

Re: File Dialog API – UNC Path

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.modulesdaovba/2007-07/msg00696.html>

- *From:* RD <progr@xxxxxxxx>
 - *Date:* Fri, 20 Jul 2007 14:36:50 -0700
-

On Thu, 19 Jul 2007 16:22:02 -0700, Tanya <Tanya@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote:

Hi All,

I am using the below piece of code created by Ken Getz for my file dialog box which works perfectly.

<http://www.mvps.org/access/api/api0001.htm>

The only problem is that i would prefer The filepath to be a UNC Filepath, at the moment it brings out the drive letter path. Does anyone know if its possible to change this api so that it provides the UNC path and if so what would i need to change and what to change it to?

Thanks

Tanya

Hi Tanya,

Here is some code that converts mapped drives to UNC's that I lifted from the web. It was some time ago and I don't remember where I got it. If anyone recognizes it, let me know.

HTH,
RD

<code>

```
Const VER_PLATFORM_WIN32s = 0 'Win32s on Windows 3.1
Const VER_PLATFORM_WIN32_WINDOWS = 1 'Win32 on Windows 95
Const VER_PLATFORM_WIN32_NT = 2 'Win32 on Windows NT
```

```
Type OSVERSIONINFO
dwOSVersionInfoSize As Long
dwMajorVersion As Long
dwMinorVersion As Long
dwBuildNumber As Long
dwPlatformId As Long
szCSDVersion As String * 128
End Type
```

Re: File Dialog API – UNC Path

```
Private Declare Function GetVersionEx Lib "kernel32" _  
Alias "GetVersionExA" (lpVersionInformation As OSVERSIONINFO) As Long
```

```
' Declare for Registry functions
```

```
Const HKEY_CLASSES_ROOT = &H80000000  
Const HKEY_CURRENT_USER = &H80000001  
Const HKEY_LOCAL_MACHINE = &H80000002  
Const HKEY_USERS = &H80000003  
Const HKEY_PERFORMANCE_DATA = &H80000004  
Const HKEY_CURRENT_CONFIG = &H80000005  
Const HKEY_DYN_DATA = &H80000006
```

```
Private Declare Function RegCloseKey Lib "advapi32.dll" _  
(ByVal hKey As Long) As Long
```

```
Private Declare Function RegOpenKeyEx Lib "advapi32.dll" _  
Alias "RegOpenKeyExA" (ByVal hKey As Long, ByVal lpSubKey _  
As String, ByVal ulOptions As Long, ByVal samDesired _  
As Long, phkResult As Long) As Long
```

```
Private Declare Function RegQueryValue Lib "advapi32.dll" Alias _  
"RegQueryValueA" (ByVal hKey As Long, ByVal lpSubKey As _  
String, ByVal lpValue As String, lpcbValue As Long) As Long
```

```
' Note that if you declare lpData as String, then it is  
' necessary to pass it with ByVal
```

```
Private Declare Function RegQueryValueEx Lib "advapi32.dll" _  
Alias "RegQueryValueExA" (ByVal hKey As Long, _  
ByVal lpValueName As String, ByVal lpReserved As Long, _  
lpType As Long, lpData As Any, lpcbData As Long) As Long
```

```
Private Declare Function RegEnumKey Lib "advapi32.dll" _  
Alias "RegEnumKeyA" (ByVal hKey As Long, ByVal dwIndex _  
As Long, ByVal lpName As String, ByVal cbName As Long) _  
As Long
```

```
Private Declare Function RegEnumValue Lib "advapi32.dll" _  
Alias "RegEnumValueA" (ByVal hKey As Long, ByVal dwIndex _  
As Long, ByVal lpValueName As String, lpcbValueName _  
As Long, ByVal lpReserved As Long, lpType As Long, _  
ByVal lpData As String, lpcbData As Long) As Long
```

```
Private Declare Function RegOpenKey Lib "advapi32.dll" _  
Alias "RegOpenKeyA" (ByVal hKey As Long, _  
ByVal lpSubKey As String, phkResult As Long) As Long
```

```
Private Declare Function GetComputerName Lib "kernel32" _  
Alias "GetComputerNameA" (ByVal lpBuffer As String, _  
nSize As Long) As Long
```

Re: File Dialog API – UNC Path

```
Private Declare Function WNetGetConnection Lib _
"mpr.dll" Alias "WNetGetConnectionA" (ByVal lpszLocalName _
As String, ByVal lpszRemoteName As String, _
cbRemoteName As Long) As Long

Function GetUNCNameNT(pathName As String) As String

Dim hKey As Long
Dim hKey2 As Long
Dim exitFlag As Boolean
Dim i As Double
Dim ErrCode As Long
Dim rootKey As String
Dim key As String
Dim computerName As String
Dim lComputerName As Long
Dim stPath As String
Dim firstLoop As Boolean
Dim ret As Boolean

' first, verify whether the disk is connected to the network
If Mid(pathName, 2, 1) = ":" Then
Dim UNCName As String
Dim lenUNC As Long

UNCName = String$(520, 0)
lenUNC = 520
ErrCode = WNetGetConnection(Left(pathName, 2), UNCName, lenUNC)

If ErrCode = 0 Then
UNCName = Trim(Left$(UNCName, InStr(UNCName, _
vbNullChar) - 1))
GetUNCNameNT = UNCName & Mid(pathName, 3)
Exit Function
End If
End If

' else, scan the registry looking for shared resources
'(NT version)
computerName = String$(255, 0)
lComputerName = Len(computerName)
ErrCode = GetComputerName(computerName, lComputerName)
If ErrCode <> 1 Then
GetUNCNameNT = pathName
Exit Function
End If

computerName = Trim(Left$(computerName, InStr(computerName, _
vbNullChar) - 1))
rootKey = "SYSTEM\CurrentControlSet\Services\LanmanServer\Shares"
ErrCode = RegOpenKey(HKEY_LOCAL_MACHINE, rootKey, hKey)
```

Re: File Dialog API – UNC Path

```
If ErrCode <> 0 Then
GetUNCNameNT = pathName
Exit Function
End If

firstLoop = True

Do Until exitFlag
Dim szValue As String
Dim szValueName As String
Dim cchValueName As Long
Dim dwValueType As Long
Dim dwValueSize As Long

szValueName = String(1024, 0)
cchValueName = Len(szValueName)
szValue = String$(500, 0)
dwValueSize = Len(szValue)

' loop on "i" to access all shared DLLs
' szValueName will receive the key that identifies an element
ErrCode = RegEnumValue(hKey, i#, szValueName, _
cchValueName, 0, dwValueType, szValue, dwValueSize)

If ErrCode <> 0 Then
If Not firstLoop Then
exitFlag = True
Else
i = -1
firstLoop = False
End If
Else
stPath = GetPath(szValue)
If firstLoop Then
ret = (UCase(stPath) = UCase(pathName))
stPath = ""
Else
ret = (UCase(stPath) = UCase(Left$(pathName, _
Len(stPath))))
stPath = Mid$(pathName, Len(stPath))
End If
If ret Then
exitFlag = True
szValueName = Left$(szValueName, cchValueName)
GetUNCNameNT = "\\ " & computerName & "\ " & _
szValueName & stPath
End If
End If
i = i + 1
Loop
```

Re: File Dialog API – UNC Path

Re: File Dialog API – UNC Path

```
RegCloseKey hKey  
If GetUNCNameNT = "" Then GetUNCNameNT = pathName
```

```
End Function
```

```
Function GetPath(st As String) As String  
Dim pos1 As Long, pos2 As Long, pos3 As Long  
Dim stPath As String  
'Stop  
pos1 = InStr(st, "Path")  
If pos1 > 0 Then  
pos2 = InStr(pos1, st, vbNullChar)  
stPath = Mid$(st, pos1, pos2 - pos1)  
pos3 = InStr(stPath, "=")  
If pos3 > 0 Then  
stPath = Mid$(stPath, pos3 + 1)  
GetPath = stPath  
End If  
End If  
End Function  
</code>
```

.