

مطلبی که در ذیل آورده شده صرفاً یک برداشت شخصی است بر اساس نقل قول‌ها و بررسی وضعیت اعضای تیم‌های مرتبط با فناوری‌های مختلف بکار گرفته شده در دات نت فریم ورک و ... نه رسمی!

ADO.NET , DataSet , DataTable و امثال آن: مرده! (مرده به معنای اینکه دیگر توسعه‌ی جدی نخواهد یافت)  
ADO.NET اولین فناوری دسترسی به داده‌ها در دات نت فریم ورک بود/است. مدل طراحی آن هم بر اساس امکانات زبان‌های آن زمان (زمان شروع به کار دات نت) بود (و تا دات نت 4 هم تغییر عمده‌ای نکرده). برای مثال در زمان ارائه اولین نگارش آن خبری از Generics نبود (در دات نت 2 اضافه شد)؛ یا LINQ وجود نداشت (در دات نت سه و سه و نیم اضافه و تکمیل شد). به همین جهت طراحی آن در حال حاضر (با وجود دات نت 4) بوی ماندگی می‌دهد (مانند استفاده از دیتاست و دیتاتیبیل) و با ORM های میکروسافت جهت استفاده از امکانات Generics و LINQ جایگزین شده است.  
البته این مورد تنها مورد مرده‌ای است که "باید" یاد گرفت؛ مهم نیست که ORMs ارائه شده‌اند. مهم این است که زیر ساخت تمام ORM های نوشته شده برای دات نت همین ADO.NET خام است.

#### LINQ to SQL : مرده!

مایکروسافت با این فناوری ORM های خودش را شروع کرد اما بعد از مدتی دید که بهتر است یک نسخه‌ی عمومی‌تر با پشتیبانی از بانک‌های اطلاعاتی دیگر مانند اوراکل، MySQL و غیره را نیز ارائه دهد. همینجا بود که آن را خیلی ساده با Entity framework جایگزین کرد و در roadmap ارائه شده صراحتاً EF به عنوان راه حل توصیه شده دسترسی به داده‌های میکروسافت اعلام شده است (+). حالا این وسط دیگر مهم نیست شما پروژه نوشته بودید یا هر چی. دیگر منتظر تغییرات خاصی در LINQ to SQL نباشید. فقط یک سری رفع باگ و نگهداری پروژه را شاهد خواهید بود. البته در همان زمان خیلی زود تکذیب کردند که LINQ to SQL مرده اما برای نمونه آقای [Damien](#) که عضو اصلی تیم LINQ to SQL بودند، اکنون در تیم XBOX مشغول به کار هستند! (+) تو خود شرح مفصل بخوان از این مجمل!

ضمناً این رو هم در نظر داشته باشید که LINQ != LINQ to SQL ؛ به عبارتی LINQ به خودی خود فقط یک language feature است.

#### Windows Forms یا به اختصار WinForms : مرده!

به نظر مظلوم‌تر از این یکی در دات نت یافت نمی‌شود! همین چند وقت پیش یکی از اعضای میکروسافت این نظر سنجی رو برگزار کرده بود که "ما چکار کنیم که شما راحت‌تر از WinForms به WPF مهاجرت کنید؟!" (+)  
در قاموس WPF ، ویندوز فرمز یعنی Canvas panel ؛ به عبارتی صلب‌ترین حالت طراحی رابط کاربری و این انتقال و مهاجرت هرچند برای کسانی که عمری را روی آن گذاشته‌اند، دردناک خواهد بود اما با وجود توانایی‌های WPF چیزی را از دست نخواهند داد. سیستم Layout (طرح بندی) در WinForms و همچنین دلفی، بر اساس قراردادی اشیاء در مختصات مطلق در صفحه است. مثلاً این دکمه‌ی خاص در آن نقطه‌ی معلوم قرار می‌گیرد و همین. این روش طرح بندی یکی از چندین روش طرح بندی در WPF است که اصطلاحاً Canvas نام دارد. توصیه اکید و مطلق در WPF آن است که از Canvas فقط برای طراحی اشیاء گرافیکی پیچیده استفاده کنید و نه طراحی رابط کاربر. چرا؟ چون برای مثال در Silverlight که برادر کوچکتر WPF محسوب می‌شود، رابط کاربری آن باید بتواند همانند HTML انعطاف پذیر باشد و با اندازه‌های مختلف مرورگر یا اندازه‌ی قلم‌های بزرگتر هماهنگ شده و مقاومت کند، بدون اینکه از ریخت بیفتد و این مورد را سایر سیستم‌های طرح بندی رابط کاربر (منهای Canvas یا همان سیستم طرح بندی WinForms) ارائه می‌دهند. مورد دیگری که در WPF و Silverlight بسیار حائز اهمیت است ، Content Controls می‌باشد. چقدر خوب می‌شد بتوان داخل یک کنترل، کلاً یک سیستم طرح بندی را تعریف کرد و اشیاء دلخواهی را داخل آن قرار داد. مثلاً ToolTip پیش فرض وجود دارد. بسیار هم خوب. بنده علاقه دارم، متن عنوان آن ضخیم باشد. کنار آن یک تصویر کوچک و در سمت چپ آن متن قرار گیرد. برای انجام اینکار در WPF لازم نیست تا شما منتظر نگارش بعدی دات نت باشید که دست اندرکاران مربوطه با افتخار در یک

سند 50 صفحه‌ای توضیح دهند که چگونه می‌توان اینکار را انجام داد. یک سیستم طرح بندی را اضافه کنید. موارد ذکر شده را در آن تعریف کنید. بدون استفاده از هیچ نوع کامپوننتی یا بدون منتظر ماندن تا نگارش بعدی این محصولات، به مقصود خود رسیده‌اید.

ASP.NET Web forms : داره نفس‌های آخرش رو می‌کشه!

از زمانیکه ASP.NET MVC آمد نسخه‌ی Web forms تقریباً فراموش شد. به وبلاگ [اسکات گاتری](#) یا [Haacked](#) و سایر اعضای اصلی دات نت که مراجعه کنید در سه سال اخیر در حد تعداد انگشتان یک دست هم در مورد Web forms مطلب ننوشته‌اند. تمام تمرکز و نوآوری‌ها بر روی MVC متمرکز شده و حتی در نسخه‌ی 4 دات نت هم فقط یک سری صافکاری مختصر را در Web forms شاهد بودیم مثلاً نام کنترل‌ها را خودتان هم در زمان رندر نهایی می‌توانید تعیین کنید! یا لطفاً کردند و قسمتی از url routing موجود در ASP.NET MVC را به ASP.NET web forms 4 هم [قرض دادند](#) (این مورد شاید مهم‌ترین تغییر قابل ذکر در ASP.NET web forms 4 است).

البته این رو هم اضافه کنم که ASP.NET MVC واقعا قابل احترام است؛ هدف از آن جدا سازی لایه‌های برنامه با الگوهای استاندارد صنعتی (و نه هر روش برنامه نویسی چند لایه من درآوردی)، ترویج کد نویسی بهتر، ترویج Unit testing، رفع یک سری مشکلات ASP.NET Web forms (مثلاً از ViewState های حجیم دیگر خبری نیست) و امثال آن است.

برای نمونه توجه شما را به مطلبی که آقای Dino Sposito در مورد ASP.NET Webforms نوشته جلب می‌کنم: ( [+](#) ) به صورت خلاصه ایشان عنوان کرده زمان طراحی ASP.NET Webforms در 10 سال قبل، هدف ما انتقال ساده‌تر برنامه نویسی‌های VB به محیط وب بود. به همین جهت دست به اختراع ViewState ، postback ، کنترل‌های سمت سرور و غیره زدیم. (بنابراین تأکید تمام این‌ها این است که webforms فناوری دهه قبل "بود" و الان بنابر نیازهای جدید دست به طراحی مجدد زده‌ایم)

در مورد "پایان پروژه ASP.NET Ajax Control Toolkit" هم [قبلاً](#) مطلب نوشته بودم و این یکی واقعا official است!

و در پایان باید متذکر شد که فلان فناوری مرده یا داره نفس‌های آخرش رو می‌کشه اصلاً مهم نیست. هنوز هم هستند سازمان‌هایی که برنامه‌های نوشته شده با ASP کلاسیک (نگارش قبل از ASP.NET Web forms) خود را دارند و خیلی هم از آن راضی هستند! این موارد رو از این جهت نوشتم که اگر می‌خواهید تازه به این جمع وارد شوید دقیقاً بدانید باید روی چه مواردی بیشتر وقت بگذارید و یادگیری کدامیک صرفاً اتلاف وقت خواهند بود (مثلاً EF بر LINQ to SQL ارجح است و اگر امروز می‌خواهید شروع کنید با EF شروع کنید، یا دیگر کم کم با وجود WPF ، بازار کاری برای WinForms نخواهد بود).

## نظرات خوانندگان

نویسنده: مصطفی دیندار  
تاریخ: ۱۳۸۹/۱۰/۱۴ ۲۳:۳۲:۱۱

جناب نصیری , مبتنی بودن شیرپوینت بر اساس وب فرم ها نمیتونه یک دلیل باشه که وب فرمها هنوز زنده بمانند ؟

نویسنده: علی اقدام  
تاریخ: ۱۳۸۹/۱۰/۱۵ ۰۰:۰۶:۰۱

دروود بر شما جناب نصیری به مقوله خوبی پرداختید

امروزه در مورد استفاده از تکنولوژی های جدید در سیستم های جدید نظرات زیاد و متناقضی هست مثلا به نظر شما تو یه سیستم که تراکنش اون بسیار بالا در حدود مثلا 1 میلیون تراکنش در روز آیا استفاده از EF عاقلانه است یا نه؟

تکنولوژی های جدید به سمت راحتی برنامه نویسان و خوانایی کد سوق داده شده اند ولی در عوض به اتلاف Performance می انجامند!

نویسنده: وحید نصیری  
تاریخ: ۱۳۸۹/۱۰/۱۵ ۰۰:۰۷:۱۵

مایکروسافت عادت داره دست به تغییرات اساسی بزنه. نمونه اش exchange server یا نمونه ی دیگر isa server که اگر به سیر تکاملی این ها دقت کنید می بینید که نگارش های جدید با نگارش های 10 سال قبل از زمین تا آسمان فرق کرده اند. در کل باید منتظر نگارش بعدی شیرپوینت بود. یکی از ایراداتی که به شیرپوینت وارد هست سنگین بودن آن برای محیط خارج از اینترنت است (حجم بالای view state).

نویسنده: وحید نصیری  
تاریخ: ۱۳۸۹/۱۰/۱۵ ۰۰:۲۰:۱۳

همین الان سایت stack overflow مبتنی بر LINQ to SQL و ASP.NET MVC است (مطابق آخرین آماری که منتشر کرده اند) و تعداد تراکنش های بسیار بالایی هم دارد. بنابراین، بله. عاقلانه است و دارد استفاده می شود : [\(+\)](#)

نویسنده: shahin  
تاریخ: ۱۳۸۹/۱۰/۱۵ ۰۸:۱۶:۳۳

سلام.

ASP.NET MVC برای پروژه های درون سازمانی و سیستم های اتوماسیون اداری مناسب هست؟  
یک مقدار درباره بازار کار آن در خارج از ایران توضیح می فرمایید نسبت به WEB Forms ?  
ممنون

نویسنده: وحید نصیری  
تاریخ: ۱۳۸۹/۱۰/۱۵ ۰۸:۴۵:۰۷

سلام، من داخل ایران زندگی می کنم ولی در کل یک سری ویدیو در مورد ASP.NET MVC هست که اکثر تهیه کنندگان آن ها افرادی هستند که مشغول تولید نرم افزار برای دولت آمریکا هستند و الان کارشون رو به MVC ارتقاء دادند (از لابلای صحبت ها قابل استخراج است): [\(+\)](#)

نویسنده: Meysam  
تاریخ: ۱۳۸۹/۱۰/۱۵ ۱۰:۱۷:۳۰

L2S تو S0 جوابش رو پس داده ولی به قیمت بسیار تخصصی! Jeff چند تا پست در مورد Performance واسه L2S داره که نشون میده قضیه چقدر بغرنج شده!

<http://www.codinghorror.com/blog/2010/03/compiled-or-bust.html>

بله موافقم، میکروسافت داره به سمت ASP.Net MVC میره.

نوشتن اتوماسیون با ASP.Net دیگه کار درستی نیست! سیلورلایت بهترین انتخابه (فقط از MVVM استفاده نکنید، اگر مشکل زمان دارید)

نویسنده: ebraminio

تاریخ: ۱۰:۴۹:۱۳ ۱۳۸۹/۱۰/۱۵

ممنون از شما، مفید بود

نویسنده: A

تاریخ: ۲۳:۴۱:۴۵ ۱۳۸۹/۱۰/۱۵

اسکات هنسلمن در یک فیلمی که در مورد MVC صحبت می‌کند، MVC را به موتور سیکلت و WebForms را به یک ماشین ون تشبیه می‌کند. این تشبیه به علت وجود خصایص کامپوزیتی در WebForms است که تشبیه بسیار خوبیست.

MVC خیلی خوب است اما برای مثال شما در آن نمی‌توانید یک کامپوننت ایجاد کرده و به راحتی در پروژه‌های بعدی استفاده کنید بدون اینکه به کدتان دست بزنید. (فقط از یک بُعد به کامپوننت نگاه نکنید. مثلاً نه فقط بر پایه jQuery و این قبیل و یا فقط کد و ...) این می‌تواند نقطه ضعف بزرگی باشد. به نظر من ASP.NET عقب افتاده است اما MVC نمی‌تواند جایگزین 100% آن باشد. چیز دیگری باید به وجود آید ولی آن MVC نیست. البته باید این را هم در نظر گرفت که شاید سیاستمداران میکروسافت به این فکر کرده‌اند که وقتی با PHP وب سایت‌های معتبری را راه می‌اندازند چه نیازی به کامپوننت گرایی و ...؟!

البته اگر از حق نگذریم وب محیطی متفاوت است و رفتار با آن مانند محیط‌های دیگر، به هر حال باعث بروز سرباره می‌شود.

نویسنده: وحید نصیری

تاریخ: ۰۱:۱۱:۵۰ ۱۳۸۹/۱۰/۱۶

بیشتر مطالعه کنید. یک نمونه از مجموعه‌های افزودنی‌های جالب ASP.NET MVC توسط شرکت معظم Telerik به صورت سورس باز و رایگان در این آدرس قابل دریافت است: [\(+\)](#) هدف از ASP.NET MVC پیاده سازی مفاهیم مهندسی نرم افزار مانند unit testing ، فریم ورک‌های IOC ، جداسازی صحیح لایه‌های برنامه از یکدیگر، دور ریختن مفاهیم web forms که اساساً تحمیل میکروسافت به وب است و نه روند طبیعی کارکردن آن و موارد شبیه به آن است.

هدف من از ارائه‌ی این مطالب چی بود؟ تلنگری به کسانی که خوابند! میکروسافت داره به این سمت میره. رفته ... من و شما خوشمون بیاد یا نیاد. همین کار رو با ASP کلاسیک کرد (سرش رو برید گذاشت یک گوشه جون بده). همین کار رو با Vb6 کرد، همین کار رو با فاکس پرو کرد و .... این یک روند تکامل تدریجی است.

نویسنده: وحید نصیری

تاریخ: ۰۱:۴۲:۴۴ ۱۳۸۹/۱۰/۱۶

یک سری آموزشی مفصل ASP.NET MVC اینجا است: [\(+\)](#) (از آقای راب کانری نویسنده ORM ایی به نام ساب سونیک) که بعدها به سایت ASP.NET هم منتقل شد: [\(+\)](#)

نویسنده: Armin

تاریخ: ۱۳:۴۴:۱۲ ۱۳۸۹/۱۰/۲۸

نظرتون درباره Silverlight و آیندش چی هستش؟

نویسنده: وحید نصیری  
تاریخ: ۱۴:۲۵:۲۹ ۱۳۸۹/۱۰/۲۸

روی سیلورلایت 5 خیلی دارند کار می‌کنند:

- API سه بعدی

- شتاب دهنده‌های مبتنی بر GPU

- ارائه‌ی نسخه‌ی 64 بیتی

و ...

جزئیات رو می‌تونید اینجا ملاحظه کنید: [\(+\)](#)

همچنین تیم سیلورلایت داره مجدداً استخدام می‌کنه! : [\(+\)](#)

و سیلورلایت 4 به عنوان بهترین فناوری RIA سال از طرف InfoWorld انتخاب شده: [\(+\)](#)

نویسنده: وحید نصیری  
تاریخ: ۹:۲۷ ۱۳۹۲/۰۴/۲۰

نسخه رسمی و به روز شده «وضعیت فناوری‌های مرتبط با دات نت از دیدگاه مرگ و زندگی!» از طرف میکروسافت:

[Summary - .NET Technology Guide for Business Applications](#)

[The .NET Technology Guide for Business Applications](#)