

The DirectX Files

Game Programming in the X Dimension

Introduction

- ◆ Introduction to DirectX II
- ◆ An Overview of Game and Graphics Programming with DirectDraw
- ◆ Equinox

Teach Yourself DirectX in About 7 1/2 Minutes

- ◆ Evolution of the DX SDK
- ◆ Device Independence - Quest for the Holy Grail
- ◆ HAL (Hardware Abstraction Layer)
HEL (Hardware Emulation Layer)
- ◆ DirectX 11 APIs

The Origin of Species

- ◆ Dos - Assembly, DOS4/GW, Mode 13h/X
- ◆ Windows - WinG, Software Hooks to Hardware
- ◆ Win95 - DirectX or DOS mode
- ◆ Windows NT and Apple Power Macs (soon)

Device Independence

- ◆ Crusades for the Holy Land
- ◆ HAL - The Great Interpreter
- ◆ HEL - The Great Deceiver
- ◆ COM - (Component Object Model)

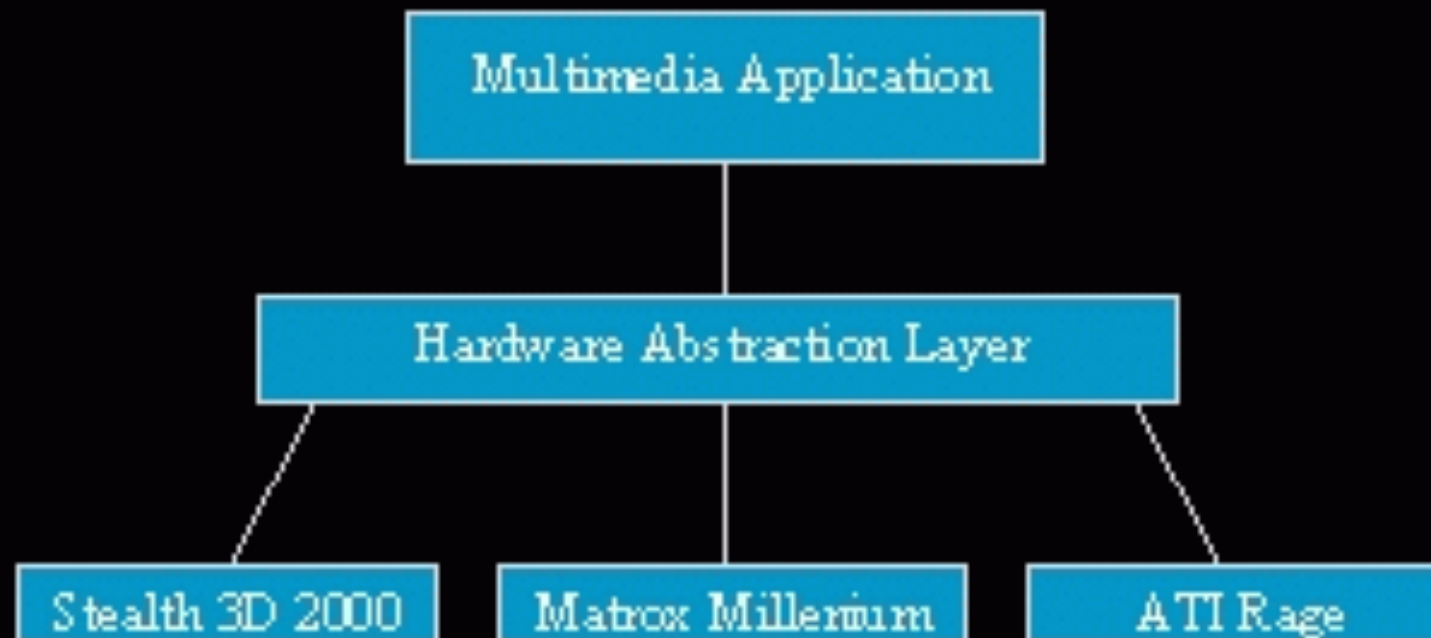
COM Interfacing

- ◆ COM - foundation of OLE
- ◆ OLE - Object Linking and Embedding:
 - allows for connectable component software
- ◆ COM is basic linking mechanism:
 - differs from programming languages
 - method is inherited, not the code

New Possibilities

- ◆ Java SDK
 - object oriented, platform independent
 - can be integrated with COM
- ◆ Result
 - applets can be extended to the COM platform

DirectX HAL



DirectX HEL

Sam's Choice 1MB VGA



System Memory



Really Slow 2MB VGA

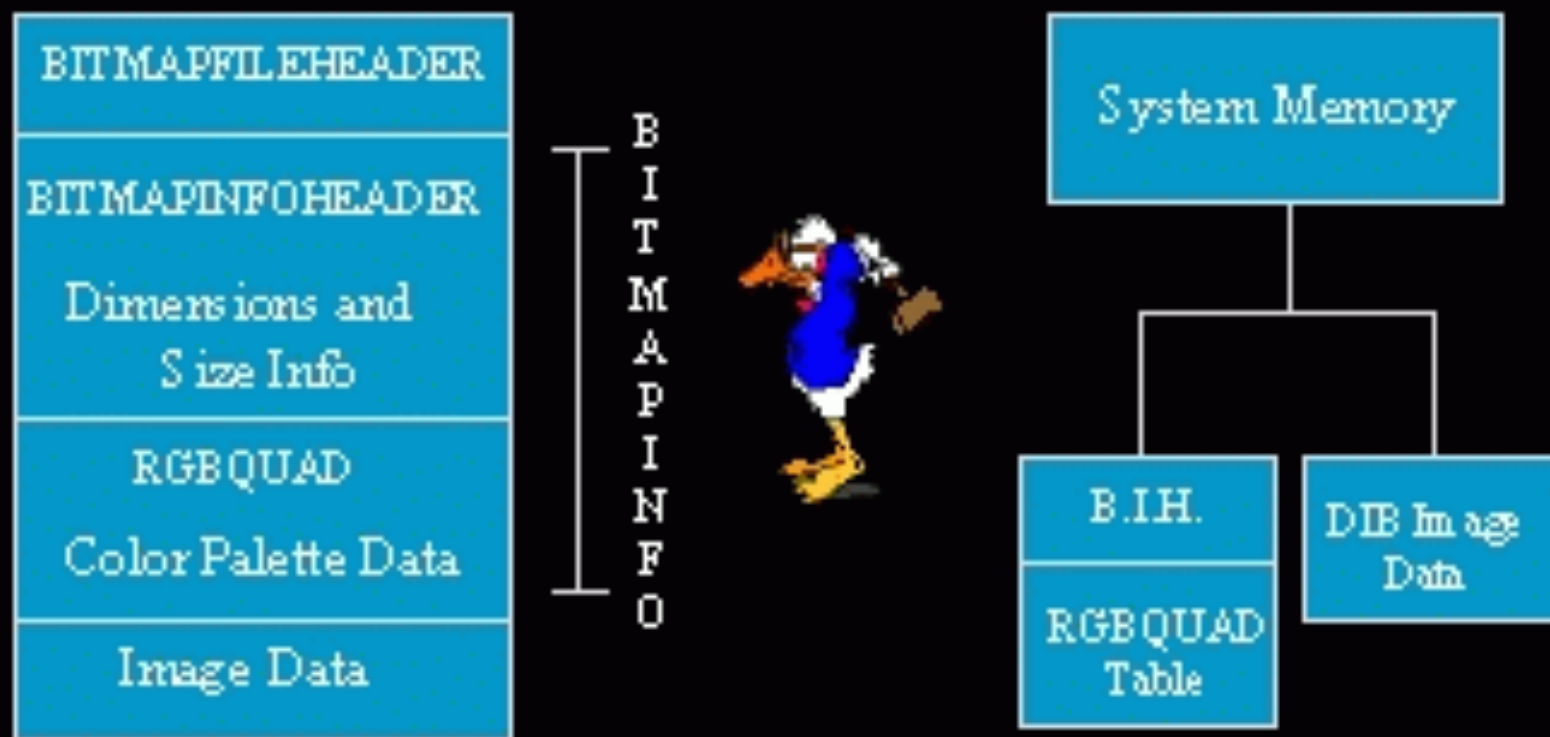
DirectX II API's

- ◆ **DirectDraw** - Graphics
- ◆ **DirectSound** - Sound
- ◆ **DirectPlay** - Networking
- ◆ **DirectInput** - Joystick Input
- ◆ **AutoPlay** - CD-ROM Auto-Startup

The Complete Idiots' Guide to Graphics Programming

- ◆ DIB, BMP, PCX, IFF, LBM, etc...
- ◆ Backgrounds and Sprites
- ◆ Surfaces and Page Flipping
- ◆ The Search for the Better Blitter

Device Independent Bitmap (DIB)



Sprites and Backgrounds



Background

Offscreen Buffer (Surface)



Surfaces and Page-Flipping

- 1.) Write to BackSurface
- 2.) Flip BackSurface to PrimarySurface
- 3.) Flip PrimarySurface to BackSurface
- 4.) Repeat



BackSurface



PrimarySurface

What Is This Blit Thing Anyway?



- 1) Compose on OffscreenSurface
- 2) Write to BackSurface
- 3) Flip Surfaces



EQUINOX

Tips, Tricks, and Secrets

- ◆ The Game
- ◆ Implementation - DirectDraw Coding Examples
- ◆ Future Plans





Time : September 26, 2007

Place : A world become completely dependent on technology - banks, industries, government, even the military - all at the mercy of the almighty microchip.

MacroCORP - buys out last independent hardware manufacturer.

Anonymous announcement : For the past five years MacroCORP has been planting a device in their hardware which - unless the US government hands over all function to MacroCORP by December 26, 2007 - will render all machines inoperable.

MacroCORP confirms announcement.

As Flynn Wolf ("WolfByte"), you must (with the assistance of four other characters) electronically infiltrate MacroCORP systems and find information needed to disable the "Equinox" chip.

The Making of *EQUINOX*

- ◆ Implements the following DirectDraw APIs
 - DirectDraw
 - DirectSound
 - DirectInput
- ◆ Programming in Microsoft Visual C++ 4.0
- ◆ Graphics - Rendered in CorelDraw Suite 6

Welcome to the Jungle!!!

- Initialize a DirectDraw Object (essentially an area in video memory)

// Instantiate Object

HRESULT ddval;

ddval = DirectDrawCreate(NULL, &lpDD, NULL);

if (ddval != DD_OK)

{ return DDError("DirectDrawCreate Failure!"); }

// Set to Exclusive Mode

ddval = lpDD->SetCooperativeLevel(hwnd,

DDSCL_EXCLUSIVE | DDSCL_FULLSCREEN);

// Set Video Mode to Desired Resolution (in this case 640 x 480 x 8)

ddval = lpDD->SetDisplayMode(640, 480, 8);

YEAH!!!! WE DID IT!!!! ... ummm... now what?!

Initializing the Surfaces

```
DDSURFACEDESC ddsd;    // Declare a DirectDraw Surface
DDSCAPS         ddscaps; // Direct Draw Surface Capabilities
HRESULT          ddval;  // Return Variable

memset( &ddsd, 0, sizeof( ddsd ) ); // Fill Surface with Zeroes
ddsd.dwSize = sizeof( ddsd );
ddsd.dwFlags = DDSD_CAPS | DDSD_BACKBUFFERCOUNT //fields in ddsd
ddsd.ddsCaps.dwCaps = DDSCAPS_PRIMARYSURFACE |
                    DDSCAPS_FLIP | DDSCAPS_COMPLEX;
ddsd.dwBackBufferCount = 1; // Create 1 back buffer
ddval = lpDD->CreateSurface( &ddsd, &lpDDSPPrimary, NULL );
ddscaps.dwCaps = DDSCAPS_BACKBUFFER;
ddval = lpDDSPPrimary->GetAttachedSurface( &ddscaps, &lpDDSBack );
```

Aren't you glad Microsoft made it easy!?!

Plans for the Future

- ◆ Complete the Game (up to 3 levels)
- ◆ Create a Web Page for Status Update
- ◆ Equinox on-line with JAVA and JavaScript
- ◆ Get some sleep...
- ◆ Graduate

Resources

◆ Books

- *Spells of Flxy* - Michael J. Norton
- *Black Art of 3D Game Programming* - Andre LaMothe
- *Programming Windows 95* - Charles Petzold

◆ Internet

- Microsoft's Games Developer Page (download DX3SDK)
– www.microsoft.com/gamesdev/
- Game Developer Magazine - www.gdmag.com
- DirectX Mailing List - listmgr@panorama.poly.edu