HeapMemView v1.02

Web site: http://www.nirsoft.net

Description

HeapMemView is a small utility that allows you to view the content of all memory blocks allocated in the heap of the process the you select.
This tool can be useful for developers that need to trace memory leaks in their software.

System Requirements

This utility can work on Windows 2000, Windows XP, Windows Server 2003, and Windows Vista (Earlier versions of Windows are not supported). there is also x64 version (for Vista x64) available as a separated download.

Versions History

- Version 1.02 - Fixed bug: The main window lost the focus when the user switched to another application and then returned back to HeapMemView.
- Version 1.01 - Added support for saving memory blocks list as comma-delimited file.
- Version 1.00 - First release.

Using HeapMemView

HeapMemView doesn't require any installation process or additional DLLs. Just copy the executable file (HeapMemView.exe) to any folder you like, and run it.
After you start HeapMemView, you have to select the process that you want to inspect.
The main window of HeapMemView has 2 panes:

- The upper pane displays the list of memory blocks allocated in the heap of the process the your selected.

- The lower pane displays the contant of the memory block that you select in the upper pane.

Tips For Using HeapMemView

- In order to decrease the number displayed memory blocks and to improve the performances, HeapMemView filter all memory blocks with size smaller than 80 bytes. You can change this default filter by using the Memory Blocks Filter window (F8).
  Be aware that without the 80 bytes filter, you may get a lots of small and meaningless memory blocks, and the loading process will be pretty slow.
- Each time that you press F5, the newly added memory blocks are marked with '*', as well as they are painted with pink color. You can clear the 'new' flag by using the 'Clear New Memory Blocks Mark' option (F6).
- In addition to the memory bytes displayed in the lower pane, You can view a sample of the memory bytes in the table of the upper pane - under 'Data Preview' column. In order to enable this feature, select Options->Data Preview Mode->ASCII or Hexadecimal. By default, the first 32 bytes of the memory block will be displayed. You can modify the default settings in Advanced Options window.
- You can export all heap memory blocks to binary files for inspecting them in the future, by selecting all memory blocks in the upper pane, and then using 'Export Memory Data To File' option (Ctrl+E)
- The 'Heap Handle' value is the handle returned by HeapCreate or GetProcessHeap API functions. The 'Data Address' value is the memory address returned by HeapAlloc API.
- When a program allocates heap memory with C/C++ memory functions (malloc or new keyword), you may see some header bytes preceding the real allocated memory data.

Translating HeapMemView to other languages

In order to translate HeapMemView to other language, follow the instructions below:

1. Run HeapMemView with /savelangfile parameter:
   HeapMemView.exe /savelangfile

A file named HeapMemView_lng.ini will be created in the folder of HeapMemView utility.

2. Open the created language file in Notepad or in any other text editor.
3. Translate all string entries to the desired language. Optionally, you can also add your name and/or a link to your Web site. (TranslatorName and TranslatorURL values) If you add this information, it'll be used in the 'About' window.
4. After you finish the translation, Run HeapMemView, and all translated strings will be loaded from the language file.
   If you want to run HeapMemView without the translation, simply rename the language file, or move it to another folder.

License

This utility is released as freeware. You are allowed to freely distribute this utility via floppy disk, CD-ROM, Internet, or in any other way, as long as you don't charge anything for this. If you distribute this utility, you must include all files in the distribution package, without any modification !

Disclaimer

The software is provided "AS IS" without any warranty, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The author will not be liable for any special, incidental, consequential or indirect damages due to loss of data or any other reason.

Feedback

If you have any problem, suggestion, comment, or you found a bug in my utility, you can send a message to nirsofer@yahoo.com