EBOARD'S IN PITTSBURGH POST WIN32

Simon Kadesh

April 5th

WHY WIN32?

- Widely supported
  - Introduced in Windows NT (1993)
  - Runs on everything since
  - Even then, very similar to the Win16 API
- Write Windows programs
  - Windows systems still make up a majority of desktop systems
  - Exposes platform specific features
  - “Fun”

INTRO TO WIN32

- A few differences from Unix programming
  - Not POSIX compliant
  - Preferred language is C++
  - Unusual type definitions
- The Windows API is object oriented
  - But not how you think it is
- Strings

SetWindowTextA takes an ANSI string.

SetWindowTextW takes a Unicode string.

C++

Copy

```
#ifdef UNICODE
 typedef LPWSTR LPTSTR;
#else
 typedef LPSTR LPTSTR;
#endif
```

TBYTE A WCHAR if **UNICODE** is defined, a CHAR otherwise.

Windows represents Unicode characters using UTF-16

Data type		
BYTE		
DWORD		
INT32		
INT64		
LONG	32 bits	Signed
LONGLONG	64 bits	Signed
UINT32	32 bits	Unsigned
UI		
UL		
UL		
W		

BOOL is a type alias for **int**, distinct from C++'s **bool**, and from other types that represent a Boolean value. The header file `WinDef.h` also defines two values for use with **BOOL**.

C++ Copy

```
#define FALSE    0
#define TRUE     1
```

Hungarian Notation

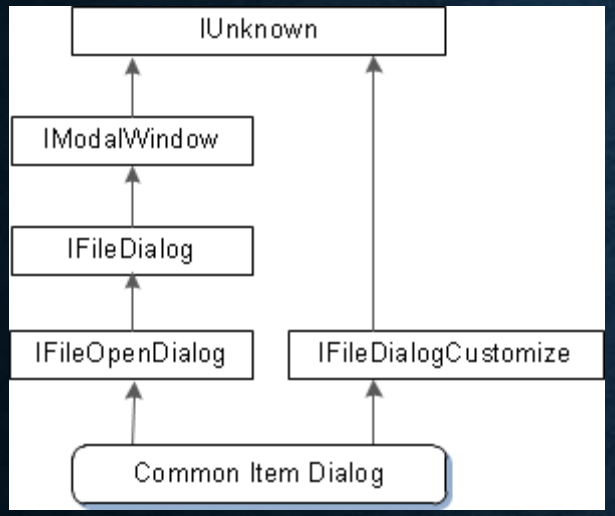
Windows defines many data types of the form *pointer-to-X*. These usually have the prefix *P-* or *LP-* in the name. For example, **LPRECT** is a pointer to a RECT, where **RECT** is a structure that describes a rectangle. The following variable declarations are equivalent.

C++ Copy

```
RECT*  rect;  // Pointer to a RECT structure.
LPRECT rect;  // The same
PRECT  rect;  // Also the same.
```

THE COM

- ABI specification for accessing objects
- Component Object Model
  - Separate objects and functions from the client application
  - Most functionality in the Windows API is accessed through COM objects
- Written in object oriented C
  - Objects are opaque pointers
  - Request "interfaces" from objects
    - Simply structs containing function pointers



C++ Copy

```
hr = pFileOpen->QueryInterface(IID_IFileDialogCustomize,
    reinterpret_cast<void**>(&pCustom));
if (SUCCEEDED(hr))
{
    // Use the interface. (Not shown.)
    // ...

    pCustom->Release();
}
else
{
    // Handle the error.
}
```

WINMAIN

- Easiest way to get the Windows API functionality we want
  - Special entry point for Windows programs

```C++
int WINAPI wWinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, PWSTR pCmdLine, int nCmdShow);
```

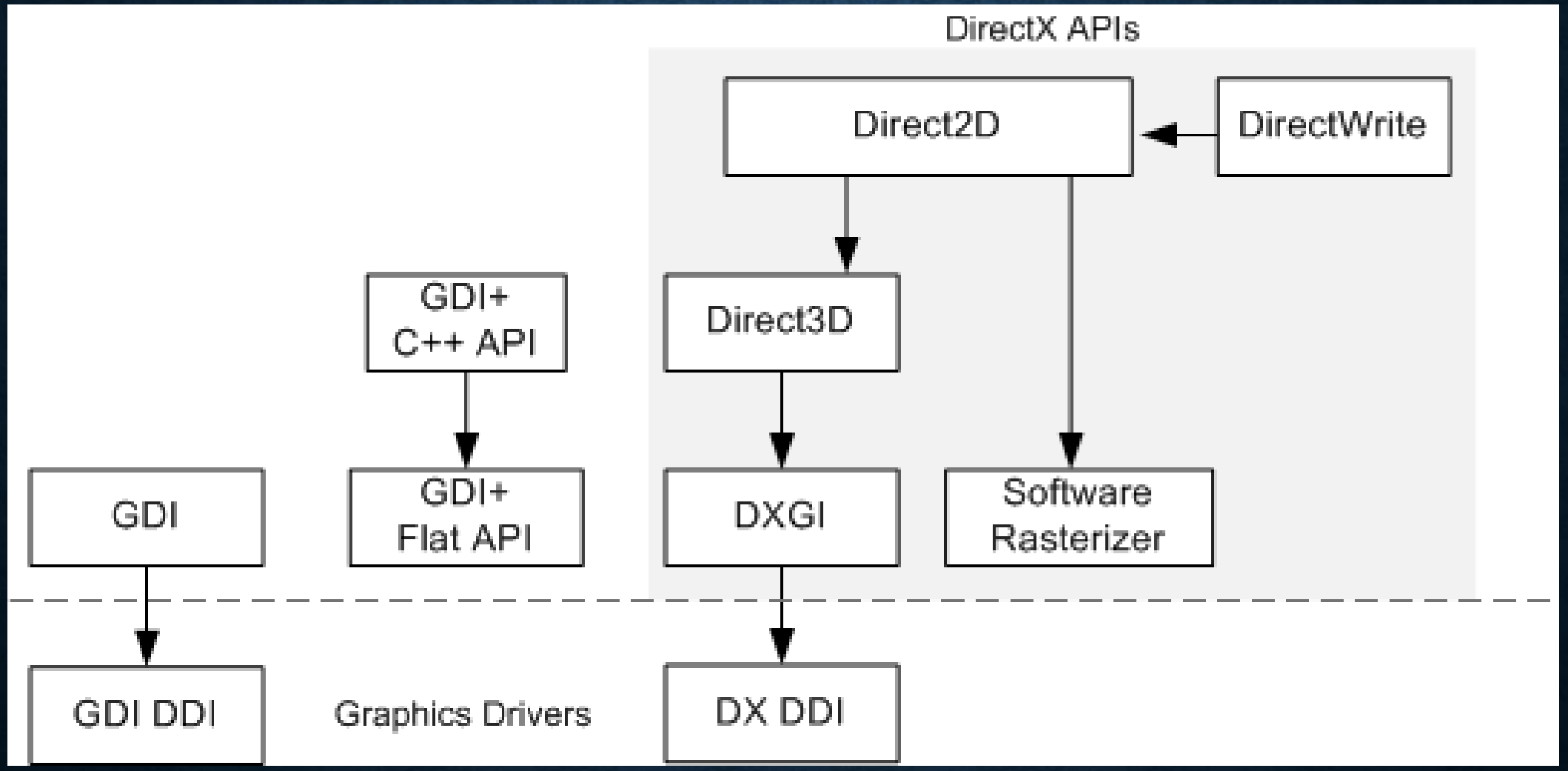
  - hInstance – Handle to instance, contains information about our process
  - hPrevInstance – Always zero, inherited from 16 bit Windows
  - pCmdLine – A string containing our command line arguments (all of them)
  - nCmdShow – Tells us whether our application is visible

BRIEF EXAMPLE INTERJECTION

USEFUL APIS IN WINDOWS

GDI/GDI+/DIRECT2D

- 2D graphics interfaces for Windows
- Original GDI is from the 16 bit days
  - Partial hardware acceleration
  - Aliasing
- GDI+ from XP onwards
  - Some improvements to GDI
- Direct2D introduced in Windows 7
  - Full hardware acceleration


DirectX APIs
Direct2D
DirectWrite
GDI+
C++ API
Direct3D
GDI
GDI+
Flat API
DXGI
Software
Rasterizer
GDI DDI
Graphics Drivers
DX DDI

DIRECT3D/DIRECTX

- 3D Graphics
  - Windows specific
  - Sometimes includes useful features
- DirectX 9 generally looks like OpenGL pre-3.3
- DirectX 10 generally looks like OpenGL 3.3
- DirectX 11 generally looks like OpenGL 4.0
- DirectX 12 generally looks like Vulkan

PROCESSES/THREADS

- Processes
  - About what you might expect
- Threads
  - Create, join, etc.
  - Pause and resume
  - Nicer thread synchronization API
- Fibers
- Thread Pools
  - Built in functions to create and manage thread pools and work queues

THE “FUN” WORLD OF WINDOWS

HELLO.C

- Windows SDK for Windows 1.0
  - Microsoft wants to demonstrate their new operating system
  - Includes code samples to show off their API
  - One of those programs is HELLO
- HELLO.C is 125 lines long
  - Also requires a 22 line long resource script

HOW MANY UI FRAMEWORKS?

- Win32
  - Older than I am
  - Have to deal with COM
- Windows Forms
  - Requires .NET
- Windows Presentation Foundation
- Universal Windows Platform
  - Dead on arrival
  - Also bad
- WinUI