tables

**S** System dump, which includes the following:

- A formatted control block dump
- A nucleus storage dump
- A dump of the translated task storage for each program of each task included in the dump file
- A dump of the virtual storage associated with the task in error
- A trace table dump
- A DFA listing

**T** All existing main and alternate trace tables and files

To list the translated storage of another task, change the address of the task block shown on the display. A system dump contains the translated storage for all tasks that were active at the time of the abnormal ending, but a task dump contains only the storage for selected tasks.

To list the translated storage of another program for this task, change the address of the request block shown on the display.

To list virtual storage, change the address of the storage block shown on the display.

The control storage data areas contain the following:

- Direct area 0 (hex 0000 through hex 007F)
- Direct area 1 (hex 1000 through hex 107F)
- Direct area 2 (hex 2000 through hex 207F)
- Direct area 3 (hex 3000 through hex 307F)
- Direct area 4 (hex 4000 through hex 407F)
- Direct area 5 (hex 5000 through hex 507F)
- Direct area 6 (hex 6000 through hex 607F)
- Direct area 7 (hex 7000 through hex 707F)
- Machine check logout area (hex 7A00 through hex 7B7F)
- Other miscellaneous areas

If DUMP MAIN is started using the EVOKE OCL statement or is run from the job queue, a system dump (S) will be printed.

# DUMP

---

**CONTROL** specifies that selected portions of a control storage dump file are to be listed. If password security is active, the operator must have service aid authority to use **DUMP CONTROL**.

If the information is displayed, the **DUMP** procedure displays the first segment of control storage. You can use the roll keys to page forward and backward through the data. You can also specify a different control storage address.

If the information is printed, the **DUMP** procedure first prompts for the control storage limits.

If **DUMP CONTROL** is started using the **EVOKE OCL** statement or is run from the job queue, all of control storage will be printed.

**DISK** specifies that selected disk or diskette sectors are to be listed. You are prompted for the address of the first sector to be displayed. If the information is printed, you are prompted for the number of sectors to be printed. If password security is active, the operator must have service aid authority to use **DUMP DISK**.

**IOC** specifies that selected portions of the input/output controller storage dump file are to be listed. If password security is active, the operator must have service aid authority to use **DUMP IOC**. If the information is to be displayed, the **DUMP** procedure displays the following:

- The first segment of controller storage for the first device found in the dump file
- The address of the first device found in the dump file

To display other portions of the dump file, you can use the roll keys to page forward and backward through the data for this device. You can use command keys 1 and 2 to page forward and backward through the devices in the dump file. You can also specify a different device address.

If the information is printed, the **DUMP** procedure prompts for the address of the device to be listed. If you enter 00, the data for all devices in the dump file are listed.

For a list of the possible device addresses, press command key 8.

If **DUMP IOC** is started using the **EVOKE OCL** statement or is run from the job queue, all the data for all the devices in the dump file is listed.

**JOBQ** specifies that the job queue is to be listed. If password security is active, the operator must have service aid authority to use **DUMP JOBQ**. If the information is to be displayed, the **DUMP** procedure displays the following:

- The first segment of the job queue.
- If F1 is specified and no file name is specified, the disk sequential sector address for the first segment is displayed.
- If I1 or a file name is specified, the address relative to the start of the job queue is displayed.

To display other portions of the job queue file, you can use the roll keys to page forward and backward through the data for this device.

If the information is printed, the **DUMP** procedure prints the entire job queue file.



**MCODE** specifies that the following microcode tables are to be listed:

- Prerequisite list (including hardware required, SSP release and modification levels, and the last SSP PTF applied)
- Microcode level table (including microcode release and modification levels)
- Patch table (including module identifiers and dates of all patches)

If password security is active, the operator must have service aid authority to use **DUMP MCODE**.

If the information is printed, the **DUMP** procedure prints all the microcode tables.

**PLD** specifies that the product level data file is to be listed or displayed.

**PTF** specifies that a library PTF log, which identifies all PTFs applied to a library, is to be listed. If no library name is specified, the system library is assumed. If the information is to be displayed, the **DUMP** procedure displays the first segment of the PTF log. To display other portions of the PTF log, you can use the roll keys to page forward and backward through the data.

If I1 or a file name (not a library name) is specified, you can use command keys 1 and 2 to page forward and backward through other PTF logs.

If the information is printed, the **DUMP** procedure prints the entire PTF log.

**SERVLOG** specifies that the system service log is to be listed. If the information is to be displayed, the **DUMP** procedure displays the newest entry in the service log. To display other entries in the service log, you can use the roll keys to page forward and backward through the data.

If the information is printed, the **DUMP** procedure prints the entire service log, beginning with the oldest entry.

**SPOOL** specifies that the one or more spool file extents are to be listed. If password security is active, the operator must have service aid authority to use **DUMP SPOOL**. If the information is to be displayed, the **DUMP** procedure displays the following:

- The first segment of the first spool file extent.
- If F1 is specified and a file name is not specified, the disk sequential sector address of the first segment is displayed.
- If I1 or a file is specified, the address relative to the start of the first spool file is displayed.

To display other portions of the spool file, you can use the roll keys to page forward and backward through the data. You can also use command keys 1 and 2 to page forward and backward through other spool file extents.

If the information is printed, the **DUMP** procedure prints all the spool file extents.

# DUMP

---

**STATUS** specifies that the status (a storage dump summary) of one or more system or task dump files is to be listed. If the information is to be displayed, the **DUMP** procedure displays the status for the most recent task dump file. To display the status of other task dump files, you can use the Roll Up (↑) key to page forward through the data. You can use the Roll Down (↓) key to page backward through the data.

If the information is printed, the **DUMP** procedure prints the status for all the task dump files, beginning with the most recent task dump file.

**TAPE** specifies that selected tape blocks are to be listed. You are prompted for the number of blocks to print. If password security is active, the operator must have service aid authority to use **DUMP TAPE**.

**TRACE** specifies that the disk trace file is to be listed. If the information is from a task dump file, the **DUMP** procedure prompts for the specified trace file to be listed. You must select one of the trace files to continue.

If the information is to be displayed, the **DUMP** procedure displays the trace file starting with the most recent entry. To display other portions of the trace file, you can use the roll keys to page forward and backward through the data.

If the information is printed, the **DUMP** procedure prints the trace file, beginning with the oldest entry.

**TWA** specifies that the task work area is to be listed. If password security is active, the operator must have service aid authority to use **DUMP TWA**. If the information is to be displayed, the **DUMP** procedure displays the following:

- The first segment of the task work area.
- If F1 is specified and a file name is not specified, the disk sequential sector address of the first segment is displayed.
- If I1 or a file name is specified, the address relative to the start of the task work area is displayed.

To display other portions of the task work area, you can use the roll keys to page forward and backward through the data. You can use command key 1 to page forward through other task work area extents. You can use command key 2 to return to the first task work area extent.

If the information is printed, the **DUMP** procedure prints the entire task work area.

**VTOC** specifies that the disk VTOC is to be listed. If the information is to be displayed, the **DUMP** procedure displays the following:

- The first segment of the disk VTOC.
- If **F1** is specified and a file name is not specified, the disk sequential sector address of the first segment is displayed.
- If **I1** or a file name is specified, the address relative to the start of the disk VTOC is displayed.

To display other portions of the disk VTOC, you can use the roll keys to page forward and backward through the data.

If the information is printed, the **DUMP** procedure prints the entire disk VTOC.

**PRINTER** specifies that output is to be printed on the session printer assigned to the display station. If the parameter is not specified, **PRINTER** is assumed.

**CRT** specifies that the output is to be displayed at the display station running the **DUMP** procedure. This parameter is not allowed if **TAPE** is specified.

**F1** specifies that the **DUMP** procedure is to process the dump information from a disk file. If the parameter is not specified, **F1** is assumed.

**I1** specifies that the **DUMP** procedure is to process the dump information from a diskette created by the **APAR** procedure, by a system dump, or by a task dump. If **DISK** was selected for the first parameter, then information from specified diskette sectors is listed for any diskette.

**TC** specifies that the **DUMP** procedure is to print the dump information from a tape in the tape cartridge drive. **TC** is valid only if **TAPE** is specified as the first parameter.

# DUMP

**file name:** If F1 is specified, this parameter specifies the file to be used by the DUMP procedure to display or print information. The file must have been created by a task dump or by the TRACE procedure. Dump files are named #DUMP.nn on disk, where nn is a number from 00 through 99. If no file name is specified, the DUMP procedure uses the following table to obtain information to be listed:

Parameter	Input Assumed if No File Name Specified
MAIN	If started by EVOKE OCL statement or run from job queue, the most recent task dump file is assumed; otherwise, the status of all dump files is displayed. You must select one of the dump files to continue.
CONTROL	The control storage library.
JOBQ	The system job queue.
MCODE	The control storage library.
PTF	The system library PTF log.
SERVLOG	The system service log.
SPOOL	The system spool files.
TRACE	If started by EVOKE statement or run from job queue, the system trace file is assumed. If DUMP procedure is run from the keyboard, the status of all task dump files that contain trace files, followed by a list of all trace files within the selected task dump file, is displayed. You must select one of the files to continue.
TWA	The system task work area.
VTOC	The system disk VTOC.
PLD	The system product level data file.

If I1 is specified, the DUMP procedure creates a resident disk file named **file name**, and copies the APARFILE file from diskette. If a disk dump file exists with the specified file name, that dump file is removed and the new dump file is created. If the existing disk file is not a dump file, a message is displayed and you can either replace the disk file with the new dump file or cancel the DUMP procedure.

**library name** specifies the library whose PTF information is to be listed. PTF must be specified for parameter 1. If I1 is specified, the PTF library information for the system library is listed from the diskette file. If no library name is specified, the system library is assumed.

**0:** If F1 is specified, 0 (zero) indicates that DUMP MAIN, DUMP STATUS, or DUMP TRACE is to use the most recent dump file. If no parameter is specified and the DUMP procedure is placed on the job queue, 0 is assumed.

If F1 is specified for DUMP MAIN and this parameter is omitted, you are shown the status of each task dump file on the system; you then choose the one to process.

## Example

This example shows how to print the dump information about main storage. The information is contained in a disk dump file named #DUMP.03, which was created because of a task dump.

```
DUMP , , , #DUMP.03
```

## ERAP Procedure

The ERAP procedure displays or prints data that was logged for the devices on the system. Depending on the device, the logged data is contained in one or more of the following tables:

- An input/output counter table that contains accumulated statistics reflecting the amount of activity for the device. (For example, the number of verify, write, read or scan read, and nonzero seek operations on a disk drive.)
- An error counter table that contains accumulated totals of specific types of errors for the device.
- An error history table that contains a data field, a time field, a system reference code (SRC) representing an error on the device, and status bytes that provide more detail about the error. The first entry in the table represents the most recent entry. After the table is filled, the oldest entry is dropped from the table each time a new entry is added.

Besides printing or displaying the logged information, the ERAP procedure allows you to reset the information in the input/output counter tables and the error counter tables.

When the ERAP procedure begins, it displays a series of menus from which you can select:

- The reports, device, or devices, for which logged information is to be displayed or printed
- The logged information to be displayed or printed
- The printer where the output is to be printed (if printing is selected)
- The time period for which logged information is to be printed or displayed

ERAP

S9020456-0

The ERAP procedure has no parameters.

### Example

This example shows how to start the ERAP procedure.

ERAP

## ICFDEBUG Procedure

The ICFDEBUG procedure controls the running of the Interactive Communications Feature (SSP-ICF) debug program. After the debug file is displayed or printed, the file is deleted.

For more information about SSP-ICF and the ICFDEBUG procedure, see the manual *Using System/36 Communications*, SC21-9082.

```
ICFDEBUG { ON
          OFF
          CRT
          CRT,job name
          PRINT
          PRINT,job name }
```

S9020134-0

**ON** specifies that the debug program is to be started. The program begins recording SSP-ICF activities in a disk file. Any previous file created by the ICFDEBUG procedure is deleted.

**OFF** specifies that the debug program is to be stopped. No further SSP-ICF activity is to be recorded and the disk file created by the ICFDEBUG procedure is deleted.

**CRT** specifies that the file previously created by the ICFDEBUG procedure is to be displayed.

**CRT,job name** specifies that entries for the named job are to be displayed from the file previously created by the ICFDEBUG procedure.

**job name** specifies the 8-character name of the job. You can use the STATUS USERS control command to display the job names.

**PRINT** specifies that the file previously created by the ICFDEBUG procedure is to be printed.

**PRINT,job name** specifies that entries for the named job are to be printed from the file previously created by the ICFDEBUG procedure.

**job name** specifies the 8-character name of the job. You can use the STATUS USERS control command to display the job names.

### Example 1

This example shows how to start the SSP-ICF debug program.

```
ICFDEBUG ON
```

### Example 2

This example shows how to print the activity recorded after the procedure command ICFDEBUG ON was entered.

```
ICFDEBUG PRINT
```

## PATCH Procedure

The PATCH procedure displays selected disk or diskette sectors and allows you to modify the data in those sectors.

The PATCH procedure prompts you for the sector number of the sector you want to display. The sector number can be specified in hexadecimal or decimal. When you display diskette sectors, you can also specify the cylinder number, head number, and record value in hexadecimal or decimal format. The first 256 bytes of the selected sector are then displayed along with the sector address. Other portions of the disk or diskette can be displayed by entering a new sector address or by using the function keys to scroll the disk or diskette storage data. You can modify disk or diskette information by replacing data displayed on the display with new data.

*Note: You should proceed with caution when using the PATCH procedure because it allows you to modify several critical system data areas.*

Because PATCH allows you to change disk data, PATCH can be run only from the system console if password security is not active. If password security is active, the procedure can be run from any display station, but running the PATCH procedure is restricted to operators who have service aid authority. For more information about service aid authority, see the *System Security Guide*.

<pre>PATCH  [ F1 ]         [ I1 ]</pre>
---

S9020457-0

**F1** specifies that the disk is to be used. If a parameter is not specified, F1 is assumed.

**I1** specifies that a diskette is to be used.

### Example

This example shows how to use the PATCH procedure to modify one or more diskette sectors.

```
PATCH I1
```



# PROBLEM

---

## PROBLEM Procedure

The **PROBLEM** procedure starts the online problem determination (OLPD) procedures. The **PROBLEM** procedure should be run from the physical system console at the system location. If the system console has been assigned to another display station, you should use the **CONSOLE GIVE** and **CONSOLE TAKE** commands to restore the physical system console.

OLPD may ask the operator to perform some tasks during the course of problem determination. The tasks may be mounting a tape or diskette, putting a modem in self test, or placing a wrap connector on a cable. Help is provided on the OLPD screens and is intended to be used by the person performing the tasks. The help contains graphics that are often not displayable on a Distributed Host Command Facility (DHCF) or other 3270 device.

PROBLEM

S9020186-0

The **PROBLEM** procedure has no parameters.

### Example

This example shows how to start the online problem determination procedures.

PROBLEM

## PTF Procedure

The PTF procedure allows you to apply program temporary fixes (PTFs) to a specified library. The PTF procedure allows you to do the following:

- Copy PTFs from a PTF diskette to a PTF library.
- Apply PTFs from a PTF library to a specified library.
- Apply PTFs directly from a PTF diskette to a specified library.
- List the PTF log for a specified library.
- Remove a PTF from a specified library.
- Save a PTF backup library on diskette.
- Restore a PTF backup library from diskette.
- Delete a PTF backup library from diskette or disk.
- Patch a library member on disk.
- Copy a microcode PTF to a PTF library.
- Apply a microcode PTF from a PTF library to the control storage library.
- Remove a microcode PTF.
- Restore the PTFNEWS library from diskette.
- Delete the PTFNEWS library from disk.
- Print or display one or more PTF newsletters in the PTFNEWS library.

PTF diskettes must be processed in ascending order by volume number. For example, volume 01 must be inserted first, followed by volume 02, until the last diskette is processed.

The suggested steps to be followed when applying a PTF to your system are:

1. PTF COPY
2. PTF APPLY
3. PTF SAVE
4. PTF DELETE

*Note: PTF APPLY requires a dedicated system with no tasks active.*

For further information about the PTF procedure, see the manual *Operating Your System*, SC21-9453.

# PTF

To copy PTFs for the SSP or IBM-supplied program products to a disk PTF library:

```
PTF      COPY, [ OLD  
              ALL  
              ptf log number ] , [ ptf type  
              ALLPTF  
              ptf file name ] , [ ptf library name ] ,  
  
        [ CHECK  
          NOCHECK ] , , , [ work library size ] , [ directory size ]
```

S9020570-0

To apply PTFs from a PTF library to the SSP or a program product library:

```
PTF      APPLY, , [ ptf type  
                 ALLPTF  
                 ptf library name ] , [ target library name ]
```

S9020459-0

To directly apply PTFs from a PTF diskette to a specified library:

```
PTF      DIRECT, [ OLD  
                 ALL  
                 ptf log number ] , [ ptf type  
                 ALLPTF  
                 ptf file name ] , [ target library name ]
```

S9020571-0

| *Note: The DIRECT parameter on the PTF procedure does not provide a backup of the programs being changed;*  
| *therefore, any PTFs installed using this parameter cannot be removed using PTF REMOVE.*

| *If you remove PTFs that were installed using PTF DIRECT, reinstall the system microcode, the SSP, and*  
| *all the installed program products.*

To list PTF information about a library:

```
PTF      LIST, target library name, [ PRINTER
                                     CRT
                                     printer id ]
```

S9020461-0

To remove a PTF from the SSP or a program product library:

```
PTF      REMOVE, [ ALL
                  ptf log number ] , [ ptf type
                                       ALLPTF
                                       target library name ] , [ backup library name ]
```

S9020462-0

To save a PTF backup library on diskette:

```
PTF      SAVE, [ ptf type
                 ALLPTF
                 backup library name ] , volume id
```

S9020572-0

To restore a PTF backup library from diskette:

```
PTF      RESTORE, [ ptf type
                   ALLPTF
                   backup library name ] , [ A1
                                             A2
                                             A3
                                             A4 ]
```

S9020573-0

# PTF

To delete a PTF backup library from disk or diskette:

```
PTF      DELETE, [ ptf type  
                  ALLPTF  
                  backup library name ], [ F1  
                                           I1 ]
```

S9020574-0

To patch a library member on disk:

```
PTF      PATCH
```

S9020466-0

To copy microcode PTFs to a PTF library:

```
PTF MCOPY, [ CSPTFLIB  
             PTF library name ], [ PTF library size ], [ directory size ]
```

S9020575-0

To apply a microcode PTF from a PTF library to the control storage library:

```
PTF MAPPLY, [ ALL  
             PTF log number ], [ CSPTFLIB  
                                 PTF library name ]
```

S9020468-0

To remove a microcode PTF:

```
PTF MREMOVE, [ PTF log number ], [ CSPTFLIB  
                                  PTF library name ]
```

S9020469-0

| To process newsletters:

PTF	NEWS,	[	ALL	,	[	DISPLAY
			RESTORE			D
			DELETE			PRINT
			PTFINDEX			P
			PTFXREF			
			BULLETIN			
			name			
			]			]

S9020607-0

**COPY** specifies that PTFs are to be copied from a PTF diskette to a PTF library. The COPY function creates a PTF library. This library contains the PTFs that are to be applied to the specified library, for SS1 for example, the library would be the system library, #LIBRARY. The name of the PTF library is PTFxxxxx, where xxxxx is the first 5 characters of the diskette file name; for example, for SS101, the name would be PTFSS101.

**APPLY** specifies that PTFs are to be applied from a PTF library to a specified library. The PTF library must have been created using the COPY function. The modules receiving PTFs are copied to a PTF backup library. The backup library contain the modules that are being replaced by PTFs; that is, the backup library contains down-level modules. The PTF modules are copied to the specified library; for example, for SS101, the system library.

The down-level modules remain in the PTF backup library until a new set of PTFs are applied to the specified library. The PTF backup library is named PTBxxxxx, where xxxxx is the last 5 characters of the PTF library name; for example, for PTF library PTFSS101, the backup library is named PTBSS101.

| If there is not enough room in the target library for the modules to be applied, the CONDENSE procedure is run for the target library before the modules are moved. #LIBRARY is always condensed, even if there is enough room.

**DIRECT** specifies that PTFs are to be applied directly from a PTF diskette to a specified library.

**LIST** specifies that the PTF log (#PTFLOG) for a library is to be printed or displayed.

**REMOVE** specifies that PTFs are to be removed from a specified library. The down level modules are copied from the PTF backup library to the specified library, replacing the PTF modules.

**SAVE** specifies that a PTF backup library is to be saved on diskette.

| **RESTORE** specifies that a PTF backup library or PTFNEWS library is to be restored from diskette.

| **DELETE** specifies that a PTF backup library or PTFNEWS library is to be deleted from disk or diskette.

**PATCH** specifies that a library or library member can be directly modified using the display station. If password security is active, PTF PATCH can be run from any display station, but only by an operator that has service aid authority.

**MCOPY** specifies that the microcode PTFs are to be copied from the PTF diskette to the microcode PTF library (CSPTFLIB).

# PTF

---

**MAPPLY** specifies that the microcode PTFs are to be applied from the PTF library (CSPTFLIB) to the control storage library.

**MREMOVE** specifies that the microcode PTF is to be removed from the control storage library. The removed PTF is in the PTF library (CSPTFLIB).

| **NEWS** specifies that the PTFNEWS library is to be processed.

**OLD** specifies that PTFs are only to be processed for existing library members. If no parameter is specified, OLD is assumed.

**ALL** specifies that all PTFs are to be processed.

**ptf log number** specifies that only the PTF corresponding to the number entered is to be processed. This number specifies the PTF log number, which is indicated in the source member named PTFXREF on each diskette.

**ALLPTF** specifies that all PTFs that are to be used with the current system configuration are to be processed. For example, if your system had the SSP, the Utilities, RPG, COBOL, and BASIC, only those PTFs for those licensed programs would be processed. Any other PTFs, such as FORTRAN, would not be processed.

**ptf type** specifies the type of PTF to be processed. The following table lists the PTF types with the associated library and a description of the library. **nn** is the SSP release level. For example, to process PTFs for release 2 of the SSP, you would enter SS102.



PTF Type	Library	Description
SS1nn	#LIBRARY	System Support Program
AP1nn	#APFLIB	Advanced Printer Function
AS1nn	#ASMLIB	Assembler Program Product
BA1nn	#BLLIB	BASIC Program Product
BH1nn	#BLHPLIB	BASIC Help Support
BNHnn	#BGUHLIB	Business Graphics Help Support
BNWnn	#BGULIB	Business Graphics Utility
BRJnn	#POPLIB	Programmer & Operator Productivity Aid (POP)
CB1nn	#COBLIB	COBOL Program Product
DF1nn	#DFULIB	Data File Utility (DFU)
DS1nn	#DSULIB	Development Support Utility (DSU)
EP1nn	#LIBRARY	3278 Emulation via IBM Personal Computer
FO1nn	#FORTLIB	FORTRAN Program Product
IG1nn	#CGULIB	Ideographic Support
IS1nn	#SRTXLIB	Ideographic Sort Utility
IW1nn	#IWLIB	PC Support/36
LC1nn	#LANLIB	Local Area Network (LAN) Feature
MA1nn	#MIGRLIB	S/34 to S/36 Migration Aid
OLPnn	#ONLPD	Online Problem Determination
QU1nn	#QRYLIB	Query/36
RG1nn	#RPGLIB	RPG II Program Product
SD1nn	#SDALIB	Screen Design Aid (SDA)
SE1nn	#SEULIB	Source Entry Utility (SEU)
TX1nn	#TMSLIB	Text & Office Management Systems (TMS & OMS)
TXSnn	#LEXLIB	Text Editor Languages
WP1nn	#TULIB	DisplayWrite/36
WP3nn	#OFCLIB	Personal Services/36
WS1nn	#WSULIB	Work Station Utility (WSU)

# PTF

---

When APPLY and DIRECT are specified, PTFs for SS1nn or ALLPTF can only be processed when:

- No other jobs are being run
- No other users are signed on at any other display stations
- Remote work stations are offline
- Interactive communication feature (SSP-ICF) sessions are disabled
- Communications lines are offline

**ptf file name** specifies the name of the file on the PTF diskette that contains the PTFs to be copied and applied to the system.

**ptf library name** specifies the name of the library that is to contain the PTFs. This library name is automatically determined by the type of PTFs that are being copied or applied. However, entering a value for this parameter overrides the automatic naming process.

**target library name** specifies the name of the library to which PTFs will be applied. This library name is automatically determined by the type of PTFs that are being copied or applied. However, entering a value for this parameter overrides the automatic naming process.

**backup library name** specifies the name of the library that will contain copies of each module that PTFs were applied to.

| **PTFINDEX** specifies that an index is displayed or printed.

| **PTFXREF** specifies that a PTF cross-reference listing is displayed or printed.

| **BULLETIN** specifies that a bulletin and a special bulletin in the PTFNEWS library is displayed or printed.

| **name** specifies that the named PTF description is displayed or printed. The **name** parameter must be in the form of PTFxxxxx, where xxxxx is a PTF log number.

| **DISPLAY** or **D** specifies that the information is displayed on the CRT.

| **PRINT** or **P** specifies that the information is printed.

**CHECK** specifies that prerequisite and release level checking is to be done as the diskette file is being copied to the PTF library on disk. If a PTF being copied has a prerequisite PTF that is not already on the system, it must also be copied into the PTF library, before applying the PTFs to the specified library.

**NOCHECK** specifies that no prerequisite or release level checking is to be performed.

**A1** specifies that the preferred disk file location is on the first disk. If space is available, the PTF backup library is placed on the first disk; otherwise, the library is placed on the second disk.

**A2** specifies that the preferred disk file location is on the second disk. If space is available, the PTF backup library is placed on the second disk; otherwise, the library is placed on A1 if you have two or four disk drives, or on A3 if you have three disk drives.

**A3** specifies that the preferred disk file location is on the third disk. If space is available, the PTF backup library is placed on the third disk; otherwise, the library is placed on A2 if you have three disk drives, or on A4 if you have four disk drives.

**A4** specifies that the preferred disk file location is on the fourth disk. If space is available, the PTF backup library is placed on the fourth disk; otherwise, the library is placed on A3.

**work library size** specifies the size (in blocks) of the temporary library that is used in the COPY process. If no parameter is specified, the necessary value is calculated. The calculated value will be at least 50 blocks.

**directory size** specifies the size (in sectors) of the directory for the temporary library. If no parameter is specified, 2 sectors will be allocated for every block.

#### **Example 1**

This example shows how to apply PTFs for release 1 of the SSP. The PTF COPY copies all the PTFs from the PTF diskette to a PTF library on disk; the library will be named PTFSS101. The PTF APPLY actually applies the PTFs to the SSP. The APPLY function also copies SSP members that are receiving PTFs to a PTF backup library, named PTBSS101.

```
PTF COPY,ALL,SS101
PTF APPLY,,SS101
```

#### **Example 2**

This example shows how to print the PTFs that have been applied to the system library.

```
PTF LIST,#LIBRARY
```

# SERVICE

---

## SERVICE Procedure

The SERVICE procedure displays a menu that allows you to do the various tasks you may be asked to do for problem determination and correction. These tasks include:

- Running the service aid procedures for determining program related problems (such as: DUMP, APAR, DFA, and TRACE)
- Running the service aid procedures for determining hardware related problems (such as: ERAP)
- Copying and applying program temporary fixes (PTFs) to the licensed programs
- Adding entries to or listing the system service log
- Running the PATCH procedure to display or change disk or diskette sectors

SERVICE

S9020452-0

The SERVICE procedure has no parameters.

### Example

This example shows how to enter the SERVICE procedure to display the SERVICE menu.

SERVICE

## SERVLOG Procedure

The SERVLOG procedure allows you to add entries to the system service log. The SERVLOG procedure can be run from any display station.

To display or print entries in the service log, see the “DUMP Procedure” on page 1-5.

```
SERVLOG 'message text to log'
```

S9020470-0

**message text to log** specifies the entry to be made into the service log. The text must be enclosed by apostrophes (').

### Example

This example shows how to add the entry NEW KEYBOARD PLACED ON SYSTEM CONSOLE to the system service log.

```
SERVLOG 'NEW KEYBOARD PLACED ON SYSTEM CONSOLE'
```

# SETDUMP

---

## SETDUMP Procedure

The SETDUMP procedure allows you to debug programs running in main storage at predetermined break points or addresses without having to stop the main storage processor. The SETDUMP procedure allows you to take a task dump of your program when an address in your program is referenced.

Dump files are named #DUMP.nn on disk, where nn is a number from 00 through 99.

SETDUMP	ADDRESS RESTART TASK OFF
---------	-----------------------------------

S9020471-0

If no parameters are specified, a menu is displayed and you can choose the task you want.

**ADDRESS** specifies that the prompt used to set the address compare values is to be displayed. A job that was previously suspended by the address compare dump facility will be resumed.

**RESTART** specifies that the address compare dump facility is to be enabled to allow another address compare dump using the previously defined address values. A job that was previously suspended by the address compare dump facility will be resumed.

**TASK** specifies that a task dump is to be taken immediately for a specified task. A job that was previously suspended by the address compare dump facility will be resumed.

**OFF** specifies that the address compare dump facility is to be disabled. A job that was previously suspended by the address compare dump facility will be resumed.

### Example

The following example shows how to enter the SETDUMP procedure to define address compare dump values.

```
SETDUMP ADDRESS
```

## TOLIBR Procedure

You may be directed by the software service representative to run this procedure. This procedure allows an IBM module to be installed from diskette to your system disk.

```
TOLIBR file name,,,REPLACE,library-name
```

S9020552-0

**file name** specifies the file containing the one or more library members to be copied into the library.

**REPLACE** specifies that if library members already exist with the specified library name, they are to be replaced. If **REPLACE** is specified, new members replace existing members with duplicate names, and no messages regarding the replacements are displayed.

If **REPLACE** is not specified, members are placed in the library until a duplicate is found, at which time the system displays a message telling the operator that a duplicate exists. In response to the message, the operator can either cancel the job or continue processing. If the job is continued, the new member replaces the existing member in the library. If other duplicates are found during the job, then existing members are automatically replaced and no messages are displayed regarding the duplicate members.

**library name** specifies the library that will contain the copied members. If a library name is not specified, the current library is assumed.

### Example

This example shows how to copy source members from a diskette file named **PAY** into a library named **MYLIB** and replace any duplicate members without a warning message being displayed.

```
TOLIBR PAY,,,REPLACE,MYLIB
```



## | TRACE Procedure

The TRACE procedure allows you to keep a history of events that occur in the system. Selected system events are recorded as they occur in a variable-length, wrap-around table in main storage. Only one copy of the utility may be active at a time.

The TRACE procedure can be run in two different ways: 1) The procedure can be run interactively; that is, the procedure will display the status of each trace table and prompt for any changes; 2) The procedure can be entered with parameters, allowing the procedure to be run from the job queue, run from another procedure, or even be one of the IPL start up procedures.

From 16 through 512 entries can be traced in the main storage trace table. Alternate trace tables can also be created or deleted. You can select tracing for any of the following events:

- Any one or a combination of main storage supervisor calls (SVCs)
- Control storage processor supervisor calls
- Task dispatches
- Task swaps
- Disk input/output operations
- Component related events, such as:
  - Work station utility (WSU) functions
  - Remote work station functions
  - Asynchronous communications support
  - BSC
  - SNA
  - SDLC
  - X.25
  - MSRJE
  - SSP-ICF
  - APPC

You can also select disk logging options, and if you have selected any communications related events, you can select the communications lines to be traced.

```
TRACE [ CRT
       BATCH ] , [ table name ] , [ OLD
                                NEW
                                MOD
                                REMOVE ] , [ table size ] , [ CURRENT
                                                                'event list' ] ,

[ START
  STOP ] , [ trace file size ] , [ comm lines ]
```

S9020472-0

If no parameters are specified, TRACE will prompt you for the parameter values.

**CRT** specifies that the procedure is to be run interactively. You will be prompted for certain TRACE parameter values. If this parameter is not entered, CRT is assumed.

**BATCH** specifies that all parameter values have been specified, whether they be the default or something different.

**table name** specifies the name of the trace table to which events will be traced. If no name is specified, the main trace table is assumed.

**OLD** specifies that the trace table already exists. If the table cannot be found, then an error message is displayed. If this parameter is not specified, OLD is assumed.

**NEW** specifies that the table named is a new alternate trace table. If a table already exists with this name, an error message is displayed.

**MOD** specifies that if the table already exists, it will be changed according to other parameters specified. If the table does not exist, then a trace table will be created with the values of the other parameters specified.

**REMOVE** specifies that the alternate trace table named is to be deleted. If logging to disk is active, then logging is stopped before the table is deleted. The trace disk file is not deleted. The main trace table cannot be deleted.

**table size** specifies the maximum number of entries the trace table can hold. The valid number of entries that can be logged to a table are 16, 24, 32, 40, 48, 56, 64, 128, 192, 256, 320, 384, 448, and 512. Any other number within the range of 16 through 512 can be specified, but will be rounded up to the next valid number. If this parameter is not specified, and the table is a new table, the table size of 16 entries is assumed.

## TRACE

---

**CURRENT** specifies that the events previously selected for tracing will be traced. If **CURRENT** is specified for a new trace table, an error message is displayed.

**'event list'** specifies the events or tracing profiles to be traced. If more than one event or profile is to be traced, then the values for this parameter must be enclosed within quotes and the events or profile names must be separated by commas. The total length of the parameter cannot exceed 82 characters including the commas and quotes. Any event listed using the interactive method of this procedure can be specified. If you want to remove events being traced by a trace table, enter a minus sign (-) in front of the event to be removed. Putting a minus sign in front of a profile does not remove the profile from the trace table, but will remove the events that the profile represents. Events are added to or removed from the trace table profile in the order that they occur in the list.

**START** specifies that the trace table should start logging to disk. The trace table name must be different from the name of an existing user disk file name or an error message is displayed.

**STOP** specifies that the trace table should stop logging to disk.

**trace file size** specifies the size of the trace file in blocks. Each block of the file will hold approximately 80 trace entries. If this parameter is not specified and logging to disk is started, and no trace file exists, then a 10-block trace file is created.

| **comm lines** specifies which communications lines have events to trace. If more than one line number is  
| specified, numbers must be enclosed in single quotes (') and separated by commas. For example, to trace  
| lines 1 and 3, specify '1,3' or '3,1'. All communications lines are traced if the listed events or profiles  
| require selecting a communications line, **comm lines** is not specified, and no previous lines were selected.

## **Chapter 2. Copy and List Library Members**

The two \$MAINT utility programs described in this chapter show you how to copy members from a library and how to list library members.

The operation control language (OCL) and utility control statements required to copy or list are described for each of the two utility programs.

## Copy Library Members

---

### Copy Members from a Library

Using \$MAINT as described below enables you to copy IBM members from disk to diskette. This can be used to back up an individual IBM module to diskette.

```
// LOAD $MAINT
// FILE NAME-file name,UNIT-I1,PACK-volume id
// RUN
// COPY FROM-library name,TO-DISK,FILE-file name,

      NAME-member name,LIBRARY- $\left\{ \begin{array}{c} S \\ P \\ R \\ O \\ ALL \end{array} \right\}$ ,PTF- $\left\{ \begin{array}{c} NO \\ YES \end{array} \right\}$ 

// END
```

S9020551-0

**NAME-file name** specifies the file to be created.

**UNIT-I1** specifies that one or more library members are to be placed in a new diskette file or added to an existing diskette file.

**PACK-volume id** specifies the volume ID of the diskette. From 1 through 6 alphameric characters can be specified.

**FROM-library name** specifies the library containing the one or more library members to be copied.

**TO-DISK** specifies that the library members are to be copied to a diskette file.

**FILE-file name** specifies the file to be created.

**NAME-member name** specifies the library member to be copied from the library.

**LIBRARY-S** specifies that only library source members are to be copied. If no parameter is specified, SOURCE is assumed.

**LIBRARY-P** specifies that only the library procedure members are to be copied.

**LIBRARY-R** specifies that only the library subroutine members are to be copied.

**LIBRARY-O** specifies that only the library load members are to be copied.

**LIBRARY-ALL** specifies that all members of the library are to be copied.

**PTF-NO** specifies that PTFs have no particular significance. If PTF is not specified, PTF-NO is assumed.

**PTF-YES** specifies that only members that have PTFs applied are to be copied.

## List Library Members and Information

By using \$MAINT as described below, you can list the IBM members in the specified library.

```
// LOAD $MAINT
// RUN
// COPY TO-PRINT,NAME- { DIR
                        member name
                        member name.ALL
                        ALL } ,LIBRARY- { S
                                        P
                                        R
                                        O
                                        ALL
                                        SYSTEM } ,DISPLAY- { MEMBERS
                                                            DIRINFO }

                        FROM-library name

// END
```

S9020550-0

**TO-PRINT** specifies that the output is to go to the system list device.

**NAME-DIR** specifies that the directory status information and the directory entries are to be listed.

**NAME-member name** specifies the library member to be listed.

**NAME-member name,ALL** specifies that the contents of the library members whose names begin with the specified characters (member name) are to be listed. Up to 7 characters can be specified for member name.

**NAME-ALL** specifies that all members of the library are to be listed.

**LIBRARY-S** specifies that only library source members or directory entries are to be listed. If no parameter is specified, **SOURCE** is assumed.

**LIBRARY-P** specifies that only the library procedure members or directory entries are to be listed.

**LIBRARY-R** specifies that only the library subroutine members or directory entries are to be listed.

**LIBRARY-O** specifies that only the library load members or directory entries are to be listed.

**LIBRARY-ALL** specifies that all types of library members are to be listed.

**LIBRARY-SYSTEM** specifies that only the status information for a library is to be listed. No library member names are listed. **LIBRARY-SYSTEM** is valid only with **NAME-DIR**.

**DISPLAY-MEMBERS** specifies that the actual contents of the specified members will be listed.

**DISPLAY-DIRINFO** specifies that only the directory information for the specified members will be listed. Specifying **DISPLAY-DIRINFO** without also specifying a member name is the same as specifying **NAME-DIR**.

**FROM-library name** specifies the library containing the members to be listed. If a library name is not specified, the current library is assumed.

---

# Index

## Special Characters

### **\$MAINT utility program**

- copying library members to files 2-2
- listing library members 2-3

## A

### **activity**

- tracing system
  - using TRACE procedure 1-30

### **adding**

- entries to system service log 1-27

### **APAR files**

- creating using APAR procedure 1-2
- listing using DUMP procedure 1-5

### **APAR procedure 1-2**

## B

### **BSC**

- tracing using TRACE procedure 1-30

## C

### **changing**

- disk storage, using PATCH procedure 1-15
- diskette storage, using PATCH procedure 1-15

### **collecting diagnostic information 1-2**

### **communications**

- debugging subsystems using ICFDEBUG procedure 1-14

### **COPY utility control statement, \$MAINT utility program**

- listing libraries 2-3

### **copying**

- APAR information 1-2
- diagnostic information 1-2
- libraries
  - from a file using TOLIBR procedure 1-29
- library members
  - from a file using TOLIBR procedure 1-29

## D

### **data areas**

- listing using DUMP procedure 1-5

### **debugging**

- communications subsystems
  - using ICFDEBUG procedure 1-14
- programs
  - using SETDUMP procedure 1-28
  - using TRACE procedure 1-30
- system
  - using the SERVICE procedure 1-26
  - using TRACE procedure 1-30

### **device error information, listing 1-13**

### **DFA procedure 1-4**

### **diagnostic information, collecting 1-2**

### **disk**

- changing using PATCH procedure 1-15

### **disk files**

- copying library members to 1-29

### **diskettes**

- changing using PATCH procedure 1-15

### **displaying**

- device error information 1-13
- libraries using the \$MAINT utility program 2-3

### **dump**

- of program storage using SETDUMP procedure 1-28

### **dump file analysis procedure 1-4**

### **dump files**

- listing using DUMP procedure 1-5

### **DUMP procedure 1-5**

## E

### **ERAP procedure 1-13**

### **error information, device, listing 1-13**

## I

### **ICFDEBUG procedure 1-14**



**L****libraries**

- copying
  - from a file using TOLIBR procedure 1-29
- displaying
  - using the \$MAINT utility program 2-3
- listing
  - using the \$MAINT utility program 2-3

**library members**

- copying
  - from a file using TOLIBR procedure 1-29

**listing**

- device error information 1-13
- libraries using the \$MAINT utility program 2-3
- system data areas using DUMP procedure 1-5
- log, system service, adding entries to 1-27

**M****\$MAINT utility program**

- copying library members to files 2-2
- listing library members 2-3

**P****PATCH procedure 1-15****PROBLEM**

- procedure 1-16

**procedures**

- APAR 1-2
- DFA 1-4
- DUMP 1-5
- ERAP 1-13
- ICFDEBUG 1-14
- PATCH 1-15
- PROBLEM 1-16
- PTF 1-17
- SERVICE 1-26
- SERVLOG 1-27
- SETDUMP 1-28
- TOLIBR 1-29
- TRACE 1-30

**program temporary fixes 1-17****programs**

- debugging
  - using SETDUMP procedure 1-28

**PTF (program temporary fixes) 1-17****PTF procedure 1-17****S****service aid procedures**

- APAR 1-2
- DFA 1-4
- DUMP 1-5
- ERAP 1-13
- PATCH 1-15
- PTF 1-17
- SERVICE 1-26
- SERVLOG 1-27
- SETDUMP 1-28
- TRACE 1-30

**service log**

- adding entries to 1-27
- listing using DUMP procedure 1-5

**SERVICE procedure 1-26****servicing the system using the SERVICE procedure 1-26****SERVLOG procedure 1-27****SETDUMP procedure 1-28****subsystems**

- debugging using ICFDEBUG procedure 1-14

**system**

- debugging using the SERVICE procedure 1-26
- testing using the SERVICE procedure 1-26
- tracing activity using TRACE procedure 1-30
- system service log, adding entries to 1-27

**T****testing**

- system using the SERVICE procedure 1-26

**TOLIBR procedure 1-29****TRACE procedure 1-30****U****utility programs**

- \$MAINT
  - listing library members 2-3

### READER'S COMMENT FORM

**Please use this form only to identify publication errors or to request changes in publications.** Direct any requests for additional publications, technical questions about IBM systems, changes in IBM programming support, and so on, to your IBM representative or to your IBM-approved remarketer. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

- If your comment does not need a reply (for example, pointing out a typing error), check this box and do not include your name and address below. If your comment is applicable, we will include it in the next revision of the manual.
- If you would like a reply, check this box. Be sure to print your name and address below.

Page number(s):

Comment(s):

**Please contact your IBM representative or your IBM-approved remarketer to request additional publications.**

Name \_\_\_\_\_

Company or  
Organization \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip Code \_\_\_\_\_

Phone No. \_\_\_\_\_  
Area Code \_\_\_\_\_

Fold and tape. **Please do not staple.**

Cut Along Line



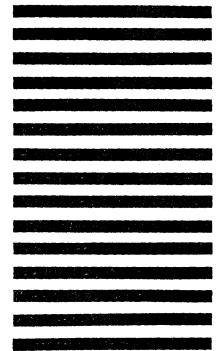
NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**

FIRST CLASS / PERMIT NO. 40 / ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

**International Business Machines Corporation**  
Information Development  
Department 245  
Rochester, Minnesota, U.S.A. 55901



Fold and tape. **Please do not staple.**

Cut Along Line





## Service Aid Procedures

International Business Machines Corporation

File Number  
S36-36

Order Number  
SC21-9494-1

Part Number  
59X4035

Printed in U.S.A.

SC21-9494-01

