

Safe multithreading in ASP.Net

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.aspnet/2004-08/1201.html>

From: who be dat? (*whatever_at_dot.com*)

Date: 08/06/04

Date: Thu, 5 Aug 2004 23:46:47 -0400

Consider the following code which enables multithreading in an ASP.Net application I'm writing:

Code in global.asx Application_start subroutine, Threadnotify is declared Globally as a new thread where it uses the address of EmailNotify below:

```
ThreadNotify.IsBackground = True
```

```
ThreadNotify.Start()
```

```
-----
```

```
Public Sub EmailNotify()
```

```
    Dim emails As New ForumEmail
```

```
    Do While ContinueNotify 'boolean value set to True so loop will continue
```

```
        emails.EmailNotify()
```

```
        Thread.Sleep(notifyinterval * 60000) 'notifyinterval is set via app  
settings in web.config file
```

```
    Loop
```

```
End Sub
```

```
-----
```

```
>From global.asx file in Application_End subroutine
```

```
ThreadNotify.Abort()
```

As you can probably tell, what happens is Threadnotify is started when the application starts. The subroutine simply loops through and calls a subroutine then it pauses in intervals of minutes. Good news: this works. Bad news: this works too good. Problem is, when does this thread get

terminated? I can set the boolean notify in the loop to false so the thread will reach the end of it's code and it will terminate. However, beyond this I have concerns. I'm writing an application on my machine and testing it. I can't get that thread to stop unless I set the boolean value to false in that loop which is bad. When I end the debug execution of the app, the thread continues and it doesn't stop. Heck, I can stop the IIS Service and the thread still continues to execute. How do I know it's executing. The goal of this thread is to send emails. I'm getting emails indicating which tells me this thread is still running.

This leads to a larger issue. When I release this application for others to use, how can I ensure this thread won't keep running even though a user might end the web service? This sucker just keeps going and going.

Suppose I upload this to a production webserver. The thread starts and executes fine. Suppose I make some changes to the application code and copy the necessary files to the production webserver. Will the thread from the previous versoin of the website continue while a new thread is started by the updated code i.e. there will be two threads doing the same thing?

The way this thread runs, I believe I can stop the webservice and destroy the webapplication. The thread would still run. This isn't acceptable. How do I ensure this thread stops?

What is this thread running under? I would've thought the thread would run under IIS and consider IIS to be the parent process. However, I'm terminating IIS and this thread still keeps going. What is it running under?

Ultimately, when the heck does this thread terminate?! The only thing I can do to stop this thread is to reboot the machine! I'm scared to publish this to my website as it may start up and never stop until they reboot their machines.

This thread is supposed to send emails after a pause determined by the user. Is there another way to do this? I though about using a timer that, when X amount of minutes passed, then the event would fire. I could capture that event and execute the email code. I wasn't sure how to do this though. Suggestions?

Thanks

Chris Smith