

eXeScope

[What is eXeScope ?](#)

[Functions of eXeScope](#)

[File Types Supported](#)

[Various Usage of eXeScope](#)

[Operation](#)

[Menu and Button](#)

[Edit Dialog Box](#)

[Edit Menu](#)

[Edit Delphi Form](#)

[Display Type Library](#)

[Find Strings](#)

[Batch Mode Editing using Log file](#)

[Create .rc file](#)

[Environment, Install](#)

[Limitations, Warning](#)

[Registration](#)

[About Copyright](#)

What is eXeScope ?

Do you want to customize an application ? For example,
to change font,
to change menu,
to change an arrangement of dialog,
etc.,

but do you think that it is impossible because you have not source files ?

eXeScope can analyze, display various information, and rewrite resources of executable files, that is, EXE, DLL, OCX, etc. without source files.

1. Analyze and display executable files (EXE, DLL, etc.)

1) Display header information

2) Display imported functions

eXeScope displays what functions in what DLLs an executable file imports. It is useful to find unused DLLs.

3) Display various resources

2. Extract resources and save them to files

eXeScope can extract Icon, Cursor, Bitmap, WAVE, MIDI, AVI and Delphi form. Other resources can be extracted as binary or text data.

3. Edit resources

eXeScope can edit windows standard resource (Icon, Cursor, Bitmap, Dialog, Menu, Accelerator, String, MessageTable) and other resources (Delphi form, WAVE, MIDI, AVI, JPEG).

Important Warning

Before editing resources, you must backup the original file. There is a possibility that edited executable file does not work, or work abnormally. Edited executable file should be tested carefully.

Author does not guarantee any disruption caused by edited executable file.

4. Confirmation of rewrite effect

In the case of Dialog resource, you can display the rewritten dialog form, and confirm it.

5. Recording Log and Batch mode editing using Log

The record of editing resource can be logged to a log file. Using the log file, same editing can be re-play. This feature can be usefull when you do same editing for the new version of target file.

6. Create a .rc file

eXeScope can create a resource source file (.rc) file from an EXE file reversely.

Functions of eXeScope

Below table lists functions which eXeScope supports.

OK:Enable, -: Disable

(*1) Can output as a text file.

(*2) Can display and rewrite as a binary dump format.

(*3) Can import and export as a binary or text file.

(*4) Can display the rewritten contents.

(*5) Required SHDocvw.dll which is included in Internet Explorer

| Item | Display | Import & Export | Edit | Log | Create .rc |
|-------------------------|---------|-----------------|--------|-----|------------|
| 1. Header | OK | (*1) | - | - | - |
| 2. <u>Import</u> | OK | (*1) | - | - | - |
| 3. <u>Export</u> | OK | (*1) | - | - | - |
| 4. Resource | | | | | |
| 4.1 <u>Cursor</u> | OK | OK | OK | OK | - |
| 4.2 <u>Bitmap</u> | OK | OK | OK | OK | - |
| 4.3 <u>Icon</u> | OK | OK | OK | OK | - |
| 4.4 <u>Menu</u> | OK | (*3) | OK | OK | OK |
| 4.5 <u>Dialog</u> | OK | (*3) | OK(*4) | OK | OK |
| 4.6 <u>String</u> | OK | (*3) | OK | OK | OK |
| 4.7 <u>MessageTable</u> | OK | (*3) | OK | OK | - |
| 4.8 <u>Accelerator</u> | OK | (*3) | OK | OK | - |
| 4.9 <u>Version</u> | OK | (*3) | OK | OK | - |
| 4.10 <u>Delphi Form</u> | OK | OK | OK | - | - |
| 4.11 <u>WAVE</u> | OK | OK | OK | - | - |
| 4.12 <u>MIDI</u> | OK | OK | OK | - | - |
| 4.13 <u>AVI</u> | OK | OK | OK | - | - |
| 4.14 <u>GIF</u> | (*5) | (*5) | (*5) | - | - |
| 4.15 <u>HTML</u> | (*5) | (*5) | (*5) | - | - |
| 4.16 <u>JPEG</u> | OK | OK | OK | - | - |
| 4.17 <u>TYPELIB</u> | OK | OK | - | - | - |
| 4.18 <u>DialogData</u> | OK | OK | OK | OK | - |
| 4.19 <u>Toolbar</u> | OK | - | - | - | - |
| 4.20 Other resources | (*2) | (*3) | (*2) | - | - |

Import

Functions which refer external DLLs.

Export

Functions which can be referred from external. DLLs have this information.

Icon
Image of Icon.

Cursor
Image of Cursor.

Bitmap
Image of Bitmap.

Version
Version Information.

Delphi Form

In the case of application which is developed by Delphi, almost resources are defined with 'Form', and 'Form' data are stored in 'RCData' of resource.

Dialog

Defined dialog box's position, size, font, and contained parts, such as buttons, labels etc.

Menu
Defined menus.

Accelerator
Defined accelerator keys.

String

Including strings displayed in message boxes, etc.

MessageTable
Defined error messages, etc.

WAVE
Recorded audio data.

MIDI (Musical Instrument Digital Interface)
Recorded MIDI audio data.

AVI (Audio Video Interleaved File)
Including AVI data.

GIF (Graphic Interface Format)
Including GIF image.

HTML (Hyper Text Markup Language)
Including HTML format data.

JPEG (Joint Photographics Experts Group Bitmap)
Including JPEG image.

TYPELIB

Including Type Library, interface information which is used with COM/DCOM. ActiveX and OCX include this information.

DialogData

Including pairs of combobox's ID and text in that dropdown.

Toolbar

Including IDs of toolbar buttons. Corresponding images are included in a Bitmap resource.

File Types Supported

eXeScope supports 32 bits PE format and 16 bits NE format files.
Typically, below files:

| Extension | File Type |
|-----------|----------------------|
| ===== | ===== |
| EXE | Executable file |
| DLL | Dynamic link library |
| OCX,OCA | OCX/ActiveX |
| VBX | VBX |
| SCR | Screen saver |
| CPL | Control panel |
| FON | Font file |
| DPL | Delphi 3 package |
| BPL | Delphi 4 package |

Various Usage of eXeScope

Change font typeface and size

Change size and position of forms or dialog boxes

Change arrangement of parts (buttons etc.) in a form

Change text in message boxes

Change text of menus

Change accelerator keys

Extract icon, cursor, bitmap, sound, or movies

Replace icon, cursor, bitmap, sound, or movies

Examine DLLs

Examine exported functions

Change font typeface and size

Edit Dialog resource. When a Delphi application, edit forms in RCData resource.

Change size and position of forms or dialog boxes

Edit forms in RCData resource if it is a Delphi application, otherwise edit dialog resource.

Change arrangement of parts (buttons etc.) in a form

Edit forms in RCData resource if it is a Delphi application, otherwise edit dialog resource.

Change text in message boxes

Edit string or message table resource.

Change text of menus

Edit forms in RCData resource if it is a Delphi application, otherwise edit menu resource.

Change accelerator keys

Edit accelerator.

Extract icon, cursor, bitmap, sound, or movies

Open Icon, Cursor, Bitmap, WAVE, MIDI, or AVI resource respectively, and select menu [File - Export].

Replace icon, cursor, bitmap, sound, or movies

Open Icon, Cursor, Bitmap, WAVE, MIDI, or AVI resource respectively, and select menu [File - Import]. In the case of Icon or Cursor, the image size must be same.

Examine used DLLs

Examine Import resource.

Examine exported functions

Examine Export resource. If there is TYPELIB resource, then examine TYPELIB also.

Operation (Menu and Button)

| | |
|--|----------------------------------|
|  Button | [File - Open]Menu |
|  Button | [File - Import]Menu |
|  Button | [File - Export]Menu |
| | [File - Output source (.rc)]Menu |
|  Button | [Edit - Edit]Menu |
| | [Edit - Insert]Menu |
| | [Edit - Delete]Menu |
| | [Edit - Copy]Menu |
| | [Edit - Save Update]Menu |
| | [Edit - Undo]Menu |
|  Button | [Edit - Log]Menu |
| | [Search - Find]Menu |
| | [Search - Find Next]Menu |
|  Button | [View - Select Font]Menu |
|  Button | [View - Binary]Menu |
|  Button | [View - Dialog Editor]Menu |
|  Button | [View - Test Display]Menu |
| | [View - Unicode/SJIS]Menu |

In the case of Menu, String, Accelerator, MessageTable or Version resource, a popup menu will be appear when the right button was clicked.

When clicked a header of each column in Import or Export resource display, then the list will be sorted with that column.

| Ordinal | Address | Name |
|---------|---------|------|
|---------|---------|------|

Open a EXE or DLL file

Note) A file can be opened to drag&drop it from Explorer to eXeScope's main form or shortcut icon on the desktop.

Replace current Icon or Bitmap etc. with an external file.

Save current Icon or Bitmap etc. to an external file.

Create a resource source file (.rc) from an EXE file reversely. Only Menu, Dialog, and String resources are supported.

Save changes of each resources to the EXE, DLL, etc. file.
A confirmation dialog will appear before actual execution. If checkbox [Permit to change the file size] is not checked, saving which changes the file size will be rejected.

Cancel the changes. This button or menu is effective before  button has been pushed.

When a Menu or Version resource is selected:

 Edit that resource item.

When a lowest level item of TreeView is selected:

 Change Language ID of that resource.

When a lowest level item of TreeView is selected, copy that resource with other language ID.

Insert a new line after the currently selected Menu or Version resource item.

Delete the currently selected Menu or Version resource item.

Search a text. This function can be used in Delphi Form, TYPELIB, or binary display of each resource.

Search a next text from the current position.

Change a font.

If pressed with CTRL or SHIFT key, then only a focused control will be changed.

Select character code (Unicode or Shift-JIS) for the status line (bottom of form).
In the case of binary dump, 32 characters from the current caret position are displayed in the status line.

Change to binary dump display. When pushed again, then return to normal display.

Display a visual dialog editor. Using the dialog editor, size and position of buttons etc. can be edited with mouse operation. When pushed again, the display will be closed.

Display the actual image of a dialog resource. For this function, it is possible to confirm the effect of change. When pushed again, the display will be closed.

(Limitation)

- * A control which is defined 'CONTROL' is displayed as a black frame.
- * Any 'ICON' is displayed as an Windows LOGO Icon.

When pressed or checked, changes to resources will be logged to a log file. The Log file name is same name of target executable file except extension '.Log'.

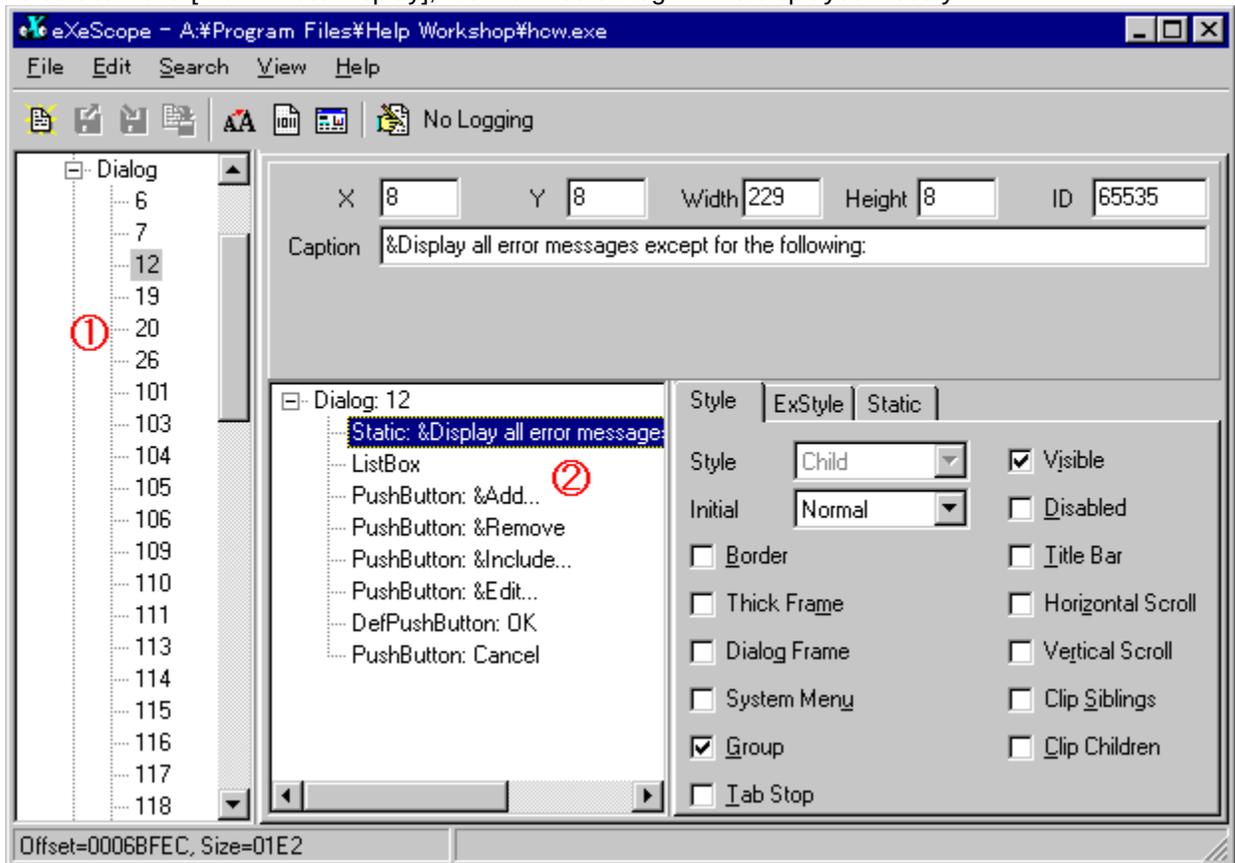
Edit a Dialog box

- 1) Open an EXE file.
- 2) Select 'Dialog' in a tree view (1) of left of display, then a list of dialog boxes will be displayed.
- 3) Select any dialog box in the list (1), then a list of controls which are included in it will be displayed at tree view (2) and an actual dialog image.
Edit the dialog with below 4) or 5).
- 4) Move and resize any controls in the actual dialog image.
- 5) Select any control on (2), then detail information, such as Caption, Font, Style, etc. will be displayed and can be edited.

Note:

- * If 'Dialog' resource is not found, then that executable file may be a Delphi application. Refer to [Edit Delphi form](#).
- * If a change is not proper, then there is a case that application does not work correctly. Please take care.
- * Represent control characters (TAB, NewLine, etc.) in text (Caption, etc.) as '\$' and two digits hex number. '\$' itself is represented as '\$\$'.
Example: TAB --> \$09, NewLine --> \$0A

To select menu [View - Test Display], the modified dialog will be displayed visually.



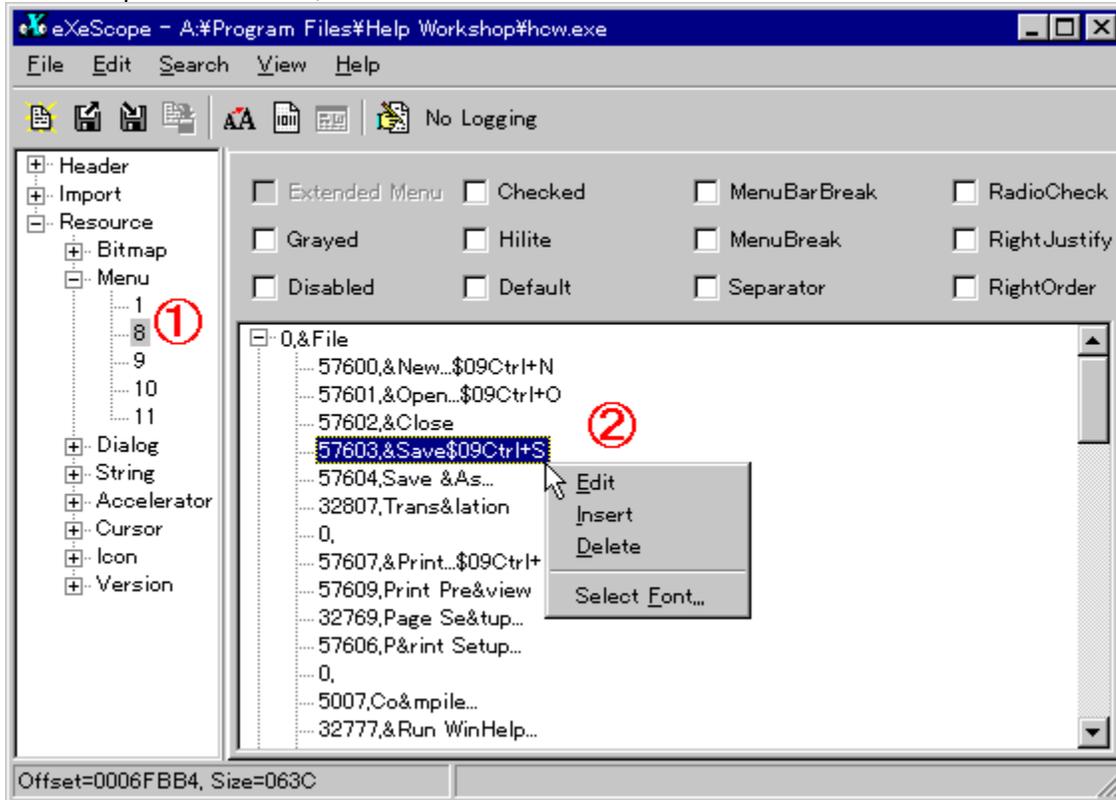
Edit a Menu

- 1) Open an EXE file.
- 2) Select 'Menu' in a tree view (1) of left of display, then a list of Menus will be displayed.
- 3) Select any menu in the list (1), then the content of that menu will be displayed at tree view (2).
- 4) Click right button on a target item which you want to edit, then a popup menu will appear and Edit, Insert, or Delete can be selected. Or Menu item can be moved by Drag&Drop.

Note:

* Represent control characters (TAB, NewLine, etc.) as '\$' and two digits hex number.

Example: TAB --> \$09, NewLine --> \$0A



Edit a Delphi form

When opened RCDData resource, then a Delphi form will be displayed like below example. To edit each property, the form can be customized.

About the meaning of each property, please refer Delphi manuals.

When updating the form, grammatical syntax check is done, but whether the property value is proper or not is not checked. Because of this, there is a possibility that the modified executable file cannot work correctly. Please test carefully by yourself.

```
object FConf: TFCConf
  Left = 262
  Top = 192
  BorderIcons = [biSystemMenu]
  BorderStyle = bsSingle
  Caption = 'Confirmation'
  ClientHeight = 118
  ClientWidth = 252
  Font.Charset = SHIFTJIS_CHARSET
  Font.Color = clWindowText
  Font.Height = -12
  Font.Name = 'MS Sans Serif'
  Font.Style = []
  Position = poScreenCenter
  PixelsPerInch = 96
  TextHeight = 12
  object Label1: TLabel
    Left = 8
    Top = 24
    Width = 233
    Height = 12
    Caption = 'Save Update, OK ?'
  end
  object OKBtn: TButton
    Left = 48
    Top = 80
    Width = 75
    Height = 25
    Caption = 'OK'
    ModalResult = 1
    TabOrder = 0
  end
  object CancelBtn: TButton
    Left = 136
    Top = 80
    Width = 75
    Height = 25
    Caption = 'Cancel'
    ModalResult = 2
    TabOrder = 1
  end
end
end
```

View Type library

Select 'TYPELIB' resource, then information of type library will be displayed.

The format of property and function are as follows:

```
property-get property-name(parameter; parameter; ...): property-type; call-type;
property-put property-name(parameter; parameter; ...); call-type;
property-putref property-name(parameter; parameter; ...); call-type;
function function-name(array of parameters): return-value-type; call-type;
```

Parameters are option, and described like below. If there is 'out' at top, it means that value is set when returned. If there is '^', it means a pointer to that value.

```
out parameter-name: type
```

Refer: [Types and Calling convention](#)

Sample

ComctlLib; // Microsoft Windows Common Controls 5.0

Class DataObject;

GUID={2334D2B2-713E-11CF-8AE5-00AA00C00905};

```
function QueryInterface(riid:^UserDefined; out ppvObj:^^void); stdcall;
```

```
function AddRef: UI4; stdcall;
```

```
    □F
```

```
function GetData(sFormat:I2): variant; stdcall;
```

```
function GetFormat(sFormat:I2): bool; stdcall;
```

```
function SetData([vValue:variant; vFormat:variant]); stdcall;
```

```
property-get Files: ^UserDefined; stdcall;
```

Dispatch ITabStrip; // Microsoft TabStrip Control

GUID={E6E17E8A-DF38-11CF-8E74-00A0C90F26F8};

```
function QueryInterface(riid:^UserDefined; out ppvObj:^^void); stdcall;
```

```
function AddRef: UI4; stdcall;
```

```
    □F
```

```
property-get MousePointer: UserDefined; stdcall;
```

```
property-put MousePointer(UserDefined); stdcall;
```

```
function Refresh; stdcall;
```

```
function OLEDrag; stdcall;
```

```
function AboutBox; stdcall;
```

Types and Calling convention

Types

| | | | |
|----------------|--------------------------|-------------|-----------------------------|
| empty | No value | null | SQL style NULL |
| I2 | 2 bytes signed integer | I4 | 4 bytes signed integer |
| R4 | 4 bytes real | R8 | 8 bytes real |
| CY | Currency | date | date type |
| BSTR | VB type string | IDispatch | IDispatch |
| error | SCODE | bool | boolean |
| variant | VARIANT | IUnknown | IUnknown |
| decimal | 16 bytes fixed decimal | I1 | 1 byte signed integer |
| UI1 | 1 byte unsigned integer | UI2 | 2 bytes unsigned integer |
| UI4 | 4 bytes unsigned integer | I8 | 8 bytes signed integer |
| INT | Signed integer | | |
| UINT | Unsigned integer | | |
| void | C type void | HResult | Standard result value |
| PTR | Pointer | SafeArray | VARIANT array |
| CARRAY | C type array | UserDefined | User defined type |
| LPSTR | NULL terminated string | LPWSTR | NULL terminated wide string |
| FileTime | FILETIME | BLOB | Binary with length |
| Stream | Stream | storage | Storage |
| StreamedObject | Object including stream | | |
| StoredObject | Object including storage | | |
| BlobObject | Object including BLOB | | |

Calling convention (Default is stdcall)

fastcall
cdecl
pascal
macpascal
stdcall
fastcall
syscall
mpwcdecl
mpwpascal

Find strings

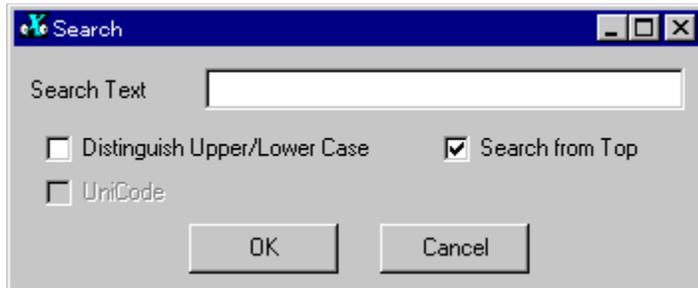
Find string function can be used in Delphi form, TYPELIB(Type Library), and binary dump display. But there are next limitations:

Binary Dump:

Uppercase/Lowercase is always distinguished. You can select ASCII or UNICODE.

Delphi form, TYPELIB:

You can select whether Uppercase/Lowercase is distinguished or not. You cannot select UNICODE.



Batch mode editing using Log file

Recording Log

To record changes to a log file, it is required to set 'Recording Log' mode by pushing  button. Changes will be logged to a file which name is same to the target executable file except extension '.Log'.

* There are resources which cannot be logged depending on it's type. Refer to [Functions of eXeScope](#).

Batch mode editing

Same editing can be done using a log file. For this, input next command line in MS-DOS window:

```
eXeBat Target-executable-filename Saved-log-filename
```

When corresponding item of Menu or Dialog in the target executable file, a warning message 'Warning, Skipped Line nn' will be displayed and that item will be neglected. For example, it is the case that the target file has new value already.

Note) Batch mode editing using log file can be done for 32 bit PE format only. 16 bit NE format cannot be edited by batch mode.

Format of Log file

Log file consists of multiple sections, and one section begins with

```
Resource,xxxxxx,yyy
```

and ends with

```
EndResource
```

The part of 'Resource,xxxxxx,yyy' is, for example, like the following:

```
Resource,Menu,120
```

This is a concatenation of each node name in left treeview of eXeScope.

Section

[Menu Section](#)

[Dialog Section](#)

[Accelerator Section](#)

[String Section](#)

[MessageTable Section](#)

[Cursor Section](#)

[Icon Section](#)

[Bitmap Section](#)

[Version Section](#)

[Comment](#)

If the top character is '#', then that line is comment.

Format of Menu resource:

```
Resource,Menu,resource_id
  menu_item
  menu_item
  -----
EndResource
```

Format of 'menu_item' is as follows. If it is a submenu, it will be indented by TAB.

MenuID,Original-Caption,New-Caption,mtFlag,mftOption,mfsOption

Unchanged part can be described by '*'. mftOption and mfsOption are used only by extended menu.

Example:

```
Resource,Menu,5
0,&file,&File,*,*,*
  57609,&update,&Update,*,*,*
  57610,"save &as","Save &As",*,*,*
  0,*,*,*,*
  57665,e&xit,E&xit,*,*,*
0,&edit,&Edit,*,*,*
  57643,&undo,&Undo,*,*,*
  ----
EndResource
```

Format of Dialog resource:

```
Resource,Dialog,resource_id
Dialog,Original-Caption,New-Caption,Font,FontSize,ID,X,Y,Width,Height,Style,ExtendedStyle
control_item
control_item
□c□c□c
EndResource
```

Format of 'control_item' is as follows:

```
Type,Original-Caption,New-Caption,ID,X,Y,Width,Height,Style,ExtendedStyle
Unchanged part can be described by '*'.
```

Example:

```
Resource,Dialog,160
Dialog,New,*,*,*,*,*,*,*
Static,"File Type :", "New File Type :",*,*,*,*,*
ListBox,*,*,*,*,*
DefPushButton,OK,*,*,*,*,*
PushButton,Cancel,*,*,*,*,*
EndResource
```

Format of Accelerator resource:

Example:

Resource,Accelerator,4

A,57642,VIRTKEY,NOINVERT,CONTROL

B,32799,VIRTKEY,NOINVERT,CONTROL

B,32799,VIRTKEY,NOINVERT,SHIFT,CONTROL

EndResource

Unchanged part cannot be described by '*'. All item must be described completely.

Format of String resource:

Example:

Resource,String,11

164,End of search

165,Cange font of selected text. \$0AFont

167,Change format

168,Standard

171,inch

173,cm

EndResource

Unchanged part cannot be described by '*'. All item must be described completely.

Format of MessageTable resource:

Example:

```
Resource,MessageTable,1  
1001,"file not found"  
1002,"unrecognized option"  
1003,"no input file specified"  
EndResource
```

Unchanged part cannot be described by '*'. All item must be described completely.

Format of Cursor resource:

Example:

```
Resource,Cursor,104,"2: 32*32"  
Import "A:\eXeScope\Hourglass.cur"  
EndResource
```

Format of Icon resource

Example:

```
Resource,Icon,128,"1: 32*32, 16 Color"  
Import "A:\eXeScope\Exescope.ico"  
EndResource
```

Format of Bitmap resource

Example:

```
Resource,Bitmap,132
```

```
Import "A:\eXeScope\Rec.bmp"
```

```
EndResource
```

Format of Version resource:

Example:

Resource,Version,1

FileVersion 4,0,950,0

ProductVersion 4,0,0,0

FileFlags 0

FileOS 4

FileType 1

FileSubType 0

StringFileInfo

041104B0

CompanyName,"Microsoft Corporation"

FileDescription,"WORDPAD MFC Application"

FileVersion,4.00.950

InternalName,wordpad

LegalCopyright,"Copyright (C) Microsoft Corp. 1995"

OriginalFilename,WORDPAD.EXE

ProductName,"Microsoft(R) Windows(TM) Operating System"

ProductVersion,4.0

VarFileInfo

Translation 11 04 B0 04

EndResource

Create .rc File

You can output Menu, Dialog, and String resource to a resource source file (*.rc). To do this, select menu [File - Output source(.rc)], then the next dialog will appear.

Check resource type which you want to output at "Resource" check box.

The created rc file can be compiled by VC++ or BRCC32.exe attached to Delphi or C++ Builder.

If you will compile by BRCC32.exe, please comment out the first line "#include <winres.h>".



Environment, Install

Environment: Windows95/98 and Windows NT 4.0

There is no specific procedure to install eXeScope. Create any directory, and extract files from archive file 'eXeSCnnn.zip' in that directory. There are next files:

eXeScope.exe ;Program file

eXeScEng.hlp ;Help file (English version)

eXeScEng.cnt ;Help content file (English version)

eXeScJpn.hlp ;Help file (Japanese version, **No need for English user**)

eXeScJpn.cnt ;Help content file (Japanese version, **No need for English user**)

If you want, create a shortcut icon to eXeScope.exe on desktop.

About target files which can be analyzed, refer [File types supported](#).

Limitations, Warning

- 1) This software will edit the target original file directly. Before editing resources, please backup the original file.
- 2) In the next case, editing which results file size increasing is impossible.
 - * Editing which increase size more than 256 KB.
- 3) If the editing is not proper, there is a possibility that edited executable file does not work, or work abnormally. Edited executable file should be tested carefully. Author does not guarantee any disruption caused by edited executable file.
- 4) If you distribute any customized file or exported icon/bitmap/etc. using eXeScope without agreement of authors of original file, you infringe a copyright. Please take care.
- 5) In the test display of dialog box, there are next limitations:
 - * A control which is defined 'CONTROL' is displayed as a black frame.
 - * Any 'ICON' is displayed as an Windows LOGO Icon.
- 6) In the test display of dialog box, when the dialog box cannot move or close because it covered over all eXeScope form, then move eXeScope form to foreground by pressing Alt + ESC key, and press  button.
- 7) Batch mode editing using log file can be done for 32 bit PE format only. 16 bit NE format cannot be edited by batch mode.
- 8) If EXE file is compressed using EXE compressor like ASPack or UPX, then eXeScope cannot view or edit that file.

Registration

eXeScope is a shareware. If you continually use eXeScope for over than 2 weeks, please pay US\$20 to be a registered user.

Non-registered user cannot edit an executable file more than one time.

You can use VISA card, Master card, AmericanExpress card, FirstVirtual, etc., and checks and cashes of various countries.

The payment will be processed by "KAGI" who is a shareware agent.

There are two ways to register:

1) Online

Online registration can be done at "<http://order.kagi.com/?V6E>".

2) E-Mail, FAX, or Postal-mail

There is the "Register" program to pay to eXeScope. This program will automatically generate the ORDER FORM when you fill forms in this program.

If you use credit card or network cash for payment, you can immediately send the ORDER FORM to KAGI. If you use check or cash, print it and send with printed ORDER FORM to KAGI's postal address.

For more information about KAGI, go to <http://www.kagi.com/>

If you can read Japanese, you can also use credit card registration by VECTOR, who is a shareware agent in Japan.

Go to <http://www.vector.co.jp>. Software number is 'sr004625'.

CopyRight

eXeScope copyright (c) 1998-1999 by Toshifumi Yamamoto (JCG01730@nifty.ne.jp).
All right reserved.

(Warning)

If you distribute any customized file or exported icon/bitmap/etc. using eXeScope without agreement of authors of original file, you infringe a copyright. Please take care.

Disclaimer of Warranty

This software and the accompanying files are sold "AS IS" and without warranties whether expressed or implied. Seller does not guarantee any disruption caused by edited executable file.
The user must assume the entire risk of using the program. Any liability of the seller will be limited exclusively to product replacement or refund of purchase price.

Development Tools

Delphi 3.1J

Support, Questions

If you have any question or request for eXeScope, please send a E-Mail to JCG01730@nifty.ne.jp.
Or see eXeScope home page <http://www.vector.co.jp/authors/VA003525/>, most recent version of eXeScope is here.

