
Subject: Windows7 compatibility

Posted by [mdelfede](#) on Mon, 27 Jul 2026 09:10:46 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

I recompiled my app for a customer and it complained about 2 missing functions:

GetDpiForWindow and GetSystemMetricsForDpi.

I found that he's still using windows7 and that providing a workaround inside Win32Proc.cpp solves the issue.

I added this:

```
static UINT CompatGetDpiForWindow(HWND hwnd)
{
    typedef UINT (WINAPI *GetDpiForWindowFn)(HWND);

    static GetDpiForWindowFn fn =
        (GetDpiForWindowFn)GetProcAddress(
            GetModuleHandleA("user32.dll"),
            "GetDpiForWindow"
        );

    if(fn)
        return fn(hwnd);

    // Windows 7: non esiste DPI per-monitor.
    HDC dc = GetDC(hwnd);
    UINT dpi = dc ? (UINT)GetDeviceCaps(dc, LOGPIXELSX) : 96;

    if(dc)
        ReleaseDC(hwnd, dc);

    return dpi ? dpi : 96;
}

static int CompatGetSystemMetricsForDpi(int index, UINT dpi)
{
    typedef int (WINAPI *GetSystemMetricsForDpiFn)(int, UINT);

    static GetSystemMetricsForDpiFn fn =
        (GetSystemMetricsForDpiFn)GetProcAddress(
            GetModuleHandleA("user32.dll"),
            "GetSystemMetricsForDpi"
        );

    if(fn)
```

```
return fn(index, dpi);

// Su Windows 7 il DPI è unico per tutto il desktop:
// GetSystemMetrics restituisce già le metriche appropriate.
return GetSystemMetrics(index);
}
```

and replaced the few calls to the compatibility functions

Ciao

Max
