

IBM

*Personal Computer
Hardware Reference
Library*

**IBM RT PC
AIX Operating System
Problem Determination Guide**

SA23-2603-0

Second Edition (February 1988)

This is a major revision of and obsoletes the first edition of *IBM RT PC AIX Operating System Problem Determination Guide*.

This edition applies to all subsequent releases until otherwise indicated in new editions or technical newsletters. Changes are made periodically to the information herein; these changes will be incorporated in new editions of this publication.

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.

International Business Machines Corporation provides this manual "as is," without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this manual at any time.

Products are not stocked at the address given below. Requests for copies of this product and for technical information about the system should be made to your authorized RT PC dealer.

A reader's comment form is provided at the back of this publication. If the form has been removed, address comments to IBM Corporation, Department 997, 11400 Burnet Road, Austin, Texas 78758. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

IBM is a registered trademark of International Business Machines Corporation.

© Copyright International Business Machines Corporation 1985, 1988

About This Book

Purpose

Use this book to do software problem determination on the RT Personal Computer (RT PC) system. If the AIX¹ Operating System is not installed and working, the tools provided in this book cannot be used. When you suspect a software problem, use this book to find the failure.

Audience

The audience for this book is the user who has some understanding of the RT PC system and is having software problems with the RT PC system programs.

How to Use This Book

Section 1, “Software Problems” provides helpful information to aid in finding the source of software problems. Descriptions of the software error codes displayed on the system’s two-digit display and the cause and recovery actions are also found in this section.

Section 2, “Dump Utilities” describes the steps needed to dump the system memory and provides instructions for using the dump formatter.

Section 3, “Error Log Analysis” describes the commands for using the error log report.

¹ Trademark of IBM

Section 4, “The Trace Utility” describes the commands and steps used in setting up the trace profile, starting and stopping trace, and producing a trace report.

A Reader’s Comment Form and Book Evaluation Form are provided at the back of this book. Use the Reader’s Comment Form at any time to give IBM information that may improve the book. After you become familiar with the book, use the Book Evaluation Form to give IBM specific feedback about the book.

Related Publications

- *IBM RT PC Problem Determination Guide* provides instructions for running diagnostic routines to locate and identify hardware problems. A problem determination guide for software and two high-capacity (1.2MB) diskettes containing the IBM RT PC diagnostic routines are included.
- *IBM RT PC Usability Services Guide* shows how to create and print text files, work with directories, start application programs, and do other basic tasks with the Usability Services. (Packaged with *Usability Services Reference*)
- *IBM RT PC AIX Operating System Commands Reference* lists and describes the AIX Operating System commands.
- *IBM RT PC Messages Reference* lists messages displayed by the IBM RT PC and explains how to respond to the messages.
- *IBM RT PC Virtual Resource Manager Technical Reference* describes the Virtual Resource Manager (VRM) routines, how to use the VRM debugger, how to develop and install code into the VRM, and defines the interface to the VRM-supplied device drivers. This book also describes the Virtual Machine Interface (VMI) between the Virtual Resource Manager and the AIX Operating System and provides information about process control, memory management, the I/O subsystem, the minidisk manager, and device drivers. (Available optionally)
- *IBM RT PC Bibliography and Master Index* provides brief descriptive overviews of the books and tutorial program that support the IBM RT PC hardware and the AIX Operating System. In addition, this book contains an index to the RT PC and AIX Operating System library.

Ordering Additional Copies of This Book

To order additional copies of this publication, use either of the following sources:

- To order from your IBM representative, use Order Number SBOF-1825.
- To order from your IBM dealer, use Part Number 22F9800.

Contents

Section 1. Software Problems	1-1
About This Section	1-3
Section 2. The Dump Utilities	2-1
About this Section	2-3
Dump Formatter	2-18
Section 3. Error Log Analysis	3-1
About This Section	3-3
Error Log Report Types	3-4
Error Log Command Format	3-5
Section 4. The Trace Utility	4-1
Using the Trace Utility	4-2
The Trace Profile	4-3
Creating the Trace Report	4-6
Section 5. Blinking Two-Digit Display Messages	5-1
About This Section	5-3
Errors Found at IPL	5-4
Errors Found During VRM Install	5-14
Appendix A. The VRM Maintenance Facility (VMF)	A-1
About this Appendix	A-4
VRM Maintenance Facility (VMF) Overview	A-5
Appendix B. Error Log Entries	B-1
About This Appendix	B-3



Section 1. Software Problems

CONTENTS

About This Section	1-3
Step 1. Determining Software Problems	1-4
Step 2. Message Displayed	1-6
Step 3. Failing Program Undetermined	1-7
VRM and AIX Operating System Dumps	1-7
Trace	1-7

About This Section

This software problem section describes the tools available for use in software problem determination. These tools will help find the cause of the problem but will not fix the problem. Once you have located the failing program, use your normal problem reporting procedures.

Step 1. Determining Software Problems

If you suspect a program has a software problem, you can do several things to determine if the problem is really software.

The Symptom Index is a list of system problems or symptoms. The section or step related to the symptom is contained in the action column of the Symptom Index.

1. Find the failure or symptom that most closely resembles your system failure.

Note: If a program has been working correctly for some time and you have made no changes, the system hardware may be failing. To check the hardware, see *IBM RT PC Problem Determination Guide* for instructions on running hardware diagnostics.

2. Follow the instructions in the Action column.
3. Follow the instructions given in the steps to complete the problem determination procedures.

Symptoms	Actions
c6 alternately displays with a two-digit abend code in the two-digit display during normal operation	Go to "Step 2. VRM Dump (System Started)" on page 2-5.
XX is blinking in the two-digit display. (XX any two digits.)	Go to Section 5, "Blinking Two-Digit Display Messages" on page 5-1.
A message is displayed for the failure	Go to "Step 2. Message Displayed" on page 1-6.
A message directed you to dump the system memory	Go to "Step 1. Dump Utilities" on page 2-4.
Incorrect system operation with no message (incorrect output, unexplained system actions)	Go to "Step 1. Summary Error Log" on page 3-6.
System is on but appears inactive (loop, no keyboard response)	Go to "Step 9. System Appears to be Hung" on page 2-14.
Failing program undetermined	Go to "Step 3. Failing Program Undetermined" on page 1-7.

Step 2. Message Displayed

If you have a message:

1. Read the message carefully
2. Do the recovery action, if any is given.

Once you have responded to the message:

3. Try the operation again.

If the program still fails:

4. Go to “Step 1. Summary Error Log” on page 3-6 to view the error log.

The error log provides a record of errors and system events and if possible, the most likely cause of the failure.

Step 3. Failing Program Undetermined

If you are unable to find the failing program in the error log, you may need to gather additional information. This information may consist of one or more of the following:

VRM and AIX Operating System Dumps

The dump utilities provides the commands needed to dump the contents of memory. Go to “Step 1. Dump Utilities” on page 2-4 for instructions on starting the dumps and for instructions on producing the dump reports.

Trace

The trace utility enables you to record certain system actions while a program that is suspected of having a problem is running. Go to Section 4, “The Trace Utility” on page 4-1 for instructions on starting and stopping trace and for instructions on producing a trace report.



Section 2. The Dump Utilities

CONTENTS

About this Section	2-3
Step 1. Dump Utilities	2-4
Step 2. VRM Dump (System Started)	2-5
Step 3. VRM Dump (User Started)	2-7
Step 4. VRM Dump (User Started From VRM Debugger)	2-9
Step 5. AIX Operating System Dump (System Started)	2-10
Step 6. AIX Operating System Dump (User Started)	2-11
Step 7. AIX Operating System Dump (Copy to Diskette)	2-12
Step 8. Core File	2-13
Step 9. System Appears to be Hung	2-14
Step 10. Full Image Dump	2-15
Dump Utility Messages	2-17
Dump Formatter	2-18

About this Section

This section contains instructions for using the AIX Operating System , VRM, and Full Image dump utilities to copy the contents of memory at the time of a program failure. The core file and crash utility are also described.

The data collected by these utilities is intended to help you, or the person servicing the system, determine the cause of the failure.

Step 1. Dump Utilities

Use the following procedure to determine where to go in this Chapter. Find the dump condition in the table and go to the associated step.

Dump Conditions	Actions
A c6 and an abend code are alternately displayed in the two-digit display.	Go to “Step 2. VRM Dump (System Started)” on page 2-5.
You want to start the VRM Dump	Go to “Step 3. VRM Dump (User Started)” on page 2-7.
Dump is started from VRM debugger	Go to “Step 4. VRM Dump (User Started From VRM Debugger)” on page 2-9.
A PANIC message was displayed on the screen	Go to “Step 5. AIX Operating System Dump (System Started)” on page 2-10.
You want to start the AIX Operating System dump.	Go to “Step 6. AIX Operating System Dump (User Started)” on page 2-11.
A core file was created	Go “Step 8. Core File” on page 2-13.
System appears to be hung (no work can be done, no response from commands or interrupt keys)	Go to “Step 9. System Appears to be Hung” on page 2-14.
You are instructed to take a Full Image dump	Go to “Step 10. Full Image Dump” on page 2-15.

Step 2. VRM Dump (System Started)

The **c6** and an abend code alternately displayed in the two-digit display is a prompt to load the dump diskette. The abend codes are described in *IBM RT PC Messages Reference* under 032-1xx, where xx is the abend code. Blinking error codes that can occur during IPL (without the **c6**) are described in Chapter 5 of this book.

Note: The dump diskette can be any formatted high-capacity diskette. A dump diskette should have been created and placed in the back of this manual when the AIX Operating System was installed.

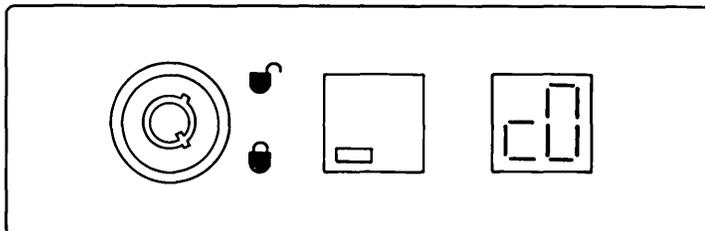
1. Remove the dump diskette from the back of this manual.
2. Insert the dump diskette into the diskette drive in location A . (The top drive if there are two.)
3. Close the diskette drive door.
4. Press **CTRL-ALT-numpad8**. (Use the left **ALT** key.)

A **c8** appears in the two-digit display while the system verifies the diskette is usable. If the diskette is not usable, a **c5** appears. In this case you should remove the bad diskette and insert another.

If the diskette is good, **c9** in the two-digit display indicates the dump is running. The dump takes approximately 5 minutes to complete.

CONTINUE to next page.

When the dump is complete, **c0** displays in the two-digit display. For any other messages displayed, see “Dump Utility Messages” on page 2-17.



IS “c0” DISPLAYED?

YES Press **CTRL-ALT-Pause** to restart the operating system. Go to “Dump Formatter” on page 2-18 and format the dump diskette.

NO There is a problem with the dump utility. Go to *IBM RT PC Problem Determination Guide* and run diskette hardware diagnostics.

If no problem is found, use your normal problem reporting procedures.

Step 3. VRM Dump (User Started)

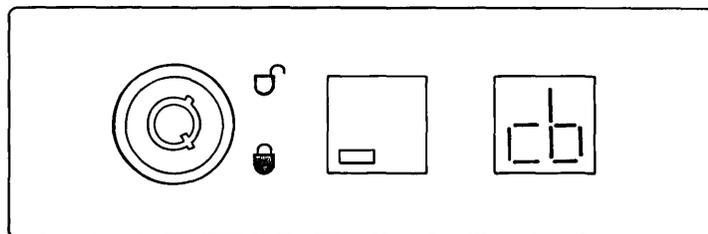
If you want to start the dump utility, follow these steps.

1. Remove the dump diskette from the back of this manual.

Note: The dump diskette may be any formatted high-capacity diskette. A dump diskette should have been created and placed in the back of this manual when the AIX Operating System was installed.

2. Press **CTRL-ALT-numpad8**. (Use the left **ALT** key.)

A **c6** alternating with abend code 00 displays in the two-digit display.



CONTINUE to next page.

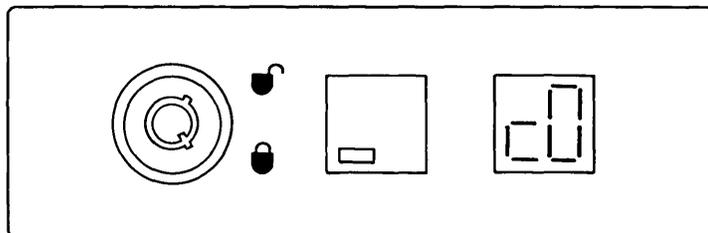
3. Insert the dump diskette into the diskette drive in location A . (The top drive if there are two.)

4. Press the **CTRL-ALT-numpad8** keys again. (Use the left ALT key.)

A **c8** appears in the two-digit while the system verifies the diskette is usable. If the diskette is not usable, a **c5** appears. In this case you should remove the bad diskette and insert another.

If the diskette is good, **c9** in the two-digit display indicates the dump is running. The dump takes approximately 5 minutes to complete.

When the dump is complete, **c0** displays in the two-digit display. For any other messages displayed, see "Dump Utility Messages" on page 2-17.



IS "c0" DISPLAYED?

YES Press **CTRL-ALT-Pause** to restart the operating system. Go to "Dump Formatter" on page 2-18 and format the dump diskette.

NO There is a problem with the dump utility, the diskette, or the diskette drive. Go to *IBM RT PC Problem Determination Guide* and run diskette hardware diagnostics.

If no problem is found, use your normal problem reporting procedures.

Step 4. VRM Dump (User Started From VRM Debugger)

Use the following instructions to start a dump from the debugger.

1. Read the debugger message generated from the crash.

2. Enter the command:

```
show
```

3. Read any dump or PANIC messages on the screen.

4. Hit any key to return to the debugger.

5. Enter the command:

```
quit dump
```

A **c6** and an abend code alternately displays in the two-digit display. The abend codes are described in *IBM RT PC Messages Reference* under 032-1xx, where xx is the abend code. Blinking error codes that can occur during IPL (without the **c6**) are described in Chapter 5 of this book.

Go to “Step 2. VRM Dump (System Started)” on page 2-5 and follow the instruction to complete the dump.

Step 5. AIX Operating System Dump (System Started)

When an error is so severe that the operating system cannot continue, a PANIC message displays. For most PANIC messages, an AIX Operating System dump, sometimes called a virtual machine dump, is written on the dump minidisk on the hardfile.

Refer to the *IBM RT PC Messages Reference* for more information about the specific PANIC message.

When the dump is complete, this message displays:

dump succeeded, xxxx blocks

Note: Use the crash utility to look at the dump.

Go to “Step 7. AIX Operating System Dump (Copy to Diskette)” on page 2-12.

Step 6. AIX Operating System Dump (User Started)

If you want to start the AIX Operating System dump utility:

1. Press the **CTRL-ALT-END** keys.
2. When the dump is written on the dump minidisk on the hardfile, this message displays:

dump succeeded, xxxx blocks

Note: The crash utility can be used to look at the dump.

CONTINUE to next page.

Step 7. AIX Operating System Dump (Copy to Diskette)

To submit AIX Operating System dump information the dump must be copied to diskettes. Follow these instructions to copy the information from the dump minidisk on the hardfile to diskettes.

Note: The following procedure attempts to dump `/dev/hd4` to one or more diskettes and `/usr/adm/ras/errfile.0` and `/usr/adm/ras/errfile.1` to another diskette. Do not be concerned if any of the three do not exist or cannot be reached.

1. Insert the **Install/Maintenance** diskette into drive 0 and IPL. The "System Management" panel appears on the screen.
2. Select option 3. "Start the Standalone Shell". A # prompt appears on the screen.
3. Remove the **Install/Maintenance** diskette from drive 0.
4. Insert a formatted, high-capacity diskette into drive 0.
5. Type `errdmp` and press **ENTER**
6. Remove the first diskette and insert additional diskettes when prompted.
7. Remove the final diskette and IPL as normal.

Use your normal problem reporting procedures to submit the dump information.

STOP, you have completed these procedures.

Step 8. Core File

A core file is the result of a terminated process when various errors occur. Common errors are: memory violation, illegal instructions, and bus errors. The core file is written in the process's working directory.

To examine the core file, use the **sdb** command. The **t** subcommand of the **sdb** command provides the name and address of the last function that was running in the program that ended abnormally. For more information on the **sdb** command, see *AIX Operating System Commands Reference*.

Note: If the program that core dumped is part of an LPP, copy the core file on a diskette in a backup format and submit the dump through your normal problem reporting procedures.

STOP, you have completed these procedures.

Step 9. System Appears to be Hung

A system is considered hung when no work can be done or no response is received from commands or interrupt keys.

When this condition occurs:

1. Press **CTRL-ALT-END** to copy the virtual machine to the dump minidisk. See “Step 7. AIX Operating System Dump (Copy to Diskette)” on page 2-12 for more information.
2. Press **CTRL-ALT-numpad8** to copy the VRM to a dump diskette. See “Step 3. VRM Dump (User Started)” on page 2-7 for more information.
3. IPL the system.
4. Submit the dump information through your normal problem reporting procedures.

STOP, you have completed these procedures.

Step 10. Full Image Dump

This step dumps all of your system's real memory and requires multiple (up to four) diskettes. A full image dump takes more time than a VRM dump and you cannot use the dump formatter to inspect the information on the diskettes. Therefore, you should not use this step unless you have been directed to do so by a support person servicing your system. Otherwise, use the VRM dump procedures described in steps 2 and 3 of this chapter.

You can start a full image dump yourself, or you can take a full image dump in place of a system-started VRM dump. In either case, have enough formatted diskettes available for your system:

- Two diskettes for a 2mb system.
- Three diskettes for a 3mb system.
- Four diskettes for a 4mb system.

Number the diskettes in the sequence you will use them. 1 for the first diskette, 2 for the second diskette, and so on.

If you want to take a system-started full image dump, go to item 2 in the following procedure.

If you want to start the full image dump, do the following:

1. Press **CTRL-ALT-numpad7**. (Use the left **ALT** key.)

A **c6** alternating with abend code **00** displays in the two-digit display.

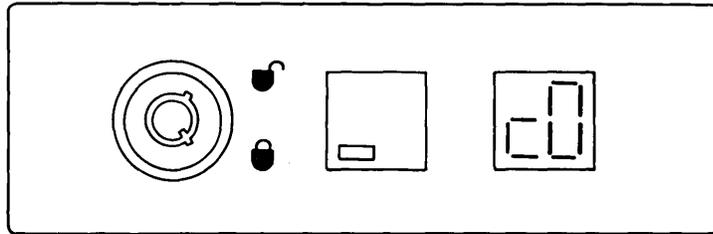
2. (Start here for a system initiated full image dump.) Insert diskette 1 into the diskette drive in location A. (The top drive if there are two.)
3. Press **CTRL-ALT-numpad7**. (Use the left **ALT** key.)

A **c8** appears in the two-digit while the system verifies the diskette is usable. If the diskette is not usable, a **c5** appears. In this case you should remove the bad diskette and insert another.

If the diskette is good, **c9** in the two-digit display indicates the dump is running. Each diskette takes approximately 5 minutes to complete.

When a diskette is full, c7 displays in the two-digit display. When that happens, remove the full diskette from the diskette drive and insert the next higher numbered diskette. A c8 appears while the diskette is inspected and c9 appears when the dump continues.

When the dump is complete, c0 displays in the two-digit display. For any other messages displayed, see "Dump Utility Messages" on page 2-17.



Is "c0" DISPLAYED?

- YES** Submit the dump diskettes to the person that requested the full image dump. You have now completed these procedures.
- NO** There is a problem with the dump utility. Go to *IBM RT PC Problem Determination Guide* and run diskette hardware diagnostics.
- If no problem is found, use your normal problem reporting procedures.

Dump Utility Messages

The messages for the dump utility are displayed on the two-digit display. These messages tell you when the dump is complete, when a dump diskette is needed, and if there is a problem found while running the dump utility.

The messages and descriptions are:

- c0 The dump utility completed.
- c4 There is a problem with the dump or with the diskette device driver. Make sure you are using a formatted diskette. If the dump does not run, go to *IBM RT PC Problem Determination Guide* and test the diskette drive.
- c5 The diskette is damaged. Insert another diskette in diskette drive A.
- c6 The system is requesting a VRM dump. Press the key sequence to start the VRM dump, or press CTRL-ALT-PAUSE to restart the system.

The characters c6 flash alternatively with a number *xx*. If this number is from 01 to 27, look at message 032-1*xx* in the messages book. For example, if **25** flashes on the system unit, look at message 032-125.

- c7 The dump diskette is full. Insert another dump diskette in diskette drive A.
- c8 The system is checking the dump diskette for damage.
- c9 The dump utility is running.

Dump Formatter

The Dump Formatter is used to format the information contained on the dump diskette into a readable form. The information can be directed to the display, the printer, or to a file.

If the dump program finds a problem, the dump formatter gives header information describing the problem.

The dump formatter can run in batch or interactive mode. For a complete description of the **dumpfmt** command see *IBM RT PC AIX Operating System Commands Reference*.

1. To run the dump formatter and produce only the header information, type the command:

```
dumpfmt -h
```

2. Press ENTER.
3. Report the problem to your service representative if you have not already done so. The header information may be required.

This screen is an example of the dump header:

```
Mon June 17 09:30:32 1985
Failing Component:      cccccccc
Module Start Address:   xxxxxxxx
Module Offset Address:  xxxxxxxx
CID = nnnnnnnn Name = yyyyyy
ABEND Code = zzzzzzzz
```

Section 3. Error Log Analysis

CONTENTS

About This Section	3-3
Error Log Report Types	3-4
Error Log Command Format	3-5
Step 1. Summary Error Log	3-6
Step 2. Summary Error Log	3-7
Step 3. Summary Error Log	3-9
Step 4. Detail Error Log	3-10

About This Section

The error log is used to help find the failing area of the system. The failures may be hardware or software. This error log is useful in finding problems when the system is failing only now and then. If the Diagnostic Routines are unable to find a suspected hardware problem, the error log may have recorded the error.

Error Log Report Types

Two types of error log reports can be produced; a summary report and a detail report.

The summary report contains one-line descriptions of system activities and errors. You can specify which error types you are looking for (such as diskette, display, or program errors), and the time the errors occurred.

The detail report displays all the information available for the error. Just as in the summary report, you can specify which error types you want to see.

Error Log Command Format

To generate an error log report, enter a command in the form:

```
errpt flags filenames
```

Note: Flags and filenames are optional parameters.

Filenames specifies the error log files. The **errpt** command accepts the following flags:

- s *date*** Ignore all records logged earlier than *date*. *Date* must be in the form *MMddhhmmyy*.
- e *date*** Ignore all records logged later than *date*.
- a** Produce a detailed report.
- n *nodename*** Include only error entries from *nodename* in the report.
- v *vmid*** Include only error entries from this system name (*vmid*) in the report.
- d *list*** Limit the report to the types of error record specified in *list*. For the values that can appear in *list*, see **errpt** in *IBM RT PC AIX Operating System Commands Reference*.

Step 1. Summary Error Log

Notes:

1. You must use the AIX Operating System to issue the following commands. Go to *IBM RT PC Problem Determination Guide* and run the hardware diagnostics if you are not using the AIX Operating System.
2. If you are using Usability Services, you may wish to run error log using Usability Services. For more information on running error log from Usability Services see *IBM RT PC Usability Services Reference*.

CONTINUE to next page.

ERROR LOG REPORT

Error log: /usr/adm/ras/errfile

Date/Time	Class	Subclass	Type	Device	Cause
Fri Aug 16 9:28:34	IPL/Shutdown	Soft IPL	Info		Info
Mon Aug 19 11:15:22	IPL/Shutdown	Manual IPL	Info		Info
Mon Aug 19 14:07:02	Hardware	Diskette	Perm	DSKT	Hardware
Tue Aug 20 17:15:45	Hardware	Diskette	Counters	DSKT	Info

IS THE ERROR LOG REPORT DISPLAYED?

YES CONTINUE to next page.

NO Enter the command again. If report still does not display, the error log file may not exist.

STOP, you have completed these procedures.

Step 3. Summary Error Log

The Error Log screen is divided into six areas: Date/Time, Class, Subclass, Type, Device, and Cause.

Look at the Date/Time column first.

ERROR LOG REPORT						
Error log: /usr/adm/ras/errfile						
Date/Time	Class	Subclass	Type	Device	Cause	
Fri Aug 16 9:28:34	IPL/Shutdown	Soft IPL	Info		Info	
Mon Aug 19 11:15:22	IPL/Shutdown	Manual IPL	Info		Info	
Mon Aug 19 14:07:02	Hardware	Diskette	Perm	DSKT	Hardware	
Tue Aug 20 17:15:45	Hardware	Diskette	Counters	DSKT	Info	

IS THERE A DATE AND TIME THAT CLOSELY MATCHES YOUR TIME OF ERROR?

YES CONTINUE to next page.

NO No entries were found in the error log.

STOP, you have completed these procedures.

Step 4. Detail Error Log

Display a detail error log report using the date and time of the suspected error.

1. Type the command:

```
errpt -a -s MMddhhmmyy
```

EXAMPLE

```
errpt -a -s 0525143585
```

A detail report for the error starting on May 25, 1985, at 2:35 p.m. will be displayed.

2. Press **ENTER**.

If there are more errors near the time you suspect the error occurred, look at the detail reports for those times also.

ERROR LOG REPORT						
Error log: /usr/adm/ras/errfile						
Date/Time	Class	Subclass	Type	Device	Cause	
Fri Aug 16 9:28:34	IPL/Shutdown	Soft IPL	Info		Info	
Mon Aug 19 11:15:22	IPL/Shutdown	Manual IPL	Info		Info	
Mon Aug 19 14:07:02	Hardware	Diskette	Perm	DSKT	Hardware	
Tue Aug 20 17:15:45	Hardware	Diskette	Counters	DSKT	Info	

CONTINUE to next page.

```

                                ERROR LOG REPORT

Error Log: /usr/adm/ras/errfile

Date/Time          Class          Subclass          Type          Device          Cause
Mon Aug 19 13:03:52  Software      Appl Prog        Perm          Software      Software

Program Error      Module_ID=ccccccc  RC=xxxxxxxx

D01 = xxxxxxxx    D02 = xxxxxxxx    D03 = xxxxxxxx    D04 = xxxxxxxx
D05 = xxxxxxxx    D06 = xxxxxxxx    D07 = xxxxxxxx    D08 = xxxxxxxx
D09 = xxxxxxxx    D10 = xxxxxxxx    D11 = xxxxxxxx    D12 = xxxxxxxx
D13 = xxxxxxxx    D14 = xxxxxxxx    D15 = xxxxxxxx    D16 = xxxxxxxx

```

IS A DETAIL ERROR REPORT SHOWN?

YES For hardware errors, read and follow the instruction in the detail report.

For software errors, use your normal problem reporting procedures.

NO Enter the command again. If command still fails, report the problem.



Section 4. The Trace Utility

Using the Trace Utility

This trace utility is used to track the system events when a program is running. By tracing, or tracking these events, a report can be made explaining the system's performance and actions.

This utility is run in background mode while using a program, to trace the steps and actions of the program. By using the trace, a possible problem with a program may be found.

The Trace Profile

Before you can start the trace, you must select the *trace profile* and the events you wish to trace. The trace profile is a file that contains the areas that can be traced and is provided with the system. Events are selected by using the system editor to change the file.

To edit the system trace profile:

1. Type the command:

```
e /etc/trcprofile
```

The trace profile is now displayed. See the following page for a sample trace profile.

2. Remove the asterisk (*) from the first column of the events you want to trace.
 3. If the event has a data field, enter that information also.
 4. File the profile.
 5. Go to “Step 1. Starting the Trace” on page 4-5 to start the trace.
-

```

*****
* SYSTEM TRACE PROFILE
*****
* To set trace on for an event class, remove the comment mark (*) from the
* first column of the line containing the event you wish to trace.
* Add a comment mark (*) in the first column of lines containing event types
* you wish to stop tracing.

***** Event
*   Type      Description

*****
          Applications

*****
          UNIX Extensions
*   36        Config

*****
          UNIX System Calls
*   60        Shared Memory
*   61        Messages
*   62        Semaphores
*   63        Signals
*   64        Time
*   65        File System
*   66        File Handling
*   67        Directory Handling
*   68        Process

*****
          VRM Components
*   110       Async
*   112       Async Interrupts
*   113       Virtual Terminal Manager
*   114       Keyboard Interrupts
*   115       Locator Interrupts

*   150       User-Defined Events

```

Step 1. Starting the Trace

After the setup of the trace profile has been completed, you can start the trace.

1. Type the command:

```
trace&
```

2. Press **ENTER**.

The trace is on, now run the program you want to trace.

Step 2. Stopping the Trace

Once you have traced the information,

1. Type the following command to stop the trace:

```
trcstop
```

2. Press **ENTER**.

Creating the Trace Report

The trace report program is used to make a readable copy of the trace information. The output can be sent to a file, the display, or a printer.

To produce the report:

1. Type the command and if needed, the options:

```
trcrpt
```

The options are:

-s *date* Starting date/time desired in the formatted file.
Date/time format is *MMddhhmmyy*.

-e *date* Ending date/time desired in the formatted file.
Date/time format is *MMddhhmmyy*.

2. Press ENTER.

Figure 4-1 on page 4-7 is an example of a trace report.

TRACE LOG REPORT

File: /usr/adm/ras/trcfile

Wed Feb 6 15:07:19 1985

System: Sys01

Node: Self

Version: 0

Machine: xxxx

TIME	SEQ	PID	IODN	IOCN	TYPE	HOOK	DATA
09:01:20.57	0001	00015			Process	alarm	seconds=585
09:01:20.58	0002	00015			Process	fork	newpid=29
09:01:20.58	0003	00029			Process	sgetpid	
09:01:20.58	0004	00029			Process	exec	errno=0
09:01:20.59	0005	00029			Process	exit	status=0
09:01:21.00	0006	00015			Time	time	time=476562057

Summary of event counts.

Event 0: 2

Event 68: 43

Event 64: 16

Total number of events: 3

Section 5. Blinking Two-Digit Display Messages

CONTENTS

About This Section	5-3
Errors Found at IPL	5-4
Errors Found During VRM Install	5-14

About This Section

This section describes the messages that may appear on the two-digit display during IPL. The cause and recovery procedures are given for each message. Only blinking two-digit messages are described in this section. If a two-digit message is displayed that does not blink, see the *IBM RT PC Problem Determination Guide* to describe the hardware displays.

Errors Found at IPL

The error numbers found in this section are errors that are found by the loadlist processor during system IPL. Some of the errors are non-fatal. When a non-fatal error is sensed, the error number will blink for about ten seconds, and then continue with the IPL.

For fatal errors, or errors that completely halt the IPL, the error number will continue to blink until the system is IPLed again.

Most of the errors found in this section are caused by programming errors. Examples of programming errors are: an error in a module that the loadlist processor loaded or executed, or an error in the loadlist processor code itself.

Since the user has the ability to install other software on the VRM minidisk, it is sometimes difficult to tell if the error occurred in IBM code, or in the user's code, if changes have been made to the VRM minidisk. For all errors that indicate a change may have been made to the VRM, the user should again install the VRM from his IBM-supplied diskette.

Once the VRM is installed again, IPL the system. If the problem still appears, use your normal problem reporting procedures.

01 Loadlist processor begins execution.

Cause: Control has been passed to the loadlist processor by ROS.

Action: This is only an information message. No action required.

02 Read error.

Cause: A read error occurred on the IPL device (either disk or diskette) during execution.

Action: If you were using a diskette to IPL, try again with a backup diskette. If you were using the disk to IPL, install the Virtual Resource Manager (VRM) again, then IPL the system. If error condition still exists, go to Problem Determination Guide and run diagnostics on the diskette drive or fixed disk.

03 Not enough memory to load module.

Cause: There was not enough free space in memory to load a module. If you have changed the VRM minidisk by adding code, then it is possible that an added routine is writing over the internal data fields of the loadlist processor.

If the VRM minidisk has not been changed, there may not be enough contiguous memory to load the system.

Action: If the VRM minidisk has been changed, install the VRM again. If this error occurs when the VRM minidisk has not been changed, go to *IBM RT PC Problem Determination Guide* and run diagnostics on the memory.

04 Not enough space for the Define Device Area (DDA).

Cause: There was not enough memory to allocate space for the DDA. See error **03** for possible causes and actions.

05 Not enough space for the Module Index Table (MIT).

Cause: The MIT contains an entry for each file that is loaded or mapped. Either there was not enough memory to allocate space for the MIT (see error **03** for more information), or there are more than 128 files in the loadlist (the files in the directory *ldlist*).

Action: If the number of files is over 128, the number must be decreased by removing files or changing permission bits, or installing the VRM again.

06 Not enough space for the Disk Table of Contents (DTC).

Cause: The DTC contains a list of disk addresses (or diskette addresses if the IPL was from diskette) for all modules that are loaded (permission bits=440 or 410) or mapped (permission bits=040). Either there was not enough memory to allocate space for the DTC (see error **03** for more information), or the amount of space reserved for the DTC was not large enough to contain all the disk addresses for all of the loaded or mapped modules.

Action: If additional code has been added to the VRM minidisk, it may be necessary to remove or change the permission bits for some of the loaded or mapped files until there is enough space in the DTC for all the disk addresses.

When IPL is from disk, each DTC disk address represents a 2,048 byte block on the disk. There is space in the DTC for 2,048 disk addresses. When IPL is from diskette, each DTC diskette address represents a 512 byte block on the diskette. There is space in the DTC for 8,192 diskette addresses.

If problems still exist it maybe necessary to install the VRM again.

07 Not enough space for the match list.

Cause: There was not enough memory to allocate space for the match list. See error **03** for possible causes and actions for this.

08 Relocation symbol type not supported (non-fatal)

Cause: One of the modules being processed by the loadlist processor has a relocation symbol type which is not supported. The *Virtual Resource Manager Technical Reference* describes the rules which must be followed for files with different permission bits in the loadlist and information on the TOC object module format.

The loadlist processor skips the module not supported and continues processing after displaying the error.

Action: If the VRM minidisk has been changed by adding code, find the failing module and remove it from the loadlist.

09 File type not supported (non-fatal)

Cause: The loadlist (directory name is *ldlist*) contains a file which has a file type other than the regular file or directory file. The loadlist processor skips this file and continues processing after displaying the error.

Action: If the VRM minidisk has been changed by adding additional code, remove the file in the loadlist which has an invalid file type.

10 Invalid device

Cause: The information contained in the POST Control Block for the IPL device does not contain a valid value. Since loadable POST routines that execute during IPL have access to the POST Control Block, any of these modules could have written the invalid value to this area of the POST Control Block.

Action: If the VRM minidisk has been changed by adding additional code, the invalid module must be found and removed, or install the original VRM to correct the problem.

11 Invalid index for address from inode.

Cause: This error may occur if the RT PC file system contained on the VRM minidisk (or VRM diskette) has been corrupted.

Action: If this error occurs when you IPL from the VRM minidisk, install the VRM again. If this error occurs when you IPL from diskette, use the backup diskette to IPL.

12 Inode out of range.

Cause: See error **11** for cause and action.

13 Address out of range.

Cause: See error **11** for cause and action.

14 File not found.

Cause: All files processed by the loadlist processor must be under a directory name *ldlist*. A directory file name *ldlist* was not found.

Action: If the VRM minidisk has been changed so that the directory name *ldlist* has been renamed or removed, correct the problem or install the original VRM again.

15 No freeblock header space.

Cause: The loadlist processor found an error in its internal data fields while trying to deallocate space.

Action: If the VRM minidisk has been changed by adding additional code, find and remove the module which is writing in the loadlist processor internal data fields, or install the original VRM again.

16 Freeing free space.

Cause: The loadlist processor is trying to deallocate space that is already free. This error can occur if an executed module writes over the internal data fields of the loadlist processor.

Action: See error **15** for recovery procedures.

17 Address already allocated.

Cause: The loadlist processor is trying to allocate space that has already been allocated for another module. Most modules that are loaded into memory by the loadlist processor are relocatable. Relocatable modules have an origin of X'60' (the origin of a module is information located in the TOC header of the module), and can be loaded into any free space in memory. If a module has a origin other than X'60', then an attempt is made to load the module into memory at the address represented by

the origin. If the module cannot be loaded because the required space is not free, the above error occurs.

Action: If the VRM minidisk has been changed by adding additional code, find and remove the failing module, or install the original VRM again.

18 Origin not on a word boundary.

Cause: A module which was to be loaded into memory by the loadlist processor has an origin which is not on a word boundary. All modules which are to be loaded are required to have an origin located on a word boundary.

Action: If the VRM minidisk has been changed by adding additional code, find and remove the module with the invalid origin, or install the original VRM again.

19 Invalid file name.

Cause: A file in the loadlist has a file name which is invalid. The *Virtual Resource Manager Technical Reference* describes the naming conventions for files processed by the loadlist processor.

Action: If the VRM minidisk has been changed by adding additional code, remove or rename the module with the invalid filename, or install the original VRM again.

20 Invalid IOCN in the file name (non-fatal)

Cause: The field containing the four character IOCN value is invalid for some filename in the loadlist. This field must contain a hexadecimal value from X'0001' to X'0400' for any module that has an IOCN. This field must be X'0000' for any module that does not have an IOCN.

The *Virtual Resource Manager Technical Reference* describes more information on the naming conventions for modules in the

loadlist. The loadlist processor will skip this module and continue processing after displaying the error.

Action: If the VRM minidisk has been changed by adding additional, remove or rename the module with the invalid filename.

21 Invalid IOCN in the DDS (non-fatal).

Cause: A module which has been executed by the loadlist processor (permission bits=700) has returned an entry for the Define Device Area (DDA) that has an invalid value for the IOCN in the Define Device Structure (DDS).

The loadlist processor will continue processing after displaying the error.

Action: If the VRM minidisk has been changed by adding additional code, remove the module that returned the invalid DDA entry.

22 Invalid length field in the DDA entry (non-fatal)

Cause: A module which has been executed by the loadlist processor (permission bits=700) has returned an entry for the Define Device Area (DDA) which has an invalid value in the length field. The length must be greater than 0 and less than or equal to 2048 bytes. The loadlist processor will continue processing after display the error.

Action: See error **21** for recovery.

23 Execution of loaded program failed (non-fatal)

Cause: A module which has been executed by the loadlist processor (permission bits=700) has returned an error code to the loadlist processor which indicates that execution of the program failed. The loadlist processor continues processing after displaying the error.

Action: If the VRM minidisk has been changed by adding additional code, find the module that returned the error and remove it.

24 No entry point was found.

Cause: The loadlist processor did not find any module with entry point X'1000'. This should be the VRM nucleus module (permission bits=450). This is the module the loadlist processor passes control to and must be present.

Action: If the VRM minidisk has been changed by adding additional code, the VRM nucleus may have been deleted or replaced. Make sure a module exists in the loadlist that can receive control from the loadlist processor, or install the original VRM again.

25 Bad load module (non-fatal).

Cause: A module that was to be loaded by the loadlist processor has invalid information in the TOC header. This module will not be loaded, and the loadlist processor will continue processing after displaying the error.

Action: If the VRM minidisk has been changed by adding additional code, find the module that caused the error and remove it.

Machine or program check error.

Cause: Either a hardware problem has caused a machine check, or a coding problem has caused a program check.

Action: If the VRM minidisk has been changed by adding additional code, the problem may have been caused by the code. Install the original VRM again.

If the problem was caused by a machine check and the problem occurs again, go to the *IBM RT PC Problem Determination Guide* and run hardware diagnostics.

If installing the VRM again does not correct the problem and no hardware problems are found, use your normal problem reporting procedures.

Errors Found During VRM Install

The errors described in this section are those found during the VRM install process. All of the errors are fatal except for number 45 (base device missing) and the numbers continue to blink until you re-IPL the system or take the appropriate action specified.

45 Base device missing.

Cause: There is not a functional adapter present for any of the display lposts that ran, or the keyboard is missing.

Action: Insert a "device diskette" for a display adapter that is present. Or install an adapter for one of the display lposts. Or plug in a keyboard.

46 VRM install error.

Cause: Internal problem in the VRM install code.

Action: Contact your local service representative.

47 VTCP command error

Cause: The **Inst.Batch** file on the "device diskette" has a VTCP command that is invalid.

Action: Contact the supplier of the "device diskette."

48 Define code failure.

Cause: A define code SVC was attempted for a file on the VTCP command line in the **Inst.Batch** file. The SVC failed and the code was not defined into the system.

Action: Contact the supplier of the "device diskette."

49 No space for file.

Cause: This is probably a logic error and installing more memory will not solve the problem.

Action: Contact the supplier of the "device diskette."

50 CP define device failed.

Cause: A define device SVC was attempted on the control program specified on the VTCP command line in the **Inst.Batch** file. The SVC failed and the control program was not added into the system.

Action: Contact the supplier of the "device diskette."

51 CP attach device failed.

Cause: An attach device SVC was attempted on the control program specified on the VTCP command line in the **Inst.Batch** file. The SVC failed and the control program was not attached to the virtual machine.

Action: Contact the supplier of the "device diskette."

52 Device driver define device failed.

Cause: A define device SVC was attempted on the device driver specified on the VTCP command line in the **Inst.Batch** file. The SVC failed and the device driver was not added into the system.

Action: Contact the supplier of the "device diskette."

53 VTRM define device failed.

Cause: A define device SVC was attempted on the VTRM. The SVC failed and the VTRM was not added into the system. The VTRM DDS used was that supplied by the control program.

Action: Contact the supplier of the "device diskette."

54 The VTMP could not be bound to its display.

Cause: The virtual display driver could not be linked to the virtual terminal.

Action: Contact the supplier of the "device Diskette."

55 Read error.

Cause: A read error occurred during the VRM install process.

Action: Try a backup diskette if you have one or contact the supplier of the diskette.

Appendix A. The VRM Maintenance Facility (VMF)

CONTENTS

About this Appendix	A-4
VRM Maintenance Facility (VMF) Overview	A-5
Command Format	A-6
Help Text	A-8
Messages	A-9
The Batch Facility	A-9
Commands	A-10
amd	A-10
cd	A-11
chmod	A-12
cpy	A-12

crmd	A-13
debug	A-15
dlmd	A-15
dosread	A-16
echo	A-17
exec	A-18
flck	A-19
format	A-19
hd	A-20
hdf	A-21
hfck	A-21
help	A-22
inst	A-22
instf	A-23
li	A-24
mc	A-24
mkfs	A-25
pause	A-26
pf	A-26
pwd	A-27
qi	A-27
quit	A-27
qvm	A-28
remark	A-28
rm	A-29
search	A-29
setattr	A-30
vfy	A-31
vm	A-31

About this Appendix

This appendix contains a description of the VRM Maintenance Facility (VMF) and its uses as a tool, and also a brief description of each of the commands that are available in the tool.

VRM Maintenance Facility (VMF) Overview

VMF is a standalone tool you can use to display and modify data on direct access storage — both fixed disk and diskette. The primary purpose of the tool is to help diagnose problems with the fixed disk when the problem has made the system inoperable. You can, however, also use VMF as a standalone tool to install VRM software or its associated device drivers. VMF runs as a diskette based, standalone, virtual machine. VMF is available on the VRM install diskette as an option on the install menu. To use VMF, select the Maintenance Facility on the Install menu and then enter VMF commands on the command line.

The following is a list of the commands that make up the tool:

amd	Alter disk or diskette data
cd	Change (or establish) the default device and directory for VMF
chmod	Change access permissions of an AIX file
cpy	Copy using relative sectors
crmd	Create a minidisk
debug	Invoke the VRM Debugger
dlmd	Delete a minidisk
dosread	Copy a file from a DOS diskette to an AIX file system
echo	Turn command echo on or off in batch mode
exec	Execute commands from a batch file
flck	Repair file system free block list
format	Initialize the track format on a disk
hd	Display data on a fixed disk or diskette
hdf	Display the contents of an AIX file
help	Display a list of VMF commands
hfck	Check fixed disk for bad blocks
inst	Copy a DOS file to a minidisk
instf	Copy an AIX file to a minidisk
li	List the names in an AIX file system directory
mc	Copy an entire minidisk to another minidisk
mkfs	Make an AIX file system
pause	Wait for the user to press a key in batch mode
pf	Patch a file in an AIX file system
pwd	Determine the present working directory
qi	List the minidisk IODN's on the system fixed disks

quit	Terminate the VMF
qvm	Display virtual machine information
remark	Make a batch file comment
rm	Delete a file from a file system
search	Search for a character string
setattr	Set minidisk attributes
vyf	Verify (compare) data on different devices
vm	IPL a virtual machine.

Command Format

You enter all VMF commands on the display command line (line 24). The basic command format is:

command *parm1* *parm2* ...

where **command** is the command itself and *parm1* and *parm2* are parameters. The number and types of parameters vary between the commands. The maximum number of parameters for any VMF command is 15 and the maximum length of the command with parameters and flags is 80 characters (the display line length). Flags are also available for some commands. Flags are optional and are preceded by the - (dash) character. A flag changes the way the command functions. For example, the **-p** flag on the **hd** command causes the output to appear on the printer rather than the display.

Parameters

The following are descriptions of the most common parameters you supply for the commands:

iodn

This is an input/output device number for a fixed disk, a diskette, or a minidisk:

- Fixed disk *iodn* values are 1, 2, and 3.
- Diskette *iodn* values are 4 and 5.
- Minidisk *iodn* values are in the range 16384 through 32767.

minidisk-ID

This is a minidisk identifier and can be either the minidisk iodn or the minidisk name. For example, **hd0** is the name for the minidisk containing the AIX root file system and **hd6** is the name for the minidisk containing the VRM file system.

devaddr

This is the device address and can be:

- A device iodn (fixed disk or diskette)
- A minidisk iodn
- A minidisk name.

dosname

This is a PC/DOS file name. DOS directories are not supported.

AIXname

This is an AIX file identifier and must contain a complete file system pathname beginning with the device and the directory on the device (**/hd0/usr** for example).

You can however, use the VMF **cd** command to save keystrokes by establishing a current default device and directory for VMF. For example, if you want to access AIX file **myfile** in directory **usr/bin** on the **hd0** minidisk, the complete *AIXname* is **/hd0/usr/bin/myfile**. If you had used the **cd** command and established **/hd0/usr/bin** as the current default device and directory for VMF, you could merely specify **myfile** for the *AIXname* parameter.

If VMF finds a / (slash) as the first character of *AIXname*, it assumes it to be a complete path name beginning with the *devaddr* (such as **/hd0**). If VMF does not find a / as the first character (such as **myfile**), and you have used the **cd** command to establish a default device and directory path for VMF, the name entered on the command line is appended to the default. If you have not established a default, the name entered is appended to / (slash). If you have established a current device and directory for VMF and you want to access an AIX file on another minidisk, you either have to supply the full file identifier for the other minidisk or use the **cd** command and establish a new default.

VMF does not support mounting of file systems onto the system root directory.

sec#

This is a sector number (decimal). The first sector number on a device is 0.

Syntax

The following characters that appear in some of the commands have the following special meaning:

- The characters < > surrounding a parameter indicate the parameter is optional. For example, <*parm2*> indicates that *parm2* is an optional parameter. A condition such as <*parm2* <*parm3*>> means that *parm2* and *parm3* are both optional but you can only supply *parm3* if you have also supplied *parm2*.
- The | character is the logical or symbol and means you can use one of two or more parameters but not more than one. For example, *parm1* | *parm2* means you can choose either *parm1* or *parm2* but not both.

You must enter characters as they are shown in the following descriptions and in the help text available on the display. All characters in the commands and parameters are case-sensitive except for DOS file names.

Help Text

This publication contains a brief description of each command. Help text for each command is available on the display when VMF is active. The help text also provides a description of each command and how to use it. To display the help text, enter the command name and a ? on the command line. For example,

```
d1md ?
```

displays the help text for the delete minidisk command.

Messages

VMF issues the following three types of messages:

- General syntax errors
- Specific user errors
- Messages on behalf of the VRM.

For syntax errors, VMF displays a short message that shows the correct syntax.

For specific user errors, VMF displays a message that identifies the parameter in error.

When VMF issues an SVC for the operator and an error occurs, pertinent information about the error is displayed.

The Batch Facility

The VMF batch facility allows you to run VMF commands from a prepared list in a file. All VMF commands are valid in batch mode and in addition, the following commands are specifically for the batch facility:

echo	Turn command echoing on and off, or display a message.
exec	Run a batch file.
pause	Wait for the operator to press the Enter key.
remark	Allow a comment in a batch file.

Batch Files

A batch file is a text file on a DOS diskette that contains VMF commands. When you run a batch file, you must leave the diskette in the drive until all of the commands have completed. You need an operating system such as AIX or PC DOS to create a batch file.

Batch Variables

You can supply the batch file up to 10 variables on the `exec` command. The variables are sequentially assigned the symbols `%0` through `%9` beginning with the batch file name. For example, if you enter

```
exec listit /hd0/usr/bin
```

on the command line, then symbol `%0` is assigned the variable `listit` and symbol `%1` is assigned the variable `/hd0/usr/bin`. If the batch file named `listit` contains

```
li %1
```

then the `li` command displays the contents of directory `/usr/bin` on minidisk `hd0`.

Commands

The following are descriptions of the VMF commands. A description of each command is available on the system when VMF is active by entering *command ?* on the command line.

amd

Use `amd` to modify data on fixed disks, diskettes, and minidisks.

Format:

```
amd devaddr block# offset w1 w2 . . .
```

Parameters:

devaddr

This is the device address.

block#

This is the block number (the first block number is 0).

offset

This is the hexadecimal offset into *block#*.

w1 w2 ...

These are the 4-byte words of data to be placed starting at *offset*. You can enter each word of data without leading zeros. Each word must be separated by one or more blanks.

cd

Use **cd** to establish or change the current default device and directory for VMF in order to save keystrokes when using the VMF commands.

Format:

cd *path*

Parameters:

path

This becomes the current default device and directory for VMF.

Example:

If you specify

```
cd /hd0/usr/bin
```

then **/hd0/usr/bin** becomes the VMF default path name prefix each time you supply *AIXname* (such as **myfile**) without a leading / (slash). In this case, the complete *AIXname* becomes **/hd0/usr/bin/myfile**.

chmod

Use **chmod** to change the read/write permission bits in a file inode.

Format:

chmod *OOO AIXname*

Parameters:

OOO

This is the octal representation of the 9 permission bits.

AIXname

This is the path name of the file to change.

cpy

Use **cpy** to copy blocks of data. If you copy from one minidisk to another, they must have the same logical block size.

Format:

cpy *devaddr1 block#1 devaddr2 block#2 #blocks*

Parameters:

devaddr1

This is the source device for the block data.

devaddr2

This is the target device for the block data. *devaddr2* can be the same as *devaddr1*.

block#1

This is the beginning sector number on the source device.

block#2

This is the beginning block number on the target device.

#blocks

This is the number of blocks of data to transfer.

crmd

Use **crmd** to create a minidisk on a fixed disk.

Format:

```
crmd iodn blksz #blks -dreald -nname -ssec# -attr
```

Flags:

-d

This option lets you specify the fixed disk to contain the new minidisk. If this option is not specified, the default is the first available space starting with fixed disk **hdisk0** (iodn 1). If you specify **0** for *reald*, it acts the same as the default.

-n

This option lets you specify the minidisk name up to a maximum of 4 characters. If you do not use this option you access the minidisk by *iodn* in subsequent operations.

-s

Use this option to specify the starting sector number for the new minidisk on the real device. This option should only be used when the **-d** flag is also specified.

-a

This option lets you specify the attributes for the new minidisk.

Parameters:

iodn

This is the *iodn* for the new minidisk.

blksz

This is a multiplier digit in the range of 1 through 16 that specifies a block size as a multiple of 512 bytes. For example, if you specified 1, the block size would be 512; if you specified 2, the block size would be 1024.

#blks

This is the size of the minidisk in blocks.

reald

This is the iodn of the device to contain the new minidisk.

name

This is the name of the new minidisk. The maximum number of characters is 4.

sec#

This is the starting sector number for the new minidisk on the device specified in *reald*.

attr

This is a character string containing the attributes for the new minidisk. The string can contain one or more of the characters **p**, **n**, **v**, **l**, **m**, or **h**. The characters have the following meaning:

p

This specifies the page space attribute.

n

This specifies the no bad block relocation attribute.

v

This specifies the write verify attribute for this minidisk.

l, m, h

This specifies the low, medium or high starting position for the new minidisk on the real device. These attributes are ignored if the **-s** option is specified.

debug

Use **debug** to invoke the VRM Debugger when the specified virtual machine is next dispatched. **debug** can only be used when VMF is IPL'ed directly from the fixed disk. See "instf" on page A-23 for a description of how to install VMF on the fixed disk.

Format:

debug *<vmid>*

Parameters:

vmid

This is the id, in the range 1 through 255, of the virtual machine to be debugged.

Example:

If you want to debug virtual machine 2 you enter

```
debug 2
```

The VRM Debugger is invoked when VM 2 is dispatched. It is not invoked immediately. When you get control in the debugger, it is the VM 2 environment.

dlmd

Use **dlmd** to delete a minidisk.

Format:

dlmd *minidisk_id* *<reald>*

Parameters:

minidisk_id

This is the iodn or name of the minidisk being deleted.

reald

This is the fixed disk IODN on which the minidisk resides. If you have more than one minidisk with the same name on different minidisks, you should specify this parameter.

dosread

Use **dosread** to copy a file from a DOS diskette to an AIX file system on a minidisk.

Format:

```
dosread dosname AIXname -mOOO -iname
```

Flags:

-m

Use this option to provide the permission bits for the new AIX file.

-i

Use this option to insert the new AIX file after the specified name in the target directory.

Parameters:

dosname

This is the name of the DOS file being copied including the diskette drive name (such as **b:**) if other than the first.

AIXname

This is the path name of the AIX file being created.

OOO

This is the octal representation of the 9 permission bits for the new AIX file.

name

The new AIX file is to be inserted after this name in the target directory.

Example:

If you enter

```
dosread vrmnuc /hd6/ldlist/vrmbase/vrmnuc.0001.00 -ilvrmnc.0000.00
```

on the command line, DOS file **vrmnuc** is copied from the first diskette drive to the filesystem on the **hd6** minidisk. The file is named **vrmnuc.0001.00** and it is placed after the name **lvrmnc.0000.00** in directory **/ldlist/vrmbase**.

If you enter

```
dosread b:myfile /hd0/usr/myusr/myfile -m444 -i..
```

on the command line, DOS file **myfile** is copied from the second (drive b) diskette drive to the filesystem on the **hd0** minidisk. The file is named **myfile** and it is placed after the **..** entry in directory **/usr/myusr**. The 9 access permission bits for the new AIX file are octal 444.

echo

Use this batch facility command to turn command echoing on or off, or to display a message on the display. When echoing is on, each command in the file is displayed on the screen as it starts to run.

Format:

```
echo <ON | OFF | string>
```

Parameters:**ON**

Turn command echoing on. This is the default.

OFF

Turn command echoing off.

string

Display this string of data on the screen.

Example:

The following is a batch file fragment that shows how you might use the **echo** command:

```
echo off
echo Insert diskette into drive 5
pause
.
.
.
```

exec

Use this batch facility command to run the commands in a VMF batch file. This command is usually entered on the command line, however, you can use this command in a batch file to transfer control to another file and start running the commands contained there. Control does not return to the original file.

Format:

```
exec dosname < var1 var2 . . . var9>
```

Parameters:

dosname

This is the name of the batch file containing the VMF commands.

var1 . . . var9

These are optional variables to be passed to the commands in the batch file *dosname*.

flick

Use the **flick** command to repair a file system free block list. This is useful when a file system has no available free blocks because the command may make more space available.

Format:

```
flick minidisk-id
```

Parameters:

minidisk-id

This identifies the minidisk containing the file system to repair.

format

Use the **format** command to initialize the track format of a new disk. This command is used only for disks that have not been initialized for the RT-PC.

Format:

```
format iodn
```

Parameters:

iodn

This identifies the drive to format. It can be a 1, 2 or 3 which corresponds to drives in slot C, D or E respectively.

VMF attempts to determine if the disk has any previous initialization data. If it does, it uses this data when re-formatting. If the disk was un-initialized, VMF presents a menu that allows specification of drive data. The parameters required are:

- Number of cylinders
- Number of heads
- Number of sectors
- Precompensation cylinder

-
- Average seek speed (actuator speed).

VMF hangs after a successful format. A re-IPL of the Install program initializes the new drive in the system. It is strongly recommended that you run the RT-PC diagnostics Format option on the drive before you put data on the drive. This stresses the drive in an attempt to find any defective tracks.

Note: This command destroys all information that resides on the disk including defect lists, directories and user data.

hd

Use **hd** to display data residing on a fixed disk or diskette.

Format:

```
hd devaddr sec#1 sec#2 -p
```

Flags:

-p

Use this option to send the data to the printer instead of the display.

Parameters:

devaddr

This specifies the device that contains the data.

sec#1

This specifies the first sector of data to be displayed.

sec#2

This specifies the last sector of data to be displayed.

hdf

Use **hdf** to display the contents of an AIX file.

Format:

```
hdf AIXname <fromoffset <tooffset>> -p
```

Flags:

-p This option sends the data to the printer instead of the display.

Parameters:

AIXname

This is the path name of the AIX file.

fromoffset

This optional parameter specifies the offset (hexidecimal) from the beginning of the file to start displaying data. If this parameter is not specified, the default is 0.

tooffset

This optional parameter (if *fromoffset* is specified) provides the offset (hexadecimal) from the beginning of the file to quit displaying data. If not specified, 128 bytes are displayed.

hfck

Use the **hfck** command to check a fixed disk bad block relocation area for bad blocks detected by the system during a read operation. This function may be able to relocate these bad blocks successfully. This is useful when a file system has no available free blocks because the command may make more space available. This command is available only when VMF is diskette based. VMF hangs after this command completes and a message is displayed telling you to restart the system.

Format:

hfck *reald* | * -v

Parameters:

reald

This identifies the fixed disk input/output device number to repair. If '*' is specified, all fixed disks are checked.

Flags:

-v

When this option is specified, VMF determines the file names in which unrelocated bad blocks reside.

help

Use the **help** command without parameters to display the complete list of VMF commands. To get help for a particular command, enter the command name followed by a ? (question mark).

Format:

help

inst

Use the **inst** command to copy a DOS file to a minidisk. The file is copied sequentially starting at sector 0 of the minidisk. The previous contents of the minidisk are destroyed.

Format:

inst *dosname minidisk_id*

Parameters:

dosname

This is the name of the DOS file to be transferred and must contain the diskette drive identifier (such as **b:**) if other than the first drive.

minidisk_id

This identifies the minidisk to receive the file.

instf

Use the **instf** command to copy an AIX file to a minidisk. The file is copied sequentially starting at sector 0 of the minidisk. The previous contents of the minidisk are destroyed.

Format:

instf *AIXname minidisk_id*

Parameters:

AIXname

This is an AIX file identifier and must contain a complete file system pathname beginning with the device and the directory on the device (**/hd0/usr** for example).

minidisk_id

This identifies the minidisk to receive the file.

You can use this command to install VMF directly on a minidisk. To do so, make sure the VRM Install diskette is in the top drive and then issue the following commands:

```
crmd iodn 550 -nvmf
instf /4/1dlist/vrinst.0050.00 vmf
```

and then use the **setattr** command to make VMF an auto-IPL minidisk.

li

Use **li** to display the contents of an AIX directory. The entries are displayed in the order they physically exist in the directory.

Format:

li *<AIXname>* **-p**

Flags:

-p This flag sends the output to the printer instead of the display.

Parameters:

AIXname

This is the path name of the directory to be listed. If not specified, the current directory is displayed.

mc

Use **mc** to copy the entire contents of one minidisk to another. Both minidisks must have the same block size but they can differ in size. The data is truncated if the target minidisk is smaller than the source minidisk. In this case, an informational message is issued but the copy proceeds. If the target is larger than the source, the extra space is unchanged.

Format:

mc *minidisk-id1 minidisk-id2*

Parameters:

minidisk-id1

This is the source minidisk.

minidisk-id2

This is the target minidisk.

mkfs

Use **mkfs** to create an AIX file system on a minidisk or diskette.

Format:

```
mkfs minidisk-id | diskette-iodn <#blocks> <#iblocks>
```

Parameters:

minidisk-id

This identifies the target if it is a minidisk.

diskette-iodn

This identifies the target if it is a diskette.

#blocks

This is the size of the new file system (the default is 540):

Minidisk — This is the number of 2K blocks for the file system. Because there is an underlying 512-byte sector size, this number should not be larger than the number of sectors on the minidisk divided by 4.

Diskette — For a diskette, this is the number of 512-byte blocks for the file system.

#iblocks

This is the number of logical blocks in the I list (the default is 5):

Minidisk — This is the number of 2K blocks in the I list.

Diskette — This is the number of 512-byte blocks in the I list.

pause

Use this batch facility command to wait until the operator presses the Enter key. When the batch facility encounters this command in a batch file, it displays the message `Press Enter to continue` and then waits for the keystroke. There are no parameters for this command.

Format:

pause

pf

Use **pf** to change data in an AIX file.

Format:

pf *AIXname offset olddata newdata newdata . . .*

Parameters:

AIXname

This is the path name of the file being changed.

offset

This is the hexadecimal offset from the beginning of the file to start changing data. This offset must be on a word boundary (divisible by 4).

olddata

This is the first word of data currently stored at *offset* in the file. You do not have to supply leading zeros. If *olddata* does not match the word of data at *offset*, VMF does not change the file.

newdata . . .

These are the words of data to be placed in the file beginning at *offset*. Note that only the first word is verified.

pwd

Use this command to determine the present working directory.

Format:

pwd

qi

Use **qi** to display a list of minidisks defined on the fixed disks. If you do not specify one or more minidisk IDs, all the minidisks are listed. If you specify 0 for one of the minidisk IDs, you get a list of all fixed disk free space.

Format:

qi <*minidisk-id1 minidisk-id2 . . .*> **-p**

Flags:

-p

This flag sends the output to the printer instead of the display.

Parameters:

minidisk-id

This is the ID of the minidisks to be listed. The default is all minidisks.

quit

Use this command to terminate the VMF.

Format:

quit

qvm

Use the **qvm** command to display information about the virtual machines running in the system. The command displays the VMID and IPL IODN for the VMID requested on the command line.

Format:

qvm *<vmid>*

Parameters:

vmid

This is a virtual machine identifier for which you want information. If not specified, information on all virtual machines running in the system is displayed.

remark

Use **remark** to put comments in batch files. The comment is treated as a null command.

Format:

remark *comment*

Parameters:

comment

This is any comment that fits on one line.

rm

Use **rm** to remove a file from an AIX file system. You can remove files and directories. However, the directories must be empty. An error message is the result if you attempt to remove a directory that is not empty.

Format:

rm *AIXname*

Parameters:

AIXname

This is the path name of the AIX file or directory to be removed.

search

Use the **search** command to search for a character string on an AIX minidisk or in an AIX file.

Format:

search *AIXname* | *minidisk_iodn string -v*

Flags:

-v

You may use this option when searching an entire minidisk for a string. Additional information is given for each occurrence of the string on the minidisk.

Parameters:

AIXname

This is an AIX file identifier and must contain a complete file system pathname beginning with the device and the directory on the device (*/hd0/usr* for example).

minidisk-iodn

This is a minidisk, input/output device number. Specify a minidisk IODN when you want to search an entire minidisk for occurrences of a string. Minidisk names cannot be used because minidisk names and AIX names are ambiguous.

string

This is the search argument. The search command searches for ascii strings or hexadecimal strings. Ascii strings are specified by starting and ending the string argument with a single quote character. To search for a single quote, put two single quote characters together in the string. For example you would specify

```
search . . . 'don't'
```

to search for the word don't. String arguments that do not begin with a single quote character must be valid hexadecimal numbers.

setattr

Use this command to set on or set off minidisk attributes as defined in the *VRM Technical Reference Manual*.

Format:

```
setattr minidisk-id
```

Parameters:

minidisk-id

This identifies the minidisk.

This command results in two menus each of which has a numbered list of minidisk attributes. Use the first menu to set on desired attributes; use the second menu to set off desired attributes. If you press **Enter** for either menu without entering any attribute numbers, no attributes are changed for that menu.

vyf

Use **vyf** to verify that two areas of data are identical. If the areas are on different minidisks, the minidisks must have the same logical block size.

Format:

```
vyf devaddr1 sec#1 devaddr2 sec#2 #sectors -p
```

Flags:

-p
Send messages to the printer instead of the display.

Parameters:

devaddr1 devaddr2

This identifies the devices that contain the data. These parameters can be the same.

sec#1

This is the starting sector number on *devaddr1*.

sec#2

This is the starting sector number on *devaddr2*.

#sectors

This is the number of sectors to verify.

vm

Use **vm** to IPL a virtual machine.

Format:

```
vm minidisk_id <vmid>
```

Parameters:

minidisk-id

This identifies the minidisk containing the virtual machine.

vmid

This is an optional parameter that specifies a VM number for the machine being IPLed. If omitted, the next available VMID is used.

Appendix B. Error Log Entries

CONTENTS

About This Appendix	B-3
Error Log Entry Types	B-3
Subclass = 02 Abnormal Termination (ABEND)	B-6
Subclass = 03 AIX Program Error	B-7
Subclass = 04 AIX Device Driver Program Error	B-7
Subclass = 07 VRM Component Program Error	B-9
Subclass = 08 Application Program Error	B-23

About This Appendix

This appendix lists some of the software entries that you could receive. There is no action you can take with most of the possible error log entries except to report the problem. However, it is thought that with the entries listed here, you might have caused the problem with your program or your operation and you might be able take action to further diagnose and solve the problem. Because of the nature of software problems, none of the actions (when one is offered) are guaranteed, and the list is not guaranteed to be complete. And just because an entry is listed here, it doesn't guarantee the problem was caused by you or your program and not by the system software.

Error Log Entry Types

Figure B-1 on page B-4 is a list of the different error log types.

Class	Type	Subclass	Mask	Subclass Type
01	Hardware	all	all	
02	Software	01		ROMP program check
		02		ABEND
			01	Dump taken
			02	No dump taken
		03		AIX Program Error
			03	AIX kernel
		04		AIX device driver program error
			01	Base LAN device driver
			02	5080 peripherals
			03	SSLA device driver
		05		AIX device driver program error
		06		VRM base program error
		07		VRM component program error
			02	Virtual terminal manager
			03	PC/AT coprocessor driver
			04	5080 display driver
			05	5080 peripherals device driver
			06	Generic block I/O device driver
		08		Applications
			01	Error log analysis
			02	Interactive workstations
			03	Base LAN
			04	Coprocessor control program
		09 through 0F		Customer Applications

Figure B-1 (Part 1 of 2). Error Log Entry Types

03	IPL/Shutdown	01	Manual IPL
		02	Soft IPL
		03	Auto IPL
		04	Shutdown
		05	Maintenance shutdown
04	General System Condition	01	Degraded configuration
		02	Set date/time
		03	Server no start
		04	Cause codes
06	User defined	All	

Figure B-1 (Part 2 of 2). Error Log Entry Types

The following are descriptions of some error log entries. Some of the descriptions provide actions you can take in an attempt to eliminate or further diagnose the problem. An attempt has been made to list only the error log entries that can be caused by external actions or application programming. If a description found here is not helpful in resolving the problem, or if there is no description here for an entry you feel is pertinent to your problem, then document the problem and submit it to your service person.

All of the entries are software and are listed by subclass and mask. Within each mask category, the entries are distinguished by module and by return code.

Subclass = 02 Abnormal Termination (ABEND)

Mask = 00 VRM ABEND Message

A machine check has occurred.

A ROMP machine check occurred that was probably caused by a hardware failure. See *Hardware Technical Reference* for definitions of machine check status.

An IAR protection exception has occurred.

An attempt was made to execute an instruction in unauthorized memory.

A page fault occurred that could not be processed by the VRM.

A page fault occurred but the VRM was not in an appropriate state to process it. The most common cause is a page fault in an interrupt handler.

All paging space is in use.

The paging space is full. Use the VRM installation utility to increase the size of the paging space minidisk.

The VRM does not have enough real memory to continue.

A process or virtual machine tried to pin too much memory.

The VRM has encountered initialization data that is not valid.

Invalid data was passed to the VRM by the loadlist processor or one of the loadable posts.

The VRM is unable to allocate a system control block.

Either too many control blocks are in use or no more real memory is available.

An I/O channel error has occurred.

This error is probably caused by a bad device on the bus.

Subclass = 03 AIX Program Error

Mask = 03 AIX Kernel

The AIX kernel issues messages that are displayed on the console. These messages appear in the error log and are documented in *Messages Reference*.

Subclass = 04 AIX Device Driver Program Error

Mask = 01 Base LAN Device Driver

RC=(errno) vec-init call

There are no more interrupt sublevels available for the device. Too many devices are using the same interrupt level.

If you created some of your own devices, move some of the devices to another interrupt level and restart the machine. See *AIX Operating System Technical Reference* under Creating Device Drivers for information about the `vec-init` call.

RC=16 ATTACH SVC

The ATTACH SVC failed.

See *Virtual Resource Manager Technical Reference* for an explanation of the return code. Possible reasons and actions to take:

-
- The device was not installed properly. Reinstall the Base PC Network Services program and try again.
 - Another device in AIX has attached to the adapter card. Only one attach is allowed per virtual machine.

RC=16 DETACH SVC

The DETACH SVC failed.

See *Virtual Resource Manager Technical Reference* for information about return codes. You may have to shut down and restart the machine in order to put the network device in a known state.

RC=256 SIO - ncb command = 05

An invalid **ncb** command was issued in the **ncb** passed to the device.

Verify that the program using the **ncb** is using a valid **ncb** command value. Use the trace facility to determine what command value was passed.

RC=257 SIO - ncb command = B0

An invalid parameter was passed to VRM in the Start I/O SVC call.

If the error persists, reinstall the Base PC Network Services program to replace the required files.

RC=258 SIO - ncb command = 35

An attempt was made to cancel an **ncb** command that is not valid to cancel.

If the error occurred in your program, verify that the program only tries to cancel **ncb** commands that are valid as described in *Base PC Network Services*. If the error occurred in the AIX kernel cleanup routine, ensure that your program does not close the device with outstanding **ncb**'s issued.

RC=259 SIO - ncb command = B0

There were insufficient resources on the adapter card when the **ncb** was issued.

Issue the **ncb** again because the resources may now be free. If the condition persists, verify that the number of **ncb**'s allowed for the adapter card is sufficient for the number of users for the card and **ncb**'s issued. Use the **devices** command to check the current value and change it if necessary.

RC=260 SIO - ncb command = 30

An **ncb** was issued while the adapter card was being reset.

Verify that other users or programs on the RT PC do not issue **ncb** reset commands. This command clears the adapter status, name tables, and session tables. The trace facility may show the reset command occurring and may help in locating the program that issued the reset. The program running should be restarted.

Subclass = 07 VRM Component Program Error

Mask = 02 VRM component - Keyboard, Locator, and Sound Device Driver

Module-ID = KDDMAIN RC = 03 D01 = 01

The main entry point was called for a **DEFINE DEVICE** request with an invalid IODN.

Module-ID = KDDMAIN RC = 03 D01 = 02

The main entry point was called for an **initialization** request with an invalid IODN.

Module-ID = KDDMAIN RC = 03 D01 = 03

The main entry point was called for a terminate device request with an invalid device ID.

Module-ID = KDDMAIN RC = 02 D01 = 04

The main entry point was called with an invalid operation type.

Module-ID = KDDMAIN RC = 01 D01 = 05

The keyboard initialization routine called **-CHANGE** and received a non-zero return code.

Module-ID = KDDMAIN RC = 01 D01 = 06

The keyboard initialization routine called **-CHANGE** and received a non-zero return code.

Module-ID = KDDMAIN RC = 04 D01 = 07

The keyboard initialization encountered a DDS specifying that the device was not working at the time of the LPOST.

Module-ID = KDDMAIN RC = 01 D01 = 08

The locator initialization routine called **-CHANGE** and received a non-zero return code.

Module-ID = KDDMAIN RC = 01 D01 = 09

The locator initialization routine called **-CHANGE** and received a non-zero return code.

Module-ID = KDDMAIN RC = 04 D01 = 10

The locator initialization encountered a DDS specifying that the device was not working at the time of the LPOST.

Module-ID = KDDMAIN RC = 01 D01 = 11

The sound initialization routine called **-CHANGE** and received a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 01 (second level interrupt handler)

-DEQUE was called to remove a **CLICK** element from the keyboard queue and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 02 (second level interrupt handler)

-DEQUE was called to remove a **VOLUME** element from the keyboard queue and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 03 (second level interrupt handler)

-DEQUE was called to remove a **KEYSTROKE** element from the keyboard queue and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 04 (second level interrupt handler)

-DEQUE was called to remove a **LOCATOR** element from the locator queue and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 05 (second level interrupt handler)

-DEQUE was called to remove a **SOUND** element from the sound queue and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 06 (second level interrupt handler)

-ENQ was called to enqueue a **KEYSTROKE** element and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 07 (second level interrupt handler)

-ENQ was called to enqueue a **LOCATOR** element to the mode processor and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 08 (second level interrupt handler)
-ENQ was called to enqueue a LOCATOR element to the screen manager and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 09 (second level interrupt handler)
-DEQUE was called to remove a KEYSTROKE element from the keyboard queue and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 10 (second level interrupt handler)
-DEQUE was called to remove a LOCATOR element from the locator queue and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 11 (second level interrupt handler)
-ENQ was called to enqueue a LOCATOR element to the mode processor and returned a non-zero return code.

Module-ID = KDDINTR RC = 01 D01 = 12 (second level interrupt handler)
-ENQ was called to enqueue a LOCATOR element to the screen manager and returned a non-zero return code.

Module-ID = KDDIOIN RC = 01 D01 = 01 (I/O initiate)
-DEQUE was called to remove an element and returned a non-zero return code.

Module-ID = KDDIOIN RC = 01 D01 = 03 (I/O initiate)
-DEQUE was called to remove an element and returned a non-zero return code.

Module-ID = KDDIOINL RC = 01 D01 = 01 (Locator I/O initiate)

-DEQUE was called to remove an element and returned a non-zero return code.

Module-ID = KDDIOINL RC = 01 D01 = 02 (Locator I/O initiate)

-DEQUE was called after activation of a virtual terminal and returned a non-zero return code.

Module-ID = KDDIOINL RC = 01 D01 = 03 (Locator I/O initiate)

-ENQ was called after activation of a virtual terminal and returned a non-zero return code.

Module-ID = KDDIOINS RC = 01 D01 = 01 (Sound I/O initiate)

-DEQUE was called to remove an element and returned a non-zero return code.

Module-ID = KDDIOINS RC = 01 D01 = 02 (Sound I/O initiate)

-DEQUE was called to remove an element and returned a non-zero return code.

Mask = 02 VRM component - Virtual Terminal Mode Processor

Module-ID = VTMMAIN RC = 0 D01 = 07

The virtual terminal mode processor received a queue element that it does not recognize.

Module-ID = VTMOUO RC = 6521

The length of the VTD header plus the minor data is greater than the total VTD length.

Module-ID = VTMOUO RC = 6522

The virtual mode processor received an invalid major type in the VTD header.

Module-ID = VTMOU RC = 6523

Some of the data within the VTD block is not valid.

Module-ID = VTMOU RC = 6524

The virtual mode processor received an invalid minor type in the VTD header.

Module-ID = VTMOU RC = 6527

The 4-byte length in the VTD header is larger than X'00FFFFFF'.

Module-ID = VTMOU RC = 6528

The 4-byte length field is less than X'06'.

Module-ID = VTMOU RC = 6532

This error occurred during a request to change thresholds and a tablet is present instead of a mouse, or during a request to change dead zones and a mouse is present instead of a tablet.

Module-ID = VTMOU RC = 6533

During a change fonts command, some of the requested fonts do not exist.

Module-ID = VTMOU RC = 6534

During a change fonts command, the requested fonts have different sizes.

Module-ID = VTMOU RC = 6537

This error occurred during a request to change indicators and the LPFKs are not attached, or during a request to change granularities and the dials are not attached.

Module-ID = VTMOU RC = 6544

The received VTD command is not valid in the current operating mode of the virtual terminal mode processor.

Module-ID = VTMOUT RC = 6546

The terminal ID passed via the change display or the change font command is not valid.

Module-ID = VTMSTS RC = 6563

This error occurred when set keyboard map tried to change a key to a function with a function ID greater than X'1FF' or to an unsupported ID.

Module-ID = VTMSTS RC = 6564

This error occurred when set keyboard map tried to re-map one of the screen manager designated keys.

Module-ID = VTMSTS RC = 6566

This error occurred when set keyboard map tried to re-map a key with an invalid position code.

Module-ID = VTMSTS RC = 6567

This error occurred because the KSR buffer definition header is too small.

Module-ID = VTMSTS RC = 6568

This error occurred because the keyboard buffer is too small.

Module-ID = VTMQRY RC = 6522

This error occurred because the virtual terminal mode processor received a query SVC with an invalid major type.

Module-ID = VTMQRY RC = 6524

This error occurred because the virtual terminal mode processor received a query SVC with an invalid minor type.

Module-ID = VTMQRY RC = 6538

This error occurred because the queried device is not configured.

Module-ID = VTMQRY RC = 6539

This error occurred because the queried device has not been selected.

Module-ID = VTMQRY RC = 6548

This error occurred because the query ASCII codes SVC designated an area with invalid coordinates.

Module-ID = VTMQRY RC = 6580

This error occurred because either the query response buffer is not large enough for all the queried data, or because invalid parameters exist in the query SVC command.

Module-ID = VTMQRY RC = 6581

This error occurred when the queried data overflowed the response buffer. The partial data remains in the query response buffer.

Module-ID = VTMQRY RC = 6583

This error occurred when the query real device was issued for an invalid device ID.

Module-ID = VTMUPD RC = 6549 D01 = 03

This error occurred because the virtual terminal mode processor received an **ERASE LINE, FIELD, or AREA** command with an invalid parameter.

Module-ID = VTMUPD RC = 6549 D01 = 04

This error occurred because the virtual terminal mode processor received an **ERASE DISPLAY** command with an invalid parameter.

Module-ID = VTMUPD RC = 6549 D01 = 05

This error occurred because the virtual terminal mode processor received a **SET MODE** or **RESET MODE** control sequence with an invalid parameter.

Module-ID = VTMUPD RC = 6549 D01 = 06

This error occurred because the virtual terminal mode processor received a **SET GRAPHIC RENDITION** control sequence with an invalid parameter.

Module-ID = VTMUPD RC = 6549 D01 = 07

This error occurred because the virtual terminal mode processor received a **SET CHARACTERSET** control sequence with invalid parameters.

Module-ID = VTMUPD RC = 6550

This error occurred because the virtual terminal mode processor detected an unsupported escape sequence.

Mask = 02 VRM component - Virtual Terminal Resource Manager

Module-ID = VTRUTLPF RC = 01

The resource controller tried to add a font to the font table in the resource structure and the IOCN for the font is unknown to the VRM.

Module-ID = VTRUTLPF RC = 02

The resource controller tried to add a font to the font table in the resource structure and either the font module symbol is not exported with the correct name or the font structure is incorrect.

Module-ID = VTRUTLPF RC = 03

The resource controller tried to add a font to the font table in the resource structure and the ID of the storage segment containing the font was unknown to the VRM.

Module-ID = VTRUTLBC RC = 04

The resource controller tried to bind the common device utilities to the virtual terminal mode processor and either one or more functions are not imported with the correct names or the parameter lists do not match.

Module-ID = VTRUTLNP RC = 05

The resource controller tried to derive requisite information for an input device and the IODN identifying the device driver is unknown to the VRM.

Module-ID = VTRUTLNP RC = 06

The resource controller tried to derive requisite information for an input device and received an unexpected VRM service return code. The return code is in D02.

Module-ID = VTRUTLNP RC = 07

The resource controller tried to derive requisite information for an input device and the IOCN of the device driver (derived from the define device structure of the device) is unknown to the VRM.

Module-ID = VTRUTLNP RC = 08

The resource controller tried to derive requisite information for an input device and either the IOCN of the device driver exported a structure different from that expected by the resource controller, or the names mismatch.

Module-ID = VTRUTLNP RC = 09

The resource controller tried to derive requisite information for an input device and either the IOCN of the device driver exported a structure different from that expected by the screen manager, or the names mismatch.

Module-ID = VTRUTLND RC = 10

The resource controller tried to derive requisite information to enter a display into the real screen table of the resource structure and the IODN identifying the device driver is unknown to the VRM.

Module-ID = VTRUTLND RC = 11

The resource controller tried to derive requisite information to enter a display into the real screen table of the resource structure and the IOCN of the

device driver (derived from the define device structure of the device) is unknown to the VRM.

Module-ID = VTRUTLND RC = 12

The resource controller tried to derive requisite information to enter a display into the real screen table of the resource structure and either the VDD expected a structure different from the one received, or the names mismatch.

Module-ID = VTRUTLND RC = 13

The resource controller tried to derive requisite information to enter a display into the real screen table of the resource structure and received an unexpected return code from **-COPY**.

Module-ID = VTRUTLND RC = 26

The resource controller tried to derive requisite information to enter a display into the real screen table of the resource structure and either the VDD does not have all of the required entry points, or the names mismatch.

Module-ID = VTRUTLND RC = 27

The resource controller tried to derive requisite information to enter a display into the real screen table of the resource structure and the attempt failed. That error was previously logged. This error occurred when the resource controller received an unexpected return code from **-COPY** while recovering resources from the previous error. The return code is in D02.

Module-ID = VTRRCICI RC = 34

During initialization, the resource controller tried to copy the module ID for various VTM components and received an unexpected VRM service return code from **-COPY**. The return code is in D02.

Module-ID = VTRRCIPR RC = 36

During initialization, the resource controller tried to set up the keyboard device and found that no keyboard is supplied in the resource controller define device structure.

Module-ID = VTRRCIPR RC = 37

During initialization, the resource controller tried to set up a display and found that none are supplied in the resource controller define device structure.

Module-ID = VTRRCIPR RC = 38

During initialization, the resource controller tried to set up a display and found that no fonts are available for that display.

Module-ID = VTRRCIPR RC = 39

During initialization, the resource controller tried to get the device ID of the sound device but the sound device driver IODN is unknown to the VRM.

Module-ID = VTRRCICS RC = 40

During initialization, the resource controller tried to bind the screen manager to the common device utilities and found that either the screen manager expected an entry point different than the one the resource controller exports or that the names mismatch.

Module-ID = VTRVTSV RC = 62

The resource controller tried to get the module ID of the replacement for the standard virtual terminal mode processor but the IOCN is unknown to the VRM.

Module-ID = VTTGADD RC = 14

An unexpected return code was received from -ENQ while attempting to update the valuator dial granularities with the requesting virtual terminal active.

Module-ID = VTTGADD RC = 15

An unexpected return code was received from **-ENQ** while attempting to update the LPFK key settings with the requesting

Module-ID = VTTSELD RC = 02

An error occurred during a **-BIND** of the virtual display driver to the the virtual terminal mode processor. It was caused by one of the following:

- One or more functions or structures are not imported with the correct names.
- The parameter lists do not match.
- The structure declarations do not match.

Module-ID = VTTSELD RC = 03

An error occurred during a **-BIND** of the virtual display driver. This error usually indicates that the parameter list declarations for the imported and exported symbol do not match. This error can cause unpredictable results.

Module-ID = VTTSELD or VTTQRD RC = 05

The size of the define device structure for the requested real display device is larger than the implementation limit for these modules.

Module-ID = VTTSELD or VTTQRD RC = 06

An error occurred while obtaining the define device structure. The return code from **-QRYDS** is in D03.

Module-ID = VTTSELD RC = 08

An unexpected return code was received from **-ENQ** while attempting to update the keyboard LEDs. The return code is in D03.

Module-ID = VTTSELI RC = 09

An unexpected return code was received from **-ENQ** while updating the keyboard mode. The return code is in D03.

Module-ID = VTTSELI RC = 10

An unexpected return code was received from **-ENQ** while updating the locator thresholds. The return code is in D03.

Module-ID = VTTSELD RC = 11

This error occurred while binding virtual terminal manager component modules to each other. The error is usually caused by one of the following:

- One or more functions or structures do not match.
- The structure declarations do not match.
- The parameter lists do not match.

Module-ID = VTTSELD RC = 12

An unexpected return code was received from **-COPY** while copying the code module for the virtual device driver. The return code is in D04.

Mask = 03 PC/AT Coprocessor Driver

Module-ID = CPDSLH RC=0

I/O Channel Check — 286 was BUS MASTER — This channel check is usually caused by a parity error on the 512K bus memory card that could indicate a bad module on the board.

A temporary solution is to remove the bus memory card and run the Coprocessor using ROMP system memory.

Module-ID = PCCDMUL8 RC=-1 -STDMA

Nonzero return code from **-STDMA** — This error indicates that an invalid address or word count was written to the DMA controller by the application running on the Coprocessor.

Attempt to determine what the DMA operation in the application is doing and correct the invalid address or word count.

Module-ID = PCCVMAIN RC=-1 -MAPSYS

Mapping of system memory failed for relocated video buffer or queue — This error indicates there was insufficient memory available for the MAPSYS operation.

You should be able to request less memory for the Coprocessor when the PCSTART command is issued. For example, 512K instead of 640K.

Module-ID = PCCVMAIN RC=-1 -PINPG

Pin page range failed — This indicates that there was insufficient memory available for the PINPGS operation.

You should request less memory for the Coprocessor when the PCSTART command is issued. For example, 512K instead of 640K.

Subclass = 08 Application Program Error

Mask = 02 Interactive Workstation

Module-ID = ATE RC=1

The send routine logs this error during a pacing file transfer if no pacing character has been received for 30 seconds.

Verify that the pacing protocol and pacing character at the remote site match those at your site. Then restart the file transfer.

Module-ID = ATE RC=2

The send routine logs this error at the start of a pacing file transfer if no pacing character is received for 100 seconds. The receiving site must transmit a specified pacing character to indicate that it is ready to receive.

Verify that both the local and remote sites are using the character pacing protocol and that the character chosen for pacing is the same for both sites. Then restart the file transfer.

Module-ID = ATE RC=3

The **receive** routine logs this error during an **XMODEM** file transfer after nine transmission errors on the same sector have occurred.

If you are using a telephone connection, terminate the current session and try again in a few minutes to get a new, less noisy line.

Module-ID = ATE RC=4

The **send** routine logs this error during an **XMODEM** file transfer when it has attempted to send a sector nine times, but the receiving site has not acknowledged receipt of the data.

Verify that both sites are using the **XMODEM** file transfer protocol. Restart the remote site and be sure that it is ready to receive. Select **send** and try the file transfer again.

Module-ID = ATE RC=5

The **send** routine logs this error during an **XMODEM** file transfer when it has attempted to send an EOT (end of transmission) nine times but the receiving site has not acknowledged receipt of the EOT.

See if the file has been received on the remote end. If it has, no further action is necessary. If it has not, restart the remote site and then select **send** and try the file transfer again.

Module-ID = 6 RC=6

The **send** routine logs this error when beginning an **XMODEM** file transfer if no start character is received for 100 seconds. The receiving site must

transmit a specified start character to indicate that it is ready to receive.

Verify that both the local and remote sites are using the **XMODEM** protocol and that the receiving site is ready to receive. Then restart the file transfer.

Module-ID = ATE RC=7

The **receive** routine logs this error when no data arrives in 30 seconds for a pacing file transfer or 100 seconds for an **XMODEM** file transfer.

Verify that both the local and remote sites are using the same file transfer protocol and that the remote site is set up to send. Then select **receive** and try the file transfer again.

Module-ID = ATE RC=8

The session has been disconnected because the system can no longer detect a carrier signal.

A break in the connection may have occurred because of one of the following:

- The telephone line has problems.
- The remote system went out of operation.
- One of the modems became disconnected.
- A wire has come loose.

Attempt to reestablish the connection.

Module-ID = ATE RC=9

The **receive** routine logs this error during and **XMODEM** file transfer when the checksum value it calculates for the sector just received does not match the checksum value calculated and sent by the remote site. This error usually occurs because of noise on the transmission line.

No action is necessary. The sector is automatically requested again up to nine times. If the errors prevent the file transfer from continuing and you are

using a telephone connection, try terminating the current session and establishing a new connection that may be less noisy.

Module-ID = ATE RC=10

The receive routine logs this error during an **XMODEM** file transfer when the same sector is received twice.

No action is necessary. The duplicate sector is automatically discarded and file transfer continues.

Module-ID = ATE RC=11

The receive routine logs this error during an **XMODEM** file transfer when it receives a sector number that is not the expected sector number.

No action is necessary. The expected sector is automatically requested again and file transfer continues.

Module-ID = ATE RC=12

The receive routine logs this error during an **XMODEM** file transfer when the sector number and the sector number complement sent by the remote site do not match.

No action is necessary. The sector is automatically requested again and file transfer continues.

Mask = 03 Base Local Area Network (LAN)

Module-ID = pcrctrl RC=2 OPEN

The **OPEN** system call to the network device failed.

The error may be because the device, **/dev/network**, does not exist. Reinstall the Base PC network program and try again. See *AIX Operating System Technical Reference* for the **OPEN** system call.

Module-ID = pcrctl RC=2 IOCTL

The **IOCTL** system call to the network device failed.

Restart the RT PC. See *AIX Operating System Technical Reference* for the **IOCTL** command.



Glossary

application. A particular task, such as inventory control or accounts receivable.

batch processing. A processing method in which a program or programs process records with little or no operator action. This is a background process. Contrast with *interactive processing*.

command. A request to perform an operation or execute a program. When parameters, arguments, flags, or other operands are associated with a command, the resulting character string is a single command.

configure. To describe (to the system) the devices, optional features, and program products installed on a system.

crash. An unexpected interruption of computer service, usually due to a serious hardware or software malfunction.

cursor. (1) A movable symbol (such as an underline) on a display, used to indicate to the operator where the next typed character will be placed or where the next action will be directed.

customize. To describe (to the system) the devices, programs, users, and user defaults for a particular data processing system.

debug. (1) To detect, locate, and correct mistakes in a program. (2) To find the cause of problems detected in software.

device. An electrical or electronic machine that is designed for a specific purpose and that attaches to your computer, for example, a printer, plotter, disk drive, and so forth.

diagnostic. Pertaining to the detection and isolation of an error.

diagnostic aid. A tool (procedure, program, reference manual) used to detect and isolate a device or program malfunction or error.

diagnostic routine. A computer program that recognizes, locates, and explains either a fault in equipment or a mistake in a computer program.

disk. A storage device made of one or more flat, circular plates with magnetic surfaces on which information can be stored.

disk drive. The mechanism used to read and write information on disk.

diskette. A thin, flexible magnetic plate that is permanently sealed in a protective cover. The diskette is used to store information copied from the disk.

diskette drive. The mechanism used to read and write information on diskettes.

display screen. The part of the display device that displays information visually on a luminescent screen by use of controlled electron beams.

dump. (1) To copy the contents of all or part of storage, usually to an output device.
(2) Data that has been dumped.

edit. (1) To modify the form or format of data.

editor. A program used to enter and modify programs, text, and other types of documents.

feature. A programming or hardware option, usually available at an extra cost.

fixed disk. See disk drive.

foreground. A mode of program execution in which the shell waits for the program specified on the command line to complete before returning your prompt.

format. (1) A defined arrangement of such things as characters, fields, and lines, usually used for displays, printouts, or files. (2) The pattern which determines how data is recorded.

formatted diskette. A diskette on which control information for a particular computer system has been written but which may or may not contain any data.

hardware. The equipment, as opposed to the programming, of a computer system.

history file. A file containing a log of system actions and operator responses.

initial program load (IPL). The process of loading the system programs and preparing the system to run jobs. See *initialize*.

label. The name in the disk or diskette volume table of contents that identifies a file. See also *file name*.

licensed program product (LPP). Software programs that remain the property of the manufacturer, for which customers pay a license fee.

load. (1) To move data or programs into storage. (2) To place a diskette into a diskette drive. (3) To insert paper into a printer.

log. To record; for example, to log all messages on the system printer.

memory. A device or part of a device that can retain data.

menu. A displayed list of items from which an operator can select tasks.

problem determination. The process of identifying why the system is not working. Often this process identifies programs, equipment, data communications facilities, or user errors as the source of the problem.

problem determination procedure. A prescribed sequence of steps aimed at recovery from, or circumvention of, problem conditions.

profile. (1) A file containing customized settings for a system or user (2) Data describing the significant features of a user, program, or device.

prompt (n.). A displayed request for information or operator action.

recovery procedure. (1) An action performed by the operator when an error message appears on the display screen. Usually, this action permits the program to continue or permits the operator to run the next job. (2) The method of returning the system to the point where a major system error occurred and running the recent critical jobs again.

routine. A set of statements in a program causing the system to perform an operation or a series of related operations.

run. To cause a program, utility, or other machine function to be performed.

setup. (1) The process or act of preparing and adjusting a machine or device for its assigned work. (2) The way in which the system unit and the external devices are arranged.

shell. See *shell program*.

shell program. A program that accepts and interprets commands for the operating system (there is an AIX shell program and a DOS shell program).

software. Programs.

system. The computer and its associated devices and programs.

system dump. A printout of storage from all active programs (and their associated data) whenever an error stops the system. Contrast with *task dump*.

system profile. A file containing the default values used in system operations.

system user. A person who uses a computer system.

task. A basic unit of work to be performed. Examples are a user task, a server task, and a processor task.

task dump. A printout of storage from a program that failed (and its associated data). Contrast with *system dump*.

trace. To record data that provides a history of events occurring in the system.

two-digit display. Two seven-segment light-emitting diodes (LEDs) on the operator panel used to track the progress of power-on self-tests (POSTs).

utility. A service; in programming, a program that performs a common service function.

Volume ID (Vol ID). A series of characters recorded on the diskette used to identify the diskette to the user and to the system.





The IBM RT PC
Hardware Reference Family

Reader's Comment Form

**IBM RT PC AIX Operating
System Problem
Determination Guide**

SA23-2603-0

Your comments assist us in improving our products. IBM may use and distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

For prompt resolution to questions regarding set up, operation, program support, and new program literature, contact the authorized IBM RT PC dealer in your area.

Comments:

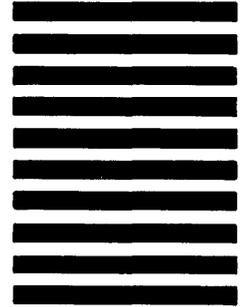


NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Department 997, Building 998
11400 Burnet Rd.
Austin, Texas 78758



Fold and tape

Fold and tape

Cut or Fold Along Line

Tape

Please Do Not Staple

Tape

© IBM Corp. 1988
All rights reserved.

International Business
Machines Corporation
11400 Burnet Road
Austin, Texas 78758

Printed in the
United States of America

SA23-2603-0

IBM
®

SA23-2603-0



22F9830