← Delphi Programming Diary - Jitendra Kumar

Delphi! is my art of programming...This blog contains technical solutions and tips for Delphi programming. I am just trying to share knowledges that I gained from others and which will help others. (Delphi Tokyo/XE7/ XE6 / XE5 / XE4 / XE3 / XE2 / XE, FireMonkey, FireDAC, DataSnap, QuickReport, DevExpress, Woll2Woll, TMS Components, Indy, REMObjects SDK....)

FASTMM and how to use in a Delphi Project.

- September 20, 2018

One of the most challenging parts of inheriting a legacy project is to fix the memory leaks that most often are hiding in the code. A while ago, while dealing with an application that managed to eat all the available memory within a few hours I found **FastMM** and it was sure a great find.

What is FastMM?

FastMM is a memory manager replacement designed to be used with Delphi. It is an Open Source project developed by Pierre Le Riche in South Africa. Starting with Delphi 2006 FastMM replaced the Borland memory manager. Unfortunately, Delphi only ships with a subset of FastMM. Most of the useful debugging reporting that can be done with FastMM has been stripped from the shipping version of Delphi, RAD Studio and BDS. But fear not! It is very simple to replace the stripped-down version of FastMM with the full version. All you have to do is follow the directions outlined below. I have also included a step to install the FastMM4 Options Interface program. That program is a very friendly way of configuring the options contained in the file FastMM4Options.inc.

How to install FastMM?

1. Download the latest FastMM source code from http://sourceforge.net/projects/fastmm/

2. Copy the content of the downloaded zip file to a folder on your computer.

3. In Delphi add a path in Tools>Options>Library – Win32 Library Path to the FastMM folder that contains the unit FastMM4.pas.

4. Copy the file **FastMM_FullDebugMode.dll** from the folder FastMM\FullDebugMode DLL\Precompiled to the Delphi install folder. For example in Delphi 2007 - C:\Program Files\CodeGear\RAD Studio\5.0\bin or in the Delphi XE install folder - C:\Program Files\Embarcadero\RAD Studio\8.0\bin

5. Optionally one can download and install the FastMM4 Options Interface program from JED software's web siteat http://jedqc.blogspot.com/2007/07/new-fastmm4-options-interface.html.
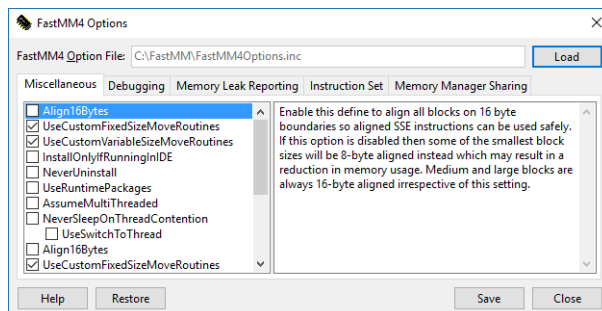
Configuring FastMM

Once you have properly installed FastMM4 you will be able to detect any memory leaks and attempts to use freed memory. To take fully advantage of FastMM4 feature, you should open **FastMM4Options.inc** file and change some conditional compilation directive. This file is heavily commented to explain all the symbols. When developing your application, you should define the symbols "**EnableMemoryLeakReporting**" and "**FullDebugMode**". And If you enable this option your application will require the **FastMM_FullDebugMode.dll** library. In the FastMM folder you will find the file FastMM4Options.inc. This is the file that controls how FastMM behaves. Each option is very well documented and it is how you set the default behavior of FastMM. One can manually edit this file or, optionally, use the *FastMM4 Options Interface*.

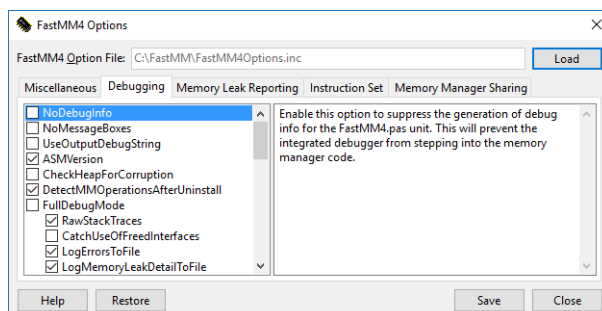Let's take a quick look to this file. There are five different sections in this file.

1. Miscellaneous Options
This section contain general settings to control memory alignment, use of fast move library, multi threaded behavior and debug only when running the IDE
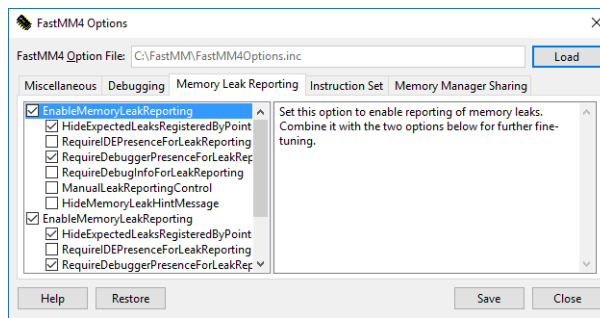

2. Debugging Options
This section contains defines that control the debugging behavior of FastMM such as logging errors to a log file, dumping of memory along with an error, stack traces and more.
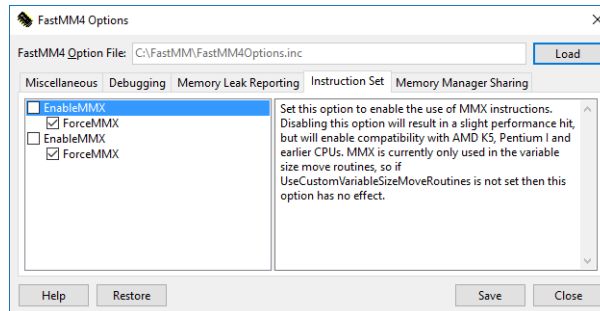

3. Memory Leak Reporting
This section controls the reporting of memory leaks, how to deal with expected memory leaks and the presence of the IDE or debug info to report errors.
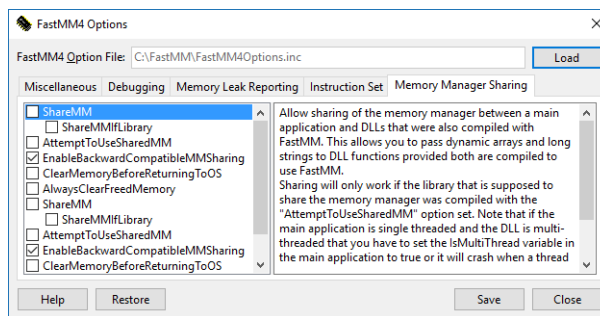
FastMM4 Options

FastMM4 Option File: C:\FastMM\FastMM4Options.inc   [Load]

Miscellaneous | Debugging | **Memory Leak Reporting** | Instruction Set | Memory Manager Sharing

☑ EnableMemoryLeakReporting
  ☑ HideExpectedLeaksRegisteredByPoint
  ☐ RequireIDEPresenceForLeakReporting
  ☑ RequireDebuggerPresenceForLeakRep
  ☐ RequireDebugInfoForLeakReporting
  ☐ ManualLeakReportingControl
  ☐ HideMemoryLeakHintMessage
☑ EnableMemoryLeakReporting
  ☑ HideExpectedLeaksRegisteredByPoint
  ☐ RequireIDEPresenceForLeakReporting
  ☑ RequireDebuggerPresenceForLeakRep

Set this option to enable reporting of memory leaks. Combine it with the two options below for further fine-tuning.

[Help] [Restore]   [Save] [Close]

4. Instruction Set Options
This section deals with using MMX instructions and this option currently only affects the variable size move routines.

FastMM4 Options

FastMM4 Option File: C:\FastMM\FastMM4Options.inc   [Load]

Miscellaneous | Debugging | Memory Leak Reporting | **Instruction Set** | Memory Manager Sharing

☐ EnableMMX
  ☑ ForceMMX
☐ EnableMMX
  ☑ ForceMMX

Set this option to enable the use of MMX instructions. Disabling this option will result in a slight performance hit, but will enable compatibility with AMD K5, Pentium I and earlier CPUs. MMX is currently only used in the variable size move routines, so if UseCustomVariableSizeMoveRoutines is not set then this option has no effect.

[Help] [Restore]   [Save] [Close]

5. Memory Manager Sharing Options
This section allows sharing of the memory manager between a main applications and DLLs.

FastMM4 Options

FastMM4 Option File: C:\FastMM\FastMM4Options.inc   [Load]

Miscellaneous | Debugging | Memory Leak Reporting | Instruction Set | **Memory Manager Sharing**

☐ ShareMM
  ☐ ShareMMIfLibrary
☐ AttemptToUseSharedMM
☑ EnableBackwardCompatibleMMSharing
☐ ClearMemoryBeforeReturningToOS
☐ AlwaysClearFreedMemory
☐ ShareMM
  ☐ ShareMMIfLibrary
☐ AttemptToUseSharedMM
☑ EnableBackwardCompatibleMMSharing
☐ ClearMemoryBeforeReturningToOS

Allow sharing of the memory manager between a main application and DLLs that were also compiled with FastMM. This allows you to pass dynamic arrays and long strings to DLL functions provided both are compiled to use FastMM.
Sharing will only work if the library that is supposed to share the memory manager was compiled with the "AttemptToUseSharedMM" option set. Note that if the main application is single threaded and the DLL is multi-threaded that you have to set the IsMultiThread variable in the main application to true or it will crash when a thread

[Help] [Restore]   [Save] [Close]

Preparing and compiling projects in Delphi
There are a few simple steps to prepare your existing projects to use the full version of FastMM in a useful way.

1. You can control how FastMM behaves in two ways:
        a. By making changes to Option Grouping in the FastMM4Options.inc so you can define different FastMM behaviors for your release version and debug version.
        b. Or by using a conditional IFDEF statement to use the complete library on the debug version and ship the Delphi supplied library in the release version. To accomplish this, in Delphi open a project and add the unit FastMM4 as the first unit in the uses clause of the .dpr file.
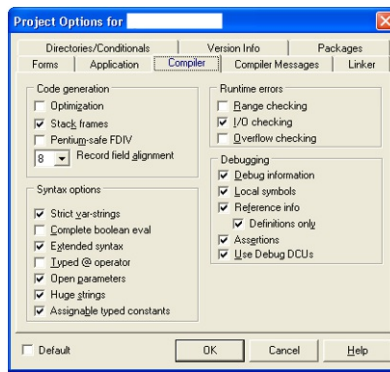…
uses
{$IFDEF DEBUG}
FastMM4,
{$ENDIF}
DB,
…

2. If your project includes EXEs and DLLs you also need to define **ShareMM**, **ShareMMIfLibrary** and **AttemptToUseSharedMM** using the *FastMM4 Options Interface* program and add **FastMM4.pas** to the top of the uses section of the .dpr for both the main application and the DLL and follow the directions outlined in item 3. This will allow FastMM to report memory leaks across EXEs and DLLs.
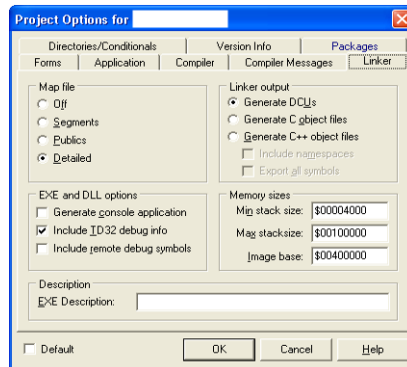
3. Optionally, set FatsMM4 debug options using the *FastMM4 Options Interface* program. Make sure to build your program every time you make changes to any FastMM4 options.

4. In order for certain debug features of FastMM4 to work you must make sure that certain debug switches are turned on. The following is a list of recommended switches:
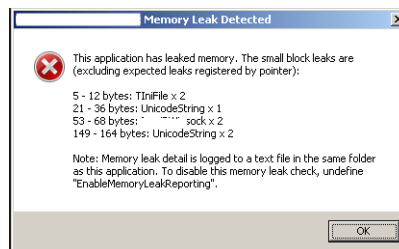   a. In the Compiler options set the following options
                ·   Debug Information
                ·   Reference Info
                ·   Use Debug DCus

b. In the Linker options make sure that one of the following options is set
· TD32 Debug info
· Map file

5. After you run the application a log of the memory manager can be found in the same folder where the application ran. The log file is named **Leaks_MemoryManager_EventLog.txt**.

AttemptToUseSharedMM   EnableMemoryLeakReporting   fastmm   FastMM_FullDebugMode.dll   FastMM4

FastMM4Options.inc   FullDebugMode   Leaks_MemoryManager_EventLog   ShareMM

Enter your comment...

Popular posts from this blog

MS Excel Automation in Delphi

- February 02, 2018

In this blog I will describe how to read and write data from and to an Excel file. Sometime in our application we use Excel for reporting purpose, for data import / export purpose and for other works. So here I will explain how to access an Excel file and use for data read / write. …

READ MORE

How to get WINDOWS special directories path in Delphi?

- August 03, 2017

Sometime we need some special directories path from Microsoft Windows system to store User data or to copy some files etc. So we can get those folder paths in Delphi in several ways. In this blog I have tried to cover all the ways. If I have left something please feel free to add comments. …

How to send Email in Delphi?

- December 03, 2014

Today in most of software program it is required to send mail to others with attached data. As some time we need to update clients about their account details, status or any other information through mail. So Delphi provides so many easiest ways to send mail from our Delphi application. I have collected some of ways which are very useful for Delphi developers. In Delphi we don't require any third-