

Save a screen snapshot to a bitmap

'If is often useful (eg. when debugging applications) to be able to save a snapshot of the screen to a file. The following code (including a demonstration routine) will save a screenshot to a bitmap file:

Option Explicit

```
Private Declare Function GetSystemMetrics Lib "user32" (ByVal nIndex As Long) As Long
Private Declare Function GetDesktopWindow Lib "user32" () As Long
Private Declare Function CreateCompatibleDC Lib "gdi32" (ByVal hDC As Long) As Long
Private Declare Function CreateCompatibleBitmap Lib "gdi32" (ByVal hDC As Long, ByVal nWidth As Long, ByVal nHeight As Long) As Long
Private Declare Function SelectObject Lib "gdi32" (ByVal hDC As Long, ByVal hObject As Long) As Long
Private Declare Function BitBlt Lib "gdi32" (ByVal hDCDest As Long, ByVal XDest As Long, ByVal YDest As Long, ByVal nWidth As Long, ByVal nHeight As Long, ByVal hDCSrc As Long, ByVal XSrc As Long, ByVal YSrc As Long, ByVal dwRop As Long) As Long
Private Declare Function DeleteDC Lib "gdi32" (ByVal hDC As Long) As Long
Private Declare Function GetWindowDC Lib "user32" (ByVal Hwnd As Long) As Long
Private Declare Function ReleaseDC Lib "user32" (ByVal Hwnd As Long, ByVal hDC As Long) As Long
Private Declare Function OleCreatePictureIndirect Lib "olepro32.dll" (PicDesc As tPicBmp, RefIID As tGUID, ByVal fPictureOwnsHandle As Long, IPic As IPicture) As Long
```

```
Private Type tPicBmp
    ISize As Long
    IType As Long
    IhBmp As Long
    IhPal As Long
    IReserved As Long
End Type
```

```
Private Type tGUID
    IData1 As Long
    IData2 As Integer
    IData3 As Integer
    abData4(7) As Byte
End Type
```

'Purpose : Captures a screen shot
'Inputs : sSaveToPath The path to save the image to
'Outputs : Returns a True if successful

```

Public Function ScreenSnapshot(sSaveToPath As String) As Boolean
    Dim lImageWidth As Long, lImageHeight As Long
    Dim lhDCMemory As Long, lhWndSrc As Long
    Dim lhDCSrc As Long, lhwndBmp As Long
    Dim lhwndBmpPrev As Long, lRetVal As Long
    Dim tScreenShot As tPicBmp
    Dim IPic As IPicture 'OR USE IPictureDisp is this doesn't compile (depending on
which VB your using)
    Dim tIDispatch As tGUID
    Const SM_CXSCREEN = 0, SM_CYSCREEN = 1

    On Error GoTo ErrFailed
    lImageWidth = GetSystemMetrics(SM_CXSCREEN)
    lImageHeight = GetSystemMetrics(SM_CYSCREEN)

    'Get a handle to the desktop window and get the proper device context
    lhWndSrc = GetDesktopWindow()
    lhDCSrc = GetWindowDC(lhWndSrc)

    'Create a memory device context for the copy process
    lhDCMemory = CreateCompatibleDC(lhDCSrc)

    'Create a bitmap and place it in the memory DC
    lhwndBmp = CreateCompatibleBitmap(lhDCSrc, lImageWidth, lImageHeight)
    lhwndBmpPrev = SelectObject(lhDCMemory, lhwndBmp)

    'Copy the screen image to the memory
    Call BitBlt(lhDCMemory, 0, 0, lImageWidth, lImageHeight, lhDCSrc, 0, 0, 13369376)

    'Remove the new copy of the the on-screen image
    lhwndBmp = SelectObject(lhDCMemory, lhwndBmpPrev)

    'Release the DC resources
    Call DeleteDC(lhDCMemory)
    Call ReleaseDC(lhWndSrc, lhDCSrc)

    'Populate OLE IDispatch Interface ID
    With tIDispatch
        .IData1 = &H20400
        .abData4(0) = &HC0
        .abData4(7) = &H46
    End With

    With tScreenShot
        .lSize = Len(tScreenShot) 'Length of structure
        .lType = 1 'Type of Picture (bitmap vbPicTypeBitmap)
        .lhBmp = lhwndBmp 'Handle to bitmap
        .lhPal = 0& 'Handle to palette (may be null)
    End With

```

```
'Create OLE Picture object
Call OleCreatePictureIndirect(tScreenShot, tIDispatch, 1, IPic)
```

```
'Return the new Picture object
SavePicture(IPic, sSaveToPath)
ScreenSnapshot = True
Exit Function
```

ErrFailed:

```
'Error occurred
ScreenSnapshot = False
End Function
```

```
'Returns the handle of the desktop
Function GetDesktopHwnd() As Long
    Static sIGetDesktopHwnd As Long 'Cache value for speed
    If sIGetDesktopHwnd = 0 Then
        sIGetDesktopHwnd = GetDesktopHwnd
    End If
    GetDesktopHwnd = sIGetDesktopHwnd
End Function
```

```
'Demonstration routine
Sub Test()
    If ScreenSnapshot("C:\Test.bmp") Then
        MsgBox("Screen Shot saved!", vbInformation)
    End If
End Sub
```