

# **MAP File Browser**

by

Software Verify

Copyright © 2015-2025 Software Verify Limited



# **MAP File Browser**

## MAP file contents inspector

by Software Verify Limited

Welcome to the MAP File Browser software tool. MAP File Browser is a software tool that allows you to inspect the contents of MAP files.

We hope you will find this document useful.

## **MAP File Browser Help**

#### Copyright © 2015-2025 Software Verify Limited

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

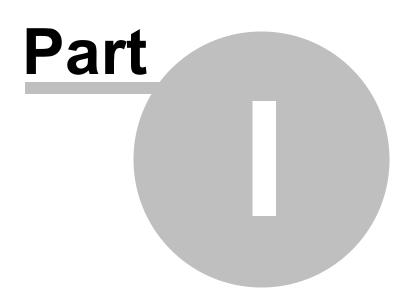
Printed: February 2025 in United Kingdom.

L

# **Table of Contents**

	Index	0
Part X	Command Line Interface	54
5	What is a load address?	46
4	Decoding an Event Viewer XML crash log	43
	Decoding a symbol relative crash address	
2	Decoding a relative crash address	
1	· · · · · · · · · · · · · · · · · · ·	36
Part IX	How to use MapFileBrowser	34
Part VIII	MAP File Information	32
4	Path Substitutions	30
3	Source code paths	28
2	Data display	27
1	Symbols	26
Part VII	Settings dialog	25
Part VI	The user interface	21
5	Неір	18
4	Software Updates	15
	Query	
	Settings	
	File	14
Part V	Menu	13
Part IV	MAP File contents	8
Part III	What is a module?	6
Part II	What does MapFileBrowser do?	4
Part I	How to get MapFileBrowser	2
	Foreword	1

Foreword	1



3

## 1 How to get MapFileBrowser

MapFileBrowser is free for commercial use. MapFileBrowser can be downloaded for Software Verify's website at https://www.softwareverify.com/product/map-file-browser/.

This help manual is available in Compiled HTML Help (Windows Help files), PDF, and online.

Windows Helphttps://www.softwareverify.com/documentation/chm/mapFileBrowser.chmPDFhttps://www.softwareverify.com/documentation/pdfs/mapFileBrowser.pdfOnlinehttps://www.softwareverify.com/documentation/html/mapFileBrowser/index.html

Whilst MapFileBrowser is free for commercial use, MapFileBrowser is copyrighted software and is not in the public domain.

You are free to use the software at your own risk.

You are not allowed to distribute the software in any form, or to sell the software, or to host the software on a website.

Contact Software Verify at:

Software Verify Limited Suffolk Business Park Eldo House Kempson Way Bury Saint Edmunds IP32 7AR United Kingdom

emailsales@softwareverify.comwebhttps://www.softwareverify.combloghttps://www.softwareverify.com/blogtwitterhttp://twitter.com/softwareverify

Visit our blog to read our articles on debugging techniques and tools. Follow us on twitter to keep track of the latest software tools and updates.



5

# 2 What does MapFileBrowser do?

MapFileBrowser allows you to inspect the contents of a linker MAP file.

You can sort the data, filter the data by name.

You can also query the data by address which can be useful for identifying what function is at a given address if all you have is a crash address and nothing else.

Query by address is supported three ways:

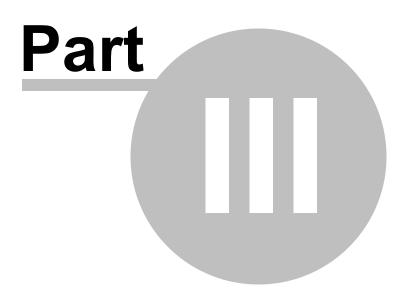
- Query by absolute address.
- Query by address offset from a DLL load address.
- Query by address offset from a symbol.

## 32 bit and 64 bit

MAP files created by 32 bit and 64 bit software are supported. On 64 bit Operating systems if a 64 bit MAP file is opened the 64 bit version MAP File Browser is automatically started.

## **History**

MapFileBrowser has been an internal tool at Software Verify for many years. We recently decided to make it a bit more user friendly and to make it available for public use.



# 3 What is a module?

A module is a contained block of executable code and data. For example, a DLL or EXE.

Some software vendors name their DLLs with different file extensions, for example .BPL, .ARX

When you call LoadLibrary to load a module, you are returned a HMODULE, which is an opaque handle to a module. The HMODULE is most often the same as the module load address, but not always. The lower few bits of the HMODULE can get OR'd with some flags to create a HMODULE value that is not the same as the module load address.

You can get the load address of a module from it's HMODULE by masking out the lower 16 bits of the HMODULE value then casting to a DWORD\_PTR.

In this documentation when you read EXE or DLL or module, we are effectively referring to the same thing. It's easier to read and write "DLLs" rather than "DLLs or EXE".



# 4 MAP File contents

MapFileBrowser can read MAP files generated by Microsoft Visual Studio, Delphi, C++ Builder and various other MAP files.

Some MAP files contain line numbers, some don't. The options for adding line numbers to MAP files have changed with different versions of compilers and linkers.

Even if a MAP file doesn't contain line numbers, the MAP can still be useful in guiding you to the correct function that is related to an address.

#### Visual Studio 6

Visual Studio 6 will add line numbers to your MAP file if you add the following option to your linker settings /MAPINFO:LINES

Project Settings	2 🔀
Settings For: Win32 Debug Non Link	General Debug C/C++ Link Resource Category: Debug Mapfile name: Generate mapfile DebugNonLink6_0/nativeExample.map Debug Info Debug Info Debug info Generate types Microsoft format GOFF format Both formats
	Project Options: //def:''.\nativeExample.def'' /out:''DebugNersLiekE.O/estiveExample.exe'' /pdbtype:sep //MAPINFO:LINES
	OK Cancel

#### Visual Studio 7 (2002) and later

Visual Studio 7 and later will not add line numbers to your MAP file. The linker option /MAPINFO:LINES is not supported.

This means you can't create MAP files containing line numbers for anything you build with Visual Studio 7 (2002) or later, or for any 64 bit builds.

You can still use MapFileBrowser to identify the function which contains a given address.

nativeExample Property Pages						?	×
Configuration: Active(Release)	✓ Platform: Ad	ctive(x64)		~	Configuratio	n Manag	jer
<ul> <li>Common Properties</li> <li>Configuration Properties         <ul> <li>General</li> <li>Debugging</li> <li>VC++ Directories</li> <li>C/C++</li> <li>Linker</li> <li>General</li> <li>Input</li> <li>Manifest File</li> <li>Debugging</li> <li>System</li> <li>Optimization</li> <li>Embedded IDL</li> <li>Advanced</li> <li>Command Line</li> <li>Manifest Tool</li> <li>Resources</li> <li>XML Document Generator</li> <li>Bivid Events</li> <li>Custom Build Step</li> </ul> </li> </ul>	Generate Debug Info Generate Program Data Strip Private Symbols Generate Map File Map File Name Map Exports Debuggable Assembly		Yes (/DEBUG) .\Release10_0_x64/ Yes (/MAP) .\Release10_0_x64/ No				
< >	Generate Debug Info The /DEBUG option creates	s debugging informa	tion for the .exe file or DLL.				
				ОК	Cancel	Ар	ply

#### Delphi

To create MAP files containing line numbers you need to specify that you want a "detailed" MAP file in the options for your Delphi or C++ Builder project.

Project Options for CppBuilder     C++ Linker     Delphi Compiler	^ Linking	× (م
Compiling Hints and Warnings Ulinking Output - C/C++ > Resource Compiler > Turbo Assembler Build Events Build Order Application Appearance Manifest Icons Services Forms Version Info > Debugger Symbol Tables Environment Block > Packages Runtime Packages > Project Properties	<ul> <li>Maximum Stack Size</li> <li>Minimum Stack Size</li> <li>Output resource string .drc file</li> <li>Place debug information in separation</li> <li>Set extra PE Header flags</li> <li>Set extra PE Header optional flags</li> <li>Set OS Version fields in PE Header</li> </ul>	<ul> <li>☐ false</li> <li>☐ false</li> <li>Detailed</li> <li>1048576</li> <li>16384</li> <li>☐ false</li> <li>☐ false</li> <li>0</li> <li>0</li> <li>5.0</li> <li>5.0</li> </ul>
		Save Cancel Help

#### C++ Builder

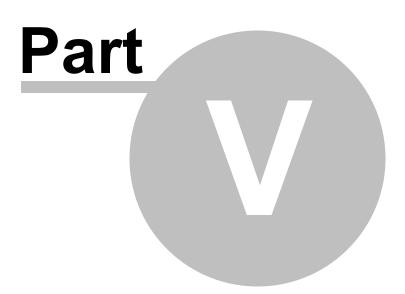
To create MAP files containing line numbers you need to specify that you want a "detailed" MAP file in the options for your Delphi or C++ Builder project.

Output       Iarget         Warnings       Iarget         Delphi Compiler       Release configuration - Windows 64-bit platform ✓ Apply       Save         > Resource Compiler       Base address       0x00400000         > Turbo Assembler       > Base address       0x00400000         Build Events       > Heap size, maximum       0x00100000         Build Order       > Heap size, minimum       0x00001000         Application       > Image comment string       > Image description         Appearance       > Image flags       > Map file type       Detailed segment map         Services       > Map with mangled names       _ false         Forms       > Stack size, maximum       0x00100000         Version Info       > Stack size, minimum       0x00100000         Symbol Tables       > Stack size, minimum       0x00000000         Environment Block       > User version       0.0         Packages       Runtime Packages       > User version       0.0         Project Properties       General	∨ C++ Linker	^	Output		
> Delphi Compiler         > Resource Compiler         > Turbo Assembler         Build Events         Build Order         Application         Appearance         Manifest         Icons         Services         Forms         Version Info         Debugger         Symbol Tables         Environment Block         Packages         Runtime Packages         Project Properties         General  Base address      Base address   May file type  Detailed segment map  Services  Map with mangled names  Section flags  Ostack size, maximum  Ox0001000      Stack size, minimum  Ox00010000  Ox00000000  Ox00010000  Ox0001000	Output		Carpar		
> Resource Compiler         > Turbo Assembler         Build Events         Build Order         Application         Applearance         Manifest         Icons         Services         Forms         Version Info         Debugger         Symbol Tables         Environment Block         Packages         Runtime Packages         Project Properties         General	Warnings		<u>T</u> arget		
> Resource Compiler         > Turbo Assembler         Build Events         Build Order         Application         Appearance         Manifest         Icons         Services         Forms         Version Info         Debugger         Symbol Tables         Environment Block         Packages         Runtime Packages         Project Properties         General	> Delphi Compiler		Release configuration - Windows 64-bit	platform V Apply	Save
S Turbo Assembler   Build Events   Build Order   Application   Appearance   Manifest   Icons   Services   Forms   Version Info   Debugger   Symbol Tables   Environment Block   Project Properties   General	> Resource Compiler		_		
Build Order       > Heap size, minimum       0x00001000         Application       > Image comment string       >         Appearance       > Image description       >         Manifest       > Image flags       >         Icons       > Map file type       Detailed segment map         Services       > Map with mangled names       _ false         Forms       > OS version       4.0         Version Info       > Section flags       >         Debugger       > Stack size, maximum       0x00010000         Symbol Tables       > User version       0.0         Project Properties       General	> Turbo Assembler		> Base address	0x00400000	
Application       > Image comment string         Appearance       > Image flags         Manifest       > Image flags         Icons       > Map file type         Services       > Map with mangled names         Forms       > OS version         Version Info       > Section flags         Debugger       > Stack size, maximum         Symbol Tables       > User version         Environment Block       > User version         Project Properties       General	Build Events		> Heap size, maximum	0x00100000	
Appearance       > Image description         Manifest       > Image flags         Icons       > Map file type       Detailed segment map         Services       > Map with mangled names	Build Order		> Heap size, minimum	0x00001000	
Appendice       Image flags         Manifest       > Image flags         Icons       > Map file type       Detailed segment map         Services       > Map with mangled names	Application		> Image comment string		
Maintest     Map file type     Detailed segment map       Icons     > Map file type     Detailed segment map       Services     > Map with mangled names	Appearance		> Image description		
Services     > Map with mangled names     false       Forms     > OS version     4.0       Version Info     > Section flags       Debugger     > Stack size, maximum     0x00100000       Symbol Tables     > Stack size, minimum     0x00002000       Environment Block     > User version     0.0       Packages     Runtime Packages       Project Properties     General	Manifest		> Image flags		
Forms     > OS version     4.0       Version Info     > Section flags       Debugger     > Stack size, maximum     0x00100000       Symbol Tables     > Stack size, minimum     0x00002000       Environment Block     > User version     0.0       Packages     Runtime Packages       Project Properties     General	lcons		> Map file type	Detailed segment map	
Version Info     > Section flags       Debugger     > Stack size, maximum       Symbol Tables     > Stack size, minimum       Environment Block     > User version       Packages       Runtime Packages       Project Properties       General	Services		> Map with mangled names	false	
Version mild     > Stack size, maximum     0x00100000       Debugger     > Stack size, minimum     0x00002000       Symbol Tables     > Stack size, minimum     0x00002000       Environment Block     > User version     0.0       Packages     Project Properties       General	Forms		> OS version	4.0	
Symbol Tables     > Stack size, minimum     0x00002000       Environment Block     > User version     0.0       Packages     Runtime Packages       Project Properties	Version Info		> Section flags		
Symbol Tables     > Stack size, minimum     0x00002000       Environment Block     > User version     0.0       Packages     Runtime Packages       Project Properties	Debugger		> Stack size, maximum	0x00100000	
Environment Block > User version 0.0 Packages Runtime Packages Project Properties General			> Stack size, minimum	0x00002000	
Runtime Packages Project Properties General			> User version	0.0	
Runtime Packages Project Properties General	Packages				
Project Properties General	-				
General	2				
Getlt Dependencies					
	Getlt Dependencies	~			
				Save Cancel	Help

## **Other Compilers**

If you find that MapFileBrowser cannot read your MAP file, please contact us at support@softwareverify.com, attaching your MAP file to the email.

We will try to add support for your MAP file format as soon as we can.



## 5 Menu

The main menu contains five menus, File, Edit, Query, Software Updates and Help.

MAP File Browser x64
File Settings Query Software Updates Help

## 5.1 File

The File menu controls loading of MAP file information, clearing the display and exiting the program.

Load MAP file
Close
Exit

File menu > Load MAP file... > loads a MAP file and displays it.

File menu > Close > clear all results, unloads the MAP file information.

File menu > Exit > closes MapFileBrowser.

## 5.2 Settings

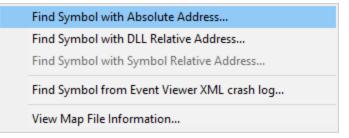
The Settings menu controls editing settings.

Edit Settings...

**Settings** menu **> Edit Settings... >** displays the settings dialog.

## 5.3 Query

The Query menu controls searching for symbols.



**Query** menu **> Find Symbol with Absolute Address... >** Use this option to turn an absolute address in a process into a symbol, filename and line number.

See Decoding an absolute crash address for more details.

**Query** menu **> Find Symbol with DLL Relative Address... >** use this option to turn a relative address inside a DLL into a symbol, filename and line number.

See Decoding a relative crash address for more details.

**Query** menu **> Find Symbol with Symbol Relative Address... >** use this option to turn a address that is relative to a symbol inside a DLL into a symbol, filename and line number.

See Decoding a symbol relative crash address for more details.

Query menu > Find Symbol from Event Viewer XML crash log... > use this option to turn an XML crash log from the Microsoft Event Viewer to a symbol inside a DLL into a symbol, filename and line number.

See Decoding an Event Viewer XML crash log for more details.

Query menu > View Map File Information... > use this option to view information about the MAP file.

## 5.4 Software Updates

The Software Updates menu controls how often software updates are downloaded.

If you've been notified of a new software release to Map File Browser or just want to see if there's a new version, this feature makes it easy to update.

Check for software updates
Configure software updates
Set software update directory

**Software Updates** menu **> Check for software updates >** checks for updates and shows the software update dialog if any exist

An internet connection is needed to be able to make contact with our servers.

Before updating the software, close the help manual, and end any active session by closing target programs.

If no updates are available, you'll just see this message:

No active session

## Software Update dialog

If a software update is available for Map File Browser you'll see the software update dialog.

Software update download confirmation		×
A software update is available for MapFileBrowse	r.	
Do you wish to install MapFileBrowser V2.74?		
Yes, Download and Install	Don't download, ask me later	Skip this version, don't tell me about it again
Software update		
Software update options		

Download and install > downloads the update, showing progress

Downloading MapFileBrowser 2.74	×
Downloading MapFileBrowser 2.74	
	Stop
3832 KB of 15241 KB (25.1%)	

Once the update has downloaded, Map File Browser will close, run the installer, and restart.

You can stop the download at any time, if necessary.

- Don't download... > Doesn't download, but you'll be prompted for it again next time you start Map File Browser
- Skip this version... > Doesn't download the update and doesn't bother you again until there's an even newer update
- Software update options... > edit the software update schedule

#### Problems downloading or installing?

If for whatever reason, automatic download and installation fails to complete:

• Download the latest installer manually from the software verify website.

Make some checks for possible scenarios where files may be locked by Map File Browser as follows:

- Ensure Map File Browser and its help manual is also closed
- Ensure any error dialogs from the previous installation are closed

You should now be ready to run the new version.

## Software update schedule

Map File Browser can automatically check to see if a new version of Map File Browser is available for downloading.

**Software Updates** menu **> Configure software updates >** shows the software update schedule dialog

The update options are:

- never check for updates
- check daily (the default)
- check weekly
- check monthly

The most recent check for updates is shown at the bottom.

Software update schedule	?	$\times$
OutputDebugStringChecker can check for software updates on	a regular ba	asis.
O Never check for updates		
C Check for updates every day		
Check for updates once a week		
$\ensuremath{\mathbb{C}}$ Check for updates once a month		
Most recent check for updates was performed on 2022-07-23		
OK	Car	ncel

### Software update directory

It's important to be able to specify where software updates are downloaded to because of potential security risks that may arise from allowing the TMP directory to be executable. For example, to counteract security threats it's possible that account ownership permissions or antivirus software blocks program execution directly from the TMP directory.

The TMP directory is the default location but if for whatever reason you're not comfortable with that, you can specify your preferred download directory. This allows you to set permissions for TMP to deny execute privileges if you wish.

**Software Updates** menu **Set software update directory shows** the Software update download directory dialog

Software update download directory	?	$\times$
Software updates will be downloaded to the location specified below:		
C:\Users\stephen\AppData\Local\Temp	Brows	e
	Rese	et
OK	Cano	el

An invalid directory will show the path in red and will not be accepted until a valid folder is entered.

Example reasons for invalid directories include:

- the directory doesn't exist
- the directory doesn't have write privilege (update can't be downloaded)
- the directory doesn't have execute privilege (downloaded update can't be run)

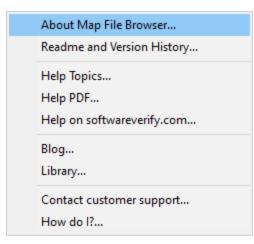
When modifying the download directory, you should ensure the directory will continue to be valid. Updates may no longer occur if the download location is later invalidated.

• Reset > reverts the download location to the user's TMP directory

The default location is c:\users\[username]\AppData\Local\Temp

## 5.5 Help

The Help menu controls displaying this help document and displaying information about Map File Browser.



Help menu > About Map File Browser... > displays information about Map File Browser.

Help menu > Readme and Version History... > displays the readme and version history.

Help menu > Help Topics... > displays this help file.

**Help** menu **> Help PDF... >** displays this help file in PDF format.

**Help** menu **> Help on softwareverify.com... >** display the Software Verify documentation web page where you can view online documentation or download compiled HTML Help and PDF help documents.

**Help** menu **> Blog... >** display the Software Verify blog.

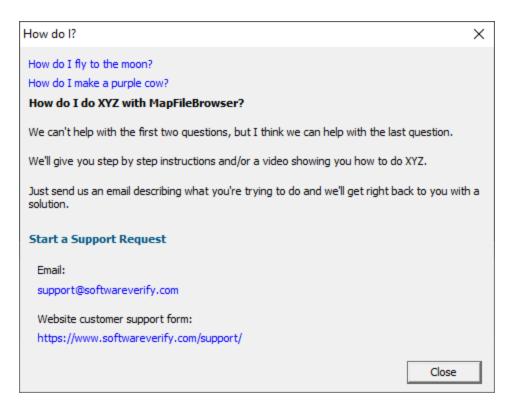
**Help** menu **> Library... >** display the Software Verify library - our best blog articles grouped by related topics.

**Help** menu **> Contact customer support... >** displays the options for contacting customer support.

Contact Software Verify Customer Support	Х		
We provide customer support via email.			
Email allows us to exchange detailed bug descriptions, detailed instructions, screenshots, files, crash dumps and other metadata that can't easily be communicated via telephone or			
Should your support request require escalation to phone, Zoom or remote computer access we will do that when required.			
There are two methods to start a support request:			
Email:			
support@softwareverify.com			
Website customer support form:			
https://www.softwareverify.com/support.php			
Close			

Click a link to contact customer support.

**Help** menu **> How do I?... >** displays the options for asking us how to do a particular task.





# 6 The user interface

The MapFileBrowser user interface is shown below.

pfile: E:\orr	n/c/memory32/examples/unativeExample	e\DebugNonLink6_0\nativeExample.map						Name Filter;	ER	er
Name (634)		A	Segment	Section	Address	Size	Type	Filename	Line	
0	public:thiscall exampleIncreme	ntDecrement::exampleIncrementDecrement(void)	0x0001	.text	0x0040A0A0	6	4 Public	E:\om\c\memory32\examples\nativeExample\TESTSVW.CPP		227
1	public: _thiscall heapNewBaseCla	ss::heapNewBaseClass(void)	0x0001	.text	0x00401040	6	4 Public	E:\om\c\memory32\examples\nativeExample\heapNewBaseClass.cpp		72
2	public:thiscall std::_Tree <unsign< td=""><td>ned long,struct std::pair<unsigned ,unsigned="" const="" lo<="" long="" td=""><td>0x0001</td><td>.text</td><td>0x004299E0</td><td>9</td><td>6 Public</td><td>C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree</td><td></td><td>122</td></unsigned></td></unsign<>	ned long,struct std::pair <unsigned ,unsigned="" const="" lo<="" long="" td=""><td>0x0001</td><td>.text</td><td>0x004299E0</td><td>9</td><td>6 Public</td><td>C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree</td><td></td><td>122</td></unsigned>	0x0001	.text	0x004299E0	9	6 Public	C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree		122
3	public:thiscall svIDataTracker::sv	/IDataTracker(char const *)	0x0001	.text	0x0042A700	9	6 Public	E:\om\c\memory32\API\sviDataTracker.cpp		17
4	public:thiscall svlDataTracker::sv	/DataTracker(unsigned short const *)	0x0001	.text	0x0042A760	9	6 Public	E:\om\c\memory32\API\svIDataTracker.cpp		33
5	public:thiscall testClass1::testCl	ass1(unsigned long)	0x0001	.text	0x00402A80	6	4 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.h		30
5	public:thiscall testClass2::testCl	ass2(unsigned short)	0x0001	.text	0x00402AC0	9	6 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.h		42
7	public: _thiscall testClass3::testCl	ass3(void)	0x0001	.text	0x00402B20	6	4 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.h		54
в	public:thiscall testClassToBeThr	own::testClassToBeThrown(class testClassToBeThrown con	0x0001	.text	0x00402120	19	2 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.CPP		149
9	public:thiscall testClassToBeThr	own::testClassToBeThrown(void)	0x0001	.text	0x004020A0	12	8 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.CPP		142
D	private:thiscall type_info::type_info(class type_info const &)		0x0001	.text	0x0042E660	6	4 Public	ti_inst.cpp		13
1	public:thiscall std::_Tree <unsigned ,unsigned="" const="" lo<="" long="" long,struct="" std::pair<unsigned="" td=""><td>0x0001</td><td>.text</td><td>0x00427D80</td><td>48</td><td>0 Public</td><td>C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree</td><td></td><td>174</td></unsigned>		0x0001	.text	0x00427D80	48	0 Public	C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree		174
2	public:thiscall ATL::CComBSTR:	~CComBSTR(void)	0x0001	.text	0x00421820	4	8 Public	C:\Program Files (x86)\Microsoft Visual Studio\VC98\ATL\INCLUDE\atlba	3	943
3	public:thiscall ATL::CComVarian		0x0001	.text	0x00421B20	6	4 Public	C:\Program Files (x86)\Microsoft Visual Studio\VC98\ATL\INCLUDE\atIba	4	175
4	public: virtualthiscall CMainFran	me::~CMainFrame(void)	0x0001	.text	0x00401880	9	6 Public	E:\om\c\memory32\examples\nativeExample\MAINFRM.CPP		56
	ddress Offset 0042A700 0 0042A70E 14	Fiename: E:Vom/c/memoy32/APN:v/DataTracker.cpp Line Function: public:thiscal:v/DataTracker:v/DataTracker( Address: 0x0042A700 9 // When this object is dele	char const "]	an tracker	will be autor	matically m	oonned fr	om the stack of tag trackers.	Cop	29
	0042A71A 26	10 // The top most rag tracker 11 // TrackerName - In- 12 // trackerName - In- 14 // FRIUM. COOS	is used ne of tag :ker(const	tö tag all tracker	locations.					

The user interface consists of a main grid showing functions in the MAP file.

Below is a display for line numbers and a source code display for viewing the source code of any function or variable that is selected.

Selecting any item in the grid populates the line numbers and source code display as appropriate.

Querying any value will select the nearest item in the main grid and populate the other displays as appropriate.

Some basic filtering functionality is also provided.

### **MAP File Information**

#	Name (634)	Segment	Section	Address	Size	Туре	Filename	Line ^
30	public: _thiscall exampleIncrementDecrement:exampleIncrementDecrement(void)	0x0001	.text	0x0040A0A0	6	4 Public	E:\om\c\memory32\examples\nativeExample\TESTSVW.CPP	227
31	public:thiscall heapNewBaseClass::heapNewBaseClass(void)	0x0001	.text	0x00401040	6	4 Public	E:\om\c\memory32\examples\nativeExample\heapNewBaseClass.cpp	72
32	public: _thiscall std::_Tree <unsigned ,unsigned="" const="" lo<="" long="" long,struct="" std::pair<unsigned="" td=""><td>0x0001</td><td>.text</td><td>0x004299E0</td><td>9</td><td>6 Public</td><td>C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree</td><td>122</td></unsigned>	0x0001	.text	0x004299E0	9	6 Public	C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree	122
33	public:thiscall svlDataTracker:svlDataTracker(char const *)	0x0001	.text	0x0042A700	9	6 Public	E:\om\c\memory32\API\svIDataTracker.cpp	17
34	public: _thiscall svlDataTracker:svlDataTracker(unsigned short const *)	0x0001	.text	0x0042A760	9	6 Public	E:\om\c\memory32\API\svIDataTracker.cpp	33
35	public:thiscall testClass1::testClass1(unsigned long)	0x0001	.text	0x00402A80	6	4 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.h	30
36	public: _thiscall testClass2::testClass2(unsigned short)	0x0001	.text	0x00402AC0	9	6 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.h	42
37	public: _thiscall testClass3::testClass3(void)	0x0001	.text	0x00402B20	6	4 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.h	54
38	public:thiscall testClassToBeThrown::testClassToBeThrown(class testClassToBeThrown con	0x0001	.text	0x00402120	19	2 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.CPP	149
39	public: _thiscall testClassToBeThrown::testClassToBeThrown(void)	0x0001	.text	0x004020A0	12	8 Public	E:\om\c\memory32\examples\nativeExample\nativeExample.CPP	142
40	private:thiscall type_info::type_info(class type_info const &)	0x0001	.text	0x0042E660	6	4 Public	ti_inst.cpp	17
41	public: _thiscall std::_Tree <unsigned ,unsigned="" const="" lo<="" long="" long,struct="" std::pair<unsigned="" td=""><td>0x0001</td><td>.text</td><td>0x00427D80</td><td>48</td><td>0 Public</td><td>C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree</td><td>174</td></unsigned>	0x0001	.text	0x00427D80	48	0 Public	C:\Program Files (x86)\Microsoft Visual Studio\VC98\INCLUDE\xtree	174
42	public:thiscall ATL::CComBSTR::~CComBSTR(void)	0x0001	.text	0x00421820	4	8 Public	C:\Program Files (x86)\Microsoft Visual Studio\VC98\ATL\INCLUDE\atIba	. 3943
43	public:thiscall ATL::CComVariant::~CComVariant(void)	0x0001	.text	0x00421B20	6	4 Public	C:\Program Files (x86)\Microsoft Visual Studio\VC98\ATL\INCLUDE\atIba	4175
44	public: virtualthiscall CMainFrame:~CMainFrame(void)	0x0001	.text	0x00401880	9	6 Public	E:\om\c\memory32\examples\nativeExample\MAINFRM.CPP	56 🗸

The MAP File information shows you the symbol name, segment number, segment name, symbol address, symbol size, symbol type, and the filename and line number for the symbol.

You can sort the data by clicking no the column header and clicking again to reverse the direction of the sort.

If you select any item in the grid the lower grids and source code display are populated with data as appropriate.

If you right click any item a context is displayed which will allow you to perform a symbol relative query.

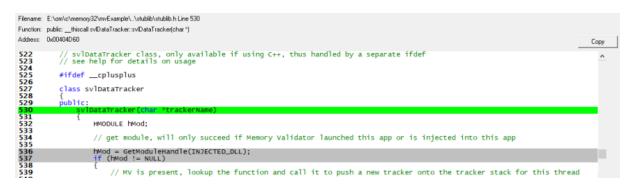
cvExample.exe:Fri Jul 03 10:09:24 2020 //

#### Line Numbers

Line #	Address	Offset
530	0x00404D60	0
536	0x00404D7F	31
537	0x00404D96	54
545	0x00404D9C	60
546	0x00404DB7	87
548	0x00404DBD	93
551	0x00404DD0	112

The line numbers section lists each line number, the address of that line and the offset of that line from the start of the owning function. Note that offsets can be negative as well as positive depending on how the compiler did it's work.

#### Source Code



The source code section displays the source code, highlights the selected line and displays information relating to filename, line number, function and address.

#### Filters

Name Filter:	TestClassNewHeap	Filter	
--------------	------------------	--------	--

The filters section allows you to filter data by symbol name.

#### Name Filter

Filtering by symbol name allows you to easily find a particular symbol. This is very useful when wanting to decode a crash address that has been provided as relative to a symbol (symbol + offset).

Map file: E:\or	Map Res E Loml-Chemony32/examples/haliveExample/DebugNorLink6_OnativeExample.map File:						
#	Name (4)	Segment	Section	Address	Size	Туре	Filename
1	public:thiscall TestClassNewHeap::TestClassNewHeap(void)	0x0001	.text	0x0040A840	96	Public	E:\om\c\memory32\examples\nativeExample\TESTSVW.CPP
2	public: virtualthiscall TestClassNewHeap::~TestClassNewHeap(void)	0x0001	.text	0x0040A920	96	Public	E:\om\c\memory32\examples\nativeExample\TESTSVW.CPP
3	public: void _thiscall TestClassNewHeap::setData1(unsigned long)	0x0001	.text	0x0040D920	64	Public	E:\om\c\memory32\examples\nativeExample\TESTSVW.CPP
4	public: voidthiscall TestClassNewHeap::setData2(unsigned long)	0x0001	.text	0x0040D960	64	Public	E:\om\c\memory32\examples\nativeExample\TESTSVW.CPP



# 7 Settings dialog

The settings dialog allows you to control the behaviour of Map File Browser.

There are three main sections: Symbols, Data display and source code paths.

Settings		?	×
Settings Symbols Data Display Source Paths Path Substitutions	Symbols         Image: Display symbols as human readable names         Image: Display symbols as compiler decorated names		
	Reset All Reset Help (F1) OK	Can	cel

# 7.1 Symbols

The Symbols settings allow you to choose how symbols are displayed.

ttings		?	×
iettings <mark>Symbols</mark> Data Display Source Paths Path Substitutions	Symbols <ul> <li>Display symbols as human readable names</li> <li>Display symbols as compiler decorated names</li> </ul>		

Depending upon the task you are performing you may wish to read the symbols as human readable symbols or compiler decorated C++ symbols.

**Reset -** Resets the settings on the current page.

**Reset All** - Resets all global settings, not just those on the current page.

# 7.2 Data display

The Symbols settings allow you to choose how symbols are displayed.

Settings			?	×
Settings Symbols Source Paths Path Substitutions	Data Display			
	Reset All Reset Help (F1)	ОК	Can	cel

For MAP files that have line numbers you may wish to view additional information as well as the line numbers.

- If line numbers in MAP file > display all information relating to symbols that have line information
  - **Display Imported Functions >** functions imported from other DLLs will be displayed
  - **Display Compiler Generated Code** > compiler generated code will be displayed
  - **Display Linked Code >** linked object code from .obj and .lib will be displayed
  - Display Functions that have no size > symbols with no size will be displayed
- If no line numbers in MAP file > display all information that was found in the MAP file

**Reset -** Resets the settings on the current page.

**Reset All** - Resets all global settings, not just those on the current page.

## 7.3 Source code paths

The Source code paths settings allow you to specify where Map File Browsers looks for source code files.

Settings		?	×
Settings Symbols	Source Search Paths		
Data Display <mark>Source Paths</mark>	Path	Add	
Path Substitutions	E:\om\c\buildNumber	Remove	
		Remove	хII
		Set Defau	ilt
	Reset All Reset Help (F1) OK	Can	cel

### Manually adding path type directories

The Path list shows all the paths that will be searched for debug information in PDB files.

You can modify the list of files for each path type in the following ways:

• Add > appends a row to the directory list > enter the directory path

Edit a directory path by double clicking the entry. The usual controls apply for removing list items:

- **Remove** > removes selected items from the list
- Remove All > clears the list
- Set Default > populates the list with each directory found in the PATH environment variable

Alternatively, press Del to delete selected items, and Ctrl + A to select all items in the list first.

**Reset -** Resets the settings on the current page.

**Reset All** - Resets all global settings, not just those on the current page.

## 7.4 Path Substitutions

The **Path Substitutions** tab allows you to specify file path substitutions to handle copying builds from build machines to development or test machines .

The default settings are shown below:

Settings				?	×
Settings Symbols	Path Substitutio	ns			
Data Display Source Paths	New Path	Old Path		Add	
Path Substitutions	G:\dev\build	e:\dev\build		Remove	
				Remove	All
	Reset All	Reset Help (F1)	ОК	Can	cel

### Path Substitutions

Some software development schemes have multiple rolling builds of their software, often enabled by using substituted disk drive naming schemes.

When you download the build to your development machine for development and testing, debugging information may reference disk drives that don't exist on your machine, for example, drive X: while your machine only has C:, D:, and E: drives.

Or you may just be copying a build from a drive on a development machine to a subdirectory on a drive on your test machine.

These options let you remap the substitution so that the MAP File Browser looks in the correct place for the source code.

Add > adds a row to the File Paths Substitutions table > enter the new path that will replace the old path in the New Path column > click in the Old Path column > enter the path that is being replaced

For example, you might enter c:\users\stephen\documents for the new path and  $f:\dev\build$  for the old path.

You can double click to edit drives and paths in the table, or remove items:

- Remove > removes selected substitutions from the list
- Remove All > removes all substitutions from the list

Alternatively, press Dell to delete selected items, and Ctrl + A to select all items in the list first.

Example: Changed disk drive	
Project originally located at	m:∖dev∖build∖testApp
Project copied to	e:∖dev\build\testApp
New Path	e:\
Old Path	m:\
Example: Project copied to a new location	
Project originally located at	f·\dev\build\testApp

Project originally located at Project copied to New Path Old Path f:\dev\build\testApp C:\Users\Stephen\Documents\testApp C:\Users\Stephen\Documents f:\dev\build

The slashes do not have to match, a forward slash will match a backslash when comparing path fragments. This is deliberate - to improve ease of use with libraries built by different compilers (LLVM and compilers that use it use forward slashes, whereas Visual Studio etc use backslashes).

**Reset All** - Resets all global settings, not just those on the current page.

**Reset -** Resets the settings on the current page.

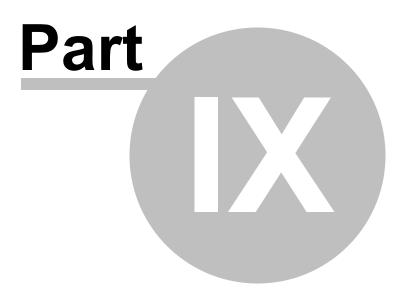


# 8 MAP File Information

The MAP File Information dialog displays information about a MAP file that is not displayed by the main display.

This information includes the DLL/EXE name, timestamp, preferred load address, entry point and a list of the segments that make up the binary image.

Aap File Information	n				×
Object F	File: nativeExam	ple			
Timesta	mp: 61811c2d (	Tue Nov 02 11:08:29 2021)			
Preferred Load Addre	ess: 0x0040000	0			
Entry Pr	oint: 0x0001:0x0	0016840			
Segments: Segment (21)	Section	Address	Length	Class	^
1	.text	0x0001:0x00000000		CODE	
2	.text\$x	0x0001:0x00017320	5565	CODE	
3	.idata\$5	0x0002:0x00000000	2024	DATA	
4	.rdata	0x0002:0x000007E8	9544	DATA	
5	.rdata\$r	0x0002:0x00002D30	68	DATA	
6	.xdata\$x	0x0002:0x00002D78	7088	DATA	
7	.idata\$2	0x0002:0x00004928	320	DATA	
8	.idata\$3	0x0002:0x00004A68	20	DATA	
9	.idata\$4	0x0002:0x00004A7C	2024	DATA	v
<			i		>
					Close



# 9 How to use MapFileBrowser

# Load MAP File Information

Use the File > Load MAP file... option to load the appropriate MAP file.

The grid displays various attributes of each symbol in the MAP file. You can sort the grid by clicking the appropriate column header. Click the same header to reverse the sort order.

Select a symbol to see information about the line numbers and source code.

# Filtering

You can filter by name by typing the name into the **Name Filter** box and clicking the **Filter** button to perform the filtering.

# Viewing function data

As each item in the list is selected the Line Numbers are updated and the source code display updates to show the source code for the function. All lines in the function that contain executable code (as indicated by the line number information) are coloured grey. The current line for the function is coloured bright green.

# **Querying data**

You can query data by using the two Query fields below the main grid.

# **Relative query**

Type the relative address (also know as address offset) into the Query by Offset field, then click Query. The symbol information is displayed.

The field accepts decimal or hexadecimcal values. Hex values must be prefixed with 0x.

# **Absolute query**

Type the absolute address into the Query by Address field, type the absolute DLL load address into the Alternate Load Address field, then click Query. The symbol information is displayed.

The fields accept decimal or hexadecimcal values. Hex values must be prefixed with 0x.

# 9.1 Decoding an absolute crash address

## Scenario:

A customer has supplied you with a crash report containing a callstack with addresses. The callstack also indicates which module relates to which address. The customer has also supplied you with a list of module load addresses.

## Example Data:

```
Exception code: C0000005 ACCESS VIOLATION
Fault address: 0x005f5eec (base 0x00400000) C:\Program Files (x86) \Software Verification \
Exception Parameters:
   0: 0x00000000 [Read Error]
   1: 0x035f0034 [Address]
Registers:
   EAX:035F0034
   EBX:0000000
   ECX:FFFDD000
   EDX:00002370
  ESI:006F7D58
  EDI:035F0034
  CS:EIP:0023:005F5EEC
   SS:ESP:002B:0018FE14 EBP:0018FE3C
   DS:002B ES:002B FS:0053 GS:002B
   Flags:00010202
StackTrace
C:\Program Files (x86)\Software Verification\C++ Memory Validator\memoryValidator.exe : 0x
C:\Program Files (x86)\Software Verification\C++ Memory Validator\memoryValidator.exe : 0x
C:\Windows\syswow64\msvcrt.dll : 0x75D70000 : 0x75D7C3E4
C:\Windows\syswow64\msvcrt.dll : 0x75D70000 : 0x75D836B6
C:\Program Files (x86)\Software Verification\C++ Memory Validator\memoryValidator.exe : 0x
C:\Windows\syswow64\kernel32.dll : 0x754D0000 : 0x754E3365
C:\Windows\SysWOW64\ntdll.dll : 0x77920000 : 0x77959F6D
C:\Windows\SysWOW64\ntdll.dll : 0x77920000 : 0x77959F40
```

C:\Windows\SysWOW64\ntdll.dll : 0x77920000 : 0x77959F40

This is data from a real crash a few years ago, from C++ Memory Validator 5.80.

## Question:

How do you decode these absolute addresses?

## Answer:

In the above data we can see a callstack containing entries for ntdll.dll, msvcrt.dll, and memoryValidator.exe.

All the modules are Microsoft DLLs except for the EXE, which is part of C++ Memory Validator, one of our tools.

To decode these values, we load memoryValidator.map into MapFileBrowser, then for each symbol we take the following actions.

For our purposes here, we're going to show how to convert one symbol. We're going to use the first symbol from memoryValidator.exe in the example data above.

0x005f5eec (base 0x00400000)

The address is 0x005f5eec. The DLL loaded at 0x00400000. You'll notice the load address for all MemoryValidator.exe entries is 0x00400000.

From the Query menu choose Find Symbol with Absolute Address....

Find Symbol with Absolute Address
Find Symbol with DLL Relative Address
Find Symbol with Symbol Relative Address
Find Symbol from Event Viewer XML crash log
View Map File Information

The Query Symbol by Absolute Address dialog is displayed.

Query Symbol by A	Absolute Address	?	×
DLL:	mvExample		
DLL Load Address:			
	What's a load address?		
Address:			
	Find Symbol		Cancel

Type the DLL load address into the DLL Load Address field. Prefix any hexadecimal addresses with 0x.

Type the symbol address into the Address field. Prefix any hexadecimal addresses with 0x.

Click the Find Symbol button.

Query Symbol by A	Absolute Address	?	×
DLL:	mvExample		
DLL Load Address:	0x00400000		
	What's a load address?		
Address:	0x005f5eec		
	Find Symbol	C	ancel

The appropriate location in the code is found and displayed.

Filename:	: E:\om\c\memory32\tabserv\MemorySettingData.cpp Line 5365	
Function:	private; intthiscall MemorySettingData;:saveSoftwareUpdateInformation[struct HKEY* const.]	
Address:	UXUUGFBEEL	Сору
5358	//-NAME	
5359	// DESCRIPTION	^
5360	//.DARAMETERS.	
5361	// PARAMIE LENS. // RETURN.CODES.	
5362	//.ĸEIUKN.CODES	
	//	
5363		
5364	int MemorySettingData::saveSoftwareUpdateInformation(const HKEY &hKey)	
5365		
5366	int ok = TRUE;	
5367		
5368	ok &= writeData(hKey, _T("suea"), softwareUpdateEmailAddress); // stephen@softwareverify.com	
5369	ok &= writeData(hKey, _T("sup"), softwareUpdatePassword); // tester	
5370	ok &= writeData(hkey, _T("suea"), softwareUpdateEmailAddress); // stephen@softwareverify.com ok &= writeData(hkey, _T("sup"), softwareUpdatePassword); // tester ok &= writeData(hkey, _T("sus"), (DWQRD &)softwareUpdateSchedule);	
5371	ok &= writeData(hKey, _T("sulc"), softwareUpdateLastCheck);	
5372		
5373	return ok:	
<		
<		,

### Results:

Repeating the process for the data shown above resulted in this information.

```
0x005f5eec (base 0x0040000) C:\Program Files (x86)\Software Verification\C++ Memory Valid
C:\Program Files (x86)\Software Verification\C++ Memory Validator\memoryValidator.exe : 0x
C:\Program Files (x86)\Software Verification\C++ Memory Validator\memoryValidator.exe : 0x
C:\Windows\syswow64\msvcrt.dll : 0x75D70000 : 0x75D7C3E4
C:\Windows\syswow64\msvcrt.dll : 0x75D70000 : 0x75D836B6
C:\Program Files (x86)\Software Verification\C++ Memory Validator\memoryValidator.exe : 0x
C:\Windows\syswow64\kernel32.dll : 0x754D0000 : 0x754E3365
C:\Windows\SysWOW64\ntdll.dll : 0x77920000 : 0x77959F6D
C:\Windows\SysWOW64\ntdll.dll : 0x77920000 : 0x77959F40
C:\Windows\SysWOW64\ntdll.dll : 0x77920000 : 0x77959F40
```

Help! I have a crash address but I don't know what the load address is? What do I do?

You need to read about load addresses.

# 9.2 Decoding a relative crash address

# Scenario:

A customer has supplied you with a crash report containing a callstack with relative offsets from DLLs. The callstack also indicates which module relates to which address.

## **Example Data:**

```
Exception code: C0000005 ACCESS VIOLATION
Fault offset: 0x00036FA3 C:\WINDOWS\system32\MSVCRT.dll
Exception Parameters:
   0: 0x00000000 [Read Error]
   1: 0x5f8f2000 [Address]
Registers:
   EAX:B3BEB6D4
  EBX:5F8CB6C8
  ECX:1508E585
  EDX:0000000
  ESI:5F8F2000
  EDI:01B98DEC
  CS:EIP:001B:77C46FA3
   SS:ESP:0023:0012F158 EBP:0012F160
  DS:0023 ES:0023 FS:003B GS:0000
  Flags:00010212
```

StackTrace

```
C:\WINDOWS\system32\MFC42u.DLL : 0x0000270a
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x000db989
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x000db1f8
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00121a83
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00121b7e
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00174ec5
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00174ec5
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00175094
C:\WINDOWS\system32\MFC42u.DLL : 0x00013724
C:\WINDOWS\system32\MFC42u.DLL : 0x0001b31
C:\WINDOWS\system32\MFC42u.DLL : 0x0008cba7
```

This is data from a real crash many years ago.

## Question:

There are no DLL load addresses and the addresses aren't addresses, but offsets from the start of a DLL. How do you decode these relative offsets?

#### Answer:

In the above data we can see a callstack containing entries for mfc42u.dll, and memoryValidator.exe.

All the modules are Microsoft DLLs except for the EXE, which is part of C++ Memory Validator, one of our tools.

To decode these values, we load memoryValidator.exe into MapFileBrowser.exe, then for each symbol we take the following actions.

For our purposes here, we're going to show how to convert one symbol. We're going to use the first symbol from memoryValidator.exe in the example data above.

C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x000db989

The relative address (or offset) is 0x000db989. We don't know the DLL load address.

From the Query menu choose Find Symbol with DLL Relative Address....

Find Symbol with Absolute Address
Find Symbol with DLL Relative Address
Find Symbol with Symbol Relative Address
Find Symbol from Event Viewer XML crash log
View Map File Information

The Query Symbol by Absolute Address dialog is displayed.

Query Sy	mbol by Relative Offset	?	×
DLL:	mvExample		
Offset:			
	Find Symbol	Ca	ncel

Type the relative address into the Offset field. Prefix any hexadecimal addresses with 0x.

Click the Find Symbol button.

Query Sy	mbol by Relative Offse	et	?	×
DLL:	mvExample			
Offset:	0x000db989			
		Find Symbol	Cance	el

The appropriate location in the code is found and displayed. In this example MapFileBrowser could not locate the source code (as the file location is not valid on this machine)

Filename: E\\om\c\coverageValidator\common\stackStuff.cpp Line 736	
Function: voidcdecl doStackTraceString(unsigned short * *_unsigned long,unsigned long,unsigned short * *_int,int,int, struct _CONTEXT *_int,int)	
Address: 0x1003486F	Сору
<pre>Unable to open file:E:\om\c\coverageValidator\common\stackStuff.cpp Unable to open file:E:\om\c\coverageValidator\common\stackStuff.cpp Unable to open file:E:\om\c\coverageValidator\common\stackStuff.cpp Interval to open file:E:\om\c\coverageValidator\coverageV</pre>	Î

Results:

Repeating the process for the data shown above resulted in this information.

```
Exception code: C0000005 ACCESS VIOLATION
Fault offset: 0x00036FA3 C:\WINDOWS\system32\MSVCRT.dll
Exception Parameters:
   0: 0x00000000 [Read Error]
   1: 0x5f8f2000 [Address]
Registers:
  EAX:B3BEB6D4
   EBX:5F8CB6C8
   ECX:150BE5B5
  EDX:0000000
  ESI:5F8F2000
  EDI:01B98DEC
   CS:EIP:001B:77C46FA3
   SS:ESP:0023:0012F158 EBP:0012F160
   DS:0023 ES:0023 FS:003B GS:0000
  Flags:00010212
StackTrace
C:\WINDOWS\system32\MFC42u.DLL : 0x0000270a
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x000db989
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x000db1f8
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00121a83
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00121b7e
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00174ec5
C:\Program Files\Software Verification\Memory Validator\memoryValidator.exe : 0x00175094
C:\WINDOWS\system32\MFC42u.DLL : 0x00013724
C:\WINDOWS\system32\MFC42u.DLL : 0x00014245
C:\WINDOWS\system32\MFC42u.DLL : 0x00001b31
C:\WINDOWS\system32\MFC42u.DLL : 0x0008cba7
```

# 9.3 Decoding a symbol relative crash address

## Scenario:

A customer has supplied you with a crash report containing a callstack with symbol relative offsets from DLLs. The callstack also indicates which module relates to which address.

## **Example Data:**

```
ntoskrnl.exe!KeSvnchronizeExecution+0x2246
ntoskrnl.exe!KeWaitForMultipleObjects+0x135e
ntoskrnl.exe!KeWaitForMultipleObjects+0xdd9
ntoskrnl.exe!KeWaitForSingleObject+0x373
ntoskrnl.exe!KeStallWhileFrozen+0x1977
ntoskrnl.exe! misaligned access+0x13f9
ntoskrnl.exe!KeWaitForMultipleObjects+0x152f
ntoskrnl.exe!KeWaitForMultipleObjects+0xdd9
ntoskrnl.exe!KeWaitForSingleObject+0x373
ntoskrnl.exe!NtWaitForSingleObject+0xb2
ntoskrnl.exe!setjmpex+0x34a3
ntdll.dll!ZwWaitForSingleObject+0xa
KERNELBASE.dll!WaitForSingleObjectEx+0x98
svlcoveragevalidatorstub.dll!sendCommandLineAndStartTimeToGUI+0x2868
svlcoveragevalidatorstub.dll!setValidatorFeedbackHookingComplete+0x1fa6
svlcoveragevalidatorstub.dll!svl sendMessageRawToUserInterface+0x21837
svlcoveragevalidatorstub.dll!svl sendMessageRawToUserInterface+0x218cb
KERNEL32.DLL!BaseThreadInitThunk+0x22
ntdll.dll!RtlUserThreadStart+0x34
```

This is real data from a bug at Software Verify Ltd. This is one thread from many in a dump relating to a deadlock bug we were investigating.

## Question:

How do you decode these symbol relative offsets?

## Answer:

In the above data we can see a callstack containing entries for ntoskrnl.exe, ntdll.dll, kernelbase.dll, kernel32.dll and svlcoveragevalidatorstub.dll.

All the modules are Microsoft DLLs except for one DLL, which is part of C++ Coverage Validator, one of our tools.

To decode these values, we load svlCoverageValidatorStub.dll into MapFileBrowser, then for each symbol we take the following actions.

For our purposes here, we're going to show how to convert one symbol. We're going to use the first symbol from svlCoverageValidatorStub.dll in the example data above.

svlcoveragevalidatorstub.dll!sendCommandLineAndStartTimeToGUI+0x2868

Type the symbol name into the **Name Filter** field, then click **Filter**. This makes it easy to find the symbol we want.

pvExample.exe (Running)

Once we have found the symbol, right click on the symbol to display the context menu and choose **Offset from this symbol...**.

cvExample.exe:Fri Jul 03 10:09:24 2020

An alternate method is to click on the symbol to select it, then from the Query menu choose **Find Symbol with Symbol Relative Address...** 

Or, from the Query menu choose **Find Symbol with Symbol Relative Address...** then choose the symbol you want from the combo box.

Find Symbol with Absolute Address
Find Symbol with DLL Relative Address
Find Symbol with Symbol Relative Address
Find Symbol from Event Viewer XML crash log
View Map File Information

Type the offset into the dialog (hex values must be prefixed with 0x) and click OK.

Query Syr	mbol by Symbol Offset		?	×
Symbol:	sendCommandLineAndStartTimeToGUI			•
Offset:	0x00002868			
	Find	d Symbol	Cancel	

The appropriate location in the code is found and displayed.

	:\om\c\coverageValidator\common\stackStuff.cpp Line 736 aidcdecl doStackTraceString[unsigned short * ".unsigned long.unsigned long.unsigned long.unsigned short * ".int,int,int,struct_CDNTEXT *.int,int)	
Address: 0	x1003486F	Сору
1 23 34 56 67 89 10 11 12 13 14 15 16 17 2	Unable to open file:E:\om\c\coverageValidator\common\stackStuff.cpp	

## Results:

Repeating the process for the data shown above resulted in this information.

```
svlcoveragevalidatorstub.dll!sendCommandLineAndStartTimeToGUI+0x2868doStackT:svlcoveragevalidatorstub.dll!setValidatorFeedbackHookingComplete+0x1fa6stubSendsvlcoveragevalidatorstub.dll!svl_sendMessageRawToUserInterface+0x21837memcpysvlcoveragevalidatorstub.dll!svl_sendMessageRawToUserInterface+0x218cbwcscpy
```

# 9.4 Decoding an Event Viewer XML crash log

# Scenario:

A customer has supplied you with data from Windows Event Viewer about a crash. The log contains XML and you don't know which values are relevant.

The event log data will have a provider name of "Windows Error Reporting" or "Application Error".

The XML data is found on the "Details" tab with the XML View radio box selected.

#### **Example Data:**

```
<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  <System>
   <Provider Name="Application Error" />
   <EventID Qualifiers="0">1000</EventID>
   <Level>2</Level>
   <Task>100</Task>
   <Keywords>0x80000000000000/Keywords>
   <TimeCreated SystemTime="2020-02-11T10:42:39.000000002" />
   <EventRecordID>260330</EventRecordID>
   <Channel>Application</Channel>
   <Computer>hydra</Computer>
   <Security />
  </Svstem>
  <EventData>
   <Data>testDeliberateCrashVS6.exe</Data>
   <Data>1.0.0.1
   <Data>5e42850d/Data>
   <Data>testDeliberateCrashVS6.exe</Data>
   <Data>1.0.0.1</Data>
   <Data>5e42850d</Data>
   <Data>c0000005</Data>
   <Data>00001d07</Data>
   <Data>1490</Data>
   <Data>01d5e0c7fa70e745/Data>
   <Data>E:\om\c\testApps\testDeliberateCrashVS6\Debug\testDeliberateCrashVS6.exe</Data>
   <Data>E:\om\c\testApps\testDeliberateCrashVS6\Debug\testDeliberateCrashVS6.exe</Data>
   <Data>390bde30-4cbb-11ea-83d3-001e4fdb3956
   <Data />
   <Data />
  </EventData>
</Event>
```

This is data from a test program that is designed to crash.

## Question:

There the event log indicates a DLL, but no load address, two different addresses, an exception code and an offset from the start of the DLL. How do you decode this relative offset?

## Answer:

MapFileBrowser has an option specifically for this occasion.

The XML data indicates the crash happened in **testDeliberateCrashVS6.exe**. Load this into MapFileBrowser being sure to load the correct build version and that the PDB file can be found so that symbols get loaded.

From the Query menu choose Find Symbol from Event Viewer XML crash log....

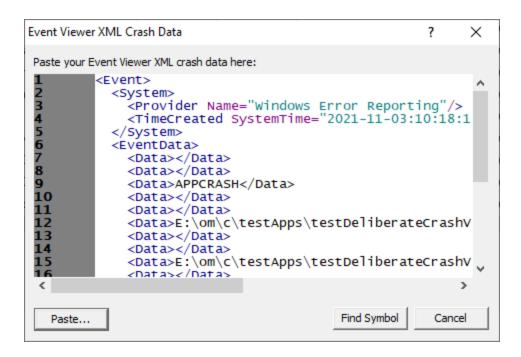
Find Symbol with Absolute Address
Find Symbol with DLL Relative Address
Find Symbol with Symbol Relative Address
Find Symbol from Event Viewer XML crash log
View Map File Information

The Query Symbol by Absolute Address dialog is displayed.

Event Viewer XML Crash Data	 ?	×
Paste your Event Viewer XML crash data here:		
Paste Find Symb	Cancel	

Paste the XML data from the Event Viewer into the text field.

Click the Find Symbol button.



The appropriate location in the code is found and displayed.

```
Filename: E:\om\c\testApps\testDeliberateCrashVS6\testDeliberateCrashVS6DIg.cpp Line 178
Function: private: void __thiscall CTestDeliberateCrashVS6DIg::causeACrash(void)
Address: 0x004029C0
170
           }
171
172
           void CTestDeliberateCrashVS6Dlg::OnOK()
173
           {
174
                causeACrash();
           }
175
176
           void CTestDeliberateCrashVS6Dlg::causeACrash()
177
178
179
                          *ptr = NULL;
                char
180
181
                 *ptr = '\0';
                                    // crash
182
183
184
185
186
```

# 9.5 What is a load address?

A load address is the address at which a DLL loads.

All versions of Microsoft Windows load modules (.dll, .exe) into address space that is reserved using a call to VirtualAlloc().

The allocation of VirtualAlloc() can be queried by calling Win32 API GetSystemInfo() and examining the value returned in **dwAllocationGranularity**. For all versions of Microsoft Windows this has been 64KB.

## Why is the load address important?

The load address is important because without it we can't calculate the offset inside the DLL so that we can obtain a symbol.

That's why a crash address with no DLL Load Address isn't very useful - we don't know which DLL the crash is in, nor do we know where the DLL was loaded.

# But I don't have a load address. What can I do?

Depending upon how your module (DLL/EXE) was built we may be able to guess the correct load address.

If the OS you are using is Windows XP or earlier, we can guess the address.

## First a brief chat about Address Space Layout Randomisation...

If the OS you are using is Windows Vista or later, we may be able to guess the load address. The reason this is not precises is because something known as Address Space Layout Randomisation (ASLR) was introduced with Microsoft Vista to improve security against many malicious computer attacks. Any program built with ASLR enabled when run on Vista (or later) will have the load address for all modules (including the .exe) randomised, making guessing the load address a waste of time.

ASLR is enabled by the /DYNAMICBASE in the linker settings of Visual Studio.

If you are using Visual Studio 2005 or earlier this setting is not available, your program is not affected by ASLR.

If you are using Visual Studio 2008 or later you will need to check to see if this option is present. If it is not present, your program is not affected by ASLR.

If you are not using Visual Studio to build your program then you may not be affected by this option, consult your compiler/linker documentation.

## If your program is not affected by ASLR...

We can try to guess the load address of your DLL/EXE. We can do this regardless of which compiler/linker you used to build your program. All the programs I mention here are free to download at the time of writing this help file.

## **VM Validator**

https://www.softwareverify.com/cpp-virtual-memory.php

This works for 32 bit and 64 bit programs.

## Method 1

- Start your program using VM Validator or attach to your running program with VM Validator.
- On the Summary tab, inspect the DLLs sub tab in the lower half of the display.
- Find the DLL name in the DLL column.
- The load address is the value in the Address column.

DLLs Page Faults						
DLL (133)	Fault Count	Address	Size	Commit	Reserve	CPU
E:\om\c\dbgHelpBrowser\Release\x86\dbgHelpBrowser.exe	0	0x00400000	1176.00 KB	1176.00 KB	0.00 KB	x86
E:\om\c\testApps\testDeliberateCrash\Release\testDeliberateCrash.exe	0	0x00640000	100.00 KB	100.00 KB	0.00 KB	x86
E:\om\c\dbgHelpBrowser\Release\x86\svIPeInfo.dll	0	0x006C0000	172.00 KB	172.00 KB	0.00 KB	x86

## Method 2

- Start your program using VM Validator or attach to your running program with VM Validator.
- Go to the Paragraphs tab.
- Find any purple entry, check the DLL name in the Description field.
- The load address is the value in the Address column.

	Summary			Virtual 🖂		Pages	⊠ Paragraphs
Address	Size	Туре	Protect	Working Set	Shared	Swap	Description
0x002C0000	64 KB	Private	Read, Write	Read/write.			Commited, Reserved
0x002D0000	152 KB	Private					Reserved, Commited, Reserved
0x00300000	1,024 KB	Private					Reserved
0x00400000	1,176 KB	Image	Read Only	Read-only. Executable and read-only.	Shared: 98		e:\om\c\dbghelpbrowser\release\x86\dbghelpbrowser.exe
0x00530000	796 KB	Mapped	Read Only	Read-only.	Shared: 49		Commited, Free

In the example above, for dbgHelpBrowser.exe, the load address is 0x00400000.

#### **Process Explorer**

https://technet.microsoft.com/en-us/sysinternals/processexplorer.aspx

This works for 32 bit and 64 bit programs.

- Start your program
- Start Process Explorer. If your program is a service or runs as administrator you'll need to start Process Explorer as administrator.
- In Process Explorer, enable View -> Show Lower Pane. Then for View -> Lower Pane Window, choose DLLs.
- Select your program in the top window.
- Find your DLL in the bottom window. Right click. Choose Properties from the Context menu.

2 Process Explorer - Sysinternals: www.sysinternals.	om (ZEl	JS\Stephen]				-		×
File Options View Process Find DLL Users	Help							
🛃 🖻 🖩 🗉 🗂 🤫 🖙 メ 🛤 🚱		<u>~</u>				. Å .	<u>.</u>	
Process	CPU	Private Bytes	Working Set	PID Description	Company Name			^
SecurityHealthSystray.exe		1,712 K	8,704 K	11880 Windows Security notificatio	Microsoft Corporation			
OneDrive.exe		29,012 K	70,728 K	12044 Microsoft OneDrive	Microsoft Corporation			
C WZQKPICK.EXE		1,496 K	8,108 K	6852 WinZip Executable	WinZip Computing, Inc.			
notepad.exe		3,112 K	16,952 K	308 Notepad	Microsoft Corporation			
🖃 🐼 svllauncher.exe		4,696 K	12,588 K	1052 svlLauncher	Software Verify Limited - w			
😡 dbgHelpBrowser.exe	12.47	28,420 K	51,624 K	6840 dbgHelpBrowser				
eventlogcrashbrowser.exe	12.48	5,896 K	16,708 K	8836 Event Log Crash Browser	Software Verify Limited - w			
vmValidator.exe	6.14	9,664 K	21,964 K	6188 vmValidator	Software Verify Limited - w			
HELPMAN.EXE	0.23	108,860 K	186,692 K	5396 Help & Manual	EC Software GmbH			~
Name Description		Company Name	P	ath				^
Data Exchange dl Data exchange		Microsoft Corpora	tion C.	Windows\SvsWOW64\DataExchange	e dll			
davcInt.dll Web DAV Client DLL		Microsoft Corpora		Windows\SysWOW64\davcInt.dll				
davhlpr.dll DAV Helper DLL		Microsoft Corpora	tion C:'	Windows\SysWOW64\davhlpr.dll				
dbgHelp.dll Windows Image Helper		Microsoft Corpora	tion E:	om\c\dbgHelpBrowser\Release\x86\	dbgHelp.dll			
dbgHelpBrowser.exe dbgHelpBrowser			E:	om\c\dbgHelpBrowser\Release\x86\	dbgHelpBrowser.exe			
dcomp.dll Microsoft DirectComposition		Microsoft Corpora		Windows\SysWOW64\dcomp.dll				
DevDispItemProvider.dll DeviceItem inproc devquery		Microsoft Corpora		Windows\SysWOW64\DevDispItemF	Provider.dll			
devobj.dll Device Information Set DLL		Microsoft Corpora		\Windows\SysWOW64\devobj.dll				
dlnashext.dll DLNA Namespace DLL	_	Microsoft Corpora		Windows\SysWOW64\dlnashext.dll				
drprov.dll Microsoft Remote Desktop	Sessio	Microsoft Corpora	tion C:	Windows\SysWOW64\drprov.dll				~
CPU Usage: 33.45% Commit Charge: 28.15% Process	100	Minnesh Company	C.	Minutes and Stream and Minutes and a stream				

• In the Properties dialog, read the load address.

dbgHelpBrow	ser.exe Properties		_		×
Image Strings					
Image					
Description:	dbgHelpBrowser				
Company:					
Version:	2.41.0.0				
Build Time:	Sun Jun 21 10:27:39 202	0			
Path:					
E: \om \c \dbgi	HelpBrowser \Release \x86 \	dbgHelpBrowser	.exe	Explore	
Autostart Loc	ation:				
n/a				Explore	
Load Address:	0x0000000000400000			Verify	
Mapped Size:	0x126000 bytes				
Mapping Type:	Image				
VirusTotal:		Submit			
Image: 32-bit					
		OK		Cancel	

In the example above, for dbgHelpBrowser.exe, the load address is 0x00400000.

# Visual Studio (any version)

https://www.visualstudio.com/

- Start Visual Studio.
- From the Project menu, choose File -> Open -> Solution. Choose your executable.
- From the Debug menu, choose Start Debugging.
- From the Debug menu, choose Windows -> Modules.

• In the Modules window, find your DLL, then read the Address column.

									- H
Path	Optimized	User Code	Symbol Status	Symbol File	Or	Version	Timestamp	Address	Process
E:\om\c\threadLockChecker\Rele	N/A	N/A	Symbols loaded.	E:\om\c\threadLockChec	1	2.01.0.0	18/05/2017 13:28	00130000-00204000	[7672] t
C:\Windows\SysWOW64\ntdll.dll	N/A	N/A	Symbols loaded	C:\Users\stephen\AppDat	2	6.3.9600.18	10/02/2016 15:50	775A0000-7770F000	[7672] t
C:\Windows\SysWOW64\kernel32	N/A	N/A	Symbols loaded	C:\Users\stephen\AppDat	3	6.3.9600.18	29/10/2014 02:58	77390000-774D0000	[7672] t
C:\Windows\SysWOW64\KernelB	N/A	N/A	Cannot find or o		4	6.3.9600.18	16/04/2017 09:16	757C0000-75897000	[7672]
E:\om\c\threadLockChecker\Rele	N/A	N/A	Cannot find or o		5	2, 0, 0, 1	28/03/2017 21:50	603A0000-60416000	[7672]
E:\om\c\threadLockChecker\Rele	N/A	N/A	Cannot find or o		6	2, 0, 0, 1	28/03/2017 20:55	1000000-10010000	[7672]
C:\Windows\SysWOW64\mfc100	N/A	N/A	Symbols loaded	C:\Users\stephen\AppDat	7	10.00.4021	11/06/2011 04:28	72080000-724BF000	[7672]
C:\Windows\SysWOW64\msvcr10	N/A	N/A	Symbols loaded.	C:\Users\stephen\AppDat	8	10.00.4021	11/06/2011 02:00	73A50000-73B0F0	[7672]
C:\Windows\SysWOW64\user32.dll	N/A	N/A	Symbols loaded	C:\Users\stephen\AppDat	9	6.3.9600.16	09/11/2016 18:25	770E0000-77233000	[7672]
C:\Windows\SysWOW64\gdi32.dll	N/A	N/A	Symbols loaded	C:\Users\stephen\AppDat	10	6.3.9600.18	07/04/2017 14:56	74DE0000-74EEE000	[7672]
C:\Windows\SysWOW64\advapi3	N/A	N/A	Symbols loaded	C:\Users\stephen\AppDat	11	6.3.9600.16	04/12/2015 15:57	75350000-753CC0	[7672]
C:\Windows\SysWOW64\shell32.dll	N/A	N/A	Symbols loaded	C:\Users\stephen\AppDat	12	6.3.9600.17	27/08/2016 17:00	75C40000-76EFB0	[7672]
Cilling dawn Swell/Oll/64 alasut?	N17A	M7A	Connet find or e		10	6 02 0600 1	16/03/2017 00.06	7520000 75247000	176731
	Elom) a thread LockChecken Rele C:Windows/SysWOW64 Intdll.dll C:Windows/SysWOW64 KemelB Elom) a thread LockChecken Rele Elom) a thread LockChecken Rele Elom) a thread LockChecken Rele C:Windows/SysWOW64 Inser32.dll C:Windows/SysWOW64 Juser32.dll C:Windows/SysWOW64 Juser32.dll C:Windows/SysWOW64 Juser32.dll C:Windows/SysWOW64 Juser32.dll C:Windows/SysWOW64 Juser32.dll C:Windows/SysWOW64 Juser32.dll C:Windows/SysWOW64 Juser32.dll	Elom/cthreadLockChecke/Acle. N/A C:\Windows\SysWOW64/ntdlLdll N/A C:\Windows\SysWOW64/kernel32 N/A C:\Windows\SysWOW64/kernelB N/A E\om\cthreadLockChecker/Rele N/A E\om\cthreadLockChecker/Rele N/A C:\Windows\SysWOW64/msvcr10 N/A C:\Windows\SysWOW64/user32.dll N/A C:\Windows\SysWOW64/asha32.dll N/A C:\Windows\SysWOW64/asha32.dll N/A	ENomicithreadLockCheckenRele         N/A         N/A           C:Windows/SysWOW64Intdl.Idll         N/A         N/A           C:Windows/SysWOW64Ixtenle32         N/A         N/A           C:Windows/SysWOW64Ixtenle8         N/A         N/A           C:Windows/SysWOW64Ixtenle8         N/A         N/A           E\orn\cithreadLockChecker/Rele         N/A         N/A           E\orn\cithreadLockChecker/Rele         N/A         N/A           C:Windows/SysWOW64Ivmcr10         N/A         N/A	ENomicithreadLockChecker/Rele         N/A         N/A         Symbols loaded.           C:Windows/SysWOW64[httdl.dll         N/A         N/A         Symbols loaded           C:Windows/SysWOW64[httdl.dll         N/A         N/A         Symbols loaded           C:Windows/SysWOW64[httdl.dll         N/A         N/A         Symbols loaded           C:Windows/SysWOW64[httdl.dll         N/A         N/A         Cannot find or o           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o           C:Windows/SysWOW64[http://www.cithreadLock/Decker/Rele         N/A         N/A         Symbols loaded           C:Windows/SysWOW64[http://www.cithreadLock/Decker/Rele         N/A         N/A         Symbols loaded <tr< td=""><td>ENomi-cithreadLockChecker/Ride         N/A         V/A         Symbols loaded         ENomi-cithreadLockChec           C:\Windows\SysWOW64Intdll.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Intdll.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Intell.ell.         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         E\om\cithreadLockChecker/Rele         N/A           C:\Windows\SysWOW64Inmcr100         N/A         N/A         Cannot find or o         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Inmcr100         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Inscr12         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Igai32.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Igai32.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Igai32.dll         N/A         N/A         Symbols loaded</td><td>ENomicithreadLockChecken/Rele         N/A         N/A         Symbols loaded.         ENomicithreadLockChecken         1           CiWindows/SysWOW64NtrdlI.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         2           CiWindows/SysWOW64NtrdlI.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         2           CiWindows/SysWOW64Nternel8         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         4           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         5           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         6           CiWindows/SysWOW64Nmcr100         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         7           CiWindows/SysWOW64Nuser32.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         9           CiWindows/SysWOW64Nuser32.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat</td><td>ENomicithreadLockChecker/Rele         N/A         N/A         Symbols loaded.         ENomicithreadLockChecker/Rele         1         201.00           C:\Windows\SysWOW&amp;Hutdll.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         2         6.3.9600.18           C:\Windows\SysWOW&amp;Hutdll.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         2         6.3.9600.18           C:\Windows\SysWOW&amp;Hutemells         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         4         6.3.9600.18           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         5         2.0.0.1           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         6         2.0.0.1           C:\Windows\SysWOW&amp;Humcr10         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         8         10.00.4021           C:\Windows\SysWOW&amp;Humcr10         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         8         3.9600.16           C:\Windows\SysWOW&amp;Humcr10         N/A         Symbols loaded         C:\Users\stephen\AppDet         6.3.9600.16</td><td>ENomicithreadLockChecker/Rele         N/A         N/A         Symbols loaded.         ENomicithreadLockCheck.         I         201.0.0         18//05/2017 13:28           CiWindows/SysWOW64Untdill.dll         N/A         N/A         Symbols loaded         C/Users/stephen/AppDat         2         6.3.3600.18         10//02/2016 15:50           CiWindows/SysWOW64Untdill.dll         N/A         N/A         Symbols loaded         C/Users/stephen/AppDat         2         6.3.3600.18         10//02/2016 15:50           CiWindows/SysWOW64Uxenel32         N/A         Symbols loaded         C/Users/stephen/AppDat         3         6.3.3600.18         10//02/2016 15:50           ENom/cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         5         2.0,0,1         28//03/2017 20:50           ENom/cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         6         2.0,0,1         28//03/2017 20:50           C/Windows/SysWOW64/user32.dll         N/A         N/A         Symbols loaded         C/Users/stephen/AppDat         7         10.00.4021         11//06/2011 02:00           C/Windows/SysWOW64/user32.dll         N/A         Symbols loaded         C/Users/stephen/AppDat         8         3.0600.16         09/11/2016 18:25</td><td>ENomicithreadLockChecker/Rele         N/A         N/A         Symbols loaded.         ENomicithreadLockCheck.         1         2.01.00         18/05/2017         3:28         00130000-00204000           CiWindows(SysWOW64)untdlLdll         N/A         N/A         Symbols loaded         C\Users\stephen\AppDat         2         6.3.9600.18         10/02/2016         15:50         77500000-7770F000           CiWindows(SysWOW64)untdlLdll         N/A         Symbols loaded         C\Users\stephen\AppDat         3         6.3.9600.18         10/02/2016         15:50         77500000-7770F000           CiWindows/SysWOW64\userBLe.         N/A         Symbols loaded         C\Users\stephen\AppDat         3         6.3.9600.18         10/02/2016         15:50         77500007-770F000           E\orn\cithreadLockChecker/Rele         N/A         Cannot find or o         5         2, 0, 0, 1         28/03/2017         21:50         603A0000-60416000           E\orn\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         5         2, 0, 0, 1         28/03/2017         21:50         603A0000-60416000           C\Windows/SysWOW64/usrsv10         N/A         N/A         Symbols loaded         C\Users\stephen\AppDat         7         10.00.4021         11/06/2011 04:28</td></tr<>	ENomi-cithreadLockChecker/Ride         N/A         V/A         Symbols loaded         ENomi-cithreadLockChec           C:\Windows\SysWOW64Intdll.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Intdll.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Intell.ell.         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         E\om\cithreadLockChecker/Rele         N/A           C:\Windows\SysWOW64Inmcr100         N/A         N/A         Cannot find or o         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Inmcr100         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Inscr12         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Igai32.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Igai32.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat           C:\Windows\SysWOW64Igai32.dll         N/A         N/A         Symbols loaded	ENomicithreadLockChecken/Rele         N/A         N/A         Symbols loaded.         ENomicithreadLockChecken         1           CiWindows/SysWOW64NtrdlI.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         2           CiWindows/SysWOW64NtrdlI.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         2           CiWindows/SysWOW64Nternel8         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         4           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         5           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         6           CiWindows/SysWOW64Nmcr100         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         7           CiWindows/SysWOW64Nuser32.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat         9           CiWindows/SysWOW64Nuser32.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDat	ENomicithreadLockChecker/Rele         N/A         N/A         Symbols loaded.         ENomicithreadLockChecker/Rele         1         201.00           C:\Windows\SysWOW&Hutdll.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         2         6.3.9600.18           C:\Windows\SysWOW&Hutdll.dll         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         2         6.3.9600.18           C:\Windows\SysWOW&Hutemells         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         4         6.3.9600.18           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         5         2.0.0.1           E\om\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         6         2.0.0.1           C:\Windows\SysWOW&Humcr10         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         8         10.00.4021           C:\Windows\SysWOW&Humcr10         N/A         N/A         Symbols loaded         C:\Users\stephen\AppDet         8         3.9600.16           C:\Windows\SysWOW&Humcr10         N/A         Symbols loaded         C:\Users\stephen\AppDet         6.3.9600.16	ENomicithreadLockChecker/Rele         N/A         N/A         Symbols loaded.         ENomicithreadLockCheck.         I         201.0.0         18//05/2017 13:28           CiWindows/SysWOW64Untdill.dll         N/A         N/A         Symbols loaded         C/Users/stephen/AppDat         2         6.3.3600.18         10//02/2016 15:50           CiWindows/SysWOW64Untdill.dll         N/A         N/A         Symbols loaded         C/Users/stephen/AppDat         2         6.3.3600.18         10//02/2016 15:50           CiWindows/SysWOW64Uxenel32         N/A         Symbols loaded         C/Users/stephen/AppDat         3         6.3.3600.18         10//02/2016 15:50           ENom/cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         5         2.0,0,1         28//03/2017 20:50           ENom/cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         6         2.0,0,1         28//03/2017 20:50           C/Windows/SysWOW64/user32.dll         N/A         N/A         Symbols loaded         C/Users/stephen/AppDat         7         10.00.4021         11//06/2011 02:00           C/Windows/SysWOW64/user32.dll         N/A         Symbols loaded         C/Users/stephen/AppDat         8         3.0600.16         09/11/2016 18:25	ENomicithreadLockChecker/Rele         N/A         N/A         Symbols loaded.         ENomicithreadLockCheck.         1         2.01.00         18/05/2017         3:28         00130000-00204000           CiWindows(SysWOW64)untdlLdll         N/A         N/A         Symbols loaded         C\Users\stephen\AppDat         2         6.3.9600.18         10/02/2016         15:50         77500000-7770F000           CiWindows(SysWOW64)untdlLdll         N/A         Symbols loaded         C\Users\stephen\AppDat         3         6.3.9600.18         10/02/2016         15:50         77500000-7770F000           CiWindows/SysWOW64\userBLe.         N/A         Symbols loaded         C\Users\stephen\AppDat         3         6.3.9600.18         10/02/2016         15:50         77500007-770F000           E\orn\cithreadLockChecker/Rele         N/A         Cannot find or o         5         2, 0, 0, 1         28/03/2017         21:50         603A0000-60416000           E\orn\cithreadLockChecker/Rele         N/A         N/A         Cannot find or o         5         2, 0, 0, 1         28/03/2017         21:50         603A0000-60416000           C\Windows/SysWOW64/usrsv10         N/A         N/A         Symbols loaded         C\Users\stephen\AppDat         7         10.00.4021         11/06/2011 04:28

In the example above, for threadLockChecker.exe, the load address is 0x00130000.

# WinDbg

https://msdn.microsoft.com/en-gb/library/windows/hardware/ff551063(v=vs.85).aspx

- Start WinDbg
- From the File menu, choose Open Executable. Choose your executable.
- Type Im, then press return.
- All modules are listed. Find your module. The start address is the load address.

		_			er\Relea Windo			gHelpB	rowser.	.exe -	WinDł	og:10	).0.19	041.	1 X8	6			_	[		×
<b>1</b>	- X			-	<b>X</b> E		· ·	• {}•	8   1	<u>)</u>   [	<u>&gt;</u> 🐺	Ş	öx		阅	10			2	10	1   A,	<b>.</b>   [
Com	mand																				Σ	2
eip	=770				)19fa2									nv	up	ei				pe no	-	
					ls=002 erBrea				fs=O	053	gs=	=002	2Ъ				1	efl	=000	0024	6	
770	aeaa 00>	2 cc				in	t	3														
sta	rt	en			<u>modu</u>																	
			1fЪ00 52600		<u>svlF</u> dbqF				ferre de		red)											
			00f00		sv19		ort		eferr													
			56900 6Ъ900		<u>qdip</u> <u>MSV0</u>		2		erred erred													
			82d00		COMO		2		erred													
			83600 31£00		<u>MSIN</u> mfc1				erred erred													
			40f00		MSVO		2		erred													
			42800 7ca00		CRYE		5E	· · · ·	erred ferre	· · · ·												
747	d000	0 74	7£000	0	Sspi	<u>Cli</u>		(defe	erred	I) (												
			86600 96700		<u>sech</u> ole3				erred erred													
74a	e000	0 743	Ь9ЪОО	0	RPCF	T4		(defe	erred	I)	• •											
			baf00 c2900		ADVA				de) erred		red)											
74c	4000	0 74	⊏9f00	0	bery	ptP:		tive	s (	def	erred	i)										
			d1700 2e600		<u>win3</u> wind		sto	(defe prage	erred /d	· · ·	rred)	)										
752	£000	0 75	2£600	0	PSAE	<u>1</u>		(defe	erred	1)	,	, 										- 1
			52400 7d700		<u>SHLV</u> USEF				erred erred													
757	e000	0 75	8c000	0	KERN		2	(defe	erred	I)												
			⊜8f00 40900		<u>msvo</u> SHEI				erred erred													
764	8000	0 76	50400	0	shee	re			erred													
			53100 55Ъ00		<u>GDI3</u> prof	_			erred erred													
			6cb00		gdi3		11		ferre													
			7ef00 7fd00		UMPI		2		erred erred													
			81300		Cryp				erred													
			9ac00 c5500		<u>msvo</u> comb		in		ferre erred													
			cf200		OLEA		2		erred													
			14300		powi				erred													
			18b00 f 8e00		<u>cfq</u> ı <u>KER</u>	ELB			erred eferr													
			19a00		ntdl				symt		)			C:>	Pro	ogr	amD	ata`	\dbg	l∕sàm,	\wnto	11
			45100 a7a00		<u>dbqł</u> svlł		<u>rool</u>	(dere L <u>Afx</u>	erred (de		red)											
									-		,											•
<																						>
): O	00>																					
						L	0.0	ol 0 S	vs 0:<1	ocals	Pro	c 000	):1bf(	οт	hrd	000:1	8dc	AS	м	OVR C	APS	NU

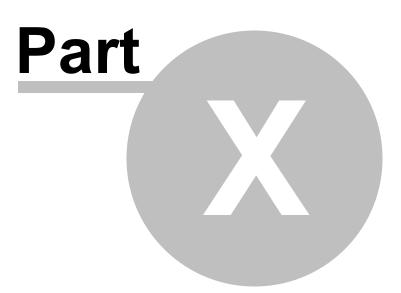
In the example above, for threadLockChecker.exe, the load address is 0x00130000.

# **Final Comments**

OK, you should now know how to find the load address of a DLL or an EXE (or any module type). Remember that a load address obtained this way is only valid for symbol decoding if the executable doesn't have ASLR applied to it.

52

If your crash reporting code only grabs crash addresses and not DLL load addresses, you need to update your code so that you grab DLL load addresses at the time of the crash. That way you know for sure what the load addresses were and you won't have to guess the load addresses in future.



# 10 Command Line Interface

MapFileBrowser can be used from the command line as well as with the GUI.

The command line options allow you to view debug information that is in a MAP file, and optionally highlight a symbol at a specified offset.

### /fileName

Specifies the module to load. This is typically a .exe or a .dll.

/fileName path-to-executable

Example: /fileName e:\om\c\test\release\test.exe

### /offset

Specifies an offset inside the executable. MapFileBrowser will highlight the symbol that occupies this location.

Typically this offset will be calculated from a crash location.

For example:

If a DLL is loaded at 0x00400000 and a crash happens at 0x00420192, the offset is calculated by subtracting the DLL load address from the crash address.

That is: 0x00420192 - 0x00400000, which gives 0x00020192.

The offset is 0x00020192.

The offset must be specified in hexadecimal with a leading 0x.

/offset value

Example: /offset 0x00020192

# **Example Command Line**

#### 32 bit applications

MapFileBrowser.exe /fileName e:\test\release\test.map /offset 0x00020192

64 bit applications

MapFileBrowser\_x64.exe /fileName e:\test\release\test.map /offset 0x00020192

