

## Re: Cannot access file, being used by another user(process)

**Source:**

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2004-08/0911.html>

---

**From:** Peter Rilling (*peter\_at\_nospam.rilling.net*)

**Date:** 08/04/04

Date: Wed, 4 Aug 2004 08:21:19 -0700

Maybe your logging should be done in a separate thread. You would need to have a single instance of the file stream and queue any write requests. That way you are guaranteed that only one thread will write to the log at any given time.

You don't necessarily want your threads to have control over when they write because you could have written half a line when another thread becomes active and begins to write.

"Chris Fink" <ChrisFink@discussions.microsoft.com> wrote in message news:823424D2-D4C7-4B4F-979A-9E3073BD60D0@microsoft.com...  
> *I have written a c# library app that writes to a log file. This app must be able to handle multiple threads, meaning it may receive 10 requests simultaneously to write to the log file. To Test this, I have spun of 10 threads and called this library app. The problem that I am running into is file locking. Occasionally, a thread will attempt to write to the file while it is open. The exception is System.IO.IOException (Cannot write to file, being used by another process).*

>

> *I would like your solutions on how you allow an application dealing with File IO to deal with the locking issue. For example, how does IIS overcome this problem when writing to the W3CService logs? IIS writes to the same file daily (appending), and I am sure that web sites with large hits would run into the same problem that I am experiencing.*

>

> *Essentially, what I am in search of is a way when I try to write to the file, if it is being used, to loop until the file is available to perform a write....or some other bulletproof method?*

>

> *Below is some sample code for the library component:*

> *//create logMMDDYY.txt for each day, if exists, append to it*

> *FileInfo t = new FileInfo(fileName);*

> *if (t.Exists)*

> *{*

> *try*

microsoft.public.dotnet.languages.csharp: Re: Cannot access file, being used by another user(process)

```
> {  
> lock(this)  
> {  
> file=t.AppendText();  
> }  
> }  
> catch(System.IO.IOException
```