

Re: Changes in 2005.

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2005-07/msg00910.html>

- *From:* "Dan Bass" <Not Listed>
 - *Date:* Thu, 7 Jul 2005 09:14:22 +0100
-

The client base I currently have makes transitioning to new technologies difficult, and I suspect my client is not unlike many other global large scale companies hesitant to roll out the latest Windows AbFab server edition and frameworks at the drop of a hat.

In a real world example, there's a standard build that's busy being rolled out over the UK which is Windows 2000 with SP4. This has been going on for the last 9 months, and there are still NT 4 systems out there. I'm new to my firm and writing up .Net windows services, web services, asp.net web pages etc is completely new for my client. .Net has been out for nearly four years already and it's still "brand new" to many large scale companies who's strict procedures don't allow the servers on which my software will run to be updated as the patches/upgrades are released. Bringing out a new set of great tools, such as VS.Net 2005, and saying to me that I can't really use any of the little quirky tools seems a little odd, all because the people I'm writing software for won't cotton on to this trend (and you can't really blame them) of updates as they're released. I think Ford would be shooting themselves in the foot bringing out a great car with a fabulous engine that runs on some fuel type that's going to take a few years to actually get into your local petrol (gas) station. If it was released as an option (with a petrol 2.0 16v engine as standard!), however, with great promises of saving the environment, costing less, then people would still buy it, but having a normal petrol tank would more than double the sales because it's using an existing fuel source.

I've not explored smart applications all that much,. but from a security point of view, maintaining one application on one server is far more attractive to us than having to try to roll out different version of applications to the 50+ servers Europe wide. Then in terms of personal desktops with in the company, the security's even tighter, trying to get an application on these machines takes months, not to mention no one's ever quite sure about the GPO's and access rights the user has on that particular build of the OS. Web pages suddenly make so much sense because every Windows machine comes with IE. I'd say around 75% of what we've done over the last 5 years has been web site development. Generating reports (yawn), moving goods around the factory, updating terminal screens, scanning all sorts of "stuff" for receipts, processing, transformations, printing of barcodes etc., all

Re: Changes in 2005.

goes through IIS. There's one box to worry about per site, and we normally have some sort of access to it.

Anyway the point to my droning is I read the article and thought that someone had penned down something we'd been focussing on and practicing for the last 5 years. Windows development is where I've come from, where I've earned my bread and butter as it were, and what I have experience implementing with, but moving from home desktop user to corporate network applications made me realise not everyone can throw an upgrade CD in a drive, or download that update, when you realise there's something that needs to (would be nice to) change. The way in which ASP.Net is now exposing the .Net GDI for example, and reshaping the way in which the backend works, I'm finding the windows and web development gap closing really fast.

"Nicholas Paldino [.NET/C# MVP]" <mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message news:uTR%23NikgFHA.2472@xxxxxxxxxxxxxxxxxxxxxxxx

> Dan,

>

> I don't think that expecting VS.NET 2005 to compile for the .NET 1.1
> runtime is reasonable. Just like with VB3/4/5/6 etc etc, you couldn't
> compile an application for a previous runtime. The only environment that
> I know of that allowed this is C++ (but I'm not completely sure about
> this), since there was no runtime to speak of.

>

> VS.NET 2003 does not compile to unmanaged code. For all managed code,
> the JIT compiler handles the compilation to unmanaged code from IL. The
> only unmanaged code that VS.NET can create is unmanaged C++, which VS.NET
> 2005 supports as well.

>

> I see it the same way I see it as the upgrade from VB5 to VB6 (I'm a
> bit young, so I wasn't around for many other major shifts like this). The
> way I see it, yes, the client has to upgrade from .NET 1.1 to .NET 2.0,
> but with Windows Update cramming the new runtimes down your throat (and WU
> will push 2.0 when it is released), as well as the fact that different
> .NET runtimes can run side-by-side, I'd MUCH rather take .NET over
> previous runtimes.

>

> I took the time to read the article posted by Mark Jerde. While I do
> think that Joel is on crack (at least when he wrote the article), to his
> credit, the article is over a year old. There are too many instances in
> the article where he lists a specific problem which can be specifically
> addressed with .NET solutions (or other solutions not .NET related even,
> just .NET makes them very easy). An example of this is his statement
> about web applications not providing the rich UI experience. Smart
> clients are a way to do this. It's not easy (but it's not overly hard
> either), it's just new, and people are too afraid to come out from under
> the protective blanket they THINK they have. While this isn't the cause
> for everyone to resist new ways of doing things (I do think ROI is a valid
> reason to not move to new technologies whenever they appear), it certainly
> is a big one, IMO.

Re: Changes in 2005.

>
> There are also a few areas in the article where he is outright wrong.
> One of the biggest errors is where he states that WinFX is a replacement
> for the Win32 API. This simply is not true. It simply provides new
> functionality. SendMessage and CreateFile are not going anywhere (the
> framework wouldn't work if that was the case). This goes back to people
> not liking new things, for (personal) security reasons. What's
> interesting is that I don't see Joel blasting MS for when they used COM to
> wrap many API functions, as well as new functionality in the OS which is
> provided only through COM interfaces. I also find it interesting that
> there is no mention of the expansion of the Windows API (for example,
> transactional NTFS), preferring to act as if there is none.
>
> He goes on to quote Paul Graham:
>
> There is all the more reason for startups to write Web-based software now,
> because writing desktop software has become a lot less fun. If you want to
> write desktop software now you do it on Microsoft's terms, calling their
> APIs and working around their buggy OS. And if you manage to write
> something that takes off, you may find that you were merely doing market
> research for Microsoft.
>
> Unfortunately, he is using a statement about the environment a decade
> ago to justify his perception of migration patterns now. This doesn't
> work. .NET wasn't around 10 years ago, and writing these kinds of apps on
> windows platforms was difficult, which is why HTML was preferred.
> However, if you are going to tell me that you would rather write an
> ASP.NET app (or web app) over a smart client app (especially with
> ClickOnce on the horizon), I would pretty much say you are out of your
> mind. God forbid if you had to display information of a hierarchical
> nature in a web app.
>
> Anyways, I could go on, but it would be more of the same. The article
> is interesting, but I put it in the same camp as the people petitioning MS
> to continue supporting VB6 (here's a hint, it's not the camp you want to
> be in). This isn't to say I think VB6 is bad, but if I had a choice to
> develop between that and .NET, the choice is easy for me (and yes I know
> the people petitioning MS for VB6 (sorry, VB.COM, my mistake) are doing it
> for the hobbyist programmer, I don't see VB.NET being that much of a
> different thing than VB6 (at least to the extent that the hobbyist
> programmer uses it for)).
>
> Flame away! =)
>
>
> --
> - Nicholas Paldino [.NET/C# MVP]
> - mvp@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
>
>
> "Dan Bass" <Not Listed> wrote in message

Re: Changes in 2005.

> news:O6gxUpjgFHA.2424@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

>>

>>> In the end, APIs will always change, and it's not always possible to
>>> make the new versions work with the old declarations. MS has always
>>> been committed to backwards compatability, so I am not surprized by them
>>> taking this approach as .NET 2.0 comes out.

>>

>> Would be nice if VS.Net 2005 compiled to .Net 1.1... After all, 2003
>> compiles to unmanaged code, all be it the code needs adjusting to get it
>> to compile.

>>

>> I find the new IDE great to use, but don't necessarily want to insist on
>> the client having .Net 2.0.

>>

>> Do you have any comments on that article posted by Mark Jerde? I found it
>> most interesting.

>>

>

>

.

• **References:**

◆ **Changes in 2005.**

◇ *From:* Mark Broadbent

◆ **Re: Changes in 2005.**

◇ *From:* Nicholas Paldino [.NET/C# MVP]

◆ **Re: Changes in 2005.**

◇ *From:* Dan Bass

◆ **Re: Changes in 2005.**

◇ *From:* Nicholas Paldino [.NET/C# MVP]

• Prev by Date: [Xml problem](#)

• Next by Date: [Re: Changes in 2005.](#)

• Previous by thread: [Re: Changes in 2005.](#)

• Next by thread: [Re: Changes in 2005.](#)

• Index(es):

◆ [Date](#)

◆ [Thread](#)