How do you get the icons out of shell32.dll?

▲

41

▼

★

13

I'd like to get the Tree icon to use for a homegrown app. Does anyone know how to extract the images out as .icon files? I'd like both the 16x16 and 32x32, or I'd just do a screen capture.

windows   icons

edited Dec 27 '15 at 11:51   asked Aug 12 '08 at 3:26

Jaqueline Vanek   Jay Mooney
826   8   18   **1,237**   1   14   21

---

9   Note that doing this is a violation of the license. "While the software is running, you may use but not share its icons, images, sounds, and media." –
Raymond Chen Mar 29 '13 at 22:45

10 Answers

▲

41

▼

✓

In Visual Studio, choose "File Open..." then "File...". Then pick the Shell32.dll. A folder tree should be opened, and you will find the icons in the "Icon" folder.

To save an Icon, you can right-click on the icon in the folder tree and choose "Export".

edited Feb 6 '16 at 13:42   answered Aug 12 '08 at 3:32

Captain Ford   jm.
340   1   9   **14.3k**   20   66   89

---

2   when I use this I only have this folder tree and all the icons listed but without a preview and the titles aren't that helpfull (1, 10, 1001, ..) Is there a way to
get a preview or do I really have to open all the Icons? – Dominik Jul 23 '15 at 10:46

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up   OR SIGN IN WITH   G Google   Facebook ✕

1   @MickyD You're welcome. ;) – RicoBrassers Sep 25 '18 at 13:29

▲

39   If anyone is seeking an easy way, just use 7zip to unzip the shell32.dll and look for the folder .src/ICON/

▼   answered Jun 14 '16 at 1:41

                                                                                        AQN
                                                                                     391   3   2

Yep, I use the 7-Zip "open archive" function very much. – Niek Aug 26 '16 at 10:48

Haven't got 7Zip, but tried this in WinRAR and it says there's no archive to open. – Hashim Oct 18 '16 at 19:04

rename to shell32.7z then try to open it up with 7-zip, Peazip, or your other fav compression tools – ronaldwidha Oct 21 '16 at 17:11

at '.rsrc/ICON/ now, works perfect thanks – J-Dizzle Jan 5 '18 at 18:10

This should be the accepted answer because it is easy and works. – Marco7757 Feb 28 '18 at 12:41

▲

18   Another option is to use a tool such as ResourceHacker. It handles way more than just icons as well. Cheers!

▼   answered Aug 12 '08 at 4:17

                                                                                        OJ.
                                                                                     25.4k   5   48   67

I tried to use Resource Hacker, but every icon has 12 duplicates (with various sizes). – HighTechProgramming15 Jan 3 '17 at 13:27

▲

      I needed to extract icon #238 from shell32.dll and didn't want to download Visual Studio or Resourcehacker, so I found a couple of
      PowerShell scripts from Technet (thanks John Grenfell and to #https://social.technet.microsoft.com/Forums/windowsserver/en-
災   US/16444c7a-ad61-44a7-8c6f-b8d619381a27/using-icons-in-powershell-scripts?forum=winserverpowershell) that did something similar

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up   OR SIGN IN WITH   G Google   Faceboo

C:\Temp\Restart.ico

238

I discovered the icon index that I needed was #238 by trial and error by temporarily creating a new shortcut (right-click on your desktop and select New --> Shortcut and type in calc and press Enter twice). Then right-click the new shortcut and select Properties then click 'Change Icon' button in the Shortcut tab. Paste in path C:\Windows\System32\shell32.dll and click OK. Find the icon you wish to use and work out its index. NB: Index #2 is beneath #1 and not to its right. Icon index #5 was at the top of column two on my Windows 7 x64 machine.

If anyone has a better method that works similarly but obtains higher quality icons then I'd be interested to hear about it. Thanks, Shaun.

```
#Windows PowerShell
Code##################################################################
# http://gallery.technet.microsoft.com/scriptcenter/Icon-Exporter-e372fe70
#
# AUTHOR: John Grenfell
#
##################################################################

<#
.SYNOPSIS
    Exports an ico and bmp file from a given source to a given destination
.Description
    You need to set the Source and Destination locations. First version of a script, I
found other examples but all I wanted to do as grab and ico file from an exe but found
getting a bmp useful. Others might find useful
    No error checking I'm afraid so make sure your source and destination locations
exist!
.EXAMPLE
     .\Icon_Exporter.ps1
.Notes
        Version HISTORY:
        1.1 2012.03.8
#>
Param ( [parameter(Mandatory = $true)][string] $SourceEXEFilePath,
        [parameter(Mandatory = $true)][string] $TargetIconFilePath
)
CLS
#"shell32.dll" 238
```

```
    $image =
[System.Drawing.Icon]::ExtractAssociatedIcon("$($SourceEXEFilePath)").ToBitmap()
    $bitmap = new-object System.Drawing.Bitmap $image
    $bitmap.SetResolution(72,72)
    $icon = [System.Drawing.Icon]::FromHandle($bitmap.GetHicon())
}
$stream = [System.IO.File]::OpenWrite("$($TargetIconFilePath)")
$icon.save($stream)
$stream.close()
Write-Host "Icon file can be found at $TargetIconFilePath"
```

answered Jan 22 '15 at 14:22

[Shaun](#)

168   5   12

Far more helpful answer than the rest. – [slashp](#) Sep 18 '15 at 12:51

Great method but it doesn't work for me. I am on Windows 7x64 on which script seems to be confirmed to work. But I get `TypeNotFound` errors for `System.Drawing.Icon` and `System.Drawing.Bitmap`. My first try to run a PS-script so maybe I do something wrong? – [Danny Lo](#) Jan 18 '16 at 1:07

1   Ok, I just needed to execute `Add-Type -AssemblyName System.Drawing` first. – [Danny Lo](#) Jan 18 '16 at 1:12

[Resources Extract](#) is another tool that will recursively find icons from a lot of DLLs, very handy IMO.

5

answered Apr 6 '13 at 19:51

[Steve Cadwallader](#)

1,951   4   24   35

---

1   While that tool is good for pulling the resource out, they also have a tool IconsExtract [nirsoft.net/utils/iconsext.html](#) that I find more handy in most cases since I can browse the icons and simply link to the system file when I want the icon in an embedded/linked document in Office. – [undrline](#) Sep 7 '18 at 18:50

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up   OR SIGN IN WITH   **G** Google   Facebook ✕

```
        Exports an ico and bmp file from a given source to a given destination
    .Description
        You need to set the Source and Destination locations. First version of a script, I
    found other examples
        but all I wanted to do as grab and ico file from an exe but found getting a bmp
    useful. Others might find useful
    .EXAMPLE
        This will run but will nag you for input
        .\Icon_Exporter.ps1
    .EXAMPLE
        this will default to shell32.dll automatically for -SourceEXEFilePath
        .\Icon_Exporter.ps1 -TargetIconFilePath 'C:\temp\Myicon.ico' -IconIndexNo 238
    .EXAMPLE
        This will give you a green tree icon (press F5 for windows to refresh Windows
    explorer)
        .\Icon_Exporter.ps1 -SourceEXEFilePath 'C:/Windows/system32/shell32.dll' -
    TargetIconFilePath 'C:\temp\Myicon.ico' -IconIndexNo 41

    .Notes
        Based on http://stackoverflow.com/questions/8435/how-do-you-get-the-icons-out-of-
    shell32-dll Version 1.1 2012.03.8
        New version: Version 1.2 2015.11.20 (Added missing custom assembly and some error
    checking for novices)
    #>
    Param (
        [parameter(Mandatory = $true)]
        [string] $SourceEXEFilePath = 'C:/Windows/system32/shell32.dll',
        [parameter(Mandatory = $true)]
        [string] $TargetIconFilePath,
        [parameter(Mandatory = $False)]
        [Int32]$IconIndexNo = 0
    )

    #https://social.technet.microsoft.com/Forums/windowsserver/en-US/16444c7a-ad61-44a7-
    8c6f-b8d619381a27/using-icons-in-powershell-scripts?forum=winserverpowershell
    $code = @"
    using System;
    using System.Drawing;
    using System.Runtime.InteropServices;

    namespace System
    {
        public class IconExtractor
```

```
    ExtractIconEx(file, number, out large, out small, 1);
    try
    {
     return Icon.FromHandle(largeIcon ? large : small);
    }
    catch
    {
     return null;
    }


    }
    [DllImport("Shell32.dll", EntryPoint = "ExtractIconExW", CharSet = CharSet.Unicode,
ExactSpelling = true, CallingConvention = CallingConvention.StdCall)]
    private static extern int ExtractIconEx(string sFile, int iIndex, out IntPtr
piLargeVersion, out IntPtr piSmallVersion, int amountIcons);

  }
}
"@

If  (-not (Test-path -Path $SourceEXEFilePath -ErrorAction SilentlyContinue ) ) {
    Throw "Source file [$SourceEXEFilePath] does not exist!"
}

[String]$TargetIconFilefolder = [System.IO.Path]::GetDirectoryName($TargetIconFilePath)
If  (-not (Test-path -Path $TargetIconFilefolder -ErrorAction SilentlyContinue ) ) {
    Throw "Target folder [$TargetIconFilefolder] does not exist!"
}

Try {
    If ($SourceEXEFilePath.ToLower().Contains(".dll")) {
        Add-Type -TypeDefinition $code -ReferencedAssemblies System.Drawing
        $form = New-Object System.Windows.Forms.Form
        $Icon = [System.IconExtractor]::Extract($SourceEXEFilePath, $IconIndexNo, $true)
    } Else {
        [void][Reflection.Assembly]::LoadWithPartialName("System.Drawing")
        [void] [System.Reflection.Assembly]::LoadWithPartialName("System.Windows.Forms")
        $image =
[System.Drawing.Icon]::ExtractAssociatedIcon("$($SourceEXEFilePath)").ToBitmap()
        $bitmap = new-object System.Drawing.Bitmap $image
        $bitmap.SetResolution(72,72)
        $icon = [System.Drawing.Icon]::FromHandle($bitmap.GetHicon())
    }
```

```
    $icon.save($stream)
    $stream.close()
} Catch {
    Throw "Error saving ICO file [$TargetIconFilePath]"
}
Write-Host "Icon file can be found at [$TargetIconFilePath]"
```

answered Nov 20 '15 at 19:56

Mr. Annoyed
343   2   12

Just open the DLL with IrfanView and save the result as a .gif or .jpg.

2

I know this question is old, but it's the second google hit from "extract icon from dll", I wanted to avoid installing anything on my workstation and I remembered I use IrfanView.

edited May 22 '14 at 14:58   answered May 22 '14 at 14:31

pocesar   MannyO
4,753   5   43   77   **21**   1

You can download freeware Resource Hacker and then follow below instructions :

2

1. Open any dll file you wish to find icons from.

2. Browse folders to find specific icons.

3. From menu bar, select 'action' and then 'save'.

4. Select destination for .ico file.

Reference : http://techsultan.com/how-to-extract-icons-from-windows-7/

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up   OR SIGN IN WITH   G Google   Facebook×

If you're on Linux, you can extract icons from a Windows DLL with [gExtractWinIcons](#). It's available in Ubuntu and Debian in the `gextractwinicons` package.

This blog article has a [screenshot and brief explanation](#).

1

answered Jul 23 '14 at 8:43

Smylers
1,390 10 12

There is also this resource available, the Visual Studio Image Library, which "can be used to create applications that look visually consistent with Microsoft software", presumably subject to the licensing given at the bottom. https://www.microsoft.com/en-ca/download/details.aspx?id=35825

1

answered Mar 22 '17 at 22:05

David Carr
344 3 7

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up   OR SIGN IN WITH   G Google   Facebook ✕