Original Post Comment Archive

Comments

Original URL: http://team.silverlight.net/announcement/the-future-of-silverlight/
As of: 7/8/2011 11:37:46 AM

1. May-28-2011 at 09:29 PM PST

   rikerion

   I have programmed in many languages and .NET/C# is outstandingly the best. I'm also a great fan of Silverlight. Having a strongly typed language like C# and a powerful declarative language like XAML for my UI, running in a browser or OOB is two orders of magnitude better than any alternatives.
   Having said that I fear that both technolgies will be relegated to niche status. I am dismayed at Apple's success with the iPhone (everyone I know has one!) and the succes of the iMac (my wife and daughters all have one!) (because the're cool and easy to use and don't have viruses etc.). I wouldn't mind so much but Objective C is a nasty langauage and I think I would choke on my own vomit if I had to use it (I am convinced Steve Jobs is no programmer or he would be equally appalled).
   Even though Nokia are switching to WP7 I can't see that beating Apple and Googl'es momentum in this area. Since Oracle are suing Google over their use of Java why not do a deal with Google. Even better put the whole C#/.NET/XAML technology set into open source or into a separate company with shared ownership by the major players.

2. May-11-2011 at 02:12 PM PST

   XAML Lover

   I'd rather bore a hole in my head than return to ASP and HTML with all of it's bolt-ons.

   That said, SL must reach other devices to be widely successful. If Android and iPhone do not get SL, it will be niche for W7 phones. LOB Apps will likely stick to WPF or XBAP. While it can survive on W7 phones alone (Objective C anyone?), but that'd be a great disappointment.

3. May-09-2011 at 07:06 PM PST

   Confused Developer

Microsoft sends mixes signals regarding the future of Silverlight and that certainly doesn't help with the platform's adoption. I wonder what the implication of the recent reorg (ScottGu heading the Azure team) are?

4. March-30-2011 at 04:47 AM PST

   James

   Listen i know there is all this hype around the web concerning html5 and what it can do. Yes alot of it is cool stuff. However, we are still going to have to leverage each browser with html as different browsers will have different html5 features. So alot of issues with html 4 programming will remain. This is why my company is sticking in with SL over the long haul. Being my team are .Net folks we are enjoying our apps working the same in both browsers and we are not concerned about alot of the tedious issues coding for css and html in multiple browsers present. Oh and by the way the standards are not even complete yet.

5. March-22-2011 at 09:50 AM PST

   Treadmill Traci

   Yes we have been spoiled. Now we only want more, bigger, better, faster technology. Nothing else will do now that we have such great ammenities. Technology has to continuously improve because that's what people expect.

6. February-04-2011 at 08:26 AM PST

   James Hancock

   Brad, this is excellent, but you're addressing the issues in a vaccum. The future of silverlight is related to Windows, WP7 and IE. What we need for any confidence in Silverlight, we have to see the grand plan. You guys need to show that you have a grand plan and tell us what it is. Until then, Silverlight is simply too great a risk for us.

7. November-19-2010 at 07:52 AM PST

   Simon

   How about building SilverLight functionality in to IE9. The Canvas tag is empty and manipulated by JavaScript procedurally. The Canvas tag could contain XAML or a link to it, in the same way the tag Script, holds JavaScript. Microsoft could use its influence on the HTML committee to argue for better use of Canvas. Their are many complaints circulating about its lack of accessibility. A XAML object tag inside the canvas tags could be manipulated by JavaScript or even .NET background code. Other browser vendors would be free to ignore it if they so chose. XAML is one of the best parts of SilverLight and it would solve one of Canvas's biggest negatives.

8. November-11-2010 at 07:30 AM PST

   Annoyed silverlight developer

   Brad this article is great and I agree with a lot what is said, I too had great commitment and faith in Silverlight and saw it as a real future technology and appeared to have good momentum.

   However can you please have a chat with Bob because he does not have the same faith as both of us. He and Mary Jo have killed whatever SL momentum there was and definitely was not on the same page. Can you get him to understand the difference in HTML5 and SL, and SL was on a very good direction before it got re-directed to the trashcan?

   Meanwhile I'll go re-train as a HTML - jQuery develeper.

   Thanks
   Former SL developer

9. November-03-2010 at 12:38 AM PST

   Re: Frederic Dumesle

   The last thing I want to see in my life is open source SL where every idiot is putting there own junk and they call it "standard". One guy wrote that open source is not free, it comes with a religion and I agree with it.
   But there still remains many questions regarding SL and it's roadmap. Currently many devs are starting to accept SL as there primary development platform for business applications. I'm one of them. Just bought kindle to read SL books.
   And just like many other MS developers, I want to know the future of SL.
   MS should come up with a clear answer, time is moving on.

10. November-02-2010 at 09:54 AM PST

   Frederic Dumesle

   Hello,
   I think MS will lose the Tool Dev Technology in the future unless they outsource the SL engine and .NET altogether! Adobe is successful at porting AIR on multiple devices
   If MS manage to get SL on Android/RIM/Chrome/iOS they are a force to be reckon with.

   Otherwise they will utterly fail! This is sad

11. November-01-2010 at 11:22 AM PST

   Brian Goldfarb

I've seen some folks asking for more clarification about PDC -- check our our most recent post.

http://bit.ly/brpkiX

12. November-01-2010 at 05:42 AM PST

blaize

ZDNET posted an article about HTML 5 and Microsoft's commitment to it and Silverlight...

http://www.zdnet.com/blog/microsoft/microsoft-our-strategy-with-silverlight-has-shifted/7834

I'd be curious to hear about the SL Team's response to this....

13. October-31-2010 at 07:38 PM PST

Mizoy

OK, can you explain this "Microsoft: Our strategy with Silverlight has shifted" : http://www.zdnet.com/blog/microsoft/microsoft-our-strategy-with-silverlight-has-shifted/7834

14. October-28-2010 at 05:31 PM PST

Marc Roussel

Silverlight is very important in all the aspect of my life since April 2007
This date changed my life almost entirely. At that time I was begining with ASP.NET and I was about to quit the world of computing because of that.

Then sudently, a MIRACLE, I had the chance to see the new baby called Silverlight 1.0 which enlighten my soul and showed me what I was waiting for since ASP
The kind of development that could bring my usual windows form application to the world. Done, Thank you so much for this technology that gives me bread on the table every day as I live of it. That's truly amazing everything we can do with Silverlight today and how easy it is. I love it so much.

Thank you a lot again and I target all the people that work on this since the begining.

15. October-24-2010 at 03:42 PM PST

Chris

Silverlight is a great technology and I have built three LOB-style applications in, respectively Silverlight 2.0, 3.0 and 4.0. But one thing that would make it more successful is to imbue it with a greater level of compatibility with WPF. There are so many nice things in WPF that simply aren't there in Silverlight and what didn't make the cut seems arbitrary. It would save me and a lot of other LOB developers out there a considerable amount of grief if the WPF and Silverlight teams got into a room and decided on a single API surface to make available whenever feasible. In fact, I'll embrace the mantra that I've heard suggesteded by others: define a client profile and treat Silverlight/WPF technologies as an implementation detail that can be selected by the compiler. I'm really tired of #if !SILVERLIGHT #else constructs littering my code and not being able to access important functionality (think data triggers, custom markup extensions, adorners....) in Silverlight that exists in WPF

16. October-14-2010 at 12:18 AM PST

    Valery

    Create a business model that let's Apple provit from providing XAP files from their App Store and let the rest of the industry provit from all the benefits Silverlight has to offer. We are already convinced Silverlight is superior to its direct competitors but as always the winner (do we really need that?) will be the one with the largest installbase (on any device). I agree HTML 5 could perfectly co-exist with Silverlight just like HTML4 and flash do.

17. October-10-2010 at 12:01 AM PST

    argarcia

    Woah! nice. I'd like to try this new apps. Thanks

18. September-27-2010 at 05:15 AM PST

    Silverlight User

    Szindbad you must not be to bright if you think Silverlight is too complex. It is easier than HTML/Javascript/CSS stack. You must be just lazy or not to bright.

19. September-23-2010 at 06:59 AM PST

    ManishKG

    Great Article! The standards/innovation cycle is illustrative enough to predict Silverlight feature. Historically as soon as there has been a www standard, the types of Internet applications needed more and this will continue (for good).

20. September-22-2010 at 12:32 PM PST

HappyFun

Does the future of Silverlight include support for 64bit browsers?
With MS's help, Adobe was able to launch 'square' (the beta flash for 64bit browsers) to coincide with the launch of IE9 Beta, but there is still no 64bit version of silverlight?

21. September-17-2010 at 04:22 AM PST

    Szindbad

    My 2 cents: Silverlight it is too complex. And because of complexity, it has bad learning curve, slow development, and most importatnly: slow running. I'm sorry to say, but you have been over-engineered it (http://en.wikipedia.org/wiki/Overengineering). That seems fatal mistake.

22. September-16-2010 at 09:40 AM PST

    Slash FX

    The proliferation of WPF on the desktop will help Silverlight. Microsoft really should be pushing the parent technology internally themselves; it's a shame Internet Explorer 9 wasn't done in WPF, but Visual Studio and Expression stand as shining examples of what can be achieved. Now if the majority of Windows applets, maybe even Explorer itself, were rendered with WPF and allowed users to plug-in custom skins and effects, maybe in Windows 8, the resulting widespread familiarity with XAML would only encourage developers to use Silverlight on the web end as well.

    In short, MS should "dogfood" WPF itself to begin with, as much as possible! XAML's "success" means Silverlight's success!

    (P.S. I had to post this comment from Firefox 4 because IE9 wouldn't let the Publish button click)

23. September-15-2010 at 05:11 PM PST

    R.D.Rush

    HTML 5 has a long way to go before it is anything comparable to Silverlight. HTML in any version is a tool to present static data. The objects that provide for media and dynamics are merely containers for said items -- end of story. Silverlight is MS answer to the need for serious, focused, direct AJAX and an easy means to navigate and construct frameworks in that technology -- period. The comparisons will be short lived through the means of "proof of concept" as web-developers will soon realize that you can do anything that HTML 5 can do with Silverlight 1 let alone 3 & 4 but, HTML 5 cannot do everything that Silverlight can do. Why? HTML 5 is for HTML where as Silverlight is web 2.0 applications development from concept to finished product. Flash and Silverlight are also apples & oranges comparisons -- they are two entirely different animals It is all competitive ignorance.

I posted this "opinion" because I know that by 2011 Silverlight will dying off for the ten-thousandth time and HTM 5 will be the next Armageddon to all things web-based. There are allot of bored, misdirected, and highly marginally accepted mind sets with far too much free time on their hands. Just read today, as a matter of fact, that Silverlight was on its last leg -- yeah, and I'm a gatrillionaire with a flying monkey for an uncle.

Vlad H has his stuff together:

Strongly typed languages cannot exist without professional, competent, security minded development teams like the staff of SL.

Application state is where everything intersects and where you will find 90% of your critical bugs and security flaws/vulnerabilities. State is where functional code touches workflow -- true??? Screw that up and OUCH!

I learned the majority of what I know listening to real industry leaders and am no expert by any stretch but, I know that it is not what you rather than what you are capable of learning.My feature recommendation (wish) supports this philosophy and it is an ideology that MS and dev-staff have helped establish and strengthen in my existence and professional pursuits. Thank you all and many thanks to my fellow developers of all levels here at the blogs.

24. September-15-2010 at 04:27 PM PST

nhwilly

H-T-M-L B-L-O-W-S. I've never seen so much incompatibility in a platform so widely deployed. The cost of developing apps for it is truly astronomical if they have any depth and don't get me started on Adobe. Priced any of their nonsense, lately.

Ever notice how Illustrator and Photoshop have so little in common in the UI and how the Windows versions are the red headed stepchildren?

End users want stuff that is reliable and renders a usable experience. Yeah, I like the iPad, too, but it's so consumer driven. If you have any heavy lifting to do, you always end up wanting a tool that makes you productive and compatible. Visual Studio rocks. So does SL and WPF.

What would be awesome?

1. Compatibility between SL and WPF. Clear direction on whether they'll outlive our development timelines.
2. Some meaningful way to convey to the lightweights that this isn't an "also ran"

technology (which this article really helped - thanks very much for that).
3. Rock solid stability. Kill the bugs. No one has time to debug someone else's platform.

Great article - I just wish more people would read it.

You guys build awesome stuff, whiners be damned.

25. September-09-2010 at 08:07 AM PST

Stefan Wenig

Is that what it's about? WinDiv pushing HTML5 as the new UI platform against DevDiv's Silverlight? If that's the case, my best wishes to DevDiv, but my money...?
http://twitter.com/MossyBlog
Just rumors? Good. What's the story then?

26. September-08-2010 at 09:07 AM PST

Vlad H

The biggest two issues of web development that SL solves are application state and strongly-typed programming language on the client. Think of all the grief caused by regular web applications having to drag application state up & down the wire (ViewState of 500K), or save it on the server! With SL there is no need to generate UI on the server and waste bandwidth on state and UI data traffic - only actual application data travels after the application is downloaded. It's really great. The same goes for the ability to write client code in a sophisticated type-safe language, compared to the cruel joke of JavaScript. So you gather that as a developer I love SL.

But, SL will be a niche, video & LOB app development tool if it loses ubiquity competition to HTML5. SL ubiquity already has a dark cloud over its head: Steve Jobs will not let SL on iPhone/iPad if he applies the same logic to SL as he did to Flash, and Google has very little incentive to put SL on its Android too. If SL won't get on Android, that's pretty much it for SL as a consumer application development tool, even if due to some miracle Windows Phone 7 turns out to be a success. To put pressure on Android and Apple, SL has to be on all popular Windows, Mac and Linux browsers, as well as on Blackberry. If SL ubiquity remains limited to mostly IE, Windows and some limited number of browsers on Mac and Linux, then despite being a better development tool and having superior feature set compared to HTML5 & JS, SL won't get much traction.

27. September-08-2010 at 02:57 AM PST

mikej

Comment stating the case even stronger than here!

http://www.i-programmer.info/news/89-net/1234-silverlight-better-than-html-5.html

28. September-08-2010 at 02:39 AM PST

    Ben Acheson

    This is very exciting. But in spite of being HTML-based, silverlight has a long way to go towards gaining an SEO-friendly reputation.

    I think Microsoft needs to to more to market the product to the SEO and digital marketing communities.

29. September-07-2010 at 06:08 AM PST

    Tatham Oddie

    Discussed on this week's Frankly Speaking, from 21 min in:
    http://www.noisetosignal.com.au/franklyspeaking/?p=256

    :)

30. September-06-2010 at 01:16 AM PST

    H_D_W

    Any News on the Next version of Silverlight?

31. September-06-2010 at 12:57 AM PST

    Stefan Wenig

    nice discussion about this blog post:
    http://www.infoq.com/news/2010/09/Silverlight-Web-Apps

32. September-05-2010 at 08:06 AM PST

    joewood

    Two interesting comments from above. I hope we hear more about these in the near future:
    "Silverlight as a GWT competitor" - to address the reach issue.
    "ECMA Script in the browser" - to address the capabilities of the client.

33. September-04-2010 at 05:03 PM PST

    andre

i ned a telesculp

34. September-04-2010 at 03:06 AM PST

    Miro MIroslavov

    Perfectly structured and comprehensive post. Thanks for it.
    One of the most important things about Silverlight for me is that developing apps/controls with it, is making me really happy. It's platform paradigms are much more natural and close to my vision about software developing. It leaves very good impression in most developers that I've talked about, because of it's programming model, data-driven development, ease of customization and so on...
    Yes, there are bugs, memory leaks (this is quite bigger problem) and so on, but if somebody can point me a bug free software...?
    On a side note, I can not imagine doing an "Application" with JavaScript.:)

35. September-03-2010 at 04:08 PM PST

    Michael Gautier

    This is well written article but too many people are having issues with Silverlight bugs popping up with Mouse events and RootVisuals to name a few. For a simple video player, Silverlight works well, but for more sophisticated scenarios, it can fall apart when least expected. Silverlight is approachable, but I cannot imagine people adopting Silverlight if there are latent flaws. However, even with better construction, the real issue is that millions of customers, users, or web visitors own and are using physical devices like the iPad, iPhone, and Droid. They do not care what technology an application is built in. They want a great application and Silverlight could deliver if it worked on these platforms. Otherwise, Silverlight is poised to become the next generation ActiveX. Good for internal business applications and a limited corporate audience, but risky for anything purporting to be accessible across the internet in the many ways people are, today, accessing it.

36. September-03-2010 at 02:28 PM PST

    grundt

    One thing that baffles me is why Microsoft so often chooses to use Flash for content on it's own website.

37. September-03-2010 at 11:12 AM PST

    Fernando Caverna

Standardize and Innovaction, simple and perfect.. I like how this blog makes greats explanations with simples perspectives. I would like see more comparisons about Silverlight and Flash technologies.

38. September-03-2010 at 05:07 AM PST

David Bolton

Silverlight development is restricted to those who have the full Visual Studio version. Encourage new developers to adopt it by making it available for free.

39. September-03-2010 at 05:04 AM PST

Stefan Wenig

Incredibly, this blog post misses the single feature that makes me wish that Silverligth (along with its OSS cousin Moonlight) might prevail: developer productivity!

No matter how many people I talk to about this specific topic, everyone seems to agree, yet nobody seems to talk about it publicly: from a developer perspective, the web has become a mess. It's part historic, part design-by-committe; a strange mix of technologies, intended to build content, not applications.

We're all held hostage by the W3C and an outspoken community, and nobody stands up to counter them: we want a clean programming model to build our apps on. Not layers above layers of abstraction, not n platforms that are compatible for the most part but still require separate testing and a tweak here and there, but a clean new start, thank you very much.

For Google et.al. this won't matter to much. In MS terminology, reaching every last user is so much more important to them. For enterprise apps, it's usually the other way around. It does matter whether development takes twice as long or more. It does matter whether I need to employ additional QA people for every new browser I need to support. Having clients install Silverlight is relatively simple. Everyone else's milage may end up somewhere in between.

In our own OSS project, re-motion, we used Silverlight for graphical workflow editors, relation designers, and more. Many developers see this and shout out "finally, a SL app that makes sense!". But none of this is impossible today with HTML (if you include SVG, which will be in IE9). It was impossible when we started, but now everything is there or will be soon enough. Still, I'd rather shoot myself in the foot than do it all in HTML. In fact, I believe I'd have to shoot myself in the foot to get there.

That's the argument I'd like to hear from Microsoft. Otherwise it's all about arguing whether feature X is important enough to leave the ground of established standards, and you'll lose that debate more often than not - if it even lasts until HTML catches up.

Existing knowledge won't buy you anything either, there's so much existing HTML knowledge out there already.

I believe Microsoft should be brave and visionary here. If you guys, along with the Mono team, offer an open, patent-free platform that co-exists with browsers, that lets people mix and match XAML and HTML, IL and JS (without interop performance penalities) people might listen.

Otherwise Silverlight's use will be limited to video and enterprise apps, and eventually people will start to wonder how long MS is going to fund this venture for the limited outcome it's getting.

related post: http://tirania.org/blog/archive/2010/May-03.html

40. September-02-2010 at 10:30 PM PST

    Janki

    Real nice post. Well Silverlight was never meant to be a "SILVER BULLET" like HTML 5. No matter how much HTML specs catch up, they can't be made specific enough. Silverlight and Flash will continue to thrive.

    Some of the features need much more work. You mentioned Tele-conference with webcam and Microphone, It's not elegant now like Flash has. For Business Applications : PDF support is needed. I am not talking about dummy PDF like 3rd party vendors are offering, I am talking about building it in the framework itself. Dummy PDF controls from 3rd party vendors don't handle PDF form filling/ Signature handling etc (because they just scrap out page as an image).

41. September-02-2010 at 07:52 PM PST

    Marcelo Negrini

    Wouldn't be possible in the future to use the Silverlight tool set to design/develop and then deploy as HTML5+JavaScript? Something like Google's GWT. It could be just as a fallback, with less features. If Google does that with Java (argh!), I bet Microsoft can too with Silverlight.

42. September-02-2010 at 05:34 PM PST

    rei miyasaka

    I find myself saying this every time someone at Microsoft asks about feature wishes -- you need better marketing. Less 90s vibe. Less gimmicks. Less mall music. Less alienating. Less targeting end users and management and more targeting developers.

It's unfortunate though, I think that a lot of the reasons people don't want to do Silverlight, aside from the stigma inherent to being a Microsoft product, is because the web has spoiled them into developing some strange habits.

For instance, there's a certain expectation of statelessness that website audiences have and developers exploit. It doesn't bother people that they have to hit the refresh button in a browser to update data sometimes, but when in a desktop application you do (e.g. in SQL Management Studio), it's really awkward. Silverlight probably suffers that same problem too, unfortunately, and I'm not sure there's anything that can be done about it.

Also, you typically don't transport XAML at runtime, but you do routinely transport HTML as part of Ajax -- which I think is bizarre, but it's an option that the HTML culture allows and the desktop development culture considers a taboo. That's another unfortunate dead-end.

Event-driven applications are just more difficult to plan out too, I think. There isn't that implicit/vague layer isolation between view and everything else; you sort of have to learn MVC/MVVM on your own. Which is fine for small projects, but I think it still leaves people with a strange aftertaste when they don't know whether or not they're doing it right.

You hear people talking about how the world is/isn't ready to part from Flash and Java and all that, but I think it's somewhat the other way around -- people just aren't ready to part from browsers.

Wow, that's a long post.

43. September-02-2010 at 05:21 PM PST

Tron++

It's nice to see the Silverlight team's side of the HTML5 Is The Future story. Not sure why you waited until 9/1/2010 to provide your perspectives on the future of user interaction with the web, which is exactly what all of this is about. For developers, there is no comparison between the numbers of people who can write HTML-based apps versus those who target the CLR via Silverlight or any other .NET platform technology. Silverlight will be a niche technology in the future, as you sort of allude to. For general purpose web programming, HTML, CSS and JavaScript will rule the web landscape (just like it does today).

As browsers become more sophisticated and capable, the need to write code that runs in a virtual machine like the CLR or the JVM or Flash hosted in what is essentially another virtual machine (the browser...) becomes essentially pointless. HTML5 promises to be a game changing platform, but the game has only just begun...

HTML5 and Silverlight are just two different software development tools with two very different goals... Why compare a hammer to a drill? Whereas you can in theory hammer a nail into a piece of wood with a drill or bore a hole with a hammer, why would you when you can use the right tool for the job?

Microsoft and the industry at large need to stop focusing on This or That is The Future of... In reality, the application platform world is a melting pot of various spices yielding numerous flavors, not a one ingredient recipe.

HTML5, Silverlight, C#, VB, C, C++, Ruby, Python, PHP, Java... Think outside the toolbox.

Finally, the future of Silverlight does not depend at all on the direction taken by HTML, CSS and EcmaScript(JS). This article is therefore mis-titled and, to be honest, not very enlightening or useful. What's the future of Silverlight, again? I think you missed that part.... You don't need to defend the lawnmower in your shed just because a new buzz saw you purchased has increased power and precision over earlier versions...

End of line.

44. September-02-2010 at 03:36 PM PST

Ethan B

Brad -- I know you guys have to put this out there given the HTML5 hype. I should note I am a big Silverlight fan, but I do have contention over one bit here.

I think it's a bit misleading to compare SL perf to Chrome V8 when the IE9 team has been hyping the DirectX-based hardware acceleration and their new Chakra JS engine.

Since I can't run the test myself without knowing the hardware / actual test you used, care to update the graphic with IE9 framerates for the same test for the sake of comparison?

45. September-02-2010 at 02:01 PM PST

Jan Conroy

Might be nice, but, Update can't update this #&%$! Troubleshooting can't troubleshoot it, either! And, if you aren't a "geek" you are totally screwed! Thanks :(

46. September-02-2010 at 01:57 PM PST

samcov

I have the same sense about Silverlight that I had with html. There is a real reason it's better than html, but it's not one clear cut answer.

1. The development environment simply can't be matched, and that's huge for larger projects, or as the web is going, more applications than websites.

2. Multithreading is also something that is a major feature, but it's hard to fully demonstrate it as a single defining reason SL is better.

3. Parallel processing, the ability to utilize all available cores would be nice to see in SL 5, but who knows.

In short, there are a lot of reasons, and it leads me to the feeling that if you keep evolving the platform, it will become even more special.

47. September-02-2010 at 01:57 PM PST

SilverLight4LOB

Sorry, but here is a rant:

You guys need to improve SilverLight making it suitable for LOB apps first.

Every little step of the way we have had to deal with pesky bugs such as:
1) Child Windows not reenabling the caller UI when they close (we shouldn't need to mess with Application.Current.RootVisual)
2) ComboBoxes not displaying the entity description (just the id)
3) ComboBoxes inside of a DataGrid not allowing direct binding to a different entity than the DataGrid's entity
4) SilverLight should come with its own Caliburn, Unity and MvvmLight.
5) For someone coming from ASP.NET, I am having to ask this question everytime: should I implement such and such in the ViewModel and communicate through messages with the View or vice-versa? In other words, should the View be dumb and have everything handled by the ViewModel or make it a little smarter?
6) Too many other little quirks to remember them all. They take time and cause frustration unlike any other development tool from MicroSoft that I know of.

As it is, SilverLight has a very low productivit and too slow to learn for LOB development. Mostly, it is because very little you've done and learned in ASP.NET can be applied to it.

48. September-02-2010 at 01:10 PM PST

TheLudditeDeveloper

Having seen a few articles (including a Mini Microsoft post) suggesting that Silverlight was yesterdays news because everyone would be switching to HTML5, it is great to hear from the Silverlight team.

One complaint. Scrollviewer (mouse scroll wheel) is not cross browser compatible. Does not work in Firefox. Even laptop mouse controls fail to scroll in Firefox. In fact scroll wheel crashed firefox plug-in on a dell studio laptop running windows 7 using firefox. You can verify problems on http://www.cheap-ipods.com

49. September-02-2010 at 12:04 PM PST

    Peter

    We wanted to develop one online Rss Control (with memory). The only technology to do this is Silverlight. I can't imagine how you can do this in HTML 5

    It's an open source project on CodePlex for the open source community

    http://magicrssmenu.codeplex.com/

    This is a Bing Search Robot

    http://www.aptools.net/feed/Search.HTML

    This is the VisualStudioMagazine in the Control

    http://www.aptools.net/feed/VisualStudioMagazine.HTML

50. September-02-2010 at 11:51 AM PST

    Rod Mac

    Great article on a fundamental question which must be running thro' most developers' minds. Can I please suggest a similar article on Silverlight vs WPF would be most appreciated. We've heard there's going to be convergence, but there's no sign of how that will happen and when exactly it will take place, and in turn it makes one think, well, better stick with ASP.NET and WinForms until MS get clear on this one. Today I really can't see how going down either the SL or WPF route is going to lead me to that truely converged app scene so I keep coming round to avoiding re-writing my ASP.NET + WinForms app in the first place. In fact I don't think SL can really be a web technology because Apple and Google will most likely block you, so why bother with 2 frameworks?

51. September-02-2010 at 10:57 AM PST

    msftdev

    Brad,
    Thanks for the great summary, very timely too. On the performance front, I've done my own Bubblemark testing on Flash/Flex and consistently get performance in the area of ~ 20 FPS, so not sure how you are arriving at < 1 FPS? I'm guessing that was a typo.