

Forum:[Hack Forums](#)

Thread: [Windows TreeView Control](#)

[Epreuvez] - January 23, 2018, 12:38 am

Hey guys,

I'm now asking for your guys help!

I have made a custom class that inherits a TreeView control. Everything works well until you enable the custom ShowFiles property, and click the Downloads folder. Maybe it just lags for me because I have a lot of files in my downloads folder, or from how I obtain the folder directory. I believe the issue occurs when retrieving the icons, but have tried many things to fix it, like running a new thread to scan for the icon, and also making sure resources are released. This only happens the first time around for me also, the second time a run the application, I have no problems. Yes, I know there is some redundant code in there, but I am working on removing it... Slowly.

Here is the code, please let me know what you think, or if you have any criticism!

[PasteBin](#)

Regards,

[Epreuvez]

P.S - The reason I started this project was for a FTP program I was going to make, I felt a custom control would look tidier, and didn't think I'd run into this much trouble... Ugh.

HFDone - January 23, 2018, 6:27 am

in the RefreshFolders function comment the drives for loop and check if its still lags

Ninja - January 23, 2018, 6:30 am

Instead of extracting icons from files you can get default icon for a given extension. Example here:

<https://stackoverflow.com/questions/2701263/get-the-icon-for-a-given-extension>

Then of course you cache the icons.

[Epreuvez] - January 23, 2018, 9:40 pm

(Jan 23, 2018 07:30 AM)Ninja Wrote: [\[CITE\]](#) Instead of extracting icons from files you can get default icon for a given extension. Example here:

<https://stackoverflow.com/questions/2701263/get-the-icon-for-a-given-extension>

Then of course you cache the icons.

I did try that actually, I had problems with .exe, .lnk, and .ico files being all the same. Now that I'm not half asleep I guess I could filter those file types out and load only .exe etc file types with SHFileInfo.

Thanks, I believe this will resolve my issue :) I'll post an update with the results...

Ninja - January 24, 2018, 6:21 am

(Jan 23, 2018 10:40 PM)[Epreuvez] Wrote: [\[CITE\]](#) I did try that actually, I had problems with .exe, .lnk, and .ico files being all the same. Now that I'm not half asleep I guess I could filter those file

types out and load only .exe etc file types with SHFileInfo.

Thanks, I believe this will resolve my issue :) I'll post an update with the results...

You could also extract icons from the files, but before you extract the icon check if you already extracted an icon from a file with the same extension, and if so then load the icon from cache.

[Eprouvez] - January 25, 2018, 4:38 pm

(Jan 24, 2018 07:21 AM)Ninja Wrote: [\[CITE\]](#) You could also extract icons from the files, but before you extract the icon check if you already extracted an icon from a file with the same extension, and if so then load the icon from cache.

In my code, it does check if the icon is already in the cache before it adds the icon to it. It filters out .exe, .lnk, .url, and .ico files to load them separately. I'm working on loading just those file types on the BeforeExpand/RefreshFolders routines, and having the file extensions loaded from registry on start up.

Thanks, will post results :)

EdwardSn0wdent - February 1, 2018, 11:18 am

Good evening. Starting from the top, until I expect a response from you. You will have to forgive if I change my approach. In this post, I will not go over your different naming standards that change with each same field etc. This is a clear insight to me you do not really understand all that is happening with the basics. We can expand on this later.

The main answer you seek is lag issues. If a UI control is lagging then to me instantly suggests that you are not adding each item in the correct manner causing the control to redraw each time.

The very first rule you should be following. Turn strict on always. How can we expect a program to run smoothly if we are writing shoddy code.

You are abusing the constructor. Public Sub New(). This is not for you to be doing work. Constructors should not do more than capture the constructor parameter. Specific user features should be simply that. User interaction. You also create a new instance of the ImageList class, then assign it to a default ImageList class that has no instance.

With that in mind. Let's see a rewritten example using the constructor correctly. Taking special note regarding RefreshFolders. Once you have posted a revised Constructor example only. We will move forward to the next step.

Microsoft has tried to move us way from the Registry for quite some time now. Please do not open a bigger can of worms for yourself.

Step gracefully through your project. It's scattered, unorganized and extremely hard to flow gracefully. I would like to see you think more about beautiful code. It's not just enough to write code to "as is".

A suggestion that might start getting you to think....

PHP Code

```
<?php
```

```
' TODO: [ESD] Put Option Strict at the top of all files.
```

```
' TODO: [ESD] Generate XML documentation; Types, members.
```

```
Complete documentation being descriptive as possible.
```

```
' TODO: [ESD] (Task) Make sure that all exceptions are handled in a graceful manner.
' TODO: [ESD] (Task) Follow constructor rules. A default constructor of private where applicable.
' TODO: [ESD] (Bug) ....
' TODO: [ESD] (Behavior) ....
```

```
Imports System.IO
Imports System.ComponentModel
Imports System.Runtime.InteropServices
```

```
Public Class WindowsTreeView
```

Was hoping to see revised first part following the guidelines I laid out.

Something like...

```
Code Option Strict On
```

```
' TODO: [ESD] Put Option Strict at the top of all files.
' TODO: [ESD] Generate XML documentation; Types, members. Complete documentation being
descriptive as possible.
' TODO: [ESD] (Task) Make sure that all exceptions are handled in a graceful manner.
' TODO: [ESD] (Bug) ....
' TODO: [ESD] (Behavior) Remove all unwanted "DUMMY" nodes. This is not helping. It's noise.
Imports System.IO
```

```
Public Class Form1
```

```
' TODO: [ESD] (Bug) Focus on getting a working tree view as specified.
' TODO: [ESD] (Behavior) Code that is not used marked with a comment should be instantly
removed.
Private ReadOnly mImageList As ImageList
' TODO: [ESD] (Task) Follow constructor rules. A default constructor of private where applicable.
' TODO: [ESD] (Task) All modifiers should be explicitly typed. You could walk into unknown bugs
by not focusing on it's default type.
' TODO: [ESD] (Task) All Methods to be given meaningful names.
' TODO: [ESD] (Task) Remove all UI interactions.
' TODO: [ESD] (Task) Remove all icon calls temporary.
Private Function EnumerateFolderNodes(rootFolderPath As String) As TreeNode()
' TODO: [ESD] (Task) Using the correct variable scope.
' TODO: [ESD] (Task) Long running tasks are background work and anything to do with a
control is inherently foreground work.
' TODO: [ESD] (Task) Provide a more efficient way to store data.
Dim nodes As New List(Of TreeNode)
```

```
Try
```

```
Dim rootFolder As New DirectoryInfo(rootFolderPath)
Dim folders = rootFolder.EnumerateDirectories()
```

```
For Each folder In folders
```

```
Dim folderPath = folder.FullName
```

```
Dim node As New TreeNode(folder.Name)
```

```
' TODO: [ESD] (Task) Handle recursive call's fore gracefully.
```

```
node.Nodes.AddRange(Me.EnumerateFolderNodes(folderPath))
```

```
nodes.Add(node)
```

```
Next
```

```
        Catch ex As UnauthorizedAccessException
            ' TODO: [ESD] (Comment) This folder is inaccessible. All we can do is ignore and move
on.
        End Try

        Return nodes.ToArray()
    End Function

    ' TODO: [ESD] (Task) Me.EnumerateFolderNodes;
    ' : [ESD]    My.Computer.FileSystem.SpecialDirectories
    ' : [ESD]    Nodes.AddRange(nodes)
End Class
```