

IMegaSoftware

referenční příručka ke knihovně:

`imswutil.dll`

`IMegaSoftwareWebUtil`

Verze: 1.5

Revize dokumentu: A

Stručný obsah

I. Základní informace.....	6
1. Datové typy:.....	6
II. IMegaSoftware.Info.....	7
2. Informační funkce:.....	7
III. IMegaSoftware.Image.....	8
3. Základní funkce pro práci s obrázky (JPEG, BMP).....	8
4. Funkce pro vytváření náhledů (JPEG, BMP).....	14
5. Funkce pro čtení hlaviček souborů (JPEG, BMP).....	15
6. Property IMegaSoftware.Image.....	16
IV. IMegaSoftware.BinaryRW.....	19
7. Funkce pro práci se soubory:.....	19
8. Funkce pro práci s daty z form-multipart:.....	23
9. Další pomocné funkce.....	26

Podrobný obsah

I. Základní informace.....	8
1. Datové typy:.....	8
long.....	8
Buffer.....	8
BSTR.....	8
__int64.....	8
uchar.....	8
double.....	8
Registrace dll knihovny.....	8
Odregistrace Registrace DLL.....	8
Předdefinované hodnoty parametrů funkcí.....	8
II. IMegaSoftware.Info.....	9
2. Informační funkce:.....	9
long GetVersion().....	9
long GetMinorVersion().....	9
long GetFalseValue().....	9
long GetTrueValue().....	9
long GetSystem().....	9
BSTR GetSystemText().....	9
BSTR GetAuthorInfo().....	9
III. IMegaSoftware.Image.....	10
3. Základní funkce pro práci s obrázky (JPEG,BMP).....	10
long LoadFromBmp(BSTR FileName).....	10
long LoadFromJpg(BSTR FileName).....	10
long CreateFromSize(long w, long h, long bgcolor = 0).....	10
long SaveToBmp(BSTR FileName).....	10
long SaveToJpg(BSTR FileName, long Quality = -1, long Progressive = -1, long Smoothing = -1).....	10
Buffer SaveToBufferBmp().....	11
Buffer SaveToBufferJpg(BSTR FileName, long Quality = -1, long Progressive = -1, long Smoothing = -1).....	11
long ResizeImg(long w, long h).....	11
DeleteImage().....	11
long GetColorFromRGB(uchar r, uchar g, uchar b).....	11
long IsValidImage().....	11
long IsNullImage().....	11
long RotateCW().....	12
long RotateCCW().....	12
long Rotate180().....	12
SetTransparentColor(long color).....	12
RemoveTransparency().....	12
long ChangeSize(long w, long h).....	12
long StretchDraw(long x, long y, long w, long h, object_image img)...	12
long Draw(long x, long y, object_image img).....	12
MoveTo(long x, long y).....	13
Line(long x1, long y1, long x2, long y2).....	13
LineTo(long x, long y).....	13
Rectangle(long x1, long y1, long w, long h).....	13
FillRect(long x1, long y1, long w, long h).....	13
LineRect(long x1, long y1, long w, long h).....	13
Ellipse(long x1, long y1, long w, long h).....	13
LineEllipse(long x1, long y1, long w, long h).....	13
TextOut(long x, long y, BSTR str).....	14
long GetTextHeight(BSTR str).....	14

long GetTextWidth(BSTR str).....	14
long GetColor(long x, long y).....	14
uchar GetRFromColor(long color).....	14
uchar GetGFromColor(long color).....	14
uchar GetBFromColor(long color).....	14
BSTR GetHtmlWH().....	14
long AssignPicture(object_image in).....	15
long GetBmp().....	15
4.Funkce pro vytváření náhledů (JPEG,BMP).....	16
long SetThumbBorder(long index, long color).....	16
long GetThumbBorder(long index).....	16
long MakeThumb(long insideW, long insideH).....	16
5.Funkce pro čtení hlaviček souborů (JPEG,BMP).....	17
long HeaderReadBmp(BSTR FileName).....	17
long HeaderReadJpg(BSTR FileName).....	17
long HeaderGetWidth().....	17
long HeaderGetHeight().....	17
BSTR HeaderGetHtmlWH().....	17
6.Property IMegaSoftware.Image.....	18
long DefaultJpgQuality.....	18
long DefaultJpgProgressive.....	18
long DefaultJpgSmoothing.....	18
long Width.....	18
long Height.....	18
long PenColor.....	18
long PenWidth.....	18
long FontColor.....	19
long FontHeight.....	19
uchar FontCharSet.....	19
BSTR FontName.....	20
uchar FontStyle.....	20
long BrushColor.....	20
uchar BrushStyle.....	20
long ThumbBgColor.....	20
long cntThumbBorder.....	20
IV.IMegaSoftware.BinaryRW.....	21
7.Funkce pro práci se soubory:.....	21
long FOpen(BSTR FileName, BSTR Type).....	21
long FOpenRead(BSTR FileName).....	21
long FOpenWrite(BSTR FileName).....	21
long FOpenAppend(BSTR FileName).....	21
FClose().....	21
long FEOF().....	21
long FSeek(long offset, long from).....	22
long FTell().....	22
long FileSize(BSTR FileName).....	22
VARIANT-int64 FileSize64(BSTR FileName).....	22
Buffer FRead(long Size).....	22
long FReaded().....	22
VARIANT FReadInt(long Size).....	22
VARIANT FReadFloat(long Size).....	23
BSTR FReadStr(long MaxSize = 255).....	23
Buffer FReadStrBin(long Size).....	23
long FWrite(Buffer data).....	23
long FWriteInt(__int64 data, long Size).....	23
long FWriteFloat(double data, long Size).....	23

long FWriteStr(BSTR string)..... 23
 long FWriteStrBin(VARIANT string, long Size, uchar fillchar = 0). 24
 BSTR BufferToStr(Buffer buffer)..... 24
 Buffer StrToBuffer(BSTR string, long AddZeroByte = 0)..... 24
 long IsBuffer(VARIANT v)..... 24
 buffer JoinBuffer(Buffer buffer1, Buffer buffer2)..... 24
 8. Funkce pro práci s daty z form-multipart:..... 25
 long FD_Init(BSTR dir="", BSTR FileName="", BSTR endl ine = "")... 25
 long FD_Decode(Buffer data)..... 26
 long FD_EndDecode()..... 26
 long FD_GetFilesCount()..... 26
 BSTR FD_GetFileName(long fileindex)..... 26
 BSTR FD_GetFileHeaderName(long fileindex)..... 26
 BSTR FD_GetFileContentType(long fileindex)..... 26
 BSTR FD_GetFileFormName(long fileindex)..... 26
 long FD_GetFileHeadersCount(long fileindex)..... 27
 BSTR FD_GetFileHeader(long fileindex, long index)..... 27
 long FD_GetItemsCount()..... 27
 BSTR FD_GetItemName(long index)..... 27
 BSTR FD_GetItemValue(long index)..... 27
 long FD_DeleteTMPFiles()..... 27
 FD_FreeData()..... 27
 9. Další pomocné funkce..... 28
 BSTR ExtractFileName(BSTR FileName, long mode = 0)..... 28
 BSTR ExtractFileExt(BSTR FileName, long mode = 0)..... 28
 BSTR ExtractFileNameOnly(BSTR FileName, long mode = 0)..... 28
 long FExists(BSTR FileName)..... 28
 long FileCopy(BSTR src, BSTR dest, long what)..... 28
 long FileDelete(BSTR FileName)..... 28
 long FileMove(BSTR src, BSTR dest)..... 28
 long DirExists(BSTR FileName)..... 28
 long DirMake(BSTR dirname)..... 29
 long DirDelete(BSTR EmptyDirName)..... 29
 BSTR CreateTMPFile(BSTR dir = "")..... 29
 long OldFilesDelete(long seconds, BSTR dir_mask)..... 29

I. Základní informace

1. Datové typy:

long

32 bitový integer se znaménkem

Buffer

VARIANT typu VT_ARRAY | VT_UI1

BSTR

Řetězec.

int64

64 bitový integer se znaménkem

uchar

8bitový integer bez znaménka rozsah 0-255.

double

64bitové reálné číslo

Registrace dll knihovny

```
regsvr32.exe x:\CelaCesta\imswutil.dll
```

Odregistrace Registrace DLL.

Zastavit veškeré používání knihovny (Stop WebServer-u).
regsvr32.exe /u x:\CelaCesta\imswutil.dll

Předdefinované hodnoty parametrů funkcí

Pokud má proměnná předdefinovanou hodnotu není nutné ji zadávat, v tomto případě bude použita tato předdefinovaná hodnota.

Např.

```
long CreateFromSize(long w, long h, long bgcolor = 0)
```

Pokud je zadáno `CreateFromSize(20,20)`, je vytvořen obrázek 20 x 20 pixelů vyplněný černou barvou.

II. IMegaSoftware .Info

2. Informační funkce:

long GetVersion()

Vrací verzi knihovny

long GetMinorVersion()

Vrací podverzi knihovny

long GetFalseValue()

Vrací hodnotu False. Je rovna 0 typu long (Není prakticky potřeba).

long GetTrueValue()

Vrací hodnotu True. Je rovna 0 typu long (Není prakticky potřeba).

long GetSystem()

Identifikuje operační systém číslem. Nabývá těchto hodnot:

- 1 - "Windows 95"
 - 2 - "Windows 98"
 - 3 - "Windows Milenium"
 - 4 - "WindowsNT 3.51"
 - 5 - "WindowsNT 4.0"
 - 6 - "Windows 2000"
 - 7 - "Windows XP"
 - 8 - "Windows 2003 Server"
 - 0 - "undefined"
-

BSTR GetSystemText()

Identifikuje operační systém textem. Nabývá těchto hodnot:

"Windows 95"
"Windows 98"
"Windows Milenium"
"WindowsNT 3.51"
"WindowsNT 4.0"
"Windows 2000"
"Windows XP"
„Windows 2003 Server"
"undefined"

BSTR GetAuthorInfo()

:)

III. IMegaSoftware .Image

3. Základní funkce pro práci s obrázky (JPEG, BMP)

long LoadFromBmp (BSTR FileName)

Načtení obrázku z BMP.
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

long LoadFromJpg (BSTR FileName)

Načtení obrázku z JPEG.
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

long CreateFromSize (long w, long h, long bgcolor = 0)

Vytvoření obrázku
Na "bgcolor" je obrázek vyplněn (0 = černá).
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

long SaveToBmp (BSTR FileName)

Uložení obrázku do BMP.
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

long SaveToJpg (BSTR FileName, long Quality = -1, long Progressive = -1, long Smoothing = -1)

Uložení obrázku do JPEG.
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.
Parametry Quality, Progressive, Smoothing jsou nepovinné a mají přednastavenou hodnotu -1.

Když Quality = -1, tak je při ukládání použita kvalita definovaná property 'DefaultJpgQuality'.
Když Progressive = -1, tak je při ukládání použita kvalita definovaná property 'DefaultJpgProgressive'.
Když Progressive = 1, tak je ukládán progresivní JPEG.
Když Progressive = 0, tak se ukládá standardní JPEG.
Když Smoothing = -1, tak se použije Smoothing = 'DefaultJpgSmoothing'.
Když Smoothing = 0, tak se nepoužije Smoothing.

Buffer SaveToBufferBmp()

Stejně jako SaveToBMP(), ale vrací binární buffer s obrázkem tak, jak by se zapsal do souboru.
Pro použití např. funkcí serveru 'BinaryWrite'.

Buffer SaveToBufferJpg(BSTR FileName, long Quality = -1, long Progressive = -1, long Smoothing = -1)

Stejně jako SaveToJpg(), ale vrací binární buffer s obrázkem, tak jak by se zapsal do souboru.
Pro použití např. funkcí serveru 'BinaryWrite'.

long ResizeImg(long w, long h)

Změna velikosti obrázku.
(Funkce potřebuje tolik paměti kolik potřebuje původní + nový obrázek)
(na OS Windows 95, změna je možná max. 4x nebo na 1/4 velikosti)

DeleteImage()

Uvolnění paměti bitové mapy.
Doporučení: Volání této funkce uvolňuje vnitřní paměť potřebnou pro obrázek. Pro obrázky je většinou potřeba nezanedbatelné množství paměti, proto je velmi vhodné volat tuto funkci vždy, když už není třeba pracovat s načteným obrázkem.

long GetColorFromRGB(uchar r, uchar g, uchar b)

Vrací indexovanou barvu vytvořenou ze složek.
Složky jsou definovány rozsahem 0-255.
0, 0, 0 = černá, 255,255,255 = bílá
128,128,128 = tm. šedá, 192,192,192 = světle šedá
255,255, 0 = žlutá, 255, 0,255 = fialová apod.

long IsValidImage()

Kontrola obrázku
Vrací 1 pokud je obrázek v pořádku.
Vrací 0 pokud je není načten žádný obrázek nebo byl načten obrázek, který má šířku či výšku 0.

long IsNullImage()

Kontrola obrázku
Vrací 1 pokud je obrázek v pořádku.
Vrací 0 pokud není načten žádný obrázek.
Doporučení: Pro kontrolu validity obrázku slouží funkce IsValidImage().

long RotateCW()

Rotace obrázku po směru hodinových ručiček.
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

long RotateCCW()

Rotace obrázku PROTI směru hodinových ručiček.
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

long Rotate180()

Rotace obrázku o 180st.
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

SetTransparentColor(long color)

Nastavení barvy, která se při použití funkce Draw nebo StretchDraw nevykresluje.

RemoveTransparency()

Zruší nastavení barvy z funkce SetTransparentColor().

long ChangeSize(long w, long h)

Zvětšení/Zmenšení obrázku.
Stávající obraz se nemění (případně je oříznut), na rozdíl od funkce Resize().
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

long StretchDraw(long x,long y,long w,long h,object_image img)

Překreslení jednoho objektu do druhého se změnou velikosti.
(Bez změny je výhodnější použít rychlejší funkci Draw())
x,y - pozice (0,0 - levý horní roh)
w,h - šířka,výška
object_image - jiná instance "ImageSoftware.Image"

long Draw(long x, long y ,object_image img)

Překreslení jednoho objektu do druhého BEZ změny velikosti.
x,y - pozice (0,0 - levý horní roh)
object_image - jiná instance "ImageSoftware.Image"

MoveTo(long x, long y)

Přestavení pozice pro funkci LineTo().

Line(long x1, long y1, long x2, long y2)

Kreslení úsečky zadané dvěma body.
-Dle aktuálního nastavení Pera. /PenXxxx()/

LineTo(long x, long y)

Kreslení úsečky do bodu.
Pro nastavení výchozí pozice použij funkci MoveTo().
-Dle aktuálního nastavení Pera. /PenXxxx()/

Rectangle(long x1, long y1, long w, long h)

Kreslí obdélník zadaný pozicí a rozměrem.
- Obvod je vykreslen dle aktuálního nastavení Pera. /PenXxxx()/
- Plocha je vyplněna dle aktuálního nastavení Štětce./BrushXxxx()/

FillRect(long x1, long y1, long w, long h)

Kreslí obdélník zadaný pozicí a rozměrem.
Vykreslení dle aktuálního nastavení Štětce. /BrushXxxx()/
Nastavení pera nemá vliv.

LineRect(long x1, long y1, long w, long h)

Kreslí obdélník zadaný pozicí a rozměrem.
- Obvod je vykreslen dle aktuálního nastavení Pera. /PenXxxx()/
Nastavení Štětce nemá vliv.

Ellipse(long x1, long y1, long w, long h)

Kreslí elipsu vepsanou do obdélníku zadaného pozicí a rozměrem.
- Obvod je dle aktuálního nastavení Pera. /PenXxxx()/
- Plocha je vyplněna dle aktuálního nastavení Štětce./BrushXxxx()/

LineEllipse(long x1, long y1, long w, long h)

Kreslí elipsu vepsanou do obdélníku zadaného pozicí a rozměrem.
- Obvod je dle aktuálního nastavení Pera. /PenXxxx()/
Nastavení Štětce nemá vliv.

TextOut(long x, long y, BSTR str)

Vykreslení textu na pozici x,y.

-Dle aktuálního nastavení Fontu /FontXxxx()/

long GetTextHeight(BSTR str)

Výška zadaného textu, která bude použita při vykreslení pomocí funkce TextOut().

Spolu s funkcí GetTextWidth() je možné získat informace o ploše využívané vykresleným textem.

long GetTextWidth(BSTR str)

Šířka zadaného textu, která bude použita při vykreslení pomocí funkce TextOut().

Spolu s funkcí GetTextHeight() je možné získat informace o ploše využívané vykresleným textem.

long GetColor(long x, long y)

Vrací barvu zadaného pixelu.

uchar GetRFromColor(long color)

Vrací složku RED z indexované barvy color. 0 - 255

0 = žádná, 255 = maximum

uchar GetGFromColor(long color)

Vrací složku GREEN z indexované barvy color. 0 - 255

0 = žádná, 255 = maximum

uchar GetBFromColor(long color)

Vrací složku BLUE z indexované barvy color. 0 - 255

0 = žádná, 255 = maximum

BSTR GetHtmlLWH()

Vrací šířku a výšku v HTML formátu obrázku pro použití v .

Vrací při chybě "", nenačten obrázek.

long AssignPicture(object_image in)

Předání obrázku do druhého objektu stejného typu.
Přenáší se jen obrázek, ne nastavení např. DefaultJpgQuality.
Vrací 1 v případě úspěchu.
Vrací 0 v případě neúspěchu.

long GetBmp()

NEPOUŽÍVAT

Jen pro vnitřní užití !!!!!.

4. Funkce pro vytváření náhledů (JPEG, BMP)

long SetThumbBorder(long index, long color)

Nastaví barvu rámečku.

Předem je potřeba připravit počet rámečku pomocí property cntThumbBorder.

Index se počítá z vnějšku a začíná od 0 do cntThumbBorder-1.

Vrací 1 v případě úspěchu.

Vrací 0 v případě neúspěchu - zadán špatný index.

long GetThumbBorder(long index)

Vrátí barvu rámečku.

Index se počítá z vnějšku a začíná od 0 do cntThumbBorder-1.

Vrací rovněž 0 v případě neúspěchu - zadán špatný index.

long MakeThumb(long insideW, long insideH)

Z aktuálního obrázku vytvoří náhled vepsaný do obdélníku insideW x insideH.

NEzdeformuje obrázek !!! - Okolí vyplní barvou ThumbBgColor.

Případně vykreslí rámeček nastavený funkcí SetThumbBorder.

Jednoduchý příklad v JavaScriptu napoví více:

```
var im = WScript.CreateObject("IMegaSoftware.Image");
im.cntThumbBorder = 7;//pocet pixelu ramecku
im.SetThumbBorder(0,im.GetColorFromRGB(100,100,100));//jednotlive barvy
im.SetThumbBorder(1,im.GetColorFromRGB(135,135,135));
im.SetThumbBorder(2,im.GetColorFromRGB(190,190,190));
im.SetThumbBorder(3,im.GetColorFromRGB(220,220,220));
im.SetThumbBorder(4,im.GetColorFromRGB(190,190,190));
im.SetThumbBorder(5,im.GetColorFromRGB(135,135,135));
im.SetThumbBorder(6,im.GetColorFromRGB(100,100,100));
im.ThumbBgColor = im.GetColorFromRGB(192,192,220);//barva pozadi
im.DefaultJpgQuality = 86;//vystupni kvalita
//zde muze byt vlozen cyklus
if(im.LoadFromJpg("NAZEVSOUBORU.jpg"))//nacteni obrazku - vyzaduje jpeg
{
    if(im.IsValidImage())//neni nutne
    {
        if(im.MakeThumb(100, 100))//generovani nahledu ctverec 100x100
        {
            if(!im.SaveToJpg("NAZEVNAHLEDU.jpg"))
                WScript.Echo("ERROR SaveToJpg");
            }else WScript.Echo("ERROR MakeThumb");
            }else WScript.Echo("ERROR No Valid Image");
        }else WScript.Echo("ERROR LoadFromFile");
    }//zde případně konec cyklu - DeleteImage, není nutné provádět v cyklu
    im.DeleteImage();
```

5. Funkce pro čtení hlaviček souborů (JPEG, BMP)

long HeaderReadBmp(BSTR FileName)

Načtení výšky a šířky z BMP (čte jen nezbytnou hlavičku).

Vrací 1 v případě úspěchu.

Vrací 0 v případě neúspěchu.

viz: HeaderGetWidth, HeaderGetHeight, HeaderGetHtmlWH

long HeaderReadJpg(BSTR FileName)

Načtení výšky a šířky z JPG (čte jen nezbytnou hlavičku).

Vrací 1 v případě úspěchu.

Vrací 0 v případě neúspěchu.

viz: HeaderGetWidth, HeaderGetHeight, HeaderGetHtmlWH

long HeaderGetWidth()

Vrací šířku obrázku.

Předem nutno volat funkci HeaderReadBmp() nebo HeaderReadJpg().

Vrací při chybě -1, nenačtena hlavička pomocí HeaderReadBmp() nebo

HeaderReadJpg()

long HeaderGetHeight()

Vrací výšku obrázku.

Předem nutno volat funkci HeaderReadBmp() nebo HeaderReadJpg().

Vrací při chybě -1, nenačtena hlavička pomocí HeaderReadBmp() nebo

HeaderReadJpg()

BSTR HeaderGetHtmlWH()

Vrací šířku a výšku v HTML formátu pro použití v .

Předem nutno volat funkci HeaderReadBmp() nebo HeaderReadJpg().

Vrací při chybě "", nenačtena hlavička pomocí HeaderReadBmp() nebo

HeaderReadJpg()

6. Property ImegaSoftware.Image

long DefaultJpgQuality

Určuje přednastavenou hodnotu při ukládání JPEGu.
Kvalita JPEGů po vytvoření instance nastaveno na 78.
Rozmezí 1 - 100. (100 = nejvyšší kvalita)

long DefaultJpgProgressive

Určuje přednastavenou hodnotu při ukládání JPEGu.
Generovat Progressive JPEG.
1 = ANO
Jiná hodnota NE (doporučeno 0).
Po vytvoření nastaveno na 0.

long DefaultJpgSmoothing

Určuje přednastavenou hodnotu při ukládání JPEGu.
Použít smoothing pro Progressive JPEG.
1 = ANO
Jiná hodnota NE (doporučeno 0).
Po vytvoření nastaveno na 1.

long Width

Šířka obrázku - doporučeno jen pro čtení.
Pro změnu velikosti použij raději funkci `ChangeSize()`.

long Height

Výška obrázku - doporučeno jen pro čtení.
Pro změnu velikosti použij fci `ChangeSize()`.
Pro změnu velikosti použij raději funkci `ChangeSize()`.

long PenColor

Barva pera pro kreslení.

long PenWidth

Šířka pera pro kreslení.

long FontColor

Barva fontu pro fci TextOut().

long FontHeight

Výška fontu pro fci TextOut().

uchar FontCharSet

Standardně není třeba přestavovat.

ANSI_CHARSET	0	ANSI characters.
DEFAULT_CHARSET	1	Font is chosen based solely on Name and Size. If the described font is not available on the system, Windows will substitute another font.
SYMBOL_CHARSET	2	Standard symbol set.
MAC_CHARSET	77	Macintosh characters. Not available on NT 3.51.
SHIFTJIS_CHARSET	128	Japanese shift-JIS characters.
HANGEUL_CHARSET	129	Korean characters (Wansung).
JOHAB_CHARSET	130	Korean characters (Johab). Not available on NT 3.51
GB2312_CHARSET	134	Simplified Chinese characters (mainland china).
CHINESEBIG5_CHARSET	136	Traditional Chinese characters (Taiwanese).
GREEK_CHARSET	161	Greek characters. Not available on NT 3.51.
TURKISH_CHARSET	162	Turkish characters. Not available on NT 3.51
VIETNAMESE_CHARSET	163	Vietnamese characters. Not available on NT 3.51.
HEBREW_CHARSET	177	Hebrew characters. Not available on NT 3.51
ARABIC_CHARSET	178	Arabic characters. Not available on NT 3.51
BALTIC_CHARSET	186	Baltic characters. Not available on NT 3.51.
RUSSIAN_CHARSET	204	Cyrillic characters. Not available on NT 3.51.
THAI_CHARSET	222	Thai characters. Not available on NT 3.51
EASTEUROPE_CHARSET	238	Includes diacritical marks for eastern european countries. Not available on NT 3.51.
OEM_CHARSET	255	Depends on the codepage of the operating system.

BSTR FontName

Jméno instalovaného fontu pro fci TextOut().

uchar FontStyle

Styl fontu pro fci TextOut().

Definováno bitovou maskou:

0x00 (0001) - Bold (tlusté)

0x02 (0010) - Italic (kurzíva)

0x04 (0100) - Underline (podtržené)

0x08 (1000) - StrikeOut (přeškrtnuté)

např. 3(0011) znamená Tlustou kurzívu

long BrushColor

Barva štětce (výplní) pro kreslení.

uchar BrushStyle

0 - Solid (Plnění barvou)

1 - Clear (Bez výplně)

2 - BDiagonal (Šrafy)

3 - FDiagonal

4 - Cross

5 - DiagCross

6 - Horizontal

7 - Vertical

long ThumbBgColor

Barva pozadí při použití funkce MakeThumb().

Po vytvoření = 0 (černá).

long cntThumbBorder

Počet "rámečků" při použití funkce MakeThumb().

Po vytvoření = 0;

Po zápisu:

```
im.cntThumbBorder = 3;
```

by mělo následovat definování barev

```
im.SetThumbBorder(0,im.GetColorFromRGB(190,190,190));
```

```
im.SetThumbBorder(1,im.GetColorFromRGB(220,220,220));
```

```
im.SetThumbBorder(2,im.GetColorFromRGB(190,190,190));
```

IV. IMegaSoftware.BinaryRW

7. Funkce pro práci se soubory:

long FOpen(BSTR FileName, BSTR Type)

Otevření souboru vrací 1 v případě úspěchu 0 při neúspěchu.
FileName - název souboru.
Type - typ otevření totožný ze specifikací C a funkce fopen().
"rb" - otevření pro Binární čtení
"wb" - otevření pro Binární zápis (neexistuje-li je vytvořen)
"ab" - otevření pro Binární zápis pro připojení, cursor je nastaven na konec (neexistuje-li je vytvořen).
Další možné varianty: "rt", "wt", "at" plus přídatné znaménko + např. "rb+" - pro čtení i zápis apod. dle specifikace C.

Doporučení:

Pro standardní práci je vhodnější použít funkce FOpenRead, FOpenWrite, FOpenAppend

long FOpenRead(BSTR FileName)

Otevření souboru pro čtení vrací 1 v případě úspěchu 0 při neúspěchu.
FileName - název souboru.
Ekvivalent FOpen("FileName", "rb")

long FOpenWrite(BSTR FileName)

Otevření souboru pro zápis vrací 1 v případě úspěchu 0 při neúspěchu.
FileName - název souboru.
Ekvivalent FOpen("FileName", "wb")

long FOpenAppend(BSTR FileName)

Otevření souboru pro připojení vrací 1 v případě úspěchu 0 při neúspěchu.
FileName - název souboru.
Ekvivalent FOpen("FileName", "ab")

FClose()

Uzavře otevřený soubor.
Každé otevření souboru by mělo mít své FClose().

long FEOF()

Vrací 1 pokud je dosaženo konce souboru jinak vrací 0.

long FSeek(long offset, long from)

Nastavuje pozici v souboru.

from:

- 0 - od začátku souboru
- 1 - od aktuální pozice
- 2 - od konce souboru

Vrací 1 v případě úspěchu, 0 v případě neúspěchu.

fseek(0, 0) - nastaví poici na začátek souboru.

long FTell()

Vrací pozici v souboru.

long FileSize(BSTR FileName)

Vrací velikost souboru max. 2Gb

Vrací -1 při chybě (souboru neexistuje).

VARIANT-int64 FileSize64(BSTR FileName)

Vrací velikost souboru v typu VARIANT s podtypem int64.

64bitový integer někdy může působit potíže, pokud je to možné použij funkci FileSize();

Vrací -1 při chybě (souboru neexistuje).

Buffer FRead(long Size)

Načte do 'Buffer' 'Size' bytů z aktuální pozice.

Pokud již v souboru je méně bytů než Size, tak jsou načteny.

Kolik bytu bylo načteno je možné zjistit funkcí FReaded().

long FReaded()

Vrací počet bytu načtených funkcí FRead() při posledním čtení.

Otevření a zavření souboru tuto hodnotu nuluje.

VARIANT FReadInt(long Size)

Načte z binárního souboru integerovou hodnotu.

Pokud se čtení nezdaří nebo byl zadán nesprávný parametr Size, vrací VARIANT typu null.

Size:

- 1 - byte (1byte)
 - 2 - short int,int16 (2byte)
 - 4 - long, int32 (4byte)
 - 8 - int64 (8byte)
-
-

VARIANT FReadFloat(long Size)

Načte z binárního souboru reálnou hodnotu číslo.
Pokud se čtení nezdaří nebo byl zadán nesprávný parametr Size, vrací VARIANT typu null.

Size:

4 - float	(4byte)
8 - double	(8byte)
10 - long double	(10byte)

BSTR FReadStr(long MaxSize = 255)

Načte z binárního souboru řetězec ukončený 0x0, nebo přeruší čtení po dosažení přečtení MaxSize bytů, což může mít za následek ztrátu kontroly nad pozicí v souboru.

Buffer FReadStrBin(long Size)

Jedná se o funkci totožnou s FRead() - vrací také Buffer NE BSTR.
Na konec je doplněn navíc znak 0x0.

long FWrite(Buffer data)

Zapíše buffer do souboru.
Vrací 1 v případě úspěchu 0 v případě neúspěchu.

long FWriteInt(__int64 data, long Size)

Zapíše integer hodnotu do souboru dle Size.

Size:

1 - byte	(1byte)
2 - short int,int16	(2byte)
4 - long, int32	(4byte)
8 - int64	(8byte)

long FWriteFloat(double data, long Size)

Zapíše reálnou hodnotu do souboru dle Size.

Size:

4 - float	(4byte)
8 - double	(8byte)
10 - long double	(10byte)

long FWriteStr(BSTR string)

Zapíše do binárního souboru řetězec a zakončí ho 0x0.
Tzn. zapíše se délka+1 znaků (bytů).
Pokud se jedná o unicode, provede se stejný zápis, jen počet bytů je dvojnásobný.

long FWriteStrBin(VARIANT string, long Size, uchar fillchar = 0)

Zapíše string (který je typu VARIANT nebo Buffer) do souboru.
Size udává jakou velikost má řetězec mít v souboru.
Data jsou oříznuta nebo doplněna znaky fillchar (0-255).
U stringu pozor na unicode.

BSTR BufferToStr(Buffer buffer)

Převéde Buffer na string.
Konverze může mít za následek ztrátu dat
(konec je na nulovém znaku 0x0, který může být v binárním bufferu
kdekoliv).

Buffer StrToBuffer(BSTR string, long AddZeroByte = 0)

Převéde string na buffer. Převedený string může být v unicode.
pokud AddZeroByte = 1 na konec je přidán nulový znak 0x0.

long IsBuffer(VARIANT v)

Zjistí, jestli je proměnná typu Buffer.
Vrací 1 v případě, že se jedná o Buffer, 0 při jiném typu.

buffer JoinBuffer(Buffer buffer1, Buffer buffer2)

Spojí dva buffery tak, že buffer2 připojí za buffer1.
Pracuje jen s typem Buffer.

8. Funkce pro práci s daty z form-multipart:

long FD_Init(BSTR dir="", BSTR FileName = "", BSTR endl ine = "")

Pro funkce začínající FD_ je třeba vždy předem zavolat tuto funkci. TESTUJ návratový kód, v případě úspěchu 1. V případě neúspěchu 0. (Neúspěch pravděpodobně nedostatek paměti, či jiná dysfunkce)

dir - Specifikuje adresář kde se vytváří stažené dočasné soubory.

Pokud je dir prázdný řetězec, použije se

1. Adresář specifikovaný pomocí systémové proměnné TMP
2. Pokud není definováno TMP, použije se systémová proměnná TEMP
3. Pokud není definováno TMP ani TEMP, použije se aktuální adresář.

FileName - je název dočasného souboru, ke kterému je přidáno pořadí staženého souboru při jednom uploadu a přípona ".tmp".

Pokud je FileName = "", tak je vytvořen systémem unikátní soubor (pokud je to možné).

Pokud je FileName = "name", tak názvy souborů jsou vytvořeny takto:

"name_1.tmp" pro první uploadovaný soubor.

"name_2.tmp" pro druhý uploadovaný soubor.

Proto by tento název měl být jedinečný (např. SessionID).

Pokud se nepodaří soubor "name_1.tmp" vytvořit, tak:

1. Vytvoří se soubor "name_xxxx_1.tmp", kde "xxxx" je hexadecimální číslo generováno z doby běhu (xxxx = 0000 první použití od startu serveru).
2. Pokud se nepodaří vytvořit "name_xxxx_1.tmp", tak se vytvoří soubor "name_xxx_yy_1.tmp" kde "yy" je minimálně dvojznakové náhodné hexadecimální číslo.
3. Pokud se nepodaří vytvořit "name_xxx_yy_1.tmp" funkce EndDecode() vrátí 11.

endl ine - je použit pro konce řádků např. "textarea"

(případně i „input type=text“ se znaky konce řádku)-

Pro konec řádku je možné použít např. "
".

Pokud je endl ine = "", tak je konec řádku definován jako 'crlf'(0D0A tzn. standardní textový dokument ve formátu dos, Windows).

Pokud je endl ine nastaveno na "{-NoLineEnd-}", tak se z value stane jeden řetězec, bez jakéhokoliv dělení.

Doporučení:

dir = Temp adresář webové aplikace.

FileName = "";

long FD_Decode(Buffer data)

Funkce přebírající binární data typu Buffer.
Umožňuje bufferování vstupu (doporučeno).
Vrací false v případě chyby nebo vstupu 0 byte.
Není třeba testovat, důležité je testovat fci FD_EndDecode(), která MUSÍ být volána po posledním volání FD_Decode().

long FD_EndDecode()

Volej vždy po posledním volání FD_Decode().
Návratový kód = 0 je vše OK.
Jiný kód, přenos neproběhl korektně.
Stávající návratové kódy jsou rezervované hodnoty -32000 až 100:
1 - Chyba zápisu do dočasných souborů.
11 - Nelze vytvořit dočasný soubor (např. již existuje).
12 - Interní chyba
-100 až -500 chyba vstupního formátu.
-1000 až -1500 nastala neočekávaná chyba - pravděpodobně chyba vstupního formátu.

long FD_GetFilesCount()

Vrací počet uploadovaných souborů.

BSTR FD_GetFileName(long fileindex)

Vrací název dočasného souboru, ve kterém je uložen stažený soubor.
Index je pořadí uploadovaného souboru.

BSTR FD_GetFileHeaderName(long fileindex)

Vrací název souboru z hlavičky, tzn. tak jak ji prohlížeč poslal.
Index je pořadí uploadovaného souboru.
PS: Někdy to je plná cesta, někdy jen název.

BSTR FD_GetFileContentType(long fileindex)

Vrací Content-Type.
Index je pořadí uploadovaného souboru.

BSTR FD_GetFileFormName(long fileindex)

Vrací název (name) formulářového prvku.
Index je pořadí uploadovaného souboru.

long FD_GetFileHeadersCount(long fileindex)

Vrací počet hlaviček k danému uploadovanému souboru.
Index je pořadí uploadovaného souboru.

BSTR FD_GetFileHeader(long fileindex, long index)

Vrací jednu hlavičku k danému uploadovanému souboru.
Fileindex je pořadí uploadovaného souboru.
Index je pořadí hlavičky.

long FD_GetItemsCount()

Vrací počet formulářových položek.

BSTR FD_GetItemName(long index)

Vrací název položky.
Index je pořadí položky.

BSTR FD_GetItemValue(long index)

Vrací hodnotu položky.
Index je pořadí položky.

long FD_DeleteTMPFiles()

Smaže všechny dočasné soubory používané při tomto uploadu.
Vrací počet smazaných souborů.

FD_FreeData()

Uvolní vnitřní alokovanou paměť objektu.

Doporučení:

Zavolej po ukončení práce s uploadovanými daty.

9. Další pomocné funkce

BSTR ExtractFileName(BSTR FileName, long mode = 0)

Vrátí název souboru z FileName bez cesty i s příponou.

mode = 0 .. beze změny

mode = 1 .. konverze na malá písmena

mode = 2 .. konverze na velká písmena

BSTR ExtractFileExt(BSTR FileName, long mode = 0)

Vrátí příponu souboru z FileName bez cesty

mode = 0 .. beze změny

mode = 1 .. konverze na malá písmena

mode = 2 .. konverze na velká písmena

BSTR ExtractFileNameOnly(BSTR FileName, long mode = 0)

Vrátí název souboru z FileName bez cesty a bez přípony.

mode = 0 .. beze změny

mode = 1 .. konverze na malá písmena

mode = 2 .. konverze na velká písmena

long FExists(BSTR FileName)

V případě, že soubor existuje vrací 1, jinak 0.

long FileCopy(BSTR src, BSTR dest, long what)

Kopíruje soubor src na soubor dest.

Pokud je what = 1 a soubor dest existuje NEbude přepsán a funkce vrátí 0.

Pokud je what = 0 a soubor dest existuje je přepsán a funkce vrátí 1.

Při zkopírování souboru vrací 1, jinak 0.

long FileDelete(BSTR FileName)

Smaže soubor.

Vrací 1 v případě úspěchu, 0 v případě neúspěchu.

long FileMove(BSTR src, BSTR dest)

Přesune soubor src do dest. Vhodnější je použít FileCopy a FileDelete.

Vrací 1 v případě úspěchu, 0 v případě neúspěchu.

long DirExists(BSTR FileName)

Vrací 1 pokud adresář existuje, jinak vrací 0.

long DirMake(BSTR dirname)

Vytvoří adresář.

Vrací 1 v případě úspěchu, 0 v případě neúspěchu.

long DirDelete(BSTR EmptyDirName)

Smaže prázdný adresář.

Vrací 1 v případě úspěchu, 0 v případě neúspěchu.

BSTR CreateTMPFile(BSTR dir = "")

Vytvoří soubor.

dir určuje cestu, kde se má soubor vytvořit.

Pokud je dir = "" tak se použije adresář:

1. Adresář specifikovaný pomocí systémové proměnné TMP
2. Pokud není definováno TMP, použije se systémová proměnná TEMP
3. Pokud není definováno TMP ani TEMP, použije se aktuální adresář.

Název souboru je vygenerován jako unikátní s koncovkou tmp.

Pokud se soubor nevytvoří je vrácen prázdný řetězec jinak cesta a název vytvořeného souboru.

long OldFilesDelete(long seconds, BSTR dir_mask)

Promaže všechny soubory v adresáři které:

Byly vytvořeny (ne změněny) před seconds (čas udaný v sekundách, max. 2147483647sec .. cca. 68 let).

Nemaže soubory, které mají nastaveny tyto atributy:

FILE_ATTRIBUTE_OFFLINE
FILE_ATTRIBUTE_HIDDEN
FILE_ATTRIBUTE_READONLY
FILE_ATTRIBUTE_SYSTEM

dir_mask - je cesta i maskou, např. "c:\kuk\temp*.tmp"

Vrací počet smazaných souborů.

Abecední rejstřík funkcí a property

BrushColor.....	18	FD_Init.....	23
BrushStyle.....	18	FEof.....	19
__int64.....	6	FExists.....	26
- A -			
AssignPicture.....	13	FileCopy.....	26
- B -			
BSTR.....	6	FileDelete.....	26
Buffer.....	6	FileMove.....	26
BufferToStr.....	22	FileSize.....	20
- C -			
cntThumbBorder.....	18	FileSize64.....	20
CreateFromSize.....	8	FillRect.....	11
CreateTMPFile.....	27	FontHeight.....	17
ChangeSize.....	10	FontCharSet.....	17
- D -			
DefaultJpgProgressive....	16	FontName.....	18
DefaultJpgQuality.....	16	FontStyle.....	18
DefaultJpgSmoothing.....	16	FOpen.....	19
DeleteImage.....	9	FOpenAppend.....	19
DirDelete.....	27	FOpenRead.....	19
DirExists.....	26	FOpenWrite.....	19
DirMake.....	27	FRead.....	20
double.....	6	FReaded.....	20
Draw.....	10	FReadFloat.....	21
- E -			
Ellipse.....	11	FReadInt.....	20
ExtractFileExt.....	26	FReadStr.....	21
ExtractFileName.....	26	FReadStrBin.....	21
ExtractFileNameOnly.....	26	FSeek.....	20
- F -			
FClose.....	19	FTell.....	20
FD_Decode.....	24	FWrite.....	21
FD_DeleteTMPFiles.....	25	FWriteFloat.....	21
FD_EndDecode.....	24	FWriteInt.....	21
FD_FreeData.....	25	FWriteStr.....	21
FD_GetFileContentType....	24	FWriteStrBin.....	22
FD_GetFileFormName.....	24	- G -	
FD_GetFileHeader.....	25	GetAuthorInfo.....	7
FD_GetFileHeaderName....	24	GetBFromColor.....	12
FD_GetFileHeadersCount..	25	GetColor.....	12
FD_GetFileName.....	24	GetColorFromRGB.....	9
FD_GetFilesCount.....	24	GetFalseValue.....	7
FD_GetItemName.....	25	GetGFromColor.....	12
FD_GetItemsCount.....	25	GetHtmlWH.....	12
FD_GetItemValue.....	25	GetMinorVersion.....	7
- H -			
- G -			
- H -			
- I -			
- J -			
- K -			
- L -			
- M -			
- N -			
- O -			
- P -			
- Q -			
- R -			
- S -			
- T -			
- U -			
- V -			
- W -			
- X -			
- Y -			
- Z -			

HeaderReadBmp..... 15
 HeaderReadJpg..... 15
 Height..... 16

- I -

IsBuffer..... 22
 IsNullImage..... 9
 IsValidImage..... 9

- J -

JoinBuffer..... 22

- L -

Line..... 11
 LineEllipse..... 11
 LineRect..... 11
 LineTo..... 11
 LoadFromBmp..... 8
 LoadFromJpg..... 8
 long..... 6

- M -

MakeThumb..... 14
 MoveTo..... 11

- O -

OldFilesDelete..... 27

- P -

PenWidth..... 16

- R -

Rectangle..... 11
 RemoveTransparency..... 10
 ResizeImg..... 9
 Rotate180..... 10
 RotateCCW..... 10
 RotateCW..... 10

- S -

SaveToBmp..... 8
 SaveToBufferBmp..... 9
 SaveToBufferJpg..... 9
 SaveToJpg..... 8
 SetThumbBorder..... 14
 SetTransparentColor..... 10
 StretchDraw..... 10
 StrToBuffer..... 22

- T -

TextOut..... 12
 ThumbBgColor..... 18

- U -

uchar..... 6

- W -

Width..... 16