

 Universidad de La Laguna Temas	<h1>Las imagenes y el Access II</h1> <p>© Jose Ignacio González Gómez</p> <p>Revisado : 1º Trimestre 2002</p>
--	---

Nota aclaratoria.....	1
Procedimientos.....	1
Desde la Base de Datos.....	2
Desde Visual Basic.....	2
Llevar la Imagen de la Base de Datos a un Archivo.....	3
Imágenes Vinculadas.....	3
Ejemplo de Tomás Boixet.....	3
Lo que se dice en las news.....	3

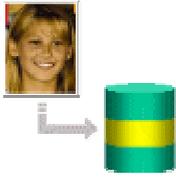
Nota aclaratoria.

La propuesta que presentamos a continuación respecto al access y el tratamiento de las imagenes la hemos recopilado de la siguiente dirección:

<http://personales.com/espana/santander/public/ComoGuardarImagenesEnUnaBDJet.htm>
 bajo el título: Imágenes en una Base de Datos. Que se debe hacer en Visual Basic para una interfaz que maneja imágenes en una Base de Datos.

A sus autores agradecer su colaboración.

Procedimientos.

	<p>Existen ciertos detalles para implementar una interfaz Visual Basic que permita administrar imágenes en una Base de datos que no están documentados explícitamente, y aunque sencillos, pueden presentarle dificultades a un programador medio. Este artículo describe la forma implementar un campo de imágenes, con un ejemplo sencillo.</p>
---	---

El caso más común, es que Usted desea colocar una foto de un ser querido o Empleado junto a sus datos, y al navegar la base de datos desea ver las fotografías. Es claro, una imagen vale más de 100 palabras. Otros casos comprenden catálogos de productos, un coleccionista organizado, etc. El ejemplo que presentare, es una colección de iconos listos para usar, para el ejemplo use Visual Basic 5.0 contra una base de datos Jet 3.5 (Access97©)

Básicamente requiere de un procedimiento rutinario en la base de datos, como es crear un campo, y de unas cuantas líneas en Visual Basic.

Desde la Base de Datos

Requiere crear o usar un campo Objeto OLE. Los campos del tipo Objeto OLE se utilizan para almacenar datos, como documentos de Microsoft Word o Microsoft Excel, imágenes, sonido y en general, otros tipos de datos binarios creados en otros programas. En Visual Basic el tipo corresponde a la constante dbLongBinary. La estructura de la tabla del ejemplo es de la siguiente manera:

Nombre de Campo	Tipo de Dato	Descripción
ID Icono	Autonumérico	Establecer como Clave Principal.
Nombre	Texto (32)	Nombre del Icono (sin extensión)
Icono	Objeto OLE	Contiene imágenes de iconos
Descripción	Texto (254)	Alguna descripción

Existen dos maneras de almacenar datos en un campo Objeto OLE, Insertar y Vincular. Brevemente, cuando se inserta una imagen forma parte del archivo de base de datos, a diferencia de Vinculados, en donde solo se guarda una referencia al objeto. Si piensa utilizar la misma imagen en varios formularios e informes, es posible que desee vincular la imagen en lugar de insertarla. Este artículo trata sobre imágenes insertadas (más adelante haré referencia a imágenes vinculadas).

Desde Visual Basic

El propósito es visualizar rápidamente las imágenes y agregar datos de la manera más sencilla posible. Para visualizar la imagen yo utilizo un PictureBox vinculado a un control Data. Simplemente la Propiedad DataSource es el Control Data interfaz DAO de la tabla o consulta de la Bases de Datos, y la propiedad DataField es el nombre del campo Objeto OLE.

Para almacenar las imágenes de manera muy sencilla, yo utilizo las capacidades Arrastrar y Soltar (Drag and Drop) de Windows95. Es decir, desde Microsoft Explorer, se arrastra el nombre del archivo y se suelta sobre el PictureBox vinculado al Campo. Cuando se archive el registro (deplazándose hacia otro registro o usando un comando Actualizar), la imagen quedará guardada en la Base de Datos. Para implementar la capacidad de Arrastrar y Soltar, se necesitan dos pasos sencillos: Primero, fijar la propiedad OLEDropMode = 1 (Manual) del PictureBox. Segundo estas líneas de código en el evento OLEDragDrop:

```
Private Sub Picture1_OLEDragDrop(Data As DataObject, _
    Effect As Long, Button As Integer, _
    Shift As Integer, x As Single, Y As Single _
)
    '//DataObject contiene datos de tipo vbCFFiles
    On Error Resume Next
    Set Picture1.Picture = LoadPicture(Data.Files(1))
End Sub
```

Listo, esta es la manera de implementar una interfaz a una base de datos con imágenes. Como una recomendación, trate de usar gráficos GIF o JPG en vez de BMP ya que reducirá dramáticamente el espacio empleado en la base de datos sin perder calidad. Para fotografías de alta calidad el formato JPG es el ideal.

Llevar la Imagen de la Base de Datos a un Archivo

Si desea llevar una imagen almacenada en la base de datos a un archivo, simplemente usa la instrucción SavePicture de Visual Basic, aplicada al PictureBox asociado a la imagen: SavePicture miPictureBox.Picture, NombreDeArchivo

Recuerde que si un gráfico se ha cargado en la propiedad Picture de un objeto desde un archivo, ya sea en tiempo de diseño o en tiempo de ejecución, y es un mapa de bits, un icono, un metarchivo o un metarchivo mejorado, se guarda con el mismo formato que el archivo original. Si se trata de un archivo GIF o JPEG, SavePicture lo guardará como un archivo de mapa de bits. Los gráficos de la propiedad Image siempre se guardan como archivos de mapas de bits (.BMP), cualquiera que sea su formato original. Use SavePicture miPicture.Image, NombreDeArchivo para archivar gráficos creados con Métodos Gráficos. De otra parte, la base de datos almacena en formato original del archivo de imagen.

Imágenes Vinculadas

Se pueden implementar imágenes vinculadas o incrustadas usando el control OLE enlazado al campo. Si acaso, esto representa mayor dificultad y sugiere estudiar la teoría OLE para lograr una interfaz OLE transparente al usuario. Sugiero usar el asistente para formularios para implementar el control OLE enlazado al campo Objeto OLE, ya que no es tan sencillo como asignar las propiedades DataSource y DataField. Personalmente nunca uso imágenes vinculadas a una base de datos, ya que si el propósito es disponer de imágenes particulares, prefiero usar algunas líneas de código para manejar directamente el archivo.

Ejemplo de Tomás Boixet

De todas formas hemos encontrado en la Web de Tomás Boixet (autentico monstruo) una aplicación en access para el tratamiento de las imágenes.zip (24 KB, v97 MDB) que nos permite visualizar imágenes (GIF, BMP, PCX) en un formulario sin ocupar espacio en la base de datos. Incluye la API GetOpenFileName, que abre la ventana de Windows para seleccionar un fichero.

Lo que se dice en las news

1.- Hola, grupo.

Tengo una base de datos de personas la cual incluye sus fotos. Para que la base no fuera inmensa, las imágenes de las fotos, en formato .jpg están vinculadas en un campo de una tabla como objeto OLE. Al ver la tabla, en el campo "foto" aparece la palabra "Paquete" y todo funciona de maravilla en los formularios, en los que aparecen los datos de la persona, y su foto. Sin embargo, no logro que un informe aparezca la foto, y en su lugar sale un icono de archivo .jpg y el nombre del archivo de la foto (Por ejemplo, "pepitoperez.jpg").

En los formularios, donde si funciona, la foto es un control Imagen que actualizo con el nombre del archivo que la contiene cada vez que ocurre el evento current del formulario (los nombres de los archivos de foto, como se ve, el el DNI de la persona (9807764.jpg, etc)):

```
Private Sub Form_Current()  
On Error GoTo ErrorFotoSocio  
FotoSocio.Picture = DNI.Value + ".jpg"  
Exit Sub  
ErrorFotoSocio:  
FotoSocio.Picture = "nofoto.bmp"  
Resume Next  
End Sub
```

Esto carga la imagen en el control FotoSocio, y si no existe la foto del socio, carga una imagen prediseñada "nofoto.bmp". Pero no soy capaz de hacer lo mismo en un informe, ya que no existe (o no conozco) un evento parecido al "current" de los formularios. El informe es un resumen de ciertos datos de cada persona, con su foto al lado. Bienvenidas sugerencias, y gracias anticipadas.

Quique.

Pues mira Quique, resulta que yo tengo exactamente lo mismo que tu comentas en una BD y todo funciona OK. En el informe tengo un cuadro de imagen y el código lo tengo puesto en el evento "Detalle - Print". Prueba a ver, a mi me va bien. Saludos Lancelot.

