

## **ERO Reader Sample Help**

**Sample Description:** [ERO Reader](#)

### **Points of Interest**

[Reading an Envelop Resource File](#)

[Writing the Displayed Bitmap to Disk](#)

[Adding a Bitmap to an ERO File](#)

[Changing a Bitmap Stored in an ERO File](#)

[Removing a Bitmap Stored in an ERO File](#)

[The ERO Reader as an Envelop Project](#)

### **Control**

Bitmap

BinaryFile

For Help on Help, Press F1

## **ERO Reader**

The ERO Reader is a stand-alone application which may be used for viewing the contents of Envelop resource object (.ero) files.

Envelop ERO files are used to store bitmap images whose **LoadType** properties are designed as "MemoryBased". Images that are used in forms that are to be a permanent part of the application and not subject to change, can be designated as "MemoryBased" and saved to a single (.ero) file when the application is saved in a text (.eto) format. If the application is saved to a binary format (.ebo), the images are saved inside the binary file.

There is an advantage of saving several bitmaps to a resource (.ero) file. Instead of several Bitmap objects reading their source (.bmp) files at initialization, they may read a single (.ero) file. As the application is deployed to other systems, only the single (.ero) file needs to be copied with the Envelop (.eto) file.

### **Reading an Envelop Resource File**

Reading an Envelop resource file is done with a BinaryFile object named **ResIdReader**. The path and filename of the (.ero) file is passed to ResIdReader's Reset method, which in turn sets the FileName property of the BinaryFile object.

A method **NextResId** is used to read the "resource id" numbers contained in the (.ero) file and append them to a list in the ERO Reader form.

## Writing the Displayed Bitmap to Disk

When a particular resource is being viewed in the ERO Reader form, the bitmap is displayed in an Image control named **imgPicture**. A SaveAs menu entry invokes the following method on the ERO Reader form.

```
Sub SaveAsBitmapFile_Click()  
    Dim saveasDlg As New SaveAsDialog  
    saveasDlg.DefaultExtension = "bmp"  
    saveasDlg.Filter = "Bitmap files (*.bmp)|*.bmp|All files (*.*)|*.*|"  
    saveasDlg.NoChangeDir = True  
    saveasDlg.Title = "Select Bitmap File"  
    If saveasDlg.Execute = IDOK Then  
        imgPicture.Picture.SaveAs(saveasDlg.FileName)  
    End If  
End Sub
```

A SaveAs dialog is used to provide a means for specifying a filename for which to save the bitmap file. If a filename is specified in the SaveAs dialog and the user clicks the OK button, the **SaveAs** method on the bitmap is executed and given the name of the file to write. Notice that `imgPicture.Picture` is an *object reference* to an embedded `bitmap1` object.

### **Adding a Bitmap to an ERO File**

An Envelop resource (.ero) file is automatically created when an Envelop text (.eto) file is created and there is a bitmap whose **LoadType** is set to "MemoryBased".

When you add an Image control to a form and double click on the control, an Open dialog is posted for you to specify a bitmap file. In addition, a Bitmap object is embedded inside the Image control and the Image control's **Picture** object reference is set to the embedded bitmap. By default, Bitmap object's LoadType property is set to "FileBased". When the FileName property is set, the bitmap file is loaded, the LoadType is automatically set to "MemoryBased".

At this point, if you were to save the Envelop project or module to disk and specify the text (.eto) format, a resource (.ero) file would be automatically created.

### **Changing a Bitmap Stored in an ERO File**

When an Envelop project or module is loaded from text (.eto) format and a resource (.ero) file exists, it is loaded automatically. If you were to look at the Bitmap object's properties you would see that the LoadType is set to "MemoryBased". You will also see a **ResId** property which contains an index value into the resource file.

To change the bitmap, simply set the LoadType property to "FileBased" and enter a bitmap (.bmp) file in the FileName property. The previous bitmap will automatically be destroyed and the new bitmap will automatically be loaded. Once the new bitmap is loaded, the LoadType property will automatically be set back to "MemoryBased". At this point, if you save the project or module to text (.eto) format, the bitmap will be written in the new resource (.ero) file.

### Removing a Bitmap Stored in an ERO File

When an Envelop project or module is loaded from text (.eto) format and a resource (.ero) file exists, it is loaded automatically. If you were to look at the Bitmap object's properties you would see that the LoadType is set to "MemoryBased". You will also see a **ResId** property which contains an index value into the resource file.

To remove the bitmap, simply set the LoadType property to "FileBased" and clear the FileName property. The previous bitmap will automatically be destroyed. At this point, if you save the project or module to text (.eto) format, the no bitmap will be written into the new resource (.ero) file, however if there are any other bitmap's in your application, they will be included in the (.ero) file.

If you wish to have an Image control always reference a FileBased bitmap file, simply set the LoadType property to "FileBased" after you load the bitmap file by setting the FileName property.

### **The ERO Reader as an Envelop Project**

The ERO Reader sample was developed both to server as an Envelop Sample as well as a stand-alone executable utility. In the sample's directory, an Envelop project (.epj) file exists as well as an Envelop executable (.exe) file.

If you double-click on the readero.exe from the File Manager, the stand-alone ERO Reader will run on the screen. The name of the application project is **ERORreader**.

The project may be loaded into Envelop by using the File/Load Project menu. In addition, the sample module may be loaded into Envelop using the File/Load Module menu.



