

## RE: Memory growth halts with use of Profiler – bug in .NET services?

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.performance/2006-10/msg00005.htm>

---

- *From:* Warren Sirota <[WarrenSirota@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:WarrenSirota@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Wed, 11 Oct 2006 09:19:03 -0700
- 

I too have a service (with about 50 threads) that seems to demand unbounded memory, unless I attach a debugger to it. I have found this discussion interesting, although I haven't used the Windbg or sos, and I could really use a pointer to a tutorial, because I'm just a simple application writer, and I actually need documentation (whcih doesn't appear to be all that present).

HOWEVER, all that aside, I have to say that when I run my code as an application instead of as a service, I have no memory problems at all. This is with \*the exact same code\* that's in my Main method of the program and in the OnStart() method of the ServiceBase descendant.

What on earth could be causing this discrepancy?

--

Warren Sirota  
[www.wsdesigns.com](http://www.wsdesigns.com)

"JeremyC" wrote:

I was using the .NET Memory Profiler and the ANTS Profiler by Red Gate. They both did the same thing. I'll try the SOS debugger and also the CLRProfiler, thanks.

Jeremy

"Dave Hiniker – MSFT" wrote:

Which memory profiler are you using? I agree that is very strange behavior.

Assuming it's managed memory growth (in the GC heap) you should be able to track it down using the SOS debugger extension. Are you familiar with this tool? If so, wait until the service has grown to hundreds of MB and then

RE: Memory growth halts with use of Profiler – bug in .NET services?

attach and run !dumpheap –stat to see what type of objects are taking up so much memory. You can then use !dumpheap –mt to find all objects of that type in the heap and !gcroot to see what chain of references is keeping each alive.

This is basically the manual version of what CLRProfiler does; the profiler also does an incredible job of aggregating the results and making it easy to see where the problem is.

Both of these tools are linked to from <https://blogs.msdn.com/maoni/archive/2004/11/08/254288.aspx>

Hope this helps!

"JeremyC" wrote:

Forgot to mention that this is in .NET 1.1 on a Windows XP SP2 platform.

Jeremy

"JeremyC" wrote:

I have .NET Windows Service that is growing and growing getting up to several hundred MB in VM Size. However, when I attach a memory profiler to the service, the growth stops and holds a steady state, usually around 40–50 MB. Requesting the GC to force a Collect does not change the behavior.

Does anyone know why a service would exhibit this behavior. I assumed a memory leak, but why would it not show up when the profiler is running? I have searched and searched and not come across this problem anywhere yet.