

גירסה 1.00 – 06.11.2005



# Direct3D באמצעות win32api - בסיס

בן זמיר

מסמך זה הורד מהאתר <http://www.underwar.co.il>

מחבר המסמך איננו אחראי לכל נזק, ישיר או עקיף, שיגרם עקב השימוש במידע המופיע במסמך, וכן לנכונות התוכן של הנושאים המופיעים במסמך. עם זאת, המחבר עשה את מירב המאמצים כדי לספק את המידע המדויק והמלא ביותר.

כל הזכויות שמורות לבן זמיר ([cxben@hotmail.com](mailto:cxben@hotmail.com)).

**מבוא**

Direct3D הוא עולם הגרפיקה של מיקרוסופט. Direct3D הוא חלק מהספריה המוכרת DirectX שתפקידה לבצע פעולות מולטימדיה. ישנן כמה שפות שDirectX מתאימה להם: Visual Basic, C++, Delphi ועוד... DirectX בנוי במקור למערכת Windows אך מתאים גם לתכנות לפלטפורמת X-Box ולימיקרוסופט PocketPC.

פעולת העבודה של Direct3D (D3D) הוא כזה:

- 1) יצירת חלון שיכיל את הסצנה.
- 2) אתחול D3D בחלון שנוצר, אתחול וטעינה של העצמים שהולכים להיות מוצגים (תמונות, מודלים, צורות ועוד...).
- 3) ציור העצמים הגרפיים עם שימוש באפקטים (מכונה רינדור Render).  
 הסצנה התלת מימדית מורכבת מנקודות המכונות Vertex (נקודה בעלת 2-3 ערכים X, Y ובתלת מימד גם Z), Lines – קווים שמחברים בין הנקודות, Polygons – דפנות העצמים שאליהן מצמידים בדרך כלל צבע או תמונה (מכונה Texture).

כמה דברים לפני שמתחילים:

1. מה צריך- ידע טוב של השפה C++, Win32Api, מהדר Visual Studio מומלץ מאוד (אני משתמש Visual Studio 2003, Visual Studio .net, DirectX9 SDK ורצון ללמוד).
2. הדוגמא בפורמט .sln של Visual Studio 7.
3. הדברים הבסיסיים של Win32API אינם מוסברים שוב. ניתן ללמוד את הנושא מספר שקונים או מאתרים שונים כגון <http://www.msdn.com>. Win32API זהו הבסיס לתכנות בסביבת Windows עם C++.

**הכנת סביבת העבודה**

טוב, עכשיו כשיש מה שצריך נתחיל: דבר ראשון נכליל בפרוייקט את קובץ הLIB. אפשר להוסיף אותו דרך התפריט במהדר, תלוי מה המהדר שלכם. אני בחרתי לעשות את זה בצורה של: #pragma

```
#pragma comment(lib, "d3d9.lib")
```

כדי להתאים לכל מהדר.

Pragma מיצגת הוראה למהדר. במקרה הזה, היא אומרת לlinker – המקשר להכליל את קובץ הספריה שצריך.

עכשיו קבצי הכותרת (שימו לב! כרגע קובץ הכותרת הוא d3d9 בהמשך נשנה אותו ל d3dx9 כדי לטעון מודלים בפורמט X למשל – פורמט של מודלים תלת מימדיים שd3ds תומך בהם בצורה ישירה. d3dx9 מכיל כלים שימושיים לטעינה לזיכרון של עצמים שנרצה להציג כגון תמונות פונטים (מודלים ועוד)

```
#include <windows.h>
#include <d3d9.h>
```

עכשיו נגדיר את המשתנים של D3D מחוץ לפונקציות כדי שהן יהיו נגישות מכל פונקציה:

```
LPDIRECT3D9 m_D3D = NULL;
LPDIRECT3DDEVICE9 m_D3DD = NULL;
```

LPDIRECT3D9 הוא המשתנה הבסיסי של D3D ממנו ניצור את הDEVICE שמשמש לפעולות הבסיסיות של D3D. שני המשתנים מוצב בהם. 0 – NULL

## אתחול Direct3D

עכשיו נתחיל לעבוד עם D3D. דבר ראשון צריך לבצע את ה-INITIALIZE שזה בעצם הגדרת D3D והכנתו לפעולה. הפונקציה D3D9Init תבצע את הפעולה. INITIALIZE יבוצע אחרי יצירת חלון (CreateWindow) והיא צריכה את הידית של החלון (HWND). נעבור על הקוד:

```
m_D3D = Direct3DCreate9( D3D_SDK_VERSION );
```

הפונקציה Direct3Dcreate9 יוצרת את האובייקט הבסיסי של D3D. היא מקבלת פרמטר אחד שמיצג את הגרסא של D3D ומחזירה NULL במקרה של שגיאה.

```
if(m_D3D == NULL)
{
    MessageBox(0, "D3D create failed", "error", 0);
    return E_FAIL;
}
```

בדיקה של המצב, במידה והפונקציה נכשלה לא ניתן להמשיך ולכן מציגים הודעת שגיאה והפונקציה חוזרת בשגיאה.

עכשיו צריך להגדיר את האפשרויות של החלון.

```
D3DPRESENT_PARAMETERS d3dpm;
ZeroMemory(&d3dpm, sizeof(d3dpm));
d3dpm.BackBufferFormat = D3DFMT_UNKNOWN;
d3dpm.Windowed = TRUE;
d3dpm.SwapEffect = D3DSWAPEFFECT_DISCARD;
```

השורה הראשונה מגדירה אובייקט שיכיל את המידע לשימוש אחר כך, השורה השניה מציבה 0 בכל האפשרויות ואז מגדירים את המשתנים שצריך. BackBuffer הוא המשטח האחורי של המסך- מה שלא מוצג.

אחרי שהגדרנו את האפשרויות ניצור את ה-DEVICE שיבצע את פעולות הרינדור והטעינה.

```
if(FAILED(m_D3D->CreateDevice(D3DADAPTER_DEFAULT, D3DDEVTYPE_HAL,
hWnd, D3DCREATE_SOFTWARE_VERTEXPROCESSING, &d3dpm, &m_D3DD)))
{
    MessageBox(0, "D3D Device creation failed", "error", 0);
    return E_FAIL;
}
```

השורות האלו עושות את הפעולה הזו ובודקות אם הייתה שגיאה. כאן מסתיים ה-INITIALIZE.

עכשיו נשאר הרינדור. ב WinProc מבצעים את הרינדור ברגע שמתקבלת ההודעה WM\_PAINT:

```
case WM_PAINT:
    Render();
```

הרינדור הבסיסי:

```
m_D3DD->Clear(0, 0, D3DCLEAR_TARGET, D3DCOLOR_XRGB(0,0,255),
1.0f, 0);
m_D3DD->BeginScene();
m_D3DD->EndScene();
m_D3DD->Present(0,0,0,0);
```

קריאה לפונקציה clear כדי לנקות את המסך לצבע מסוים.  
בין Endi Begin ו- Present יבוצע הרינדור עצמו ו- Present מציג את מה שנעשה (הופך את החוצצים)

זהו, זה הבסיס לd3d9. ציור צורות מתבצעת מיד לאחר האתחול ואז מצוירת בrender. טעינת  
קבצים למיניהם יתבצע עם פונקציות עזר של d3dx9 כמו D3DXLoadMeshFromX או על ידי  
למידת הפורמט וטעינה עצמית שלו (שימוש ב fopen כדי לפתוח ו fread כדי לקרוא נתונים  
ולהעבירם למשתנים)