
Subject: Re: SysInfo - some bug fixes
Posted by [luoganda](#) on Tue, 06 Dec 2016 14:07:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Window_top:

on windowsxp it doesn't show up, at least not programmatically with many windows on - yet to check,
but suggested ver works always

Window_SaveCapture:

Excelent example is Reference/CaptureScreenDll - it properly DeSelects oldBM, but it is a little buggy,
so here is patched version, without the patch img is invisible:

```
Image ScreenShot(int x, int y, int cx, int cy){  
HDC dcScreen = CreateDC("DISPLAY", NULL, NULL, NULL);
```

```
RGBA *pixels;
```

```
Buffer<byte> data;  
data.Alloc(sizeof(BITMAPINFOHEADER) + sizeof(RGBQUAD)*256);  
BITMAPINFOHEADER *hi = (BITMAPINFOHEADER *) ~data;;  
memset(hi, 0, sizeof(BITMAPINFOHEADER));  
hi->biSize = sizeof(BITMAPINFOHEADER);  
hi->biPlanes = 1;  
hi->biBitCount = 32;  
hi->biCompression = BI_RGB;  
hi->biSizeImage = 0;  
hi->biClrUsed = 0;  
hi->biClrImportant = 0;  
hi->biWidth = cx;  
hi->biHeight = -cy;
```

```
HBITMAP hbmp = CreateDIBSection(dcScreen, (BITMAPINFO *)hi, DIB_RGB_COLORS, (void  
**)&pixels, NULL, 0);  
HDC dcMem = ::CreateCompatibleDC(dcScreen);  
HBITMAP hbmpOld = (HBITMAP) ::SelectObject(dcMem, hbmp);  
HDC hdcCompatible = CreateCompatibleDC(dcScreen);  
::BitBlt(dcMem, 0, 0, cx, cy, dcScreen, x, y, SRCCOPY);
```

```
//ver-1: use this version or down 2 - whichever is faster  
//Image img; <= direct to image could be used here to optimize, since now it's into buffer and then  
to Image  
//mint 32on32Bit/64on64bit machine dependant int to speed up  
//for(register mint i=0,j=cx*cy; i<j; i++)pixels[i].a=0xff;
```

```
ImageBuffer ib(cx, cy);
memcpy(~ib, pixels, cx * cy * sizeof(RGBA));

//ver-2: use this version or up 1 - whichever is faster
//mint 32on32Bit/64on64bit machine dependant int to speed up
dword *pix=(dword*)~ib;
for(register mint i=0,j=cx*cy; i<j; i++)pix[i]=*(dword*)&pixels[i]|0xff000000;

::DeleteObject(::SelectObject(dcMem, hbmpOld));
::DeleteDC(dcMem);
::DeleteDC(dcScreen);

return ib;
};
```

And the test code:

```
Image img=ScreenShot(200,200,400,400);
ImageCtrl ic; ic.SetImage(img); //qbox(img.ToString());
TopWindow w; w.Add(ic.TopPos(0,400).LeftPos(0,400)); w.LeftPos(0,440).TopPos(0,440);
w.Execute();
```
