

Forum:[Hack Forums](#)

Thread: [SingleInstance Application \[C#/VB\]](#)

Nikey646 - October 1, 2014, 4:01 am

First off: The VB Version crashes. I have no clue as to WHY it crashes, it's something related to the Mutex not properly stating that it's owned as far as i can tell. I'll let a wizard like [TheBarret](#) or [Aeonhack](#) figure it out ^_^

What does this do:

It uses Mutex and NamedPipeStreams to prevent multiple instances of an application from occuring. It will also send the CMD information to the existing instance of the application, so that it can be processed.

Purpose:

Allows you to open files to queue work on them, or create a URL Based scheme eg: MyApp://Command1/Parameter1/Command2/Parameter2 with a few lines of code and this class.

Examples:

I've created a WinForms application in VB and C#, which outputs each parameter onto a new line of a textbox.

C# Example:

Spoiler ([Click to View](#))

Form1.cs: ([Credits of DoUI](#) to [GvS](#)[StackOverFlow])

```
Code    public Form1()
    {
        Program.InstanceManager.ArgumentsRecieved += (s, e) => DoUI(() =>
        {
            foreach (var Arg in e.Arguments)
            {
                textBox1.Text += Arg;
                textBox1.Text += Environment.NewLine;
            }
        });
        InitializeComponent();
    }

    private delegate void InvokeAction();
    private void DoUI(InvokeAction call)
    {
        if (IsDisposed)
        {
            return;
        }
        if (InvokeRequired)
        {
            Invoke(call);
        }
        else
```

```

    {
        call();
    }
}

```

Program.cs:

Code public static Singleton InstanceManager;

```

/// <summary>

```

```

/// The main entry point for the application.

```

```

/// </summary>

```

```

[STAThread]

```

```

static void Main(string[] Args)

```

```

{

```

```

    InstanceManager = new Singleton("{245A13C8-8588-410F-9213-63642A0188F8}",

```

```

    Args);

```

```

    if (!InstanceManager.FirstInstance) InstanceManager.Release();

```

```

    else

```

```

    {

```

```

        Application.EnableVisualStyles();

```

```

        Application.SetCompatibleTextRenderingDefault(false);

```

```

        Application.Run(new Form1());

```

```

    }

```

```

}

```

VB Example

Spoiler ([Click to View](#))

Form1.vb:

Code Public InstanceManager As Singleton

```

Public Sub New()

```

```

    InitializeComponent()

```

```

    InstanceManager = New Singleton("{245A13C8-8588-410F-9213-63642A0188F8}",

```

```

    My.Application.CommandLineArgs.ToArray())

```

```

    If Not InstanceManager.FirstInstance Then

```

```

        InstanceManager.Release()

```

```

    Else

```

```

        AddHandler InstanceManager.ArgumentsRecieved, AddressOf ProcessArguments

```

```

    End If

```

```

End Sub

```

```

Private Sub ProcessArguments(ByVal sender As Object, ByVal e As

```

```

ArgumentsRecievedEventArgs)

```

```

    Invoke(Sub()

```

```

        For Each Arg As String In e.Arguments

```

```

            textBox1.Text += Arg

```

```

            textBox1.Text += Environment.NewLine

```

```

        Next

```

Source (Bitbucket):

[Full Solution](#)

[C#](#) | [VB.Net](#)

Did i miss anything?

TheBarret - October 1, 2014, 4:08 am

Oh thats quit intresting, this is similair to what they had in the past, called DDE (http://en.wikipedia.org/wiki/Dynamic_Data_Exchange) connection between apps.

ill have a go with it, ill comment back when i find probs.

Nikey646 - October 1, 2014, 4:27 am

(Oct 1, 2014 04:08 AM)TheBarret Wrote: [\[CITE\]](#) Oh thats quit intresting, this is similair to what they had in the past, called DDE (http://en.wikipedia.org/wiki/Dynamic_Data_Exchange) connection between apps.

ill have a go with it, ill comment back when i find probs.

Was working with DDE (Not directly, but through the NDde library) for getting Firefox's URL a while ago. It's interesting to see that something from so long ago is used in one of the 3 Major Browsers of today.

Btw, it's probs something insanely stupidly simple, i havn't coded in VB for a bit, mostly C#, and hand converted it. (Making sure it works with Options Strict <_< >_>) By the end of the various error had i to fix while converting it, i was tired of working with VB again, and just gave up xD

He B TeMy - October 1, 2014, 5:45 am

What's wrong with "Enable single instance application" ?

Nikey646 - October 1, 2014, 6:15 am

(Oct 1, 2014 05:45 AM)HeBTeMy Wrote: [\[CITE\]](#) What's wrong with "Enable single instance application" ?

Besides not passing the Command Parameters from the old instance of the application to a new one, nothing.

If you know of a better way to do this, then please go ahead and enlighten me.

TheBarret - October 1, 2014, 6:29 am

Well not all the peeps arround here had a clue what this does or should perform, perhaps this could be tagged as a challenge ^^

CodeAssassin - October 1, 2014, 7:37 am

Hmm this is quite interesting, Never had the need to pass the args from one single instance app to another, but i can defiantly see the need in certain situations. Thanks for the heads up will have a little look myself :)

He B TeMy - October 1, 2014, 8:17 am

(Oct 1, 2014 06:15 AM)Nikey646 Wrote: [\[CITE\]](#) Besides not passing the Command Parameters from the old instance of the application to a new one, nothing.

If you know of a better way to do this, then please go ahead and enlighten me.

Wtf are you talking about with command parameters?

Nikey646 - October 1, 2014, 8:26 am

(Oct 1, 2014 08:17 AM)HeBTeMy Wrote: [\[CITE\]](#) Wtf are you talking about with command parameters?

Well. I may not be explaining it that *well*, but two other's seem to have understood what i meant?

Command = Command line (CMD.exe)

If i need to explain Parameters...sigh

If you are wondering what a Real World Everyday use of a system like this is, Spoiler [\(Click to View\)](#)
Most likely every major browser will implement some form of Inter-Process Operation's to read the new Command Line Parameters, and process them. So when you tell your Browser to open a URL (http://, https://, ftp://, etc), from an external program (EG: Skype), It'll likely receive it as something like Code Process.Open("Browser.exe", "https://hackforums.net")Which is then sent to the running Instance, where it can open that new webpage.

He B TeMy - October 1, 2014, 8:34 am

I figured it quite out, i don't figure is why you would need such a thing, let's face it, most programs won't, so this is kinda pointless.

No one's gonna code a "so-called browser" and for other programs there's other ways i could think of.

Although this may have value, don't call me dumb at the first reply and don't be so defensive for a rather OLD topic.

Nikey646 - October 1, 2014, 8:54 am

(Oct 1, 2014 08:34 AM)HeBTeMy Wrote: [\[CITE\]](#) I figured it quite out, i don't figure is why you would need such a thing, let's face it, most programs won't, so this is kinda pointless.

No one's gonna code a "so-called browser" and for other programs there's other ways i could think of.

Although this may have value, don't call me dumb at the first reply and don't be so defensive for a rather OLD topic.

How is this topic "OLD", in any way?

If you figured it out however cannot think of any usage's of such a program, then why didn't you try to think of a new usage?

In a short-term application there is no usages, however If you are trying to replicate an entire website's functionality in an application (Like i am for several projects, with and without owner approval), generating and using links that use a *name*:// format (URI Scheme) which will automatically open your application and respond to it is quite a useful user interaction feature.

The same could can be done with files, if you associate .*name* file's to your application (Say a Image Converter that adds a Watermark), instead of creating an entirely new instance of your application, it can load it up in a new tab, saving memory and load time.

This topic is not old, and quite a lot of Professionally made applications still use some form of inter-process operations to date. Steam, TS3, Origin, Firefox, Thunderbird, Outlook, GIMP, Any program that can open multiple files into a SINGLE instance.

Unless i've miss understood what you are implying Old to be, this is quite a relevant snippet of code, it's just not usable in any of the normal rinse-repeat code you find the average skid looking for here, aka Malicious code for RATs, Keyloggers, Spammers, etc.

Edit: Also i put this code out there more so if i ever lose it locally it'll be online. It's not like i'm forcing you to use this code in your next application, and infact, if anyone used this code in a proper application, i would be surprised, since it seems this forum's Creativity is limited to Malicious code.

He B TeMy - October 1, 2014, 9:11 am

(Oct 1, 2014 08:54 AM)Nikey646 Wrote: [\[CITE\]](#) How is this topic "OLD", in any way?

If you figured it out however cannot think of any usage's of such a program, then why didn't you try to think of a new usage?

In a short-term application there is no usages, however If you are trying to replicate an entire website's functionality in an application (Like i am for several projects, with and without owner approval), generating and using links that use a `*name*://` format (URI Scheme) which will automatically open your application and respond to it is quite a useful user interaction feature.

The same could can be done with files, if you associate `*name*` file's to your application (Say a Image Converter that adds a Watermark), instead of creating an entirely new instance of your application, it can load it up in a new tab, saving memory and load time.

This topic is not old, and quite a lot of Professionally made applications still use some form of inter-process operations to date. Steam, TS3, Origin, Firefox, Thunderbird, Outlook, GIMP, Any program that can open multiple files into a SINGLE instance.

Unless i've miss understood what you are implying Old to be, this is quite a relevant snippet of code, it's just not usable in any of the normal rinse-repeat code you find the average skid looking for here, aka Malicious code for RATs, Keyloggers, Spammers, etc.

"In any way?" it is, fairly old, since vb6 at least, no need for the "extra feature" for most of it imo.

I wasn't talking about usages of this program, i was talking about other usages that may accomplish this. (in perhaps a more dirty way depending on my brain-mood)

There you go, "website's functionality" , how many vb programs use that? Less than 5% right? (that is, if you want to take a wild guess.)

I'm not saying other programs don't use it, i'm not saying the use of it is old, i'm saying you're comming up to that solution which only brings an extra possible-feature to an older solution (or even the .net built-in one which i was talking about) just for the use referred above.

I'm not saying this is bad, i didn't even see the code (perhaps that's what started this.) i was just referring to the Title basically.

(Oct 1, 2014 08:54 AM)Nikey646 Wrote: [\[CITE\]](#) Edit: Also i put this code out there more so if i ever lose it locally it'll be online. It's not like i'm forcing you to use this code in your next application, and infact, if anyone used this code in a proper application, i would be surprised, since it seems this forum's Creativity is limited to Malicious code.

The Creativity is limited. Period.

Nikey646 - October 1, 2014, 9:23 am

(Oct 1, 2014 09:11 AM)HeBTMy Wrote: [\[CITE\]](#) --snip--

I'm not saying other programs don't use it, i'm not saying the use of it is old, i'm saying you're comming up to that solution which only brings an extra possible-feature to an older solution (or even the .net built-in one which i was talking about) just for the use referred above.

--snip--

Ahh, Okay so that's what you mean.

The only problem with your example of the "built-in" solution for Single Instances, is it doesn't tell the running instance of the application the launch commands of the original (Because there is no way to know if it's designed to handle that), which is what i really wanted out of this code. (However, to primary way of implementing that was to create a Mutex to find out if the application is running or not. So it already implements the solution from Visual Studio..(afaik))

Another thing to take into account, is WPF, which as far as i can tell, it actually doesn't have the

Single Instance option and expects you to take care of it yourself.

(Oct 1, 2014 09:11 AM)HeBTeMy Wrote: [\[CITE\]](#) --snip--

There you go, "website's functionality" , how many vb programs use that? Less than 5% right? (that is, if you want to take a wild guess.)

--snip--Well, for one website i'm actually using the API to create a Download Manager for the content. That's implementing the website functionality (Since the API will expose everything that can be accessed from the website...when it's finished lol). In the case of this forum, implementing a website functionality is severely limited Shells, Booters, Resolvers, and an Occasional Pastebin spammer.

(Oct 1, 2014 09:11 AM)HeBTeMy Wrote: [\[CITE\]](#) --snip--

The Creativity is limited. Period. So sad, but so true...Very few people can think outside of the box on this forum...(Hell, even me sometimes :P)

DBots - October 2, 2014, 1:33 am

@Nikey646, what I am understanding is you have an application which is a single-instance app. When you send command-line arguments to it, it does not detect them.

If that is the case then I had the same problem and the solution I found was using the [StartUpNextInstance](#) event.

Here is a sample of the code I used:

<http://pastebin.com/NV6RcDB7>

420 Blaze It - October 2, 2014, 1:35 am

Thanks for the source, I used exit.app and it kept crashing, now it didn't.

Nikey646 - October 2, 2014, 1:50 am

(Oct 2, 2014 01:33 AM)DBots Wrote: [\[CITE\]](#) @Nikey646, what I am understanding is you have an application which is a single-instance app. When you send command-line arguments to it, it does not detect them.

If that is the case then I had the same problem and the solution I found was using the [StartUpNextInstance](#) event.

Here is a sample of the code I used:

<http://pastebin.com/NV6RcDB7>

This was originally designed in C#, and i didn't want to import the Microsoft.VisualBasic namespace like you have. Weird errors due to an extra namespace can cause a few problems. I just converted it to VB for fun...then regretted doing so about 30 minutes later while trying to fix errors.

So while you can use StartUpNextInstance with no problems in VB, in C# it'll start causing Intellisense to act up, and further more, the Compiler will get just as confused as Intellisense. (EG: Try to create a new button in code while in a WPF project, but have System.Windows.Forms imported to use the NativeWindow class.)

unknown123456 - October 2, 2014, 3:19 am

(Oct 1, 2014 06:29 AM)ThePussyBarret Wrote: [\[CITE\]](#) Well not all the peeps around here had a clue what this does or should perform, perhaps this could be tagged as a challenge ^^

I bet you have no idea aswell . You just like to talk like a tough guy

Only3 - October 2, 2014, 4:20 am

with MyApplication u use it like
Edit - ? just saw dbots paste link

Code Namespace My
Partial Friend Class MyApplication

```
Private mtx As Threading.Mutex = Nothing
Private Sub MyApplication_Startup(sender As Object, e As
ApplicationServices.StartupEventArgs) Handles Me.Startup
Dim bExists As Boolean = True
mtx = New Threading.Mutex(False, "Local\UniqueMutexName", bExists)
If Not bExists Then
' An instance is already opened.
e.Cancel = True
Else
' No instances opened yet.
End If
End Sub

'Private Sub MyApplication_StartupNextInstance(sender As Object, e As
ApplicationServices.StartupNextInstanceEventArgs) Handles Me.StartupNextInstance
' Dim inputArgument As String = "/input="
' Dim inputName As String = ""

' For Each s As String In e.CommandLine
' If s.ToLower.StartsWith(inputArgument) Then
' inputName = s.Remove(0, inputArgument.Length)
' End If
' Next

' If inputName = "" Then
' MsgBox("No input name")
' Else
' MsgBox("Input name: " & inputName)
' End If
'End Sub
End Class
End Namespace
```

TheBarret - October 2, 2014, 6:09 am

(Oct 2, 2014 03:19 AM)unknown123456 Wrote: [\[CITE\]](#) I bet you have no idea aswell . You just like to talk like a tough guy

Aww how cute ^^ now hush hush, go play with the others kids.

Nikey646 - October 2, 2014, 1:12 pm

(Oct 2, 2014 03:19 AM)unknown123456 Wrote: [\[CITE\]](#) I bet you have no idea aswell . You just like to talk like a tough guy

Sighs

TheBarret has shown knowledge about a protocol unmentioned in this thread until he pointed out the similarity of it to my original post, that has been around since the early stages of Windows, and what it's proper use for it is. He is also able to create applications that are of no use to anyone, but very few users of this forum could replicate to such completeness. What do you have under your name? A Public Exploit(Patched) converted to VB Code, UDP Flooder, and various other

seen-before-malicious-crap.

(P.S: I respect his knowledge of VB, that is all.)

(Oct 2, 2014 04:20 AM)Only3 Wrote: [\[CITE\]](#) with MyApplication u use it like
Edit - ? just saw dbots paste link

Code Namespace My

Partial Friend Class MyApplication

```
Private mtx As Threading.Mutex = Nothing
Private Sub MyApplication_Startup(sender As Object, e As
ApplicationServices.StartupEventArgs) Handles Me.Startup
    Dim bExists As Boolean = True
    mtx = New Threading.Mutex(False, "Local\UniqueMutexName", bExists)
    If Not bExists Then
        ' An instance is already opened.
        e.Cancel = True
    Else
        ' No instances opened yet.
    End If
End Sub

'Private Sub MyApplication_StartupNextInstance(sender As Object, e As
ApplicationServices.StartupNextInstanceEventArgs) Handles Me.StartupNextInstance
    ' Dim inputArgument As String = "/input="
    ' Dim inputName As String = ""

    ' For Each s As String In e.CommandLine
    '     If s.ToLower.StartsWith(inputArgument) Then
    '         inputName = s.Remove(0, inputArgument.Length)
    '     End If
    ' Next

    ' If inputName = "" Then
    '     MsgBox("No input name")
    ' Else
    '     MsgBox("Input name: " & inputName)
    ' End If
End Sub
End Class
End Namespace
```

Then i presume you also saw my response to DBo's post, which is above your post as well. It also states that the primary purpose of this code was for C#, and your Microsoft.VisualBasic namespace solution is meaningless. Unless your telling me that i should use the Microsoft.VisualBasic namespace in a C# application, upon which i reply to you, Go use the Microsoft.CSharp namespace in your VB application...Oh wait...that's pointlessly stupid...