



Recuperare il percorso delle cartelle di sistema

http://www.vbsimple.net/activity/act_22.htm

Richiesta di: [Blisset](#) - 13 Maggio 2002

Difficoltà: 2 / 5

Come faccio da VB ad accedere alla cartella di sistema di Windows o alla cartella del menu Avvio veloce (in Win98 o XP differiscono)?

In tutti i sistemi operativi Windows sono presenti alcune variabili di ambiente, recuperabili tramite l'istruzione *Environ*. Tuttavia queste indicano soltanto alcune cartelle di sistema e cambiano da versione a versione.

Esistono invece alcune funzioni [API](#) dedicate al recupero delle cartelle di sistema. La principale di queste funzioni è **SHGetSpecialFolderPath** ma, poiché su alcuni sistemi operativi non riesce a recuperare alcune cartelle, abbiamo preferito farla appoggiare da altre tre funzioni: **GetWindowsDirectory**, **GetSystemDirectory**, **GetTempPath** per recuperare rispettivamente la cartella in cui è installato Windows (in genere Windows o WinNT), la cartella di sistema (System o System32) e la cartella per i files temporanei.

Il nostro progetto di esempio si compone di un modulo standard e di un form. Iniziamo con lo studio del codice presente all'interno del modulo:


```
1. Option Explicit
2.
3. Private Declare Function SHGetSpecialFolderPath Lib "shell32.dll" Alias
   "SHGetSpecialFolderPathA" (ByVal hwnd As Long, ByVal pszPath As String, ByVal csidl
   As Long, ByVal fCreate As Long) As Long
4. Private Declare Function GetWindowsDirectory Lib "kernel32" Alias
   "GetWindowsDirectoryA" (ByVal lpBuffer As String, ByVal nSize As Long) As Long
5. Private Declare Function GetSystemDirectory Lib "kernel32" Alias
   "GetSystemDirectoryA" (ByVal lpBuffer As String, ByVal nSize As Long) As Long
6. Private Declare Function GetTempPath Lib "kernel32" Alias "GetTempPathA" (ByVal
   nBufferLength As Long, ByVal lpBuffer As String) As Long
7.
8. Private Const MAX_PATH = 260
9.
```

La prima funzione **SHGetSpecialFolderPath** richiede la specifica di un **IDL** (IDentifier List) ovvero un identificatore della cartella di cui recuperare il percorso. I vari IDL delle cartelle di sistema sono riportati in seguito.

Le altre 3 funzioni API richiedono soltanto il passaggio del [buffer](#) dati [allocato](#) e la sua ampiezza. La costante alla riga 8 indicherà la lunghezza massima in caratteri di un percorso.

```
10. Public Enum CSIDL_Cartelle
11.     [_CSIDL_PRIMO] = &H0
12.     CSIDL_DESKTOP = &H0
```

```
13. CSIDL_INTERNET = &H1
14. CSIDL_PROGRAMS = &H2
15. ...
```

Alla riga 10 è definita un'[enumerazione](#)  di nome **CSIDL_Cartelle** e conterrà al suo interno tutti gli IDL standard più alcuni nuovi aggiunti. Il primo valore di quest'enumerazione sarà **[_CSIDL_PRIMO]** ed indicherà il primo valore dell'enumerazione.

```
59. [_CSIDL_ULTIMO] = &H31
60. End Enum
61.
```

Alla stessa maniera l'ultimo valore della lunga enumerazione sarà **[_CSIDL_ULTIMO]** ed indicherà l'ultimo valore dell'elenco.

Questi due valori speciali saranno utilizzati per limitare l'enumerazione ed effettuare un ciclo di tutti i suoi elementi. Il nome di questi due valori che è stato dato è circondato da parentesi quadre perché il loro primo carattere è un trattino di sottolineatura.

L'aver dato un nome con un carattere di sottolineature all'inizio determinerà che il valore venga nascosto all'utente che usufruisce dell'enumerazione. In particolare il sistema [Intellisense](#) non ne rileverà la presenza.

```
62. Public Function TrovaCartella(ByVal IDLCartella As CSIDL_Cartelle) As String
63.     TrovaCartella = String$(MAX_PATH, 0)
64.     Select Case IDLCartella
65.         Case CSIDL_TEMP
66.             Call GetTempPath(MAX_PATH - 1, TrovaCartella)
67.         Case CSIDL_WINDOWS
68.             Call GetWindowsDirectory(TrovaCartella, MAX_PATH - 1)
69.         Case CSIDL_SYSTEM
70.             Call GetSystemDirectory(TrovaCartella, MAX_PATH - 1)
71.         Case Else
72.             Call SHGetSpecialFolderPath(ByVal 0&, TrovaCartella, IDLCartella, ByVal 0&)
73.     End Select
74.     TrovaCartella = Left$(TrovaCartella, InStr(1, TrovaCartella, Chr$(0)) - 1)
75.     If Len(TrovaCartella) > 0 And Right$(TrovaCartella, 1) <> "\" Then TrovaCartella
       = TrovaCartella & "\"
76. End Function
```


Alla riga 62 è definita la funzione principale di questo modulo che prende il nome di **TrovaCartella**; essa richiede soltanto il passaggio di un valore dell'enumerazione appena vista e restituisce il percorso della cartella recuperata.

Alla riga 63 è allocato il buffer utilizzato dalle varie funzioni API. In base all'argomento **IDLCartella** fornito sarà richiamata una delle quattro funzioni API: *GetTempPath* per **CSIDL_TEMP**, *GetWindowsDirectory* per **CSIDL_WINDOWS**, *GetSystemDirectory* per **CSIDL_SYSTEM** ed infine *SHGetSpecialFolderPath* per tutti gli altri valori.

Al termine del recupero del percorso, alla riga 74, dovrà essere ripulito il buffer dai caratteri aggiuntivi. Sarà quindi estratto tutto ciò che si trova prima del primo carattere [NULL](#). Solo in seguito sarà aggiunta la barra rovesciata al percorso estratto nel caso essa non dovesse essere presente. Questo al fine di assicurare che tutti i percorsi terminino con un backslash e non solo quelli che puntano alla radice del disco.

Possiamo quindi passare al semplice form dimostrativo del funzionamento: si compone di soli quattro controlli: una *Label* di nome

lblPercorso e la corrispondente *TextBox*  di

nome **txtPercorso**; l'altra *Label* **lblCSIDL** e la corrispondente *HScrollBar*  di nome **hsbCSIDL**.



Ad ogni cambio di valore del controllo *HScrollBar* corrisponderà un IDL e la corrispondente cartella estratta e riportata nella *TextBox*. Il codice si compone di due sole routine:

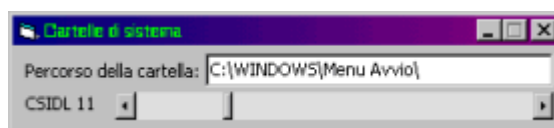
```
1. Option Explicit
2.
3. Private Sub Form_Load()
4.     hsbCSIDL.Min = [_CSIDL_PRIMO]
5.     hsbCSIDL.Max = [_CSIDL_ULTIMO]
6. End Sub
7.
```

All'avvio del form sarà inizializzata la *HScrollBar* con i valori limite e nascosti dell'enumerazione **CSIDL_Cartelle**.

```
8. Private Sub hsbCSIDL_Change()
9.     lblCSIDL.Caption = "CSIDL " & CStr(hsbCSIDL.Value)
10.    txtPercorso.Text = TrovaCartella(hsbCSIDL.Value)
11. End Sub
```

Così allo cambiamento del valore della barra di scorrimento sarà aggiornata la *Label* **lblCSIDL** con il numero dell'IDL indicato (riga 9). Alla riga 10 invece sarà recuperata la cartella di sistema tramite la funzione **TrovaCartella** e l'IDL indicato dalla barra di scorrimento.

Il progetto è davvero semplicissimo: basterà spostare la barra per scorrere tutte le cartelle di sistema.



La soluzione proposta è davvero molto semplice e consente il recupero di molte cartelle di sistema. Tuttavia il funzionamento può cambiare da un sistema operativo all'altro.

[Fibia FBI](#)
15 Maggio 2002



[Torna all'introduzione delle Richieste dei lettori](#)