

Re: Microsoft MVPs Say They Want Old VB Back

Source: <http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.general/2005-03/2014.html>

From: Jim Hubbard (*reply_at_groups.please*)

Date: 03/17/05

Date: Thu, 17 Mar 2005 17:16:24 -0500

Hi Gerry,

>> *Again, you are right. But, who is really in the market to produce web*

>> *applications?*

>

> *In my view, these are the only apps to bother with. The problem for me, is*

> *that people all over the world need to use my apps at anytime of day or*

> *night – they all have different operating systems Windows/Mac/Linux/Sun*

> *and different browsers IE/Safari/WebTV/Mozilla.*

I agree....to a point. Web applications are definitely preferable to the installation headaches of multiple user configurations – especially when looked at globally. However, web applications lack the power of desktop apps if you are talking about a graphics-oriented application or very rich UI and they lack the ease-of-use that you can achieve by settling on one desktop and coding to take advantage of said desktop OS (whether it be XP, OSX or Linux).

>

> *I did all the internal company stuff in 3 tier web apps, and it's great*

> *because it works just the same whether they're at their desk or on the*

> *other side of the world. I also never need to worry about re-installing or*

> *patching every time someone buys a new computer/laptop – they just type in*

> *the URL and off they go.*

That works fine for databases, simple text and financial applications, but a rich user interface that could be built in a RAD environment (until .Net) was not really possible on the web. Of course you can do Flash or Shockwave to create rich UIs, but that takes more art talent than most programmers have on tap. And, they are not exactly RAD tools.

>

>> *The web is unstable.*

>

> *Uh?? I wonder if Google and Amazon are aware of this fact?*

Sure they are. That's why their UIs and applications are so simple (when compared to an N-tier enterprise application).

>
>> *If your internet connection is down, what happens to the web-based applications you need to do your job?*
>
> *Well you could argue "email is unreliable" because "what if the internet is down".*

But, email is hardly a component that will shut down a company if it isn't working at the moment. On the other hand, if you use a web-based point-of-sale terminal to ring up your customers and the internet connection is lost – now you have a problem.

While Amazon can handle this down time (if it doesn't last too long), it is an entirely different matter for a sales clerk with a customer standing in line at a small business to deal with. It is also far more financially devastating to the small business owner to be down for a day than it is for Amazon or Google.

>
>> *So, what's the big deal with .Net? It's a little long-winded, but check out my blog at <http://poderthis.blogspot.com/> and I'll show you.*
>
> *I don't really have time to read blogs, but can you give us a short summary?*

No problem.....

- 1) .Net was not written for you or I, it was written by and for Microsoft to further their "software as a service" goals.
- 2) .Net is using the Microsoft programming community as the world's largest beta test group for its new programming tools that it will then use to wring more money out of those same users by changing to "rentalware" instead of software as we understand and purchase it today.
- 3) .Net does have value for those individuals that want to do "software as a service" – but they are a minority at this point.
- 4) Microsoft doesn't care what's good for programmers outside Microsoft or the world IT community in general. It ignores the pleadings of its customer base and can do so simply because it has become too powerful. (Breaking up Microsoft is still a good idea.)

>
>> *You could use them with C++, Visual Basic or Borland's C++ or Delphi.*
>
> *Let's not forget JScript.*

Ahhhh.....sorry about that.

>> *In order to have true interoperability across languages, there is a significant trade-off in security.*
>
> *Security is NOT the right word to use here!*

What is? I consider protecting my intellectual property security.

>
>> *Not the security of the environment, but the security of intellectual
>> property rights. In .Net, it is quite easy to disassemble another
>> person's work and steal their ideas and code – even if an obfuscator is
>> used. Binary code was much more secure in this aspect.*
>>
>> *This one flaw in .Net virtually forces you to make your applications
>> web–applications to hide your proprietary code on the servers.*
>
> *I see it the other way round. I have all my stuff coded up in JScript; 3
> tier apps, Enterprise management, Change and Configuration management,
> Software deployment (GPO Alternate), and W3C compliant front–ends. If
> anyone wants the source code, just ask, or find me on groups like WMI. I
> post it all the time. Why should I care who can see the code? The way I
> see it is we should all post our code, then we learn from others.*

Do you post the source for your entire application (all n tiers)?

Most companies will not. It is not in their best financial interests to give away their corporate model or source code.

Open source – as a model – will fail. You can already see it with Linux. Linux has fragmented into so many distros that software written on one distro (especially if it takes advantage of hooks or APIs built into that distro) will not run on many others.

This is where Linux went wrong (and where Java is soon to screw up). Even Microsoft talking about opening up the source code is a big mistake. But, there is a right way to accomplish the goal of interoperability without sharing the source code.

Simply share ALL of the OS's APIs. If Linux had opened ALL APIs to developers and kept the source code patented for its use and the use of its customers only, Linux would not be fragmented into so many distros and would have already put Microsoft on the ropes through the use of more simplification through APIs and hooks into the desktop and underlying kernel.

This would've also allowed Linux to grow much faster because all Linux developers would be working on the same desktop – making the market of supported applications available much larger than it currently is. (Yes, I know there are thousands of Linux titles out there, but there aren't nearly that many that have professional teams backing them up – and that is required by any serious company. Freeware just doesn't cut it in most business scenarios in the US.)

If Microsoft would open up ALL of their code's APIs, they wouldn't need to show their source code and people could write more compatible code to work with the OS and Office suites. And, that's all developers really want....to

microsoft.public.dotnet.general: Re: Microsoft MVPs Say They Want Old VB Back

have a level playing field with Microsoft APIs. We don't even need the source code.

Open source leads to fragmentation of the application and user base – resulting in higher application costs in the long run.

This is just my opinion, of course.

Jim Hubbard