

# Re: COM Object performance in different Threads

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2005-06/msg02374.html>

---

- *From:* "Nicholas Paldino [.NET/C# MVP]" <[mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Tue, 14 Jun 2005 12:36:24 -0400
- 

Ben,

COM objects have apartment affinity. By default, the UI thread (the main thread in your .NET application) is a STA thread, and when you create your COM object, it is associated with that apartment (it is an STA thread because of the STAThread attribute attached to your main method).

Now, when you create your new thread, by default, it doesn't live in a COM apartment. When you make the first call to COM interop, if you haven't set the ApartmentState property on the Thread to a value, it will place the thread in the multithreaded apartment. Now, when you make calls to the COM object from that thread, if you marshaled the reference correctly (which you are not, I am sure, and I don't believe COM interop does this by default), then all calls will be made through a proxy so the call will be executed on the correct thread.

In the end, you are either going to have to marshal the reference correctly (through the use of the global interface table, and a lot of interop), or create the COM objects on the thread they are going to be used (and make sure to set the ApartmentState property of the thread to STA BEFORE you create any COM objects on that new thread).

Hope this helps.

--  
- Nicholas Paldino [.NET/C# MVP]  
- [mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)

"Ben Childs" <[bchild@wpidotedu](mailto:bchild@wpidotedu)> wrote in message  
[news:%23UFJquPcFHA.720@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:%23UFJquPcFHA.720@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)

> Hi,

>

> I am writing an application in C# (VS.NET 2002) that runs performance  
> intensive analyses on data sets. In order to keep the UI responsive I have  
> created a separate thread to do the analysis.

>

> I am using my own DLLS to load the data sets from files and run the  
> analyses. But I have run into a bit of a problem. If I load the data set

## Re: COM Object performance in different Threads

- > in the main thread and then run the analysis in the separate thread the
- > analysis takes about 20 times longer than usual. In addition when I try to
- > display the results in the gui they take forever to display. So it seems
- > that accessing COM objects from different threads than they were created
- > in is really slow. Is there any way around this?
- >
- > One solution for me would be to create a thread that handled all of the
- > com objects, however this would be a lot of work and I was wondering if
- > there was a simple solution.
- >
- > Thanks,
- >
- > -Ben Childs

---

- **References:**

- ◆ **COM Object performance in different Threads**

- ◆ *From:* Ben Childs

- Prev by Date: **DirectoryServices Add/Remove Manager to Direct Report**
- Next by Date: **Re: Complexty of algorithm**
- Previous by thread: **COM Object performance in different Threads**
- Next by thread: **Re: COM Object performance in different Threads**
- Index(es):
  - ◆ **Date**
  - ◆ **Thread**