

# Re: File attributes

---

*Source:*

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2006-12/msg03524.html>

---

- *From:* "Bob Phillips" <[bob.NGs@xxxxxxxx](mailto:bob.NGs@xxxxxxxx)>
  - *Date:* Sat, 16 Dec 2006 23:15:42 -0000
- 

That should be in a class module, not a standard module. The class module should be called clsFileVersion because it is referred to in the code

```
Dim cVersion As clsFileVersion
```

```
Set cVersion = New clsFileVersion
```

which does go in a standard code module.

BTW, I hope that the truncated class code was just a snip, it is required.

--  
---  
HTH

Bob

(change the xxxx to gmail if mailing direct)

"Joergen Bondesen" <[bondesen@xxxxxxxx](mailto:bondesen@xxxxxxxx)> wrote in message  
[news:em1sfm\\$6fl\\$1@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:em1sfm$6fl$1@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)

Hi Bob

I am not sure what you mean by that, because I do not know what class code means.

In a MODULE I have:

```
' *****
' Copyright ©1995–2001 Karl E. Peterson, All Rights Reserved
' http://www.mvps.org/vb
' *****
' You are free to use this code within your own applications, but you
' are expressly forbidden from selling or otherwise distributing this
' source code without prior written consent.
```

Re: File attributes

'\*\*\*\*\*'

Option Explicit

'

' API declarations

'

Private Declare Function GetFullPathName Lib "kernel32" Alias

"GetFullPathNameA" ( \_

ByVal lpFileName As String, \_

..... ending with

Private Function PointerToStringB \_

(lpString As Long, nBytes As Long) As String

Dim Buffer As String

If nBytes Then

Buffer = Space(nBytes)

CopyMemory ByVal Buffer, ByVal lpString, nBytes

PointerToStringB = Buffer

End If

End Function

Sub test()

// Compile error: User-Defined type not defined (for next line)

Dim cVersion As clsFileVersion

Set cVersion = New clsFileVersion

cVersion.FullPathName = \_

"C:\Program Files\Tools\system\DSO File\dsofile.dll"

MsgBox cVersion.FileVersion

MsgBox cVersion.ProductVersion

End Sub

--

Best regards

Joergen Bondesen

"Bob Phillips" <bob.NGs@xxxxxxx> wrote in message

[news:%23\\$NySITIHHA.1248@xxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:%23$NySITIHHA.1248@xxxxxxxxxxxxxxxxxxxxxxxx)

You need the class code that was also posted in my previous reply.

--

---

HTH

Re: File attributes

Bob

(change the xxxx to gmail if mailing direct)

"Joergen Bondesen" <bondesen@xxxxxxx> wrote in message  
[news:em1047\\$2rep\\$1@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:em1047$2rep$1@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)

Hi Bob.

This is running:

```
' Try the DSO OLE file method, you have to install DSO.  
' You can get it at  
' http://support.microsoft.com/?id=224351  
' and set a reference to  
' "DSO OLE Document Properties Reader 2.0." in the  
VBIDE
```

```
Sub test2()  
Dim FSO As Object  
Set FSO = CreateObject("Scripting.FileSystemObject")
```

```
Dim this As Workbook  
Set this = ActiveWorkbook
```

```
Dim file As Variant  
Set file = FSO.GetFile("C:\Program Files\Tools\system\DSO  
File\dsofile.dll")  
MsgBox file.Size
```

```
End Sub
```

This is NOT running:

```
Sub test()  
'// Compile error: User-Defined type not defined (for next  
line)  
Dim cVersion As clsFileVersion  
Set cVersion = New clsFileVersion
```

```
cVersion.FullPathName = "C:\Program  
Files\Tools\system\DSO  
File\dsofile.dll"
```

```
MsgBox cVersion.FileVersion  
MsgBox cVersion.ProductVersion  
End Sub
```

I do hope you can help me.

Re: File attributes

Best regards  
Joergen Bondesen

"Bob Phillips" <bob.NGs@xxxxxxx> wrote in message  
news:O3hx2j2HHHA.1280@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

If you trace Randy's article back to Karl  
Peterson, you can get this  
simple class, and use code like so to get it

```
Dim cVersion As clsFileVersion
```

```
Set cVersion = New clsFileVersion
cVersion.FullPathName = "C:\Program
Files\Tools\system\DSO
File\dsofile.dll"
MsgBox cVersion.FileVersion
MsgBox cVersion.ProductVersion
```

```
'
'*****
' Copyright ©1995–2001 Karl E. Peterson,
' All Rights Reserved
' http://www.mvps.org/vb
'
'*****
' You are free to use this code within your
' own applications, but you
' are expressly forbidden from selling or
' otherwise distributing this
' source code without prior written consent.
'
'*****
Option Explicit
'
' API declarations
'
Private Declare Function GetFullPathName
Lib "kernel32" Alias
"GetFullPathNameA" ( _
ByVal lpFileName As String, _
ByVal nBufferLength As Long, _
ByVal lpBuffer As String, _
lpFilePart As Long) As Long
```

## Re: File attributes

```
Private Declare Function GetFileVersionInfo
Lib "Version.dll" Alias
"GetFileVersionInfoA" ( _
ByVal lpstrFilename As String, _
ByVal dwhandle As Long, _
ByVal dwlen As Long, _
lpData As Any) As Long
Private Declare Function
GetFileVersionInfoSize Lib "Version.dll"
Alias
"GetFileVersionInfoSizeA" ( _
ByVal lpstrFilename As String, _
lpdwHandle As Long) As Long
Private Declare Function VerQueryValue
Lib "Version.dll" Alias
"VerQueryValueA" ( _
pBlock As Any, _
ByVal lpSubBlock As String, _
lplpBuffer As Any, _
puLen As Long) As Long
Private Declare Function VerLanguageName
Lib "kernel32" Alias
"VerLanguageNameA" ( _
ByVal wLang As Long, _
ByVal szLang As String, _
ByVal nSize As Long) As Long
Private Declare Function
GetSystemDirectory Lib "kernel32" Alias
"GetSystemDirectoryA" ( _
ByVal Path As String, _
ByVal cbBytes As Long) As Long
Private Declare Sub CopyMemory Lib
"kernel32" Alias "RtlMoveMemory" ( _
Destination As Any, _
Source As Any, _
ByVal Length As Long)
Private Declare Function lstrlenA Lib
"kernel32" ( _
ByVal lpString As Long) As Long
Private Declare Function lstrlenW Lib
"kernel32" ( _
ByVal lpString As Long) As Long '

' API structures.
'
Private Type VS_FIXEDFILEINFO
dwSignature As Long
dwStrucVersionl As Integer ' e.g. = &h0000
= 0
dwStrucVersionh As Integer ' e.g. = &h0042
= .42
```

## Re: File attributes

```
dwFileVersionMSI As Integer ' e.g. =
&h0003 = 3
dwFileVersionMSh As Integer ' e.g. =
&h0075 = .75
dwFileVersionLSI As Integer ' e.g. =
&h0000 = 0
dwFileVersionLSh As Integer ' e.g. =
&h0031 = .31
dwProductVersionMSI As Integer ' e.g. =
&h0003 = 3
dwProductVersionMSh As Integer ' e.g. =
&h0010 = .1
dwProductVersionLSI As Integer ' e.g. =
&h0000 = 0
dwProductVersionLSh As Integer ' e.g. =
&h0031 = .31
dwFileFlagsMask As Long ' = &h3F for
version "0.42"
dwFileFlags As Long ' e.g. VFF_DEBUG
Or VFF_PRERELEASE
dwFileOS As Long ' e.g.
VOS_DOS_WINDOWS16
dwFileType As Long ' e.g. VFT_DRIVER
dwFileSubtype As Long ' e.g.
VFT2_DRV_KEYBOARD
dwFileDateMS As Long ' e.g. 0
dwFileDateLS As Long ' e.g. 0
End Type
'
' API constants.
'
Private Const MAX_PATH = 260
' ----- VS_VERSION.dwFileFlags -----
Private Const VS_FFI_SIGNATURE =
&HFEEF04BD
Private Const VS_FFI_STRUCVERSION =
&H10000
Private Const VS_FFI_FILEFLAGSMASK
= &H3F&
' ----- VS_VERSION.dwFileFlags -----
Private Const VS_FF_DEBUG = &H1
Private Const VS_FF_PRERELEASE =
&H2
Private Const VS_FF_PATCHED = &H4
Private Const VS_FF_PRIVATEBUILD =
&H8
Private Const VS_FF_INFOINFERRED =
&H10
Private Const VS_FF_SPECIALBUILD =
&H20
' ----- VS_VERSION.dwFileOS -----
```

Re: File attributes

```
Private Const VOS_UNKNOWN = &H0
Private Const VOS_DOS = &H10000
Private Const VOS_OS216 = &H20000
Private Const VOS_OS232 = &H30000
Private Const VOS_NT = &H40000
Private Const VOS_DOS_WINDOWS16 =
&H10001
Private Const VOS_DOS_WINDOWS32 =
&H10004
Private Const VOS_OS216_PM16 =
&H20002
Private Const VOS_OS232_PM32 =
&H30003
Private Const VOS_NT_WINDOWS32 =
&H40004
' ----- VS_VERSION.dwFileType -----
Private Const VFT_UNKNOWN = &H0
Private Const VFT_APP = &H1
Private Const VFT_DLL = &H2
Private Const VFT_DRV = &H3
Private Const VFT_FONT = &H4
Private Const VFT_VXD = &H5
Private Const VFT_STATIC_LIB = &H7
' **** VS_VERSION.dwFileSubtype for
VFT_WINDOWS_FONT ****
Private Const VFT2_FONT_RASTER =
&H1&
Private Const VFT2_FONT_VECTOR =
&H2&
Private Const VFT2_FONT_TRUETYPE =
&H3&
' ----- VS_VERSION.dwFileSubtype for
VFT_WINDOWS_DRV -----
Private Const VFT2_UNKNOWN = &H0
Private Const VFT2_DRV_PRINTER =
&H1
Private Const VFT2_DRV_KEYBOARD =
&H2
Private Const VFT2_DRV_LANGUAGE =
&H3
Private Const VFT2_DRV_DISPLAY =
&H4
Private Const VFT2_DRV_MOUSE = &H5
Private Const VFT2_DRV_NETWORK =
&H6
Private Const VFT2_DRV_SYSTEM =
&H7
Private Const VFT2_DRV_INSTALLABLE
= &H8
Private Const VFT2_DRV_SOUND = &H9
Private Const VFT2_DRV_COMM = &HA
```

## Re: File attributes

```
'  
' Member variables.  
'  
Private m_PathName As String  
Private m_Available As Boolean  
Private m_StrucVer As String ' Structure  
Version – NOT USED  
Private m_FileVer As String ' File Version  
Private m_ProdVer As String ' Product  
Version  
Private m_FileFlags As String ' Boolean  
attributes of file  
Private m_FileOS As String ' OS file is  
designed for  
Private m_FileType As String ' Type of file  
Private m_FileSubType As String '  
Sub-type of file  
Private m_VerLanguage As String  
Private m_VerComments As String  
Private m_VerCompany As String  
Private m_VerDescription As String  
Private m_VerFileVer As String  
Private m_VerInternalName As String  
Private m_VerCopyright As String  
Private m_VerTrademarks As String  
Private m_VerOrigFilename As String  
Private m_VerProductName As String  
Private m_VerProductVer As String  
Private m_VerPrivateBuild As String  
Private m_VerSpecialBuild As String  
Private m_VerTrademarks1 As String  
Private m_VerTrademarks2 As String  
  
Public Enum VersionInfoStrings  
viPredefinedFirst = 0  
viLanguage = 0  
viComments = 1  
viCompanyName = 2  
viFileDescription = 3  
viFileVersion = 4  
viInternalName = 5  
viLegalCopyright = 6  
viLegalTrademarks = 7  
viOriginalFilename = 8  
viProductName = 9  
viProductVersion = 10  
viPrivateBuild = 11  
viSpecialBuild = 12  
viLegalTrademarks1 = 13 'Used by Office  
apps only?  
viLegalTrademarks2 = 14 'Used by Office
```

## Re: File attributes

```
apps only?
viPredefinedLast = 14
End Enum

'
*****
' Initialize and Terminate
'
*****
Private Sub Class_Initialize()
'
' All member variables can be left to
defaults.
'
End Sub

Private Sub Class_Terminate()
'
' No special cleanup required.
'
End Sub

'
*****
' Public Properties
'
*****
Public Property Let FullPathName(ByVal
NewVal As String)
Dim Buffer As String
Dim nFilePart As Long
Dim nRet As Long
'
' Retrieve fully qualified path/name specs.
'
Buffer = Space(MAX_PATH)
nRet = GetFullPathName(NewVal,
Len(Buffer), Buffer, nFilePart)
If nRet Then
m_PathName = Left(Buffer, nRet)
Refresh
End If
End Property

Public Property Get FullPathName() As
String
' Returns fully-qualified path/name spec.
FullPathName = m_PathName
End Property

Public Property Get Available() As Boolean
```

## Re: File attributes

```
' Returns whether version information is
available
Available = m_Available
End Property

'
*****
' Standard Version Information
'
*****
Public Property Get FileFlags() As String
FileFlags = m_FileFlags
End Property

Public Property Get FileOS() As String
FileOS = m_FileOS
End Property

Public Property Get FileType() As String
FileType = m_FileType
End Property

Public Property Get FileSubType() As String
FileSubType = m_FileSubType
End Property

Public Property Get VerFile() As String
VerFile = m_FileVer
End Property

Public Property Get VerProduct() As String
VerProduct = m_ProdVer
End Property

Public Property Get VerStructure() As String
VerStructure = m_StructVer
End Property

'
*****
' Better Version Information
'
*****
Public Property Get PredefinedName(ByVal
Which As VersionInfoStrings)
As String
Select Case Which
Case viLanguage
PredefinedName = "Language"
Case viComments
PredefinedName = "Comments"
```

Re: File attributes

```
Case viCompanyName
PredefinedName = "Company Name"
Case viFileDescription
PredefinedName = "File Description"
Case viFileVersion
PredefinedName = "File Version"
Case viInternalName
PredefinedName = "Internal Name"
Case viLegalCopyright
PredefinedName = "Legal Copyright"
Case viLegalTrademarks
PredefinedName = "Legal Trademarks"
Case viOriginalFilename
PredefinedName = "Original Filename"
Case viProductName
PredefinedName = "Product Name"
Case viProductVersion
PredefinedName = "Product Version"
Case viPrivateBuild
PredefinedName = "Private Build"
Case viSpecialBuild
PredefinedName = "Special Build"
Case viLegalTrademarks1
PredefinedName = "LegalTrademarks1"
Case viLegalTrademarks2
PredefinedName = "LegalTrademarks2"
End Select
End Property
```

```
Public Property Get PredefinedValue(ByVal
Which As VersionInfoStrings)
As String
Select Case Which
Case viLanguage
PredefinedValue = m_VerLanguage
Case viComments
PredefinedValue = m_VerComments
Case viCompanyName
PredefinedValue = m_VerCompany
Case viFileDescription
PredefinedValue = m_VerDescription
Case viFileVersion
PredefinedValue = m_VerFileVer
Case viInternalName
PredefinedValue = m_VerInternalName
Case viLegalCopyright
PredefinedValue = m_VerCopyright
Case viLegalTrademarks
PredefinedValue = m_VerTrademarks
Case viOriginalFilename
PredefinedValue = m_VerOrigFilename
```

## Re: File attributes

```
Case viProductName
PredefinedValue = m_VerProductName
Case viProductVersion
PredefinedValue = m_VerProductVer
Case viPrivateBuild
PredefinedValue = m_VerPrivateBuild
Case viSpecialBuild
PredefinedValue = m_VerSpecialBuild
Case viLegalTrademarks1
PredefinedValue = m_VerTrademarks1
Case viLegalTrademarks2
PredefinedValue = m_VerTrademarks2
End Select
End Property
```

```
Public Property Get Comments() As String
Comments = m_VerComments
End Property
```

```
Public Property Get CompanyName() As
String
CompanyName = m_VerCompany
End Property
```

```
Public Property Get FileDescription() As
String
FileDescription = m_VerDescription
End Property
```

```
Public Property Get FileVersion() As String
FileVersion = m_VerFileVer
End Property
```

```
Public Property Get InternalName() As
String
InternalName = m_VerInternalName
End Property
```

```
Public Property Get Language() As String
Language = m_VerLanguage
End Property
```

```
Public Property Get LegalCopyright() As
String
LegalCopyright = m_VerCopyright
End Property
```

```
Public Property Get LegalTrademarks() As
String
LegalTrademarks = m_VerTrademarks
End Property
```

Re: File attributes

```
Public Property Get OriginalFilename() As String
OriginalFilename = m_VerOrigFilename
End Property
```

```
Public Property Get ProductName() As String
ProductName = m_VerProductName
End Property
```

```
Public Property Get ProductVersion() As String
ProductVersion = m_VerProductVer
End Property
```

```
Public Property Get PrivateBuild() As String
PrivateBuild = m_VerPrivateBuild
End Property
```

```
Public Property Get SpecialBuild() As String
SpecialBuild = m_VerSpecialBuild
End Property
```

```
,
*****
' Public Methods
'
*****
Public Sub Refresh()
Dim nDummy As Long
Dim nRet As Long
Dim sBuffer() As Byte
Dim nBufferLen As Long
Dim lplpBuffer As Long
Dim udtVerBuffer As VS_FIXEDFILEINFO
Dim puLen As Long
Dim nLanguage As Integer
Dim nCodePage As Integer
Dim sSubBlock As String
Dim sTemp As String
'
' Get size
'
nBufferLen =
GetFileVersionInfoSize(m_PathName,
nDummy)
If nBufferLen Then
m_Available = True
Else
```

## Re: File attributes

```
m_Available = False
Exit Sub
End If
'
' Store info to udtVerBuffer struct
'
ReDim sBuffer(nBufferLen) As Byte
Call GetFileVersionInfo(m_PathName, 0&,
nBufferLen, sBuffer(0))
Call VerQueryValue(sBuffer(0), "\",
lpBuffer, puLen)
Call CopyMemory(udtVerBuffer, ByVal
lpBuffer, Len(udtVerBuffer))
'
' Determine Structure Version number –
NOT USED
'
m_StructVer =
Format$(udtVerBuffer.dwStrucVersionh) &
"." & _
Format$(udtVerBuffer.dwStrucVersionl)
'
' Determine File Version number
'
m_FileVer =
Format$(udtVerBuffer.dwFileVersionMSH)
& "." & _
Format$(udtVerBuffer.dwFileVersionMSL,
"00") & "."
If udtVerBuffer.dwFileVersionLSH > 0 Then
m_FileVer = m_FileVer &
Format$(udtVerBuffer.dwFileVersionLSH,
"00") & _
Format$(udtVerBuffer.dwFileVersionLSL,
"00")
Else
m_FileVer = m_FileVer &
Format$(udtVerBuffer.dwFileVersionLSL,
"0000")
End If
'
' Determine Product Version number
'
m_ProdVer =
Format$(udtVerBuffer.dwProductVersionMSH)
& "." & _
Format$(udtVerBuffer.dwProductVersionMSL,
"00") & "."
If udtVerBuffer.dwProductVersionLSH > 0
Then
m_ProdVer = m_ProdVer &
```

## Re: File attributes

```
Format$(udtVerBuffer.dwProductVersionLSh,
"00") & _
Format$(udtVerBuffer.dwProductVersionLSl,
"00")
Else
m_ProdVer = m_ProdVer &
Format$(udtVerBuffer.dwProductVersionLSl,
"0000")
End If
'
' Determine Boolean attributes of File
'
m_FileFlags = ""
If udtVerBuffer.dwFileFlags And
VS_FF_DEBUG _
Then m_FileFlags = "Debug "
If udtVerBuffer.dwFileFlags And
VS_FF_PRERELEASE _
Then m_FileFlags = m_FileFlags & "PreRel
"
If udtVerBuffer.dwFileFlags And
VS_FF_PATCHED _
Then m_FileFlags = m_FileFlags &
"Patched "
If udtVerBuffer.dwFileFlags And
VS_FF_PRIVATEBUILD _
Then m_FileFlags = m_FileFlags & "Private
"
If udtVerBuffer.dwFileFlags And
VS_FF_INFOINFERRED _
Then m_FileFlags = m_FileFlags & "Info "
If udtVerBuffer.dwFileFlags And
VS_FF_SPECIALBUILD _
Then m_FileFlags = m_FileFlags & "Special
"
If udtVerBuffer.dwFileFlags And
VFT2_UNKNOWN _
Then m_FileFlags = m_FileFlags +
"Unknown "
m_FileFlags = Trim(m_FileFlags)
'
' Determine OS for which file was designed
'
Select Case udtVerBuffer.dwFileOS
Case VOS_DOS_WINDOWS16
m_FileOS = "DOS-Win16"
Case VOS_DOS_WINDOWS32
m_FileOS = "DOS-Win32"
Case VOS_OS216_PM16
m_FileOS = "OS/2-16 PM-16"
Case VOS_OS232_PM32
```

Re: File attributes

```
m_FileOS = "OS/2-16 PM-32"  
Case VOS_NT_WINDOWS32  
m_FileOS = "NT-Win32"  
Case Else  
m_FileOS = "Unknown"  
End Select  
'  
' Determine type of file  
'  
Select Case udtVerBuffer.dwFileType  
Case VFT_APP  
m_FileType = "Application"  
Case VFT_DLL  
m_FileType = "DLL"  
Case VFT_DRV  
m_FileType = "Driver"  
Select Case udtVerBuffer.dwFileSubtype  
Case VFT2_DRV_PRINTER  
m_FileSubType = "Printer drv"  
Case VFT2_DRV_KEYBOARD  
m_FileSubType = "Keyboard drv"  
Case VFT2_DRV_LANGUAGE  
m_FileSubType = "Language drv"  
Case VFT2_DRV_DISPLAY  
m_FileSubType = "Display drv"  
Case VFT2_DRV_MOUSE  
m_FileSubType = "Mouse drv"  
Case VFT2_DRV_NETWORK  
m_FileSubType = "Network drv"  
Case VFT2_DRV_SYSTEM  
m_FileSubType = "System drv"  
Case VFT2_DRV_INSTALLABLE  
m_FileSubType = "Installable"  
Case VFT2_DRV_SOUND  
m_FileSubType = "Sound drv"  
Case VFT2_DRV_COMM  
m_FileSubType = "Comm drv"  
Case VFT2_UNKNOWN  
m_FileSubType = "Unknown"  
End Select  
Case VFT_FONT  
m_FileType = "Font"  
Select Case udtVerBuffer.dwFileSubtype  
Case VFT2_FONT_RASTER  
m_FileSubType = "Raster Font"  
Case VFT2_FONT_VECTOR  
m_FileSubType = "Vector Font"  
Case VFT2_FONT_TRUETYPE  
m_FileSubType = "TrueType Font"  
End Select  
Case VFT_VXD
```

Re: File attributes

```
m_FileType = "VxD"
Case VFT_STATIC_LIB
m_FileType = "Lib"
Case Else
m_FileType = "Unknown"
End Select
'
' Get language translations
'
If VerQueryValue(sBuffer(0),
"\VarFileInfo\Translation", lpBuffer,
puLen) Then
If puLen Then
'
' Handle the language/codepage as a dword.
'
nRet = PointerToDWord(lpBuffer)
nLanguage = LoWord(nRet)
nCodePage = HiWord(nRet)
'
' Determine language
'
m_VerLanguage = Space(256)
nRet =
VerLanguageName(CLng(nLanguage),
m_VerLanguage,
Len(m_VerLanguage))
If nRet Then
m_VerLanguage = Left(m_VerLanguage,
nRet)
Else
m_VerLanguage = ""
End If
'
' Microsoft really screwed up in a number of
their
' applications, including Office 95 and
Office 97,
' by storing the resources under the U.S.
English
' codepage after identifying the language as
neutral.
' Test to see if no company name shows up
in these
' cases, then try English to see if that works
better.
'
sSubBlock = "\StringFileInfo\" & _
FmtHex(nLanguage, 4) & _
FmtHex(nCodePage, 4) & "\"
m_VerCompany =
```

## Re: File attributes

```
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"CompanyName")
If Len(m_VerCompany) = 0 Then
' Try U.S. English...?
sTemp = "\StringFileInfo\" & _
FmtHex(&H409, 4) & _
FmtHex(nCodePage, 4) & "\"
m_VerCompany =
GetStdValue(VarPtr(sBuffer(0)), sTemp &
"CompanyName")
If Len(m_VerCompany) > 0 Then
' We probably found the MS version bug.
sSubBlock = sTemp
End If
End If
'
' Get remaining predefined version resources
'
m_VerComments =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"Comments")
m_VerDescription =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"FileDescription")
m_VerFileVersion =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"FileVersion")
m_VerInternalName =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"InternalName")
m_VerCopyright =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"LegalCopyright")
m_VerTrademarks =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"LegalTrademarks")
m_VerOrigFilename =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"OriginalFilename")
m_VerProductName =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"ProductName")
```

Re: File attributes

```
m_VerProductVer =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"ProductVersion")
m_VerPrivateBuild =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"PrivateBuild")
m_VerSpecialBuild =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"SpecialBuild")
m_VerTrademarks1 =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"LegalTrademarks1")
m_VerTrademarks2 =
GetStdValue(VarPtr(sBuffer(0)), sSubBlock
&
"LegalTrademarks2")
End If
End If
End Sub

'
*****
' Private Methods
'
*****
Private Function FmtHex(ByVal InVal As
Long, ByVal OutLen As Integer)
As String
' Left pad with zeros to OutLen.
FmtHex = Right$(String$(OutLen, "0") &
Hex$(InVal), OutLen)
End Function

Private Function GetStdValue(ByVal
lpBlock As Long, ByVal Value As
String) As String
Dim lplpBuffer As Long
Dim puLen As Long
If VerQueryValue(ByVal lpBlock, Value,
lplpBuffer, puLen) Then
If puLen Then
GetStdValue = PointerToString(lplpBuffer)
End If
End If
End Function

Private Function LoWord(ByVal LongIn As
```

## Re: File attributes

```
Long) As Integer
Call CopyMemory(LoWord, LongIn, 2)
End Function
```

```
Private Function HiWord(ByVal LongIn As
Long) As Integer
Call CopyMemory(HiWord, ByVal
(VarPtr(LongIn) + 2), 2)
End Function
```

```
Private Function PointerToDWord(ByVal
lpDWord As Long) As Long
Dim nRet As Long
If lpDWord Then
CopyMemory nRet, ByVal lpDWord, 4
PointerToDWord = nRet
End If
End Function
```

```
Private Function
PointerToStringW(lpStringW As Long) As
String
Dim Buffer() As Byte
Dim nLen As Long
```

```
If lpStringW Then
nLen = lstrlenW(lpStringW) * 2
If nLen Then
ReDim Buffer(0 To (nLen - 1)) As Byte
CopyMemory Buffer(0), ByVal lpStringW,
nLen
PointerToStringW = Buffer
End If
End If
End Function
```

```
Private Function PointerToString(lpString
As Long) As String
Dim Buffer As String
Dim nLen As Long
```

```
If lpString Then
nLen = lstrlenA(lpString)
If nLen Then
Buffer = Space(nLen)
CopyMemory ByVal Buffer, ByVal
lpString, nLen
PointerToString = Buffer
End If
End If
End Function
```

Re: File attributes

```
Private Function PointerToStringB(lpString  
As Long, nBytes As Long) As  
String  
Dim Buffer As String
```

```
If nBytes Then  
Buffer = Space(nBytes)  
CopyMemory ByVal Buffer, ByVal  
lpString, nBytes  
PointerToStringB = Buffer  
End If  
End Function
```

--

----

HTH

Bob

(change the xxxx to gmail if mailing direct)

"Alan"

<Alan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

wrote in message

[news:D3CC6718-A3AC-4EB8-B255-D706B62FDA6D@xxxxxxxxxxxxxxxxxxxx](mailto:news:D3CC6718-A3AC-4EB8-B255-D706B62FDA6D@xxxxxxxxxxxxxxxxxxxx)

Many thanks to all  
I am still unsure how to  
retrieve the 'Product  
Version' attribute from  
a  
file, however will continue  
to plug away. Im not sure  
why Version and  
Product  
Version are different in  
some of the files I am  
listing, but will try  
to  
source an answer to this  
elsewhere

Thanks again

Re: File attributes

"Bob Phillips" wrote:

File size is  
simple

Set FSO =  
CreateObject("Scripting.FileSystemObject")

Set this =  
ActiveWorkbook  
Set file =  
FSO.GetFile("C:\Program  
Files\Tools\system\DSO  
File\dsofile.dll")  
MsgBox  
file.Size

but version  
seems  
tricky I  
admit

--  
---

HTH

Bob

(change the  
xxxx to  
gmail if  
mailing  
direct)

"Alan"

<Alan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

wrote in  
message

[news:D0187F5E-96F8-42C2-8F4B-8D20C9E5CDA9@xxxxxxxxxxx](mailto:news:D0187F5E-96F8-42C2-8F4B-8D20C9E5CDA9@xxxxxxxxxxx)

Many  
Thanks  
Bob

Almost  
there

...

Re: File attributes

although  
still  
cant  
find  
the  
'Product  
Version'  
attribute

...  
my  
file  
has  
a  
value  
also  
I  
noted  
that  
the  
DSO  
OLE  
doesnt  
give  
me  
the  
option  
for  
size  
as  
a  
comparison  
??

Once  
again  
your  
assistance  
is  
very  
much  
appreciated

"Bob  
Phillips"  
wrote:

Try  
the  
DSO

Re: File attributes

OLE  
file  
method,  
you  
have  
to  
install  
DSO.  
You  
can  
get  
it  
at  
<http://support.microsoft.com/?id=224351>  
and  
set  
a  
reference  
to  
"DSO  
OLE  
Document  
Properties  
Reader  
2.0."  
in  
the  
VBIDE

Option  
Explicit

Const  
COL\_Application  
As  
String  
=  
1  
Const  
COL\_Author  
As  
String  
=  
2  
Const  
COL\_Version  
As  
String  
=  
3

Re: File attributes

Const  
COL\_Subject  
As  
String  
=  
4  
Const  
COL\_Category  
As  
String  
=  
5  
Const  
COL\_Company  
As  
String  
=  
6  
Const  
COL\_Keywords  
As  
String  
=  
7  
Const  
COL\_Manager  
As  
String  
=  
8  
Const  
COL\_LastSavedBy  
As  
String  
=  
9  
Const  
COL\_WordCount  
As  
String  
=  
10  
Const  
COL\_PageCount  
As  
String  
=  
11  
Const  
COL\_ParagraphCount  
As

Re: File attributes

String  
=  
12  
Const  
COL\_LineCount  
As  
String  
=  
13  
Const  
COL\_CharacterCount  
As  
String  
=  
14  
Const  
COL\_CharacterCountspaces  
As  
String  
=  
15  
Const  
COL\_ByteCount  
As  
String  
=  
16  
Const  
COL\_PresFormat  
As  
String  
=  
17  
Const  
COL\_SlideCount  
As  
String  
=  
18  
Const  
COL\_NoteCount  
As  
String  
=  
19  
Const  
COL\_HiddenSlides  
As  
String  
=  
20

Re: File attributes

Const  
COL\_MultimediaClips  
As  
String  
=  
21  
Const  
COL\_DateCreated  
As  
String  
=  
22  
Const  
COL\_DateLastPrinted  
As  
String  
=  
23  
Const  
COL\_DateLastSaved  
As  
String  
=  
24  
Const  
COL\_TotalEditingTime  
As  
String  
=  
25  
Const  
COL\_Template  
As  
String  
=  
26  
Const  
COL\_Revision  
As  
String  
=  
27  
Const  
COL\_IsShared  
As  
String  
=  
28  
Const  
COL\_CLSID  
As

Re: File attributes

```
String
=
29
Const
COL_ProgID
As
String
=
30
Const
COL_OleFormat
As
String
=
1
Const
COL_OleType
As
String
=
32
```

```
Sub
ListFileAttributes()
Dim
FSO
As
Object
Dim
i
As
Long
Dim
sFolder
As
String
Dim
fldr
As
Object
Dim
Folder
As
Object
Dim
file
As
Object
Dim
Files
```

Re: File attributes

```
As  
Object  
Dim  
this  
As  
Workbook  
Dim  
aryFiles  
Dim  
cnt  
As  
Long  
Dim  
sh  
As  
Worksheet
```

```
Set  
FSO  
=  
CreateObject("Scripting.FileSystemObject")
```

```
Set  
this  
=  
ActiveWorkbook  
sFolder  
=  
"C:\MyTest"  
Set  
Folder  
=  
FSO.GetFolder(sFolder)  
Set  
Files  
=  
Folder.Files  
cnt  
=  
0  
ReDim  
aryFiles(1  
To  
33,  
1  
To  
1)  
For  
Each
```

Re: File attributes

```
file
In
Files
If
file.Type
=
"Microsoft
Excel
Worksheet"
Then
Call
DSO(file.Path,
aryFiles)
End
If
Next
file

On
Error
Resume
Next
Set
sh
=
Worksheets("ListOfFiles")
On
Error
GoTo
0
If
sh
Is
Nothing
Then
Worksheets.Add.Name
=
"ListOfFiles"
Else
sh.Cells.ClearContents
End
If

For
i
=
LBound(aryFiles,
2)
To
```

Re: File attributes

```
UBound(aryFiles,  
2)  
Cells(i  
+  
1,  
"A").Value  
=  
aryFiles(COL_Author,  
i)  
Next  
i  
Columns("A:C").AutoFit
```

```
End  
Sub
```

```
Sub  
DSO(ByVal  
FileName  
As  
String,  
ByRef  
aryData)  
Static  
notFirstTime  
As  
Boolean  
Dim  
fOpenReadOnly  
As  
Boolean  
Dim  
DSO  
As  
DSOFile.OleDocumentProperties  
Dim  
oSummProps  
As  
DSOFile.SummaryProperties  
Dim  
oCustProp  
As  
DSOFile.CustomProperty  
Dim  
iNext  
As  
Long
```

Re: File attributes

```
If
notFirstTime
Then
iNext
=
UBound(aryData,
2)
+
1
Else
iNext
=
UBound(aryData,
2)
notFirstTime
=
True
End
If
ReDim
Preserve
aryData(1
To
33,
1
To
iNext)

Set
DSO
=
New
DSOFile.OleDocumentProperties
DSO.Open
FileName,
fOpenReadOnly,
dsoOptionOpenReadOnlyIfNoWriteAccess

'Get
the
SummaryProperties
(these
are
built-in
set)...
Set
oSummProps
=
DSO.SummaryProperties
```

Re: File attributes

```
aryData(1,  
iNext)  
=  
oSummProps.ApplicationName  
aryData(2,  
iNext)  
=  
oSummProps.Author  
aryData(3,  
iNext)  
=  
oSummProps.Version  
aryData(4,  
iNext)  
=  
oSummProps.Subject  
aryData(5,  
iNext)  
=  
oSummProps.Category  
aryData(6,  
iNext)  
=  
oSummProps.Company  
aryData(7,  
iNext)  
=  
oSummProps.Keywords  
aryData(8,  
iNext)  
=  
oSummProps.Manager  
aryData(9,  
iNext)  
=  
oSummProps.LastSavedBy  
aryData(10,  
iNext)  
=  
oSummProps.WordCount  
aryData(11,  
iNext)  
=  
oSummProps.PageCount  
aryData(12,  
iNext)  
=  
oSummProps.ParagraphCount  
aryData(13,  
iNext)  
=
```

Re: File attributes

oSummProps.LineCount  
aryData(14,  
iNext)  
=  
oSummProps.CharacterCount  
aryData(15,  
iNext)  
=  
oSummProps.CharacterCountWithSpaces  
aryData(16,  
iNext)  
=  
oSummProps.ByteCount  
aryData(17,  
iNext)  
=  
oSummProps.PresentationFormat  
aryData(18,  
iNext)  
=  
oSummProps.SlideCount  
aryData(19,  
iNext)  
=  
oSummProps.NoteCount  
aryData(20,  
iNext)  
=  
oSummProps.HiddenSlideCount  
aryData(21,  
iNext)  
=  
oSummProps.MultimediaClipCount  
aryData(22,  
iNext)  
=  
oSummProps.DateCreated  
aryData(23,  
iNext)  
=  
oSummProps.DateLastPrinted  
aryData(24,  
iNext)  
=  
oSummProps.DateLastSaved  
aryData(25,  
iNext)  
=  
oSummProps.TotalEditTime  
aryData(26,  
iNext)

Re: File attributes

```
=
oSummProps.Template
aryData(27,
iNext)
=
oSummProps.RevisionNumber
aryData(28,
iNext)
=
oSummProps.SharedDocument
'Add
a
few
other
items
that
pertain
to
OLE
files
only...
If
DSO.IsOleFile
Then
aryData(29,
iNext)
=
DSO.CLSID
aryData(30,
iNext)
=
DSO.progID
aryData(31,
iNext)
=
DSO.OleDocumentFormat
aryData(32,
iNext)
=
DSO.OleDocumentType
End
If

'Now
the
custom
properties
For
Each
oCustProp
```

Re: File attributes

```
In
DSO.CustomProperties
aryData(33,
iNext)
=
CStr(oCustProp.Value)
Next
oCustProp
```

```
Set
oCustProp
=
Nothing
Set
oSummProps
=
Nothing
Set
DSO
=
Nothing
```

```
End
Sub
```

```
--
---
```

HTH

Bob

(change  
the  
xxxx  
to  
gmail  
if  
mailing  
direct)

"Alan"  
<Alan@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>  
wrote  
in  
message  
[news:3335A02B-6743-4A4A-9704-45E9D47D56E](mailto:news:3335A02B-6743-4A4A-9704-45E9D47D56E)

Re: File attributes

Martin

'File  
Version'  
and  
'Product  
Version'  
are  
file  
attributes  
selectable/viewable  
within  
windows  
explorer.

I  
note  
NickHK  
comments  
using  
API  
and  
will  
maybe  
need  
to  
try  
that  
although  
would  
rather  
write  
to  
a  
worksheet  
than  
display  
in  
a  
window

Many  
thanks  
for  
your  
response

"Martin  
Fishlock"  
wrote:

Re: File attributes

Alan,

Where  
are  
the  
'File  
Verion'  
and  
'Product  
Version'  
items?

Are  
they  
custom  
attributes?

Please  
explain  
a  
little  
more.

--  
Hope  
this  
helps  
Martin  
Fishlock  
Please  
do  
not  
forget  
to  
rate  
this  
reply.

"Alan"  
wrote:

I  
have  
managed  
to  
use  
a  
macro  
(VBA)

Re: File attributes

to  
list  
the  
file  
structure  
of  
a  
particular  
folder  
and  
would  
now  
like  
this  
list  
to  
have  
a  
number  
of  
file  
attributes  
associated  
with  
each  
file  
including  
File  
Verion  
and  
product  
Version.

Can  
anyone  
point  
me  
in  
the  
right  
direction...  
please  
...  
I  
am  
using  
the  
Filesearch  
method  
to  
get

Re: File attributes

the  
list

Many  
Thanks