Windows NT DrWatson Postmortem Debugger

Welcome to the Windows NT DrWatson Postmortem Debugger help file. This help file can help you get the best use from DrWatson in diagnosing your application errors.

Help Contents

What is DrWatson?

DrWatson is a Windows NT postmortem debugger. A postmortem debugger is a program that detects application errors, diagnoses the error, and logs the diagnostic information.

The information obtained and logged by DrWatson is the information needed by Product Support Personnel to diagnose the application error. The log file is a text file that can be printed, e-mailed, copied to a floppy disk, or otherwise delivered to Support Personnel.

Options That Control DrWatson's Behavior

The options listed here are all of the options that you can use to change the behavior of DrWatson. The options can all be changed through DrWatson's main dialog. To access the main dialog run DrWatson from a command window or Program Manager. The options data is stored in the system's registryunder the key \\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\DrWatson.

Options

Logfile Location
Name of a Wave file for Sound Notification
Number of Machine Instructions to Disassemble
Number of Application Errors to Save in the Eventlog
Dump the Symbol Table for Each Thread
Create a State Dump for Each Thread
Append to the Logfile or Create a New Logfile
Visual Popup Notification
Sound Notification

Logfile Location

The logfile location is a valid path on your local machine. The default path is the Windows Directory.This path is where DrWatson will create the log file containing the diagnostic information about application errors.

Be sure that the path specified is one that all users have read/write priviliges to. If DrWatson cannot use the path specified when a log file is created, a file open dialog box is presented and a new path must be specified.

Name of a Wave file for Sound Notification

The wave file name is used by DrWatson to play a sound when an application error occurs. The file name must be a .WAV file and must conform to the Microsoft wave file format. If you can play the wave file with Media Player then the file is a valid wave file.

Number of Machine Instructions to Disassemble

This is a number that tells DrWatson how many machine instructions to disassemble before and after the current Program Counterfor each thread's state dump.

Number of Application Errors to Save in the Event Log

When DrWatson dectects an application error extensive diagnostic information is logged into the DrWatson log file. DrWatson also records an entry in the Windows NT Application Eventlog.DrWatson records the application name, date, time, exception number, exception name, program counter,and function name at the current program counter.

Dump the Symbol Table for Each Thread

This option determines whether DrWatson dumps the symbol table for the module at the current program counterfor the thread being dumped. The symbol table dump contains the address and name for each symbol.

 Beware that this option can cause your log file to be quite large.

Create a State Dump for Each Thread

This option controls how many threads DrWatson dumps state information for. If this option is set DrWatson logs a state dump for each thread is the faulting application. Otherwise DrWatson logs only the thread that caused the application error.

Append to the Logfile or Create a New Logfile

This option determines if DrWatson appends diagnostic information to the end of the DRWTSN32.LOG log file or creates a new log file with each application error.

 Beware that this option can cause your log file to be quite large.

Visual Popup Notification

This option determines whether DrWatson provides a popup dialog box when an application error is detected. The popup has an OK button that requires user interaction. However, if a user does not respond to the popup within 5 minutes the popup is removed.

Sound Notification

This option determines whether DrWatson plays a sound when an application error is detected. The sound that is played is the WAVE file specified in the wave file option.

The registry is the Windows NT repository for system and application configuration information.

The eventlog is the Windows NT repository for system and application event information.

The Windows Directory is the directory that Windows NT is installed into.

The program counter is a machine register that contains the memory location for a thread's current point of execution. For Intel x86 machine the register is EIP, for MIPS machines the register is FIR, and for DEC Alpha the register is FIR.

Log File Description

```
Application exception occurred:
```

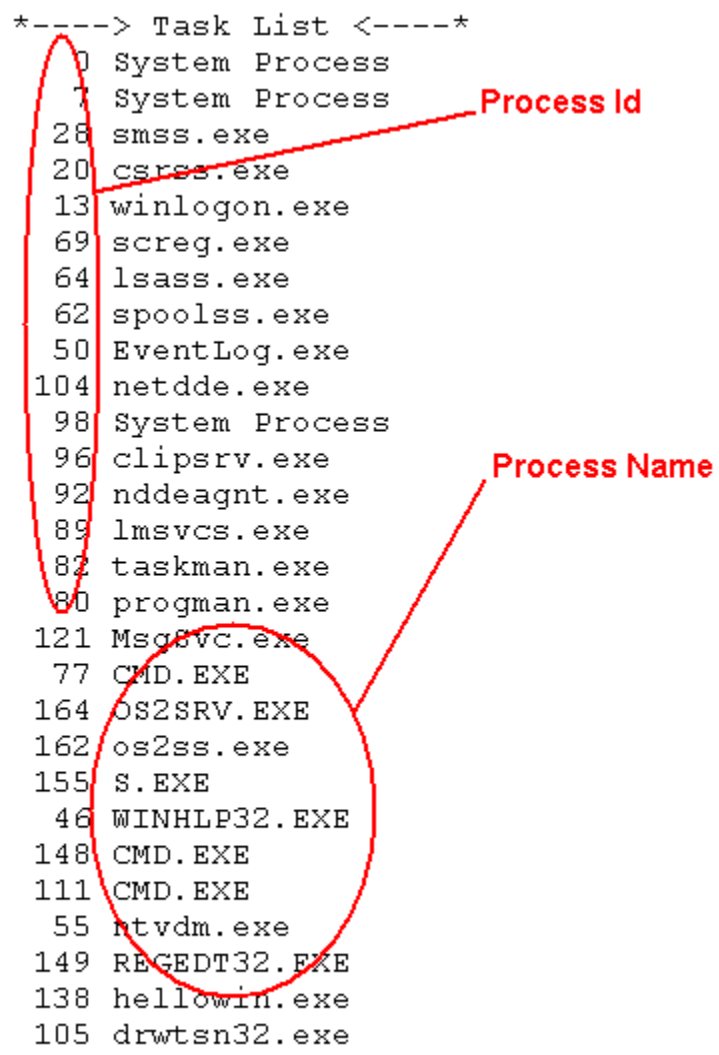
This part of the log file contains exception information. The exception number listed is the exception generated by the system.

```
        App: hellowin.exe
         When: 5/1/1993 @ 15:33:42.810
         Exception number: c0000005 (access violation)
```

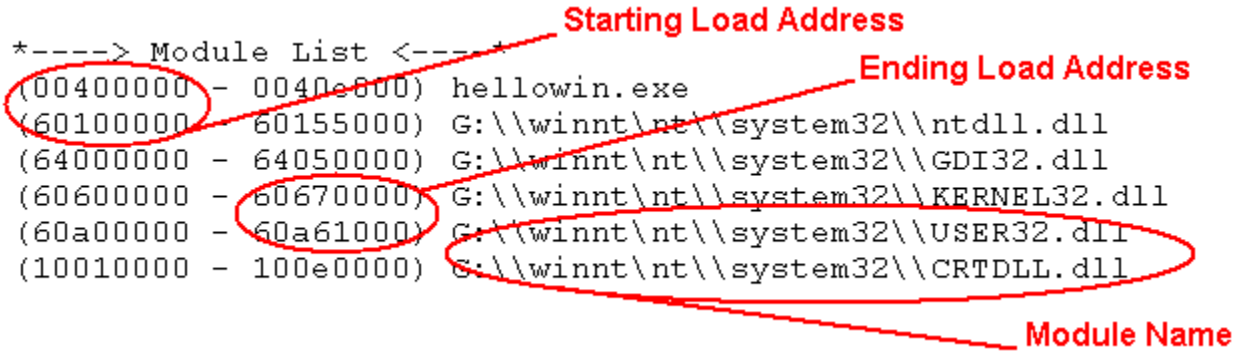
This part of the log file contains system information about the user and the computer on which the application faulted.

```
*----> System Information <----*
         Computer Name: tester
         User Name: bob
         Number of Processors: 1
         Processor Type: Intel 486
         Windows Version: 3.10
```

This part of the log file contains the list of tasks running on the system at the time the application faulted.



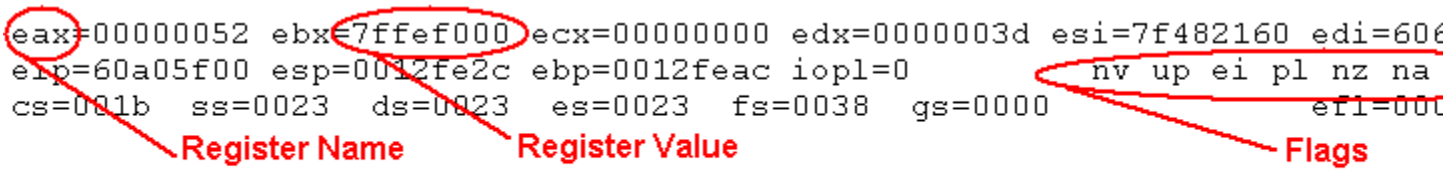
This part of the log file contains the list of modules that the faulted application loaded.



This part of the log file contains the state dump for the thread id listed. The state dump consists of a register dump, disassembly of the code surrounding the current program counter, and a stack back trace.

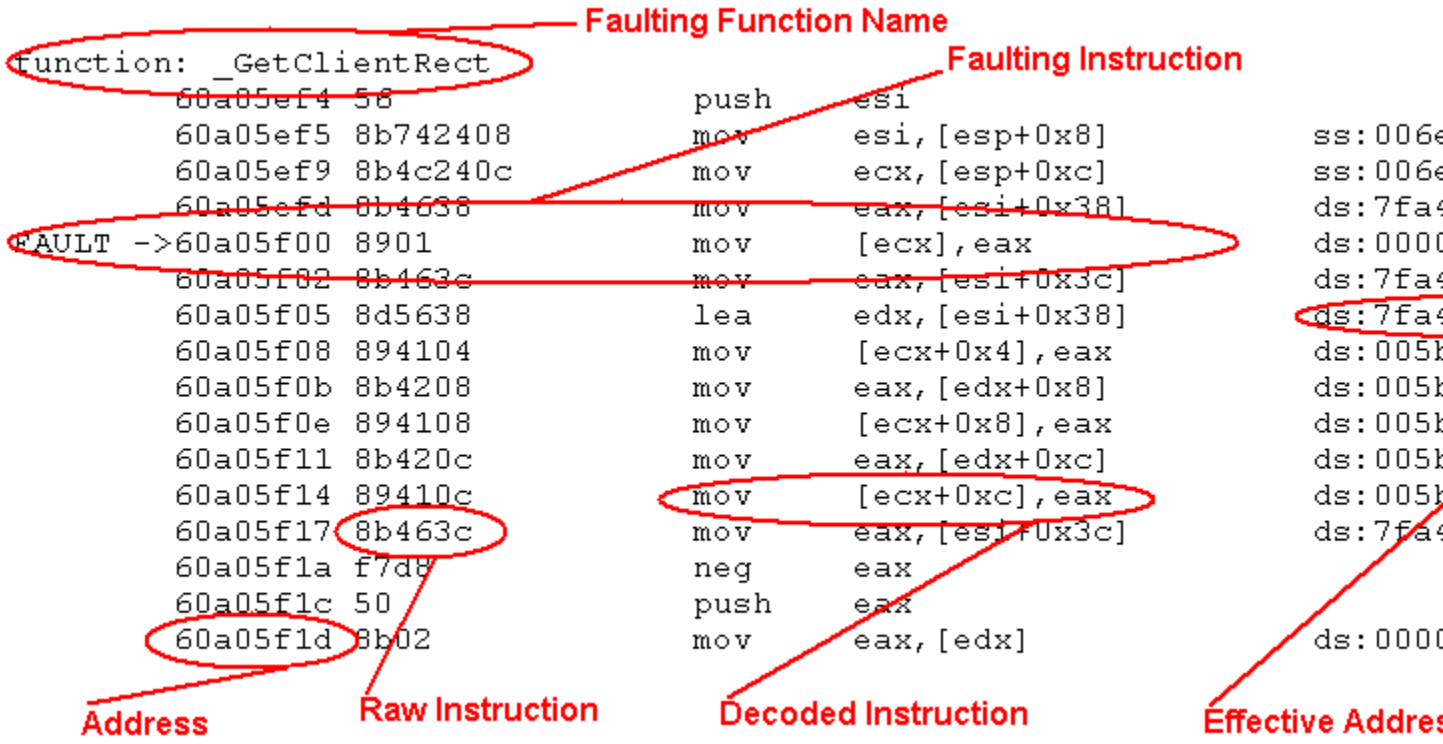
```
State Dump for Thread Id c6
```

This part of the log file contains the register dump.



Note: this example shows a register dump for a Intel 486 Processor

This part of the log file contains the instruction disassembly.



This part of the log file contains the stack back trace.

```
*----> Stack Back Trace <----*
RetAddr  FramePtr Param#1   Param#2   Param#3   Param#4
60a05f00 0012fe2c 7f482160 00000000 0040112d 003d017a _GetClientRect
60a0614f 0012fe38 003d017a 00000000 60637680 002c0044 GetClientRect
0040112d 0012feac 003d017a 0000000f 00000000 00000000 WndProc
60a05de4 0012fec0 003d017a 0000000f 00000000 00000000 DispatchClientMess
60a06c62 0012fed8 002c0058 002c0044 0012ff10 60103bdc __fnPAINT
60103b14 0012fee8 002c0044 60637680 002c0000 7ffef000 CsrpProcessCallbac
60103bdc 0012ff10 ffffffff 60a06ad7 0012ff28 002c0000 CsrClientSendMessa
60a046eb 0012ff18 0012ff28 002c0000 002c0000 002c0024 CCSMakeCall
60a06ad7 0012ff34 003d017a 60637680 ffffffff 7ffef000 UpdateWindow
004010c1 0012ff94 00400000 00000000 00130d3e 0000000a WinMain
004012cd 0012fff0 7ffef000 00000000 00000049 00000100 WinMainCRTStartup
```

Return Address

Frame Pointer

First 4 Parameter to the Functions

Function N

This part of the log file contains the symbol table.

```
*----> Symbol Table <----*
60a01000    AbortProcYield
60a01080    UserRegisterWowHandlers
60a01150    DispatchDlgProc
60a011b0    MenuWndProcA
60a01210    MenuWndProcW
60a01270    InitClientDrawing
60a01350    fnHRGNDWORD
60a01390    CallWindowProcAorW
60a01470    ClientFrame
60a01540    WOWLoadBitmapA
60a015e0    DisconnectConv
60a01600    UnlinkConvFromOthers
60a01750    WaitForZombieTerminate
```

Beginning Address

Function Name

How to Install DrWatson

The DrWatson program (DRWTSN32.EXE) is pre-installed on your system when Windows NT is setup. The file is located in your system directory, typically c:\winnt\system32. The default options are setup the first time that DrWatson is run, either when an application fault occurs or when it is run from Program Manager.

When an application error occurs on Windows NT the system searches for an application software exception handler. If an exception handler is not found the system verifies that the application is not currently being debugged. If the application is not being debugged then the exception is considered unhandled. The system processes unhandled exceptions by looking in the registry for a application error debugger. The system looks in \\HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\AeDebug for a value named Debugger and Auto. If the Debugger value is present it specifies a debugger that is executed to analyze the faulted application. The value named Auto is then checked for a value of "0" or "1". If Auto is set to "0" the system generates a popup dialog containing a message that says the application has faulted. If the Debugger value contains a valid debugger (such as WinDbg or NTSD) the popup contains 2 buttons OK and Cancel. If OK is pressed then the application is terminated. If Cancel is pressed then the debugger specified in the Debugger value is executed. If the Debugger value is empty then the popup only contains an OK button (no debugger is ever executed). If the Auto value is set to "1" then the system does not generate a popup and the debugger specified in the Debugger value is executed (if one is specified).

When Windows NT is setup on your system the Auto value is set to "1" and the Debugger value is set to DRWTSN32. This means that when an application faults on your system DrWatson will catch the fault and log the appropriate diagnostic information.

In summary, the requirements for having DrWatson log your application errors is twofold. The DrWatson program (DRWTSN32.EXE) must be in your system directory and the above mentioned registry values must be set correctly.

Assertion Error

The Assertion dialog box is presented when an unexpected error occurs in DrWatson. It does not mean that DrWatson is going to fault or that the error is catastrophic. The dialog box shows the expression that failed the assertion (source code), the last system error, the source file, and line number where the assertion failed.

The dialog box has 4 push buttons; Abort, Retry, Ignore, and Help. The abort button causes DrWatson to terminate immediatly. The Retry button causes a breakpoint to occur. If you are attached to a debugger such as WinDbg the debugger will get control. If you are not attached to a debugger then a popup will be presented indication an unknown software exception. The Ignore button simple ignores the assertion and proceedes as if nothing happened.