Computer Crash Analysis Tool User Guide

This guide explains how to use the Computer Crash Analysis Tool (CCAT) to analyze crash files on the supported operating systems.

Product Name:	Computer Crash Analysis Tool (CCAT)
Product Version:	5.1.1
Operating Systems:	Microsoft® Windows® 2000 and XP HP Tru64 UNIX® versions 4.0F, 4.0G, 5.0A or higher HP OpenVMS Alpha versions 7.2–2 or higher
Document Date:	4 December 2003

Contents Page

List of Figures

List of Tables

1 Introduction

Computer Crash Analysis Tool (CCAT) is a software application that enables Hewlett-Packard customer service engineers and system administrators to analyze operating system crashes.

CCAT matches information collected about a crash against a set of operating system-specific rules to determine if the footprint of the collected crash data matches any known crash data footprints for which a solution or corrective action has been found. Use of CCAT greatly reduces customer downtime by shortening the time required to analyze system crashes and eliminates the need for customer site visits.

1.1 Gathering Crash Data

The method used to gather crash data varies depending on your operating system.

1.1.1 Windows

In order to generate a crash data file that CCAT can use, Windows systems must have Crash Analysis Data Collector (CADC) installed.

CADC reads the binary crash information stored in the memory.dmp file created by the operating system in the event of a crash. CADC processes the memory.dmp file and creates a new file named NtFootPrint.txt. CCAT can only process crash files that have been pre-processed by CADC. CCAT cannot process a raw memory.dmp file.

Note

The current version of CCAT will not work with the original version of CADC. You must have version 3.1 or higher of CADC installed in order to use CCAT. You can install CADC either before or after you install CCAT.

You can download CADC from this URL:
http://www.compaq.com/support/svctools/webes/ccat/cadc.html

Once CADC is installed, you will need to configure your machine to create a memory.dmp file when/if it crashes for CADC (and subsequently CCAT) to work. For Windows, these settings can be found in the Control Panel, under the System utility.

For Windows NT, choose the tab labeled Startup/Shutdown from the System window.

- Check the box labeled Write debugging information to.
- Do NOT change the default name of MEMORY.DMP in the text window.

- Check the box labeled Overwrite any existing file option.
- Optionally check the box labeled Automatically reboot.

For Windows 2000, select the Advanced tab, then click the Startup and Recovery button. In the Write Debugging Information section, do the following:

- In the first drop down list, choose Complete, Kernel, or Small memory dump. If the machine you are configuring has more than 512 meg of memory, choose Small Memory Dump or Mini Dump.
- Do not change the default name displayed in the text window for the dump file.

Once installation and configuration is complete, each time your Windows system crashes, CADC reads and processes the memory.dmp file, and creates a new NtFootPrint.txt file. Once CADC has created the footprint, CCAT can process the crash data.

1.1.2 Tru64 UNIX

Each time your Tru64 UNIX system crashes, a system utility collects data about the crash and saves it in a crash data file.

1.1.3 OpenVMS

Each time your OpenVMS system crashes, a system utility collects data about the crash and saves it in a crash data file.

1.2 CCAT Functionality

Once the footprint has been created, CCAT can perform the following functions automatically:

- Email information about the crash to the system administrator or other designated local addressee.
- Analyze the crash file and produce a results file.
- Electronically forward a copy of the footprint and the results file to the Customer Support Center (CSC). If you would like CCAT to email the footprint and analysis results file to the CSC, you must also have DSNlink or Proactive Remote Service (PRS) installed and running.

  For more information about DSNlink, contact the CSC or see the following web site: http://www.compaq.com/support/svctools/connectivity

  For more information about PRS, contact the CSC or see the following web site: http://www.compaq.com/manage/remoteservices

CCAT can also be run at any time as a GUI, enabling you to manually process crash data files.

1.3 Security and Required Permissions

In order to enhance security, only privileged users can access the WEBES directory tree and run Compaq Analyze commands. The requirements for each operating system are given here.

1.3.1 Windows

The following actions are restricted to privileged users:

- Running any of the WEBES programs from Start | Programs | Hewlett-Packard Service Tools
- Running any WEBES or Compaq Analyze command (**desta** or **wccat** commands from the command prompt)
- Accessing any files within the WEBES install directory tree (C:\Program Files\hp\svctools by default)

To perform restricted actions, your user ID must be either:

- A member of the Administrators group on that machine.
- A member of a group that is a member of the Administrators group on that machine. For example, if your user ID is a Domain Admin, and you have added Domain Admins to the Administrators group on the local machine, you will have the necessary permissions. (The *WEBES Install Guide* describes how to add a group to the Administrators group.)

1.3.2 Tru64 UNIX

The following actions are restricted to privileged users:

- Running any WEBES or Compaq Analyze commands (**desta** or **wccat** commands from the command prompt)
- Viewing the WEBES directory tree on a system

Only the “root” user can perform these actions. The /usr/opt/hp/svctools directory is owned by root, and has rwx (read, write, and execute) permissions for root (owner), and no permissions for any other user (group or world).

1.3.3 OpenVMS

Commands—To execute any Compaq Analyze commands (**desta** or **wccat** commands from the command prompt), the user needs all of the following OpenVMS privileges. Note that

these are a subset of the privileges required to install, upgrade, or uninstall WEBES as described in the *WEBES Installation Guide*:

ALTPRI	DIAGNOSE	SYSPRV
BUGCHK	IMPERSONATE	TMPMBX
CMKRNL	NETMBX	

Files—File access is restricted in the WEBES installed directory tree pointed to by the SVCTOOLS_HOME logical (SYS$COMMON:[HP] by default). To view these files, you must be a member of the System group, your user ID must have all privileges, or you must issue the SET PROCESS /PRIV=ALL command.

All directories and files in the SVCTOOLS_HOME tree are owned by the System user, and have System, Owner, and Group permissions of RWED (Read, Write, Execute, and Delete). There are no permissions for World.

1.4 Intended Audience

The *Computer Crash Analysis Tool User Guide* is intended for use by system administrators and Hewlett-Packard Customer Services engineers who use the CCAT software on all supported operating systems, including Windows 2000 and XP, Tru64 UNIX, and OpenVMS Alpha.

1.5 Further Information

CCAT is a member of the Web-Based Enterprise Services (WEBES) suite of products. For more information on the other WEBES applications, visit the support web site: http://www.compaq.com/support/svctools/webes

2 Running CCAT Automatically

This manual tells you how to use CCAT to process crash files manually. However, CCAT is used most efficiently as an automatic process requiring no input from the user. This section describes the automatic operation of CCAT.

2.1 Automatic Mode Process

Note

If you are running an older unsupported operating system or OpenVMS VAX, you must use the Crash Analysis Data Collector (CADC) for operating system diagnostics. For more information about installing and using CADC, see the CADC user documentation for your operating system.

The automated CCAT process begins when a system crashes and consists of the following steps:

1. When the system reboots, a system utility or other software collects data about the cause of the crash and creates a crash file.

    - On Windows systems, you must install the Windows version of CADC in order for the crash file to be created.
    - On Tru64 UNIX systems, the crashdc utility creates the crash file.
    - On OpenVMS/Alpha systems, the Clue utility creates the crash file.

2. CCAT automatically starts when a system reboots from a crash, and detects that there is a crash file to process. CCAT analyzes the crash file against the local CCAT knowledge base and produces a results file which contains the crash parameters, and may include the possible cause and solution for the system crash.
3. CCAT sends an email message to the system administrator or other specified local addressee containing information about the crash.

Note

In order for CCAT to perform the following functions automatically, either DSNlink or PRS must be installed and running on the system.

4. CCAT opens a service request containing the crash parameters and the crash data analysis file at the Customer Support Center (CSC) using DSNlink or PRS. (If neither DSNlink not PRS is available, the customer can provide the crash data analysis file to the CSC via ftp, email, or storage medium (e.g., diskette or tape).
5. The crash is analyzed again when the message containing the crash parameters and the results file arrives at the CSC, in case the CCAT server at the CSC may have updated rule sets that can provide additional insight into the cause of the crash and problem resolution.
6. The results of the analysis performed at the customer site and at the CSC are entered into the Call Handling System.

7. The CSC monitors open calls in the Call Handling System, and notifies the customer of the final analysis results via email or by means of a call from a crash analysis specialist.

2.2 Configuring CCAT To Run Automatically

If you want CCAT to process a footprint automatically and send the footprint and the results to the CSC, you must do the following:

- Make sure your system is equipped to generate a footprint when it reboots from a crash. In order for CCAT to determine the cause of a system crash and log a call to the CSC for further analysis, your system must be able to collect information about what was happening at the time of the crash into a crash file, or footprint. This data collection may be done by a utility included with your operating system, or by software you install for that purpose.
    - The Windows operating systems do not include a utility for collecting crash files. On Windows systems, you must install the Crash Analysis Data Collector (CADC) software in order for the footprint to be created. For more information about CADC for Windows, see the CADC user documentation.
    - All Tru64 UNIX operating systems supported by CCAT include system utilities which collect the crash data into a footprint.
    - All OpenVMS operating systems supported by CCAT include system utilities which collect the crash data into a footprint.
- Make sure your system is equipped to email the footprint, the analysis results, or both to specified email addresses.
- Make sure DESTA Director starts and is running before CCAT starts.
  The WEBES installation modifies your startup procedure so that when your system reboots after a crash, the DESTA Director starts before CCAT. In the unlikely event that the DESTA Director fails to start, CCAT will not start.
- Make CCAT a part of the startup procedure.
  The WEBES installation modifies your startup procedure so that when your system reboots after a crash, CCAT starts after the DESTA Director.
- Make sure either DSNlink or PRS is installed and running before CCAT starts.
  For more information, refer to the DSNlink or PRS user documentation.

3 Using the CCAT GUI

The CCAT GUI is an interactive tool you can use to analyze crash files manually. It is important to keep in mind that the CCAT GUI is used only for onsite manual tasks. It does not log calls or send crash parameters or results files to the CSC, nor does it send email notification to anyone.

The CCAT GUI allows you to perform the following tasks:

- Input operating system crash data parameters for a manual CCAT analysis.
- Produce and save results files.
- View the saved results files.

3.1 Verifying the WEBES Director

The DESTA Director must be running before you start the CCAT GUI. Ordinarily, the WEBES Common Components installation configures your startup procedure so that the DESTA Director starts every time your system reboots. If the DESTA Director fails to start at system startup, you will not be able to analyze crash files.

You can verify that the DESTA Director is running by executing the following command:

desta status

If circumstances require it, you can manually start the Director by following the instructions for your operating system.

3.1.1 Windows

To start the WEBES Director, start the DESTA_Service Windows service using one of the following methods:

- Select Programs | Hewlett-Packard Service Tools | Web-Based Enterprise Service | Start Director from the Start menu.
- Enter **net start DESTA_Service** in a Command Prompt window.
- Start DESTA_Service from the Services utility in the Control Panel.

Using the desta start command on Windows systems is unsupported. Using the desta start command will start the Director, but will also generate error messages. Starting the director this way is is not recommended because:

- Closing the command prompt window used to issue the command or logging out of the Windows session, will forcibly but incompletely kill the Director, leaving running processes behind (see the *WEBES Release Notes* if this situation occurs). In addition, open files may not be saved correctly, resulting in data corruption.
- Text log output from the Director process will only be displayed on the screen and will eventually scroll past the buffer.

On Windows, the desta start/stop functionality is only intented to be used as a tool for investigating WEBES operational problems. If the Director is started with `desta start`, it must be stopped with `desta stop`.

3.1.2 Tru64 UNIX

Enter **/usr/sbin/desta start** at a shell prompt.

On TruClusters, you can run the /usr/sbin/webes_install_update program and choose the Start WEBES Director option to start the Director on either all the nodes in the cluster or a selected group of nodes that you choose.

3.1.3 OpenVMS

Enter **DESTA START** at the OpenVMS command line prompt.

On OpenVMS clusters, you can uses the SYSMAN utility to issue the command **do desta start** on either all the nodes in the cluster or a specific group of nodes that you choose.

3.2 Starting the GUI

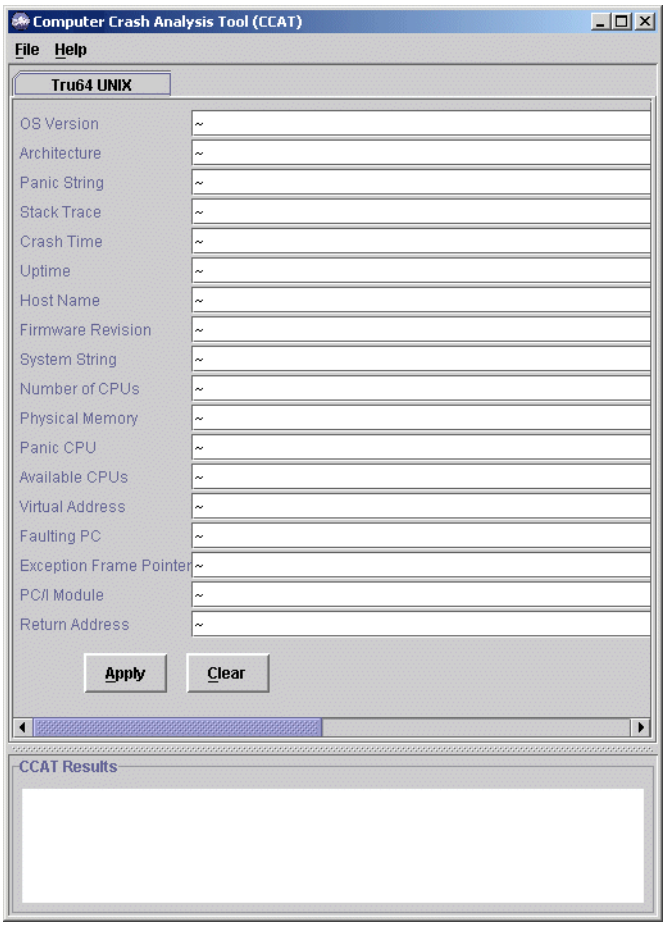
Start the CCAT GUI according to your operating system:

Windows:	Start | Programs | Hewlett-Packard Service Tools | Computer Crash Analysis Tool | Computer Crash Analysis Tool
Tru64 UNIX:	# **/usr/sbin/wccat gui**
OpenVMS:	(Before you start the CCAT GUI, make sure your user account page file quota is set to at least 300,000 blocks.) $ **@SVCTOOLS_HOME:[COMMON.BIN]WCCAT GUI**

3.3 CCAT GUI

Starting the GUI displays the CCAT window (Figure 3–1).

Figure 3–1 Computer Crash Analysis Tool Window



Note the horizontal scroll bar at the bottom of the upper frame of the CCAT window. You can resize the CCAT window to best suit your needs and the size of your monitor. Use the scroll bar to view information in the crash data parameter fields that falls outside the frame area.

3.4 Performing a Manual Crash Analysis

To analyze a crash manually, you must enter the parameters from the crash data file into the fields in the CCAT window.

3.4.1 Crash Data Parameters

The crash data parameters that you need to enter vary depending on your operating system.

Windows

Table 3–1 Windows Crash Data Parameters

Parameter	Explanation
OS Version	The version number of the failed Windows operating system
Minor Version	The NT build number (for NT 4.0, 1381; for Windows 2000, 2195)
Service Pack	The number of the Service Pack installed on the failed machine
Machine Image Type	“intel”
BugCheckCode	The number of the stop that occurred, which can be used to determine what trap occurred
BugCheckParam #1 BugCheckParam #2 BugCheckParam #3 BugCheckParam #4	The four parameters normally included with the BugCheckCode that give clues to the nature of the BugCheckCode
Failing Module	The name of the driver that failed
Failing Module Offset	The offset of the failed driver
Failing Module Timestamp	The date and time the failed driver was built
Crash Process Name	The name of the process that was running when the system crashed
Failing Routine	The name of the failing routine
Failing Routine Offset	The failing address location within the failing routine, offset from the start of the routine
Pool Information	The address within a Page or NonPage pool, depending on the stopcode

Table 3–1 Windows Crash Data Parameters (continued)

Parameter	Explanation
Canonical Stopcode Parameter 1 Canonical Stopcode Parameter 2 Canonical Stopcode Parameter 3 Canonical Stopcode Parameter 4	Address or status register variables (see the Kanalyze documentation for more information)
Keyword 1 Keyword 2 Keyword 3 Keyword 4	Items on the stack that point to the cause of the failure (see the Kanalyze documentation for more information)
Driver List	The Driver Name, Driver Load Address, Driver Size and Driver Date. These values are derived from the failing address information contained in the Bugcheck Parameter fields. Which Bugcheck Parameter field you use depends on the Bugcheck Code. The Driver List corresponds to the driver base address when compared to the address of the Stopcode.
Stack Trace	A list of the functions the system was executing when it crashed, with the ending line of code for each
Call Site List	Addresses taken from the Stack Trace used to identify failing areas

Tru64 UNIX

Table 3–2 Tru64 UNIX Crash Data Parameters

Parameter	Explanation
OS Version	The version number of the failed operating system
Architecture	The hardware architecture (e.g., alpha)
Panic String	A brief description of why the system crashed
Stack Trace	A list of the functions the system was executing when it crashed, with the ending line of code for each
Crash Time	The time of the system crash
Uptime	How long the system that crashed had been running since the last reboot
Host Name	The node on which the crash occurred
Firmware Revision	The machine hardware type of the failed CPU
System String	The System Information String, e.g., AlphaServer 4100 5/400 4MB
Number of CPUs	The number of CPUs available to the system
Physical Memory	The memory in megabytes
Panic CPU	The CPU that caused the system to crash

Table 3–2 Tru64 UNIX Crash Data Parameters (continued)

Parameter	Explanation
Available CPUs	The CPUs that are currently being used
Virtual Address	The virtual address that caused a kernel memory fault, and subsequent system crash (valid for kernel memory fault panics only)
Faulting PC	The PC on which the fault occurred
Exception Frame Pointer	A pointer to the exception frame that contains register information about the state of the failed CPU (valid prior to V4.0 only)
PC/I Module	The Program Counter/Instruction at the time of the trap or exception that led to the system crash (valid prior to V4.0 only)
Return Address	The address of the instruction immediately prior to the trap or exception that led to the system crash (valid prior to V4.0 only)

OpenVMS Alpha

Table 3–3 Open VMS Alpha Crash Data Parameters

Parameter	Explanation
OS Version	The version number of the failed operating system
Crash Time	The date and time the system crash occurred
Bugcheck	The type of diagnostic check logged by the operating system
Host Name	The node on which the crash occurred
CPU Type	The model number of the failed CPU
Process Name	The name of the process active at the time of the crash
Image Name	The name of the image being executed at the time of the crash
Signal Array	The Signal Array count. The Signal Array contains the exception code, zero or more exception parameters, the PC, and the PSL.
Condition Code	The symbolic value assigned to the specific condition
Reason Mask	The longword mask
Virtual Address	The virtual address the failing instruction tried to reference
Exception PC	The instruction whose attempted execution resulted in the unexpected executive or kernel mode exception
Exception PSL	Processor Status Longword (PSL) at the time of the exception
Module Name	The name of the failed module
Module Offset	The offset of the failed module
Instruction	The failing instruction corresponding to the exception PC
Map Module	The name of the map module in use when the crash occurred

Table 3–3 Open VMS Alpha Crash Data Parameters (continued)

Parameter	Explanation
Map Offset	The beginning memory location where the map module driver resides
Caller Module	The first module identified on the stack below the failing PC
Caller Module Offset	The first module offset identified on the stack below the failing PC
Instruction M1	The instruction executed immediately before the Failing Instruction (helps to locate the Failing Instruction precisely in the code)
Instruction M2	The next-to-last instruction executed before the Failing Instruction (helps to locate the Failing Instruction precisely in the code)
Instruction P1	The first instruction that would have been executed after the Failing Instruction (helps to locate the Failing Instruction precisely in the code)
Instruction P2	The second instruction that would have been executed after the Failing Instruction (helps to locate the Failing Instruction precisely in the code)

3.4.2 Entering Parameters

You can enter crash data parameters in any of the following ways:

- Select and open a crash file.
- Type the parameter in the appropriate field in the CCAT window.

Note

To edit the contents of a parameter field, click on the field and use the arrow and Backspace keys to remove unwanted characters. Do not click on the Clear button. The Clear button clears all of the parameter fields.

3.4.2.1 Selecting And Opening a Crash File

To populate the crash data parameter fields by selecting and opening a crash file, follow these steps:

1. Choose Select Crash File.... from the File pull-down menu.
2. Select the desired crash file using the appropriate procedure for your operating system.

    - Windows—Select the desired file from the Open window. If the file is not in the default directory (C:\Program Files\hp\svctools\common\ccat\data), use the pull-down menu in the Look in field to navigate to the correct location.

- Tru64 UNIX—Select the desired file from the Open window. If the file is not in the default directory (/usr/opt/hp/svctools/common/ccat/data), use the pull-down menu in the Look in field to navigate to the correct location.
- OpenVMS Alpha—Enter the fully qualified path name to the OpenVMS crash file. You cannot use logical names such as SYS$ERRLOG for the disk address.

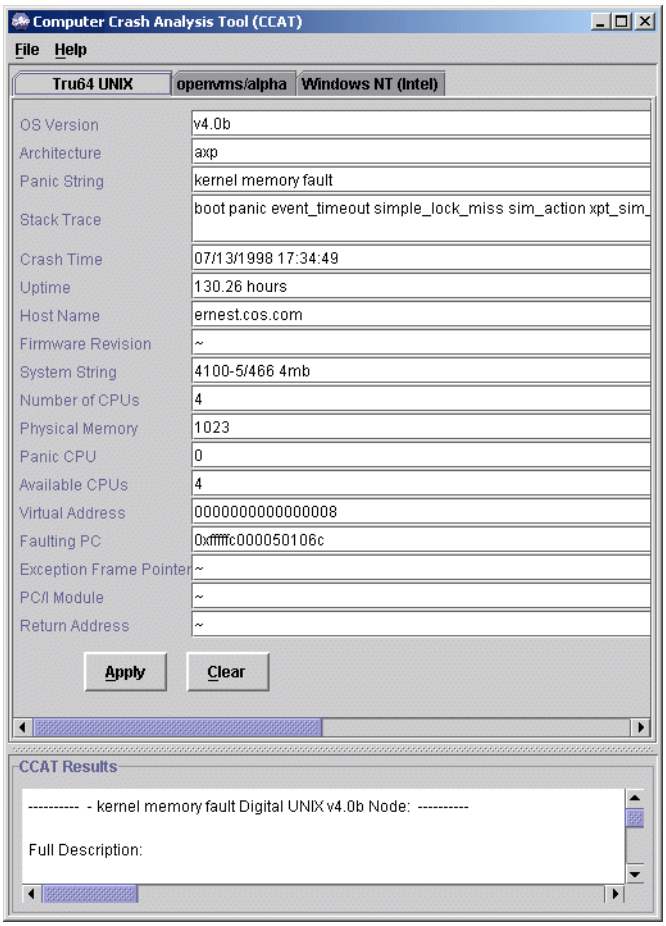
3. Once you have the correct file selected or entered, click the Open button. The CCAT Message Processing window appears, telling you that the crash file is being analyzed.

During analysis, CCAT populates the parameter fields. When analysis is complete, the results appear in the CCAT Results frame at the bottom of the CCAT window, as shown in Figure 3–2.

Note

When the results file is displayed in the CCAT Results frame, the frame at the top of the CCAT window may be grayed out. To display the contents of this frame, click on the operating system tab.

Figure 3–2 CCAT Analysis Results



3.4.2.2 Typing In Crash Parameters

To enter a crash parameter manually, click on the appropriate field and type the parameter exactly as it appears in the crash data file or use the Copy and Paste functions to copy information into the fields.

When you are entering crash parameters manually, it is important to remember the following:

- Crash data information is case sensitive.
- CCAT does not automatically zero-fill. For example, if you are entering the crash parameter "00000005" you must type in the seven zeroes. CCAT does not interpret "5" as "00000005."

If you make a mistake or need to edit the contents of a parameter field, click on the field and use the arrow and Backspace keys to remove unwanted characters. Do not click on the Clear button. The Clear button clears all of the parameter fields.

The crash data file may not contain all of the parameters listed in the CCAT window. When the crash data file does not contain a parameter, leave the tilde (~) in the field to indicate that the parameter is not available.

Once you have entered all the crash data parameters available to you, click on the Apply button on the right side of the CCAT window to start the crash analysis.

When CCAT has completed the crash analysis, the results file is displayed in the frame at the bottom of the CCAT window. You can resize the window and use the scroll bar to view the file.

3.4.3 Saving the Results File

If you want to save the results file so you can view it again later, make sure the file is still displayed in the frame at the bottom of the CCAT window. Then follow these steps:

1. Select Save Results File As from the File pull-down menu. The Save window appears.
2. Use the Look In field to select the directory where you want to save the results.
3. Enter the name you want to assign to the saved results file in the File Name field and click the Save button.

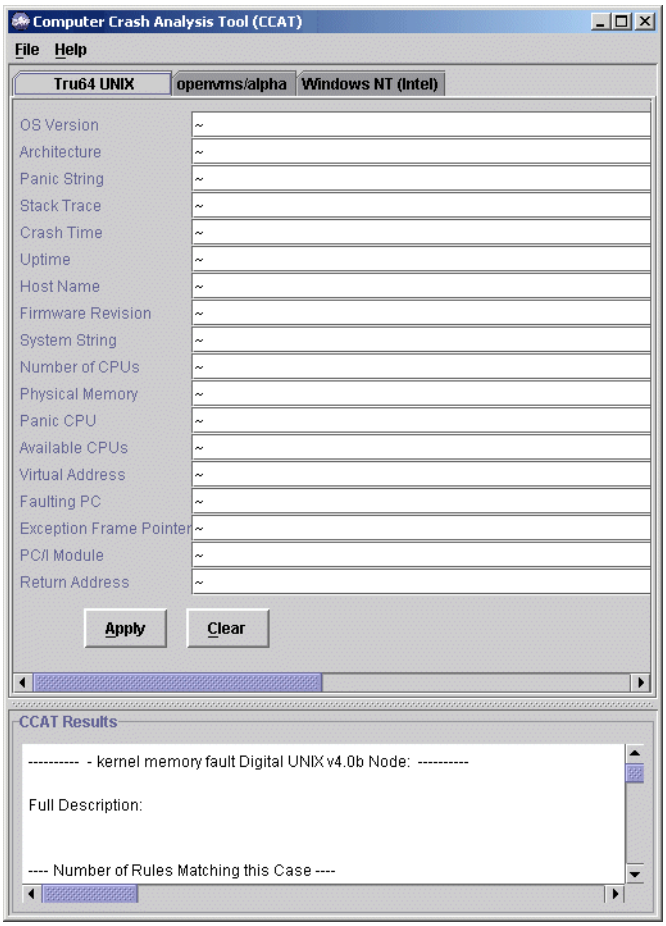
3.5 Viewing Saved Results Files

To view a previously saved results file, follow these steps:

1. Select View Saved Results File from the File pull-down menu. The Open window appears.
2. Use the Look In field to select the directory where the results file is saved.
3. Click on the results file you want to display. The name of the file you selected appears in the File Name.
4. Click on Open.

CCAT displays the results file in the Results frame at the bottom of the CCAT window, as shown in Figure 3–3.

Figure 3–3 Typical CCAT Analysis Results



3.6 Exiting From the CCAT GUI

To exit from the CCAT GUI, select Exit from the File pull-down menu.

A CCAT Information message window appears, telling you that the communication interface has been shut down, as shown in Figure 3–4.

Figure 3–4 Exit CCAT Information Window

Click on OK to exit from CCAT.

Copyright Statement