

Forum:[Hack Forums](#)

Thread: [Trying to make my crosshair work.](#)

the best guy ever - May 7, 2014, 9:40 am

Hey guys, I'm trying my luck at C# here. I'm more of a C++ guy, but learning doesn't hurt.

Right now, I'm trying to make a crosshair. Can you please review my code, and help me figure out where I'm getting the error on line 3? It would be appreciated.

```
Code using System;using System.Runtime.InteropServices;namespace Crosshair{public static
class Win32{public enum Bool{False= 0,True};[StructLayout(LayoutKind.Sequential)]public struct
Point{public Int32 x;public Int32 y;public Point(Int32 x, Int32 y) { this.x = x; this.y = y;
}}[StructLayout(LayoutKind.Sequential)]public struct Size{public Int32 cx;public Int32 cy;public
Size(Int32 cx, Int32 cy) { this.cx = cx; this.cy = cy; }}[StructLayout(LayoutKind.Sequential, Pack =
1)]struct ARGB{public byte Blue;public byte Green;public byte Red;public byte
Alpha;}[StructLayout(LayoutKind.Sequential, Pack = 1)]public struct BLENDFUNCTION{public byte
BlendOp;public byte BlendFlags;public byte SourceConstantAlpha;public byte
AlphaFormat;}[StructLayout(LayoutKind.Sequential)]public struct RECT{public int Width{get { return
(Right - Left); }}public int Height{get { return (Bottom - Top); }}public int Left;public int Top;public int
Right;public int Bottom;}public const Int32 ULW_COLORKEY = 0x00000001;public const Int32
ULW_ALPHA = 0x00000002;public const Int32 ULW_OPAQUE = 0x00000004;public const byte
AC_SRC_OVER = 0x00;public const byte AC_SRC_ALPHA = 0x01;[DllImport("user32.dll",
ExactSpelling = true, SetLastError = true)]public static extern Bool UpdateLayeredWindow(IntPtr
hwnd, IntPtr hdcDst, ref Point pptDst, ref Size psize, IntPtr hdcSrc, ref Point pprSrc, Int32 crKey, ref
BLENDFUNCTION pblend, Int32 dwFlags);[DllImport("user32.dll", ExactSpelling = true,
SetLastError = true)]public static extern IntPtr GetDC(IntPtr hwnd);[DllImport("user32.dll",
ExactSpelling = true)]public static extern int ReleaseDC(IntPtr hwnd, IntPtr
hDC);[DllImport("gdi32.dll", ExactSpelling = true, SetLastError = true)]public static extern IntPtr
CreateCompatibleDC(IntPtr hdc);[DllImport("gdi32.dll", ExactSpelling = true, SetLastError =
true)]public static extern Bool DeleteDC(IntPtr hdc);[DllImport("gdi32.dll", ExactSpelling =
true)]public static extern IntPtr SelectObject(IntPtr hdc, IntPtr hObject);[DllImport("gdi32.dll",
ExactSpelling = true, SetLastError = true)]public static extern Bool DeleteObject(IntPtr
hObject);[DllImport("user32.dll")]public static extern int FindWindow(string lpClassName, string
lpWindowName);[return: MarshalAs(UnmanagedType.Bool)][DllImport("user32.dll")]public static
extern bool GetWindowRect(IntPtr hwnd, out RECT lpRect);}}
```

Thank you in advance..

Edit: Wow, I figured it out a minute after posting. I was missing a } after { this.cx = cx; this.cy = cy; } so the code should have looked like { this.cx = cx; this.cy = cy; }

Hopefully, this can be a reference to anyone else who encounters this issue in the future.

Cornflower Blue ???-?— - May 7, 2014, 9:52 am

What the hell is that mess!

Look up how to make a DirectX overlay.
Or even XNA.

the best guy ever - May 7, 2014, 10:45 am

(May 7, 2014 10:30 AM)Visual Basic. Wrote: [\[CITE\]](#) This is probably the worst copy paste i have ever

seen.

It's not a copy paste. It's my own format.

Edit: Unless you mean copy pasted from visual studio, then yes. But this is how I format my own code, sorry if there was a problem.

(May 7, 2014 09:52 AM)lhendrey4.5 Wrote: [\[CITE\]](#) What the hell is that mess!

Look up how to make a DirectX overlay.

Or even XNA.

I know how to make a DirectX overlay. I'm no noob to programming. Why do I need XNA? It has nothing to do with what I -was- trying to achieve.

Weakky. - May 7, 2014, 11:50 am

(May 7, 2014 10:45 AM)the best guy ever Wrote: [\[CITE\]](#) It's not a copy paste. It's my own format.

Edit: Unless you mean copy pasted from visual studio, then yes. But this is how I format my own code, sorry if there was a problem.

I know how to make a DirectX overlay. I'm no noob to programming. Why do I need XNA? It has nothing to do with what I -was- trying to achieve.

Assuming this is your code, and according to the way you're "formatting" (if it's even possible to call that formatting), then sorry man, but you're a noob.

"Any fool can write code that a computer can understand. Good programmers write code that humans can understand." - Martin Fowler. I've never seen something that messy and horrible so far. Even javascript eval packed function are clearer than that

TreeTopSky - May 7, 2014, 11:53 am

Perhaps his messy code is a good thing.

Perhaps someone was to view the source of his program, and see that code.

Perhaps, the viewer will be swayed to stop.

;))

the best guy ever - May 7, 2014, 12:01 pm

(May 7, 2014 11:53 AM)TreeTopSky Wrote: [\[CITE\]](#) Perhaps his messy code is a good thing.

Perhaps someone was to view the source of his program, and see that code.

Perhaps, the viewer will be swayed to stop.

;))

That's some nice thinking you got there! If only others did!?

Noone viewed the source of my program, except you guys now! Except that's just a snippet that I needed help with, but I got it. Thanks again!

Cornflower Blue ???-?— - May 7, 2014, 12:47 pm

(May 7, 2014 10:45 AM)the best guy ever Wrote: [\[CITE\]](#) It's not a copy paste. It's my own format.

Edit: Unless you mean copy pasted from visual studio, then yes. But this is how I format my own code, sorry if there was a problem.

I know how to make a DirectX overlay. I'm no noob to programming. Why do I need XNA? It has nothing to do with what I -was- trying to achieve.

You're not a noob to programming but you cant draw an image on a screen? Seems legit.

Also, XNA is a wrapper for DirectX, you can make a game overlay using XNA.

p-xy - May 7, 2014, 1:25 pm

Thought I would just pretty print for all the lazy.

PHP Code

```
<?php
using System;
using System.Runtime.InteropServices;
namespace Crosshair
{
    public static class Win32
    {
        public enum Bool
        {
            False= 0,True
        }
        ;
        [StructLayout(LayoutKind.Sequential)]public struct Point
        {
            public Int32 x;
            public Int32 y;
            public Point(Int32 x, Int32 y)
            {
                this.x = x;
                this.y = y;
            }
        }
        [StructLayout(LayoutKind.Sequential)]public struct Size
        {
            public Int32 cx;
            public Int32 cy;
            public Size(Int32 cx, Int32 cy)
            {
                this.cx = cx;
                this.cy = cy;
            }
        }
        [StructLayout(LayoutKind.Sequential, Pack = 1)]struct ARGB
        {
            public byte Blue;
            public byte Green;
            public byte Red;
            public byte Alpha;
        }
        [StructLayout(LayoutKind.Sequential, Pack = 1)]public struct BLENDFUNCTION
        {
            public byte BlendOp;
            public byte BlendFlags;
            public byte SourceConstantAlpha;
            public byte AlphaFormat;
```

```

}
[StructLayout(LayoutKind.Sequential)]public struct RECT
{
    public int Width
    {
        get
        {
            return (Right - Left);
        }
    }
    public int Height
    {
        get
        {
            return (Bottom - Top);
        }
    }
    public int Left;
    public int Top;
    public int Right;
    public int Bottom;
}
public const Int32 ULW_COLORKEY = 0x00000001;
public const Int32 ULW_ALPHA = 0x00000002;
public const Int32 ULW_OPAQUE = 0x00000004;
public const byte AC_SRC_OVER = 0x00;
public const byte AC_SRC_ALPHA = 0x01;
[DllImport("user32.dll", ExactSpelling = true, SetLastError = true)]public static
extern Bool UpdateLayeredWindow(IntPtr hwnd, IntPtr hdcDst, ref Point pptDst, ref Size psize,
IntPtr hdcSrc, ref Point pprSrc, Int32 crKey, ref BLENDFUNCTION pblend, Int32 dwFlags);
[DllImport("user32.dll", ExactSpelling = true, SetLastError = true)]public static
extern IntPtr GetDC(IntPtr hWnd);
[DllImport("user32.dll", ExactSpelling = true)]public static extern int ReleaseDC(IntPtr hWnd,
IntPtr hdc);
[DllImport("gdi32.dll", ExactSpelling = true, SetLastError = true)]public static
extern IntPtr CreateCompatibleDC(IntPtr hdc);
[DllImport("gdi32.dll", ExactSpelling = true, SetLastError = true)]public static
extern Bool DeleteDC(IntPtr hdc);
[DllImport("gdi32.dll", ExactSpelling = true)]public static extern IntPtr SelectObject(IntPtr hdc,
IntPtr hObject);
[DllImport("gdi32.dll", ExactSpelling = true, SetLastError = true)]public static
extern Bool DeleteObject(IntPtr hObject);
[DllImport("user32.dll")]public static extern int FindWindow(string lpClassName,
string lpWindowName);
[return: MarshalAs(UnmanagedType.Bool)][DllImport("user32.dll")]public static
extern bool GetWindowRect(IntPtr hWnd, out RECT lpRect);
}
}

```

0xDEAD10CC - May 8, 2014, 4:11 pm

XNA is old and deprecated already, there's no future use for it.

Banks? - May 8, 2014, 4:52 pm

(May 8, 2014 04:11 PM)0xDEAD10CC Wrote: [\[CITE\]](#) XNA is old and deprecated already, there's no future use for it.

It may not be getting anymore updates, but is it 'old' yet? I don't think so. Anyways, there's an open source version that continues to get updates: [MonoGame](#)

0xDEAD10CC - May 8, 2014, 4:55 pm

It's old news then, there's no sense in arguing the semantics here, technology is considered old a year after most devices first come out. The first stable release was 2 years ago, and the fact still remains that it is already deprecated... What old means to various people, is subjective, and I'm not here to debate that. That's silly.

Banks? - May 8, 2014, 5:05 pm

(May 8, 2014 04:55 PM)0xDEAD10CC Wrote: [\[CITE\]](#) It's old news then, there's no sense in arguing the semantics here, technology is considered old a year after most devices first come out. The first stable release was 2 years ago, and the fact still remains that it is already deprecated... What old means to various people, is subjective, and I'm not here to debate that. That's silly.

DirectX 11 was released in 2009. DirectX 11.1 was released in 2012. That's a 3 year gap without any major updates. MonoGame's last stable release was a month ago, and the last stable release before that was about a year ago.

0xDEAD10CC - May 8, 2014, 5:41 pm

You're still not getting it... DirectX isn't deprecated, regardless of how many updates are made to it in X amount of time. MonoGame has built around the XNA framework. These statistics have no relevance to what I was pointing out. The XNA framework IS deprecated on the other hand, and you're comparing wrappers around XNA to the thing XNA was built around. How is that even logical? That's like comparing Windows programs to the Windows Operating System... At its core, although MonoGame has expanded, it uses XNA, which is an implementation or framework around DirectX.

Gundam X - May 8, 2014, 5:46 pm

Who needs obfuscation when you can use this guy's "formatting"!

Banks? - May 8, 2014, 7:24 pm

(May 8, 2014 05:41 PM)0xDEAD10CC Wrote: [\[CITE\]](#) You're still not getting it... DirectX isn't deprecated, regardless of how many updates are made to it in X amount of time. MonoGame has built around the XNA framework. These statistics have no relevance to what I was pointing out. The XNA framework IS deprecated on the other hand, and you're comparing wrappers around XNA to the thing XNA was built around. How is that even logical? That's like comparing Windows programs to the Windows Operating System... At its core, although MonoGame has expanded, it uses XNA, which is an implementation or framework around DirectX.

MonoGame is not a 'wrapper around' XNA; it IS XNA.

0xDEAD10CC - May 8, 2014, 8:46 pm

(May 8, 2014 07:24 PM)Banks? Wrote: [\[CITE\]](#) MonoGame is not a 'wrapper around' XNA; it IS XNA. Do you even know what a wrapper is?... You obviously do not, as that statement shows you have absolutely no clue. If MonoGame was exactly XNA it wouldn't provide the same functionality as XNA and more (i.e. being cross platform). Just the same as a WebClient is a wrapper for WebRequest.

They've implemented and expanded on the functionality of the XNA 4.0 framework. Explain how that is not a wrapper around XNA, but rather, "IS" XNA? Was it developed by Microsoft? Didn't think so.

Banks? - May 9, 2014, 6:08 am

(May 8, 2014 08:46 PM)0xDEAD10CC Wrote: [\[CITE\]](#) Do you even know what a wrapper is?... You obviously do not, as that statement shows you have absolutely no clue. If MonoGame was exactly XNA it wouldn't provide the same functionality as XNA and more (i.e. being cross platform). Just the same as a WebClient is a wrapper for WebRequest.

They've implemented and expanded on the functionality of the XNA 4.0 framework. Explain how that is not a wrapper around XNA, but rather, "IS" XNA? Was it developed by Microsoft? Didn't think so.

WebClient is a wrapper for WebRequest because it uses WebRequests internally. MonoGame IS XNA in the sense that it does everything that XNA does, but it doesn't use it. If you swapped out XNA for MonoGame, you wouldn't have to change anything.

Something doesn't have to be made by the same people for it to be the same thing. If it literally is the exact same thing, and functions the same, then it doesn't matter who it is made by.

0xDEAD10CC - May 9, 2014, 4:50 pm

(May 9, 2014 06:08 AM)Banks? Wrote: [\[CITE\]](#) WebClient is a wrapper for WebRequest because it uses WebRequests internally. MonoGame IS XNA in the sense that it does everything that XNA does, but it doesn't use it. If you swapped out XNA for MonoGame, you wouldn't have to change anything.

Something doesn't have to be made by the same people for it to be the same thing. If it literally is the exact same thing, and functions the same, then it doesn't matter who it is made by.

Quote:MonoGame IS XNA in the sense that it does everything that XNA does, but it doesn't use it.

Really? :S And MonoGame is only Windows platform specific just like XNA then I would assume?
:S And that although Microsoft discontinued XNA, the developers of XNA believe that Microsoft was really going to take XNA in a cross platform specific direction in order to call their implementation 'XNA' itself?

It's extended; an implementation on top of XNA 4.0 functionality. Earlier you tried to argue how DirectX hasn't been updated in a while and how MonoGame has... ?

MonoGame on Windows (in order of inheritance): MonoGame -> SharpDX -> DirectX.

Comparing updates to MonoGame vs DirectX is ridiculous.

Quote:MonoGame is free software used by game developers to make their Windows and Windows Phone games run on other systems. It currently supports Mac OS, Linux, iOS, Android, PlayStation Mobile, and the OUYA console. It implements the Microsoft XNA 4 Application programming interface.[3] It has been used for several bestselling games, including Bastion and Fez.

<http://www.monogame.net/about/>

Quote:MonoGame is an Open Source implementation of the Microsoft XNA 4 Framework

Quote:Something doesn't have to be made by the same people for it to be the same thing.

Sure, but would you call an operating system developed by a 3rd party to look like Windows, "Windows"? Absolutely not. Irregardless of any legalities. To argue such would be no different than saying that a clone of Angelina Jolie is the same person, when in fact it's not; not the same history, experiences, etc... We're arguing common sense here.

ManDoR - May 9, 2014, 5:41 pm

(May 9, 2014 04:50 PM)0xDEAD10CC Wrote: [\[CITE\]](#) Really? :S And MonoGame is only Windows platform specific just like XNA then I would assume? :S And that although Microsoft discontinued XNA, the developers of XNA believe that Microsoft was really going to take XNA in a cross platform specific direction in order to call their implementation 'XNA' itself?

It's extended; an implementation on top of XNA 4.0 functionality. Earlier you tried to argue how DirectX hasn't been updated in a while and how MonoGame has... ?

MonoGame on Windows (in order of inheritance): MonoGame -> SharpDX -> DirectX.

Comparing updates to MonoGame vs DirectX is ridiculous.

<http://www.monogame.net/about/>

Sure, but would you call an operating system developed by a 3rd party to look like Windows, "Windows"? Absolutely not. Irregardless of any legalities. To argue such would be no different than saying that a clone of Angelina Jolie is the same person, when in fact it's not; not the same history, experiences, etc... We're arguing common sense here.

Both of you are wrong. Monogame isn't XNA because its source code totally differs from the original XNA's source code.

But Monogame isn't a wrapper of XNA, because Monogame doesn't use any components of XNA, at all. All of them has been redesigned and re-coded.

Mono game is the sequel of XNA, a clone, because the use of it is the same as the use of XNA.

the best guy ever - May 9, 2014, 7:03 pm

everyone who is still posting is being very informative and on topioc thank u , mwah

Cornflower Blue ???-?— - May 9, 2014, 7:04 pm

(May 9, 2014 05:41 PM)ManDoR Wrote: [\[CITE\]](#) Both of you are wrong. Monogame isn't XNA because its source code totally differs from the original XNA's source code.

But Monogame isn't a wrapper of XNA, because Monogame doesn't use any components of XNA, at all. All of them has been redesigned and re-coded.

Mono game is the sequel of XNA, a clone, because the use of it is the same as the use of XNA.

This guy is correct, well from what I have read anyway.

It uses the same method names so you can just convert your XNA game to this with out making any major changes. You loose all your content pipeline stuff which is unique to the XNA framework.

And Monogame uses the Mono framework to make it crossplatform, this is why it isn't a wrapper, it's a whole new thing, just like a phone and a computer- Does the same thing, just in a different way.

(May 8, 2014 04:11 PM)0xDEAD10CC Wrote: [\[CITE\]](#) XNA is old and deprecated already, there's no future use for it.

I recommended to use XNA because working with DirectX is a pain, making an overlay with XNA is easily done (Done it my self) I tried to do it with just DX and there isn't many tutorials on it so getting started is really hard. For what he want's done, XNA being deprecated or not doesn't matter, it is still there to be used, just like Windows XP & vista - They may be old, but it does the job relatively well.

0xDEAD10CC - May 9, 2014, 7:05 pm

(May 9, 2014 05:41 PM)ManDoR Wrote: [\[CITE\]](#) Both of you are wrong. Monogame isn't XNA

because its source code totally differs from the original XNA's source code.

But Monogame isn't a wrapper of XNA, because Monogame doesn't use any components of XNA, at all. All of them has been redesigned and re-coded.

Mono game is the sequel of XNA, a clone, because the use of it is the same as the use of XNA.

I never said it embedded XNA, so you have no point here. I took it from their developer site where it says they implement XNA 4.0, which is true, and if you would say that this is wrong, you would be wrong. This is the whole reason why porting XNA to and from MonoGame is relatively direct. They implement XNA and extend it, that's the point of a wrapper, if not to make a complex implementation easier to use... Take the WebClient class as an example for my case.

I'll say it once again, in a more succinct context, MonoGame IS an extended wrapper for a custom implementation of XNA 4.0 for easy porting to & from XNA with cross platform features/capabilities.

Cornflower Blue ???-?— - May 9, 2014, 7:11 pm

(May 9, 2014 07:05 PM)0xDEAD10CC Wrote: [\[CITE\]](#) I never said it embedded XNA, so you have no point here. I took it from their developer site where it says they implement XNA 4.0, which is true, and if you would say that this is wrong, you would be wrong. This is the whole reason why porting XNA to and from MonoGame is relatively direct. They implement XNA and extend it, that's the point of a wrapper, if not to make a complex implementation easier to use... Take the WebClient class as an example for my case.

I'll say it once again, in a more succinct context, MonoGame IS an extended wrapper for a custom implementation of XNA 4.0 for easy porting to & from XNA with cross platform features/capabilities. I would love to see your sources where it says that MonoGame IS a wrapper for XNA.

Edit: You should read theses

[http://en.wikipedia.org/wiki/Wrapper_\(programming\)](http://en.wikipedia.org/wiki/Wrapper_(programming))

<http://darkgenesis.zenithmoon.com/xna-to-monogame-and-beyond/>

Spoiler [\(Click to View\)](#)

And maybe even this

<http://stackoverflow.com/questions/20573246/what-is-monogame-supposed-to-be>

0xDEAD10CC - May 9, 2014, 7:17 pm

(May 9, 2014 07:11 PM)lhendrey4.5 Wrote: [\[CITE\]](#) I would love to see your sources where it says that MonoGame IS a wrapper for XNA.

Would you ask the same for references to common sense? ? Do you know what the programmer's definition of a wrapper is?

Cornflower Blue ???-?— - May 9, 2014, 7:24 pm

(May 9, 2014 07:17 PM)0xDEAD10CC Wrote: [\[CITE\]](#) Would you ask the same for references to common sense? ? Do you know what the programmer's definition of a wrapper is?

I just edited the old post with a link saying what it is.

<http://stackoverflow.com/questions/3293752/where-and-how-is-the-term-used-wrapper-in-programming-what-does-it-help-to-do>

Quote:The term 'wrapper' gets thrown around a lot. Generally its used to describe a class which contains an instance of another class, but which does not directly expose that instance. The wrapper's main purpose is to provide a 'different' way to use wrapped object (perhaps the wrapper provides a simpler interface, or adds some functionality).

"The wrapper's main purpose is to provide a 'different' way to use wrapped object"

Monogame can't use the XNA framework because XNA is managed code which can only be ran on windows operation systems. Monogame is cross-platform, and MAC != windows, therefor MAC cant use XNA, so what is it going to do? Oh yeah, use Monogames methods which have the same name as XNA's and do the same thing just using OpenTK and not directX.

the best guy ever - May 9, 2014, 7:25 pm

thank you i hdenre 4,5 for the good information

Cornflower Blue ???-? - May 9, 2014, 7:26 pm

(May 9, 2014 07:25 PM)the best guy ever Wrote: [\[CITE\]](#) thank you i hdenre 4,5 for the good information

"i hdenre 4,5" - love it!

You're welcome for what little help I have given you.

Ta-kun - May 9, 2014, 7:26 pm

I don't mean to hate, but if you are going to code being organized is a must especially if you are doing a group project. Glad to see you figured out your problem though! :D

the best guy ever - May 9, 2014, 7:27 pm

(May 9, 2014 07:26 PM)Cplusplus Wrote: [\[CITE\]](#) I don't mean to hate, but if you are going to code being organized is a must especially if you are doing a group project. Glad to see you figured out your problem though! :D

thanks that was good really good help i appreciate it

0xDEAD10CC - May 9, 2014, 7:38 pm

(May 9, 2014 07:24 PM)lhendrey4.5 Wrote: [\[CITE\]](#) I just edited the old post with a link saying what it is.

<http://stackoverflow.com/questions/3293752/where-and-how-is-the-term-used-wrapper-in-programming-what-does-it-help-to-do>

"The wrapper's main purpose is to provide a 'different' way to use wrapped object"

Monogame can't use the XNA framework because XNA is managed code which can only be ran on windows operation systems. Monogame is cross-platform, and MAC != windows, therefor MAC cant use XNA, so what is it going to do? Oh yeah, use Monogames methods which have the same name as XNA's and do the same thing just using OpenTK and not directX.

The term is subjective, but the general idea remains the same. Hence why terms like "generally" were thrown into his post. They wrote a cross-platform implementation of XNA 4.0, and extended it.

I also remember saying absolutely nothing about MonoGame using XNA too, but rather, an implementation of XNA 4.0... Which is no different than the wording on their developer site. I still don't see where you're pulling these assumptions from, for where you got that I said MonoGame specifically uses Microsoft's XNA. ?

Quote:and do the same thing just using OpenTK and not directX

OpenTK is used only when the cross-platform requirement includes a platform that isn't Windows (metro)... MonoGame aims to implement SharpDX for Windows app development, and OpenTK is

used otherwise. SharpDX = managed DirectX. OpenTK is a managed (.NET), and unmanaged (Mono) implementation of OpenGL. You don't even know how MonoGame works on Windows if you're saying that it doesn't use DirectX.

the best guy ever - May 9, 2014, 7:41 pm

so far good thread im learning a lot here

Cornflower Blue ???-? — May 10, 2014, 3:13 am

(May 9, 2014 07:38 PM)0xDEAD10CC Wrote: [\[CITE\]](#) I also remember saying absolutely nothing about MonoGame using XNA too, but rather, an implementation of XNA 4.0... Which is no different than the wording on their developer site. I still don't see where you're pulling these assumptions from, for where you got that I said MonoGame specifically uses Microsoft's XNA. ?You said that MonoGame was a wrapper of XNA, you must of go confused on the differences between implementation and wrapper.

Quote:OpenTK is used only when the cross-platform requirement includes a platform that isn't Windows (metro)... MonoGame aims to implement SharpDX for Windows app development, and OpenTK is used otherwise. SharpDX = managed DirectX. OpenTK is a managed (.NET), and unmanaged (Mono) implementation of OpenGL. You don't even know how MonoGame works on Windows if you're saying that it doesn't use DirectX.

Yep, I wrote a 1,848 word essay on this. I was talking about how the MAC uses MonoGame, and how it doesn't use DirectX (because it cant).

Quote:it doesn't use DirectX.

For windows it can use DirectX or OpenGL, but it uses DirectX by default because it runs faster and is more supported by windows (Except windows XP, and Vista if it used DirectX9)

ManDoR - May 10, 2014, 8:36 am

(May 9, 2014 07:05 PM)0xDEAD10CC Wrote: [\[CITE\]](#) I never said it embedded XNA, so you have no point here. I took it from their developer site where it says they implement XNA 4.0, which is true, and if you would say that this is wrong, you would be wrong. This is the whole reason why porting XNA to and from MonoGame is relatively direct. They implement XNA and extend it, that's the point of a wrapper, if not to make a complex implementation easier to use... Take the WebClient class as an example for my case.

I'll say it once again, in a more succinct context, MonoGame IS an extended wrapper for a custom implementation of XNA 4.0 for easy porting to & from XNA with cross platform features/capabilities. In the programming world, taking your words, a wrapper is something built around something else, hence it embed this "something else".

But here, MonoGame doesn't use XNA at all.

Quote:I also remember saying absolutely nothing about MonoGame using XNA too, but rather, an implementation of XNA 4.0...

You are saying that MonoGame is a wrapper of XNA, but here you are saying that MonoGame is an implementation of XNA. You are contradicting yourself. A wrapper is not, and can't be, an implementation. These two are both very different things.

Cornflower Blue ???-? — May 10, 2014, 10:47 am

(May 10, 2014 08:36 AM)ManDoR Wrote: [\[CITE\]](#) You are saying that MonoGame is a wrapper of XNA, but here you are saying that MonoGame is an implementation of XNA. You are contradicting yourself. A wrapper is not, and can't be, an implementation. These two are both very different things. He knows he is wrong that's why he changed his mind and tried to cover it up xD

0xDEAD10CC - May 10, 2014, 1:37 pm

(May 10, 2014 03:13 AM)lhendrey4.5 Wrote: [\[CITE\]](#) For windows it can use DirectX or OpenGL, but it uses DirectX by default because it runs faster and is more supported by windows (Except windows XP, and Vista if it used DirectX9)

You still don't know how it works. It doesn't use DirectX by default unless you are writing a metro app. They had the OpenGL template's much before DirectX was even used.

(May 10, 2014 08:36 AM)ManDoR Wrote: [\[CITE\]](#) You are saying that MonoGame is a wrapper of XNA, but here you are saying that MonoGame is an implementation of XNA. You are contradicting yourself. A wrapper is not, and can't be, an implementation. These two are both very different things. <https://hackforums.net/showthread.php?tid=4216167&pid=39996431#pid39996431>

Quote:I'll say it once again, in a more succinct context, MonoGame IS an extended wrapper for a custom implementation of XNA 4.0 for easy porting to & from XNA with cross platform features/capabilities.

MonoGame does provide more methods on top of the XNA functionality.

<http://encyclopedia2.thefreedictionary.com/wrapper>

Quote:wrapper

A data structure or software that contains ("wraps around") other data or software, so that the contained elements can exist in the newer system. The term is often used with component software, where a wrapper is placed around a legacy routine to make it behave like an object. This is also called "encapsulation" or "wrapper encapsulation," but is not the same as "object encapsulation," a fundamental concept of object technology

You're all saying that I said MonoGame specifically uses Microsoft's XNA framework. Would you mind taking your head's out of your asses and read?

Cornflower Blue ???-?— - May 10, 2014, 3:52 pm

(May 10, 2014 01:37 PM)0xDEAD10CC Wrote: [\[CITE\]](#) You still don't know how it works. It doesn't use DirectX by default unless you are writing a metro app. They had the OpenGL template's much before DirectX was even used.

<https://hackforums.net/showthread.php?tid=4216167&pid=39996431#pid39996431>

MonoGame does provide more methods on top of the XNA functionality.

<http://encyclopedia2.thefreedictionary.com/wrapper>

You're all saying that I said MonoGame specifically uses Microsoft's XNA framework. Would you mind taking your head's out of your asses and read?

Listen, you are wrong and you know you are, now I am leaving this argument because people like you make me sick, just learn to accept your mistakes.

0xDEAD10CC - May 10, 2014, 4:43 pm

(May 10, 2014 03:52 PM)lhendrey4.5 Wrote: [\[CITE\]](#) Listen, you are wrong and you know you are, now I am leaving this argument because people like you make me sick, just learn to accept your mistakes.

You both twisted my wording around on my initial statement, I know what I meant, and so I'm not seeing your logic here.