

## ◆ ASP.Net

چهارشنبه ۲۴ آبان ۱۳۸۵

مقدمه

با به وجود آمدن ASP.NET ما شاهد تغییر در روش های متعارف اسکریپت نویسی بسوی شروع برنامه نویسی حرفه ای بر خط (online) هستیم.

اسکریپت های VB تنها انتخاب نیستند زیرا برنامه نویسان میتوانند از تمام امکانات موجود در .NET مانند VB و C++ و تمام زبانهای موجود در آن استفاده کنند.

نمیتوان انکار نمود که .NET مورد توجه بسیار توسعه دهندگان نرم افزار قرار گرفته است و این نشانه قالب مناسب و استراتژی .NET برای تداوم در دنیای برنامه نویسی میباشد. معرفی راه حل نرم افزاری که هر کس را قادر میسازد با هر زبان دلخواه که با محیط .NET سازگاری داشته باشد برنامه نویسی کند حداقل کاری است که میتوان در مورد آن انجام داد.

در این مقاله نگاهی به شروع و سرگذشت صفحات فعال سرور در دو سال اخیر داریم و اینکه چگونه برنامه نویسان را به خود جذب کرده. البته مشکلاتی دارد ولی طراحان .NET راه حلی برای

تعداد زیادی از مشکلات برنامه - نویسی یافته اند.

همچنین به منظور فراهم سازی اطلاعاتی برای کاربر که شما میخواهید آنها را ارائه نمایید، تحولاتی در زمینه کارکرد ASP با سرور و کلاینت انجام گرفته است.

هر چند که نسخه فعلی نسخه پایدار بتا میباشد و بسیاری از مردم تصور میکنند که آنچه که ارائه میشود پایدار میباشد ولی به دلیل طبیعت بتای آن دارای مشکلاتی میباشد. یادگیری با مشکلات موجود در چهار چوب امکان آمادگی در برابر نسخه جدید را فراهم میکند.

### یادگیری از تاریخچه ASP

برای دانستن تاریخچه ASP میتوان به سال ۱۹۹۵ و در محیطی خاص که ماکروسافت دریافت که به علت عدم پشتیبانی اینترنت در محصولات خود از این صنعت باز مانده باز گشت. قبل از آن ماکروسافت در پی بدست آوری فن آوری و ابزار و پرتکلهای اختصاصی خود برای شبکه های ماکروسافت بود. اما ناگهان به استراتژی اینترنت احتیاج پیدا کردند.

ماکروسافت توانست خود را از شرایط عقب مانده به شرایط ایده آلی برساند با تولید مرورگر وب

IE (internet explorer) بازار مرورگر وب را بدست آورد و IIS (internet information

server) را در ۱۰۰۰ شرکت مهم نصب نمود.

## پیدایش ASP

بازگشت به اواسط دهه ۹۰ ابتدای دنیای وب نشان میدهد در آن زمان برای تولید وب کسی که میخواست وب سایت خود را محیطی مناسب برای تجارت قرار دهد ابزارهای زیادی برای انتخاب نداشت و انتخابها محدود بودند به محیطهای برنامه نویسی سمت سرور و ابزار برنامه نویسی سمت کاربران محدود میشد. در انتها برنامه نویس باید برنامه های رابط دروازه عمومی CGI(common gateway interface) را بکار برد. در این برنامه ها از زبانهای کامپایلی نظیر دلفی و ویژوال بیسیک یا زبانهای اسکریپت نویسی تفسیرشدنی مانند پرل یا رکسو اسکریپت های لایه سیستم عامل مانند UNIX استفاده می گردد.

ابتدای ۱۹۹۶ ماکروسافت شروع به اصلاح وضعیت موجود کرد برای رسیدن به اهداف خود رابط برنامه

## نویسی کاربردی

سرور اینترنت (Internet Server Application Programming Interface) ISAPI را به عنوان

بخشی از سرور اطلاعات اینترنت در نظر گرفت ISAPI. حاصل توسعه Win32 API ویندوز میباشد.

این فن آوری برای تولید نرم افزار سرور وب تولید شده بود. که بتواند با IIS ارتباط داشته باشد. که

باعث پنج برابر شدن کارایی سیستم بود. همان گونه که شما با این توضیحات تصور میکنید این

افزایش کارایی اثرات جانبی دارد بر روی پیچیدگی برنامه نویسی برای برنامه نویسان میشود.

این به علت ضعف برنامه نویسان نبود بلکه احتیاج به دانش پایه ای ودقیق برنامه نویسی داشت تا

برنامه های ISAPI درست کار کنند. شرکت ماکروسافت همانگونه برنامه نویسان را به استفاده  
ISAPI تشویق میکرد آنها را به استفاده از فن آوری اتصال دهنده پایگاه داده اینترنت  
Database Connector (IDC) تشویق میکرد. این فن آوری روشی جدید برای ایجاد ارتباط  
سایتهای اینترنتی با پایگاههای داده سرور به واسطه ارتباط پذیری باز با پایگاه داده (ODBC) Open  
Database Connectivity میباشد.  
فن آوری های IDC و ISPI از خامی ماکروسافت کاسته و برای اولین بار باعث تبدیل سرور وب  
ضعیف به محیط سرور برنامه های محاوره ای ابتدایی شد.

برنامه نویسان دیگر ابزارهای دیگری در اختیار داشتند بسیاری از آنها از جمله Netscape  
Livewire از محبوبیت زیادی برخوردار بودند Livewire فن آوری قابل اجرا تحت سرور  
Netscape بود که از فن آوری جاوا اسکریپت که در آن از اجزا جاوا نیز استفاده شده بود برای  
منطق در صفحه استفاده میکرد. محدودیتهای ISAPI در Livewire نیز به علت کاملی بودن  
وجود داشت و برای اجاد تغییر نیاز به قطع کردن برنامه و شروع مجدد بود.

چرا ASP مورد نیاز بود؟

تمامی برنامه نویسان وب ، توانایی برنامه نویسی لازم برای نوشتن نرم افزارها ISAPI را نداشتند و چون در SAPI به کامپایل برنامه ها نیاز بود. مراحل زیادی در تولید سایت مبتنی بر ISAPI وجود داشت که سبب کند شدن مراحل تولید سایت میشد. برنامه نویسان تازه کار دریافتند که یادگیری زبانی صنعتی و قدرتمند مانند ++C ، کامپایل حتی ساده ترین منطق صفحه خود به فایل های dll نیاز دارد.

برنامه های ویژوال بیسیک علی رغم سهولت تولیدشان ، در صورت بکارگیری برای CGI بسیار ضعیف عمل نموده و تمام منابع را در اختیار خواهند گرفت .زبانهای دیگر از جمله پرل برای آغاز یک برنامه خط دستور جداگانه و به منظور تفسیر اجرای اسکریپتهای درخواست شده به سرور نیاز دارند که این امر موجب افزایش زمان بارگذاری صفحه و کاهش کارایی سرور خواهد شد.

CGI خود تمام منابع را در اختیار میگیرد چون درخواست هر صفحه سرورهای وب را وادار میکند تا فرایندهای جدید را متوقف نموده و از طریق این فرایندها ارتباط را برقرار نمایند. این امر موجب اتلاف وقت و مصرف حافظه بارزش RAM میگردد مشکل دیگری که فراروی تیم برنامه نویسی در اواسط دهه 90 قرار داشت ، این مسئله بود که سایت در واقع ترکیبی از زبان نشانه گذاری

فرامتنی (Hypertext Markup Language (HTML و منطق است. آنها به روشی برای ادغام

کدهای برنامه نویسان با HTML طرح صفحه طراح و طراحی بدون ایجاد آشفتگی در ساختار دیگری نیاز داشتند. راه حل های متعددی برای رفع این مشکل از سیستمهای قابل اختصاصی گرفته

تا عبارات شامل سمت سرور Server Side Include (SSI) وجود داشتند. در هر یک از این راه حل ها بر سرور اعلام میشود که کد مبتنی بر تگ های توضیحات ویژه HTML را اجرا نماید.

ایجاد قابلیت محاوره ای که بتواند از سوی پایگاه داده هدایت شود ، نیز مشکلی دیگر در بهینه سازی برنامه نویسی بود. تقاضا برای سایت های وب پیچیده آغاز شده بود و برنامه نویسان به منظور رفع این تقاضا به روشی قابل مدیریت نیاز داشتند ولی ابزارهای موجود تسهیلی در انجام این کار ایجاد نمیکردند. افرادی که میتوانستند به این روش دست یابند ، خواهان پاداشی متناسب با سختی انجام کاری که از آنها خواسته شده بودند.

آنچه که لازم بود ، راه حلی برای بقیه ما بود . این راه حل می بایست فن آوری ساده ای مبتنی بر متن و دارای اسکریپتی همانند پرل باشد ، تا برنامه نویسان بتوانند صفحات خود را بدون نیاز به کامپایل و با کمک ابزار ساده ویرایش متن مانند Notepad تغییر دهند. این روش باید در حین حفظ کارایی لازم ، به حداقل منابع نیاز داشته باشد. بنابراین باید آن را درون سرور همانند ISAPI ولی بدون پیچیدگی اجرا نمود ، تیمهای طراحان و تنظیم کنندگان علاقمند بودند که این روش شامل امکانات و خصوصیات قالبها و SSI باشد تا راحتتر بتوان یکپارچه سازی آرایش صفحات را مدیریت نمود. این روش برای اینکه بتواند با استقبال مواجه شود ، می بایست زبانی را اجرا نماید که براحتی

بتوان آن را آموخت و برای بسیاری از برنامه نویسان آشنا باشد.

چرا از ابتدا ASP بکار گرفته نشد؟

ASP موفقیتی ناگهانی نبود، با این حال توانست، بخش عظیمی از جامعه تولیدکنندگان نرم افزار را به خود جذب کند. بویژه افرادی که قبلاً تجربه های خوبی در برنامه نویسی ویژوال بیسیک برای اسکریپت های کاربردی داشتند.

افراد دیگری که برای یادگیری ویژوال بیسیک سرمایه گذاری نکرده بودند، دریافتند که محدودیتهایی در ویژوال بیسیک و اسکریپت نویسی ویژوال بیسیک وجود دارد که مانع استفاده از این فن آوری میگردد. کمبودهای موجود شامل مدیریت ضعیف حافظه، عدم قابلیت های مدیریت قدرتمند رشته مانند عبارات منظم است که در سایر زبانهای موجود وجود داشت ASP. در مقایسه با CGI پرل، چنین کمبودی ندارد در آن زمان سرور اطلاعات در مراحل اولیه خود بسر میبرد و توسعه آن به آرامی صورت میگرفت، هر چند که نیروی قدرتمند روابط عمومی مایکروسافت پس از پیشرفت شگفت آوری که از سوی شرکت گزارش شد، کاملاً به جریان افتاده بود، سرور اطلاعات در مقایسه با نسخه های موجود این نرم افزار کاملاً ضعیف به نظر میرسد ولی هنوز هم از لحاظ کارایی قابل رقابت است.

تا سال ۱۹۹۷، بخش اعظم برنامه نویسی سرور در اختیار پرل و CGI بود. سایت های وب با کارایی بالا معمولاً از ترکیبی از برنامه های کامپایل شده C برای موتورهای تجاری واقعی و برای پردازش های سبکتر از فرم ها استفاده میکردند. در آن زمان تردیدها و سوءظن هایی پیرامون اقدامات و فعالیت های اینترنتی ماکروسافت از جمله IIS و Internet Explorer وجود داشت و ISAPI بیش از تشویق بخش وسیعی از جامعه برنامه نویسان نرم افزار، کاربیشتری انجام نداد. علی رغم وجود چنین فضای نامطمئنی، ماکروسافت با خرید چندین مجوز Windows NT4 برای میزبانی وب و توسعه تولید نرم افزار روبرو شد.

پشتیبانی شرکتهای ثالث برای هر چیز دیگری به جز اجزا کوچک در ابتدا به کندی آغاز شد ولی همانند تمامی محصولات ماکروسافت که پس از عرضه دو نسخه از محصول، شرکت همه مشکلات را برطرف میسازد، ASP نیز از این قاعده مستثنی نبود.

در حالی که بخش عمده ای از برنامه نویسان از جمله شخص برجسته ای چون Larry Wall از پرل استفاده میکردند. برنامه نویسان ASP تا آن زمان تحت حمایت مناسبی قرار نداشتند. برنامه نویسان پرل از طرف مقام های بالا دعوت به مشارکت و باز گذاشتن کد هایشان میشدند و به همین علت این گروه ها با پیشرفت مواجه میشدند، به طوری که تمامی راه های ممکن فقط با چند کلیک



در سایت شبکه آرشیو جامع پرل از لحاظ رقابتی و مالی منجر به ایجاد نظریاتی پیرامون وجود فروشندگان ثالث اجزا در دنیا نرم افزار های توزیع یافته اینترنتی ویندوز (DNA) گردید. البته این موجب توسعه ASP به صورت برنامه مورد علاقه امروزی نگردید ولی در موفقیت فعلی آن سهمیم بوده است.

### نیاز به مدل جدید ASP

بدیهی بود ماکروسافت به تغییر اصولی برای رساندن ASP به سطح ایده آل برنامه نویسی قدرتمند و

صنعتی نیاز داشت. صفحات فعال وب مبتنی بر اصول COM است. فن آوری COM و

ActiveX بخش مهمی از قدرت ASP را فراهم میکنند ولی محدودیتهایی نیز برای آن خواهند

داشت. ماکروسافت باید تلاش زیادی در جهت بررسی COM و یافتن روشهای بهبود آن انجام

میداد و این تغییرات می بایست بر روی ASP نیز تاثیر بگذارند. در همان زمان ماکروسافت در یافت

که زمینه کاری برنامه نویسان تغییر یافته است و بر اساس استانداردهای جدید که هر لحظه عرضه

میشوند به ویژه در نرم افزارهای توزیع یافته و اشتراک اطلاعات همچون Simple (SOAP)

XML\_RPC و Object Access Protocol ، که در آنها از XML استفاده گردیده تجدید

میشود.

خدمات وب متداول شد، جاوا در همه جا حضور داشت و XML محبوبیت زیادی در میان برنامه

نویسان پیدا کرده بود. نسخه جدیدی از ASP برای رفع این نیازها کفایت نمیکرد. اگر تغییرات فقط به قصد شکست دادن حریفان نبود و میتوانستند در مقابل سختیها مقاومت کنند ولی باید فن آوری دور از دسترسی را در اختیار میگرفتند.

Windows DNA و ASP که بر پایه فن آوریهای Win32 API و COM ابتدای دهه ۱۹۹۰ بودند، نمیتوانستند معماری فنی و منطقی برای نرم افزارهای توزیع یافته جدید فراهم نمایند در حالی که شرکت SUN با استفاده از Java 2 Enterprise Edition (J2EE) مجموعه ای از فن آوریهایی را در اختیار داشت که برنامه نویسان میتوانند با دنبال نمودن آنها پروژه های کوچک

را همراه با Standard Edition آغاز نموده و آن را برای Enterprise Java Beans کاملاً تغییر اندازه دهند.

در دنیای امروز، تنها رقابت برای مرورگرهای مختلف دیده نمیشود، بلکه رقابت در بخشهایی چون کانالهای توزیع، روشهای اجرا در کامپیوترها و تلفنهای موبایل، تلوزیون دیجیتال محاوره ای، لوازم خانگی هوشمند، منازل مجهز به شبکه های دیجیتال و شاید برای تبدیل صفحات وب به نرم افزارها و خدمات وب یکبار مصرف نیز وجود دارد. بدون تردید، در حالی که ماکروسافت در حال بررسی

پیرامون فن آوری های مخصوص خود میباشد ، باید رقابت را نیزمورد تجزیه تحلیل قرار دهد . همان طور که ماکروسافت چارچوب کاری NET را عرضه نمود ، زبان جدیدی را نیز برای قرن بیست و یکم به نام C# ارائه کرد و NRT بر تمام انتقادات غلبه کرد و روشی کاملاً جدید برای بررسی نرم افزارها و وب فراهم ساخت و برای هر چیز موجود شامل ویژوال بیسیک و ، ویژوال C++ و صفحات فعال سرور جای گزین قرار داده اند.

پیش از ASP.NET نگاهی به تاریخچه مختصر و پر حادثه ASP خواهیم داشت :

در دسامبر ۱۹۹۵ ، ماکروسافت با انجام تحولی کامل بیان نمود که تمامی تیم تولیدی محصولات خود را برای در برگرفتن انترنیت در ابزارهای خود بکار خواهد گرفت . تا آن زمان آنها از بازار انترنت صرف نظر کرده بودند و در این رقابت کاملاً عقب افتاده بودند.

فوریه ۱۹۹۶ ، ماکروسافت سرور اطلاعاتی انترنت را برای بارگذاری رایگان عموم منتشر کرد.سخنگوی ماکروسافت ادعا کرد این که این سرور ۴ برار سرور Netscape Netsite | کارایی دار و شامل IDC ISAPI نیز می باشد.

با گسترش Windows NT4 نسخه بعد IIS نیز همراه با آن گسترش یافت.

اکتبر ۱۹۹۶ ، ماکروسافت نسخه بتای IIS3 را برای ارتقا IIS 2 در اختیار قرار داد تغییر اصلی در

این نسخه پشتیبانی از صفحات فعال سرور بود که این پروژه را "Denali" نامیده بود. ماکروسافت

ادعا کرد بر بازار Netscape 2.1 غلبه خواهد کرد.

آگوست ۱۹۹۷، ماکروسافت 2 ASP را به همراه IIS 2 منتشر کرد IIS . در خود کنسول مدیریت ماکروسافت (MMC) را برای مدیریت ساده سرور داشت و سرور SMTP به عنوان برنامه همراه در نظر گرفته شده بود SMTP. قبلاً قسمتی از بسته Commercial محسوب میشد. ÷ و ASP در این مرحله بطور کامل با Microsoft Transaction Server یکپارچه شده و این در واقع بمنزله برداشتن قدمی به سوی تهیه برنامه ای بود که انتخابی معتبر برای راه اندازی شبکه های بزرگ محسوب میشد.

1998 تا ۲۰۰۰ ماکروسافت انتشار نسخه های بالاتر موتورهای زبان اسکریپت نویسی (Language Scripting Engines) را آغاز نمود و بدون نیاز به UPDATE کارایی ASP را بالا برد بطور مثال عبارات منظم را برای برنامه نویسان اسکریپتهای VB افزود.

با انتشار ویندوز ۲۰۰۰ همراه با IIS 5 نسخه 3 ASP عرضه شد ASP 3 امکان کنترل دوبار سمت سرور، و پشتیبانی بهتر از خطاها و استفاده از ADO 2.5 و XML و مخفی سازی کد کامپایل شده را در اختیار قرار میدهد IIS 5. این امکان را میدهد که پردازها را تحلیل کند و از خرابی و از کار افتادن سیستم جلوگیری کند.

جولای سال ۲۰۰۰ NET. اطلاعاتی عمومی خود را منتشر نمود و C# را نیز معرفی کرد که نشان

دهند انعطاف و پایداری بود.

مروری بر پایه محیط ASP.NET

ماکروسافت با ارائه ASP و زبانهای قدیمتر خود به صورت NET. در قرن بیست و یکم قدم مهمی

برداشت ASP.NET. که از VB.NET را برای کد نویسی در اختیار قرار میدهد به برنامه حرفه

ای و شی گرا برای تولید صفحات وب تبدیل شده و به این فن آوری کمک بسیار میکند. ولی زبانهایی

که از گذشته به ارث رسیده اند نمی توانند به اندازه NET. کارایی داشته باشند بنابراین NET.

زبان C# را ارائه کرده که برای محیط NET. مناسب باشد.

C# در NET. به عنوان زبانی کاملاً جدید ارائه شده C#. در عین اینکه برای برنامه-نویسان C

آشنا میباشد ولی جدید است C#. به دلیل داشتن خصایص RAS مورد علاقه برنامه نویسان VB

نیز میباشد. برخی میگویند C# زبانی ارائه شده از سوی ماکروسافت برای رقابت با Java شرکت

SUN میباشد.

اگر ماکروسافت تولید نرم افزار را بخوبی تولید نماید و C# را با VS.NET عرضه کند آنگاه قطعاً C#

به معیارهای مورد نظر خواهد رسید C#. در واقع زبانی جدید با تمامی امکانات دلخواه شما مانند

شی گرای کامل، مدیریت خود کار حافظه و مرتب سازی محیط کار می باشد.

بعضی از نکات مهم در مورد ASP.NET عبارتند از:

ASP.NET بخشی اصلی از حوزه ابتکاری Microsoft.NET یا برنامه ای جدید برای تولید نرم

افزار می باشد.

ASP.NET در واقع یک طراحی کاربردی برای جایگزینی به جای مدل DNA ویندوز و همچنین

مجموعه ای از ، خدمات ، ابزارها ، نرم افزارها و سرورها بر پایه چهارچوب کاری ASP.NET و زبان

اجرائی مشترک می باشد.

ASP.NET به جای که جایگزین ASP 4 بشود یا نسخه ارتقا یافته تدریجی آن باشد خود باز

نویسی کامل از ابتدا محسوب می شود و از تمامی امکانات پیشرفته برای در دسترس قرار دادن

ASP.NET استفاده شده است.

ASP.NET می تواند از هر چیز که توسط ASP.NET ارائه می شود از جمله پشتی بانی از تعداد ۲۰ یا

بیشتر زبان ( ASP.NET از Perl.NET تا C# ) و مجموعه ای کامل کتابخانه های نرم افزاری چارچوب

کاری ASP.NET بهره ببرد .

نرم افزارهای وب نوشته شده در ASP.NET سریع ، کارآمد ، قابل مدیریت ، مقایسه پذیر و انعطاف

پذیر هستند علاوه بر تمامی موارد فوق درک و کدنویسی آنها نیز بسیار ساده می باشد.

تمامی اجزا و نرم افزارهای وب کامپایل می شوند . اشیا ASP.NET همگی به زبانهای یکسانی نوشته

شده و همه آنها عملکرد یکسانی دارند ، در نتیجه به دلایل اجرائی نیازی به ترک محیط ASP وجود

ندارد.

نیاز کمتری به استفاده از ابزار خارجی می باشد با نوشتن فقط چند خط کد ، می توان میان ASP.NET و XML را بر قرار کرد یا شرطی گذاشت که ASP.NET به عنوان نوعی وب

سرویس عمل کند ، یا از آن استفاده کند فایل ها را منتقل کند یا صفحه سایتی را از راه دور پاک کند یا تصویری روی آن قرار دهد.

استفاده از انعطاف ASP.NET

ماکروسافت با کمک چارچوب کاری NET و ASP.NET نه تنها خود را به عنوان رقیبی در فن

آوری های تولید وب مطرح نمود بلکه بسیاری از منتقدان نیز بر این عقیده بودند که ماکروسافت

توانسته سایرین را پشت سر بگذارد ASP.NET. برای انجام هر کاری که بخواهید از ساخت

صفحات وب تا سایت های بزرگ تجارت الکترونیکی ، در نظر گرفته شده.

ماکروسافت دقیق شد تا انعطاف و قدرت مورد نظر برنامه نویسان با داشتن خاصیت سادگی استفاده

از آن رعایت شود.

با استفاده از ASP.NET توانایی انتخاب زبانها را دارید. همه زبانهای NET. از کتابخانه و امکانات

یکسان استفاده میکنند. قدرتهای یکسان از جهت ارث بری و کار با اجزا COM دارند

شما می توانید با اطلاعات و کد نویسی یکسان برای هر کار از تولید سایت اینترنتی تا ایجاد

سیستمهای بزرگ استفاده کنید. اختلاف میان API ها برای برنامه نویسان هیچ مشکلی ایجاد نمیکند و برنامه نویسان لازم نیست نگران تبدیل متغیرها باشند.

در ASP.NET تمامی استانداردهای جدید و مهم مانند SOAP و XML به علاوه ADO.NET و کتابخانه های کلاس اصلی ترکیب شده اند و پیاده ساز آنها نسبت به سایر فن آوری ها ساده تر خواهد بود.

یک برنامه نویس ASP.NET فقط به کامپیوتری که دارای Notepad و قابلیت FTP میباشد برای نوشتن کد ASP احتیاج دارد ولی با استفاده از کنترل دستور محیط NET و قالبهای متنی بر

XML برنامه هاین عمل ساده تر انجام می پذیرد. ماکروسافت امکانات بسیار قدرتمند کلاس های

کتابخانه، چون توابع قابل استفاده در شبکه برای عملکرد با پروتکل کنترل مخابره، پروتکل اینترنت

(TCP/ P I Transmission Control Protocol/ Internet Protocol) و سیستم نام

دومین (DNS) و خدمات وب و داده XML و تصویر کشیدن تصاویر گرافیکی را در چارچوب کاری

NET در نظر گرفته است.

در گذشته ، منظور از محدودیت های اسکریپت نویسی ASP اجزا بود که بنا به دلایل مربوط بعمل



کرد و نه دلایل طراحی مورد نیاز بودند ASP.NET. به عملکرد یکسانی دسترسی داشته و از زبانهای یکسانی برای تولید اجزا استفاده میکنند در نتیجه اجزا را میتوان انتخابی از لحاظ طراحی دانست.

یک برنامه نویس ASP.NET نسبت به تغییرات انجام گرفته در سیستم عامل زمینه و API محافظت میشود و از آنجایی که فن آوری NET در رابطه با نحوه پیاده سازی کدهای شما میباشد شما با استفاده از Common Type System نباید نگران بکار گیری پیاده سازی های مختلف یک رشته یا عدد صحیح در زبان مورد استفاده در جز ایجاد شده خود باشید.

#### نوشتن برنامه های کاربردهای ASP.NET

در نسخه های قبلی ASP، نصب و راه اندازی نرم افزار به برنامه ریزی دقیقی نیاز داشت به خصوص در شرایطی که سیستم بزرگ و پیچیده بود این امر به علت وجود عوامل مختلفی چون نیاز به انتقال، نصب، ثبت اجزا تولید شده لزوم متوقف ساختن و آغاز سرور وب و تضمین بابت اینکه این نسخه درست را در اختیار دارید دیده میشود. شما مجبور به پیکر بندی سرور وب از طریق کنسول مدیریت ماکروسافت ADSI یا ابزار خط فرمان بودید که بعضی اوقات شما را وادار به متوقف ساختن و آغاز دوباره خدمات میکرد.

اما با استفاده از ASP.NET تمامی این موارد ساده شده است. فایل‌های ASP، اجزا و گزینه‌های پیکربندی همگی فایل‌هایی هستند که شما آنها را با هم منتقل خواهید کرد. نیازی به ثبت اجزا وجود

نداشته و شما می‌توانید با استفاده از فایل‌های مبتنی در قالب XML تغییرات پیکر بندی را مشخص

نمایید ASP.NET. از طریق ایجاد قابلیت میزبانی برای نسخه‌های مختلف یک جز بر روی یک

سیستم وابستگی‌های نسخه‌های نرم افزاری را نیز ساده کرده است.

تبدیل کد به زبانهای چندگانه

ASP.NET و محیط کاری .NET ارائه شده از سوی ماکروسافت شامل سه زبان اصلی:

VB.NET, C# و Jscript.NET میباشد. البته سایر تولید کنندگان زبانهای بسیار بیشتری مانند

Perl.NET و یک Cobol.NET نسخه از Python را نیز در اختیار دارند.

Jscript به منظور تبدیل به زبانی کارآمد و دارای شی گرای .NET. مجبور به بازسازی شد. برنامه

نویسان با تجربه Jscript نسبت به موارد افزوده شده احساس راحتی و رضایت میکنند.

VB.NET جایگزین Vbscript شده است و از لحاظ عملکرد شباهت زیادی به آن دارد بطوری که

یادگیری آن برای برنامه نویسان Vbscript مشکل نخواهد بود. این زبان به همراه Jscript

دستیابی کامل به هر آنچه که NET. میتواند ارائه کند (مانند شی گزایی کامل برای اولین بار) را فراهم مینماید.

C# همانند J++ mark 2 توصیف شده. ولی در واقع C# بسیار فراتر از آن است. C# را در حقیقت باید ++C ای دانست که از ابتدا طراحی شده. مشکلات ++C کاملاً شناخته شده اند و در نتیجه نیازی به پرداختن به آن در اینجا نیست. تنها نکته ای که باید بدان اشاره نمود این است که در ++C، شی گزایی آخرین راه حل انتخابی بوده است. در حالی که در C#، این مسئله به عنوان پایه در نظر گرفته شده است.

تمامی عملکردها و پشتیبانی از محیط کاری NET. برای تمامی زبانهای NET. فراهم گردیده و علاوه بر این اشیایی که در یک زبان نوشته میشوند تحت هر زبان دیگری قابل استفاده و تولد بوده و به ارث میرسند. این قابلیت مفهومی بسیار قدرتمند بوده و نظریه عدم وابستگی زبان را مطرح میکند. این قابلیت از طریق فن آوری Common Language Runtime قابل دستیابی میباشد CLR.

کرد زبان NET. را دریافت کرده و آن را دریافت نموده و آن را به زبان واسطه ای (زبان واسط ماکروسافت) MSIL تبدیل می کند، پس این زبان واسط به کدی دودویی ویژه دستگاه مقصد کامپایل می شود. ویژگی زبان واسطه یکی از سیستم های عامل و برنامه های غیر پنجره های مانند Portable.NET و Mono در حوزه منبع باز و محصولات ارائه شده از Borland و Corel در راه هستند.

## مقایسه ASP.NET و مدل‌های قبلی ASP

اولین تفاوتی که یک برنامه نویس با تجربه ASP میفهمد این است که پشتیبانی از VBscript به نفع VB.NET کم شده است. البته این مسئله بر خلاف ظاهر آن مشکل ایجاد نمیکند چون کد آنها بسیار به یکدیگر شباهت دارند و VB.NET زبانی کاملاً قدرتمند است بطوریکه محیطی بسیار کامل تر از VBscript دارد.

همان طور که در بالا اشاره شد، تمامی زبانهای ASP.NET شی گرا بوده و در سرور کامپایل میشوند این امر فواید بسیاری از جمله در بخشهایی که به بهبودهای بیشتری نیاز است مانند کارایی ماندگاری و قابلیت مدیریت به همراه داشته است.

با استفاده از ASP قدیمی شما باید تمامی برنامه خود را از ابتدا کد نویسی کنید ASP.NET دارای قابلیت های زیادی در صرفه جویی نیروی کار به منظور آسانی انجام کار میباشد. فرم های وب با نگاهی جدید به فرم ها در صفحات وب عرضه شده اند که در واقع شیوه جدید VB Raid Development میباشد.

برنامه نویسان با استفاده از فرم های وب از اجزا جدید فرم بهره میبرند که شما میتوانید آنها را به طریق مرسوم یا از طریق کد بیافزایید. این اجزا به برنامه نویسی قابلیت برنامه نویسی رویداد گرا را در

سمت سرور میدهند. شما میتوانید کد طرح سیستم و وظایف را با بکارگیری کد در پشت صفحاتی که

از ساختار وراثتی برای افزایش متدها به فرم استفاده میکنند، جداسازی نمایید. کنترل - های فرم

NET. حالت جلسه را حفظ میکنند به طوری که ورود کاربران حین ارسال صفحه حفظ میشود و

مقادیر ویژه کنترل ها در کدهای ASP در دسترسی قرار میگیرند ، بدون اینکه در درخواست برای

شی خواسته شده ، دوباره مرتب شوند.

کتابخانه های اصلی در بردارنده توانایی های جدید که تا قبل توسط شرکت های دیگر ارائه مانند

System میشود . ابزارهای ترسیم که امکان ساخت تصاویر متحرک را میدهد ، امکان ارسال فایل

وابسته به مرورگر به سرور و خدمات سیستمی شبکه برای کار با TCP/IP و DNS از این خدمات

میباشد.

با کمک وب سرویسهای پشتیبانی قرار داده شده برای SOAP شما میتوانید نرم افزارها و کد را

توزیع کنید . اسکریپت های ASP.NET شما میتوانند خدمات موجود در سراسر وب را مورد استفاده

قرار دهند و زیر برنامه ها را بعنوان نوعی خدمات در معرض نمایش بگذارند و یا اینکه آنها را منتشر

کنند.

نصب و راه اندازی شامل پیکر بندی سرورهای انتقالی که پیکر بندی آنها قبلاً از طریق MMC در دسترس قرار نگرفته و اکنون توسط فایل های XML پیاده سازی شده اند ، دارای اهمیت میباشد .  
اکنون دیگر نیازی به ثبت یا حذف اجزا وجود ندارد و سرور میتواند نسخه های متعدد یک جز را بدون ایجاد ناهماهنگی مورد استفاده قرار دهد.

خدمات حساس مربوط به وظایف ، پشتیبانی از برقراری تعادل بار و انتخابهای متعدد در مدیریت وضعیت را افزایش داده است . این انتخابها شامل قابلیت ذخیره سازی اطلاعات وضعیت در یک پایگاه داده سرور SQL و عبور دادن ID جلسه بر روی URL به منظور جلوگیری از لزوم ظهور بلوک های داده ای (Cookie) برای کاربر میباشد.

چگونگی اجرا فایل های ASP توسط سرور

هنگامی که بازدیدکننده سایتی آدرس یک صفحه وب را درخواست میکند ، مرورگر با سرور وب مشخص شده در آدرس URL تماس برقرار نموده و درخواستی برای صفحه ای که در درخواست HTTP به صورت فرمول اعلام میکند. سرور وب به محض دریافت درخواست ، نوع فایل درخواستی را مشخص میکند و پردازش را به سرور وب رسیدگی کننده اختصاصی میفرستد . فایل های

ASP.NET در صورت لزوم به کلاس های صفحات NET. کامپایل شده و سپس اجرا میشوند و نتایج به مرورگر فرستاده خواهد شد.

کامپایل نمودن به این معنی است که در بار گذاری اول برنامه های ASP.NET به مدت زمان

بیشتری برای حاضر شدن نیاز است ولی بعد از کامپایل نمایش آنها به اندازه زیادی سریعتر میشود.

کامپایل و تحویل صفحات وب

عملیات کامپایل و تحویل صفحات ASP.NET طی مراحل زیر انجام می گیرد:

1- IIS میان URL مورد در خواست با فایل موجود در سیستم فیزیکی فایل (دیسک سخت) مقایسه

ای را انجام میدهد ، این مقایسه با ترجمه مسیر واقعی ( به عنوان مثال ( `site/index.aspx` )به

مسیری نسبت به ریشه وب سایت (به عنوان مثال

) `wwwroot\site\index.aspx` صورت میپذیرد.

2- هنگامی که فایل بدست آمد ، پسوند فایل ( `.aspx` ) با فهرستی از انواع معین فایلها مقایسه

میشوند تا معین گردد که آیا فایل باید برای بازدید کننده ارسال شود و یا برای پردازش فرستاده

شود.

3- اگر نسبت به زمانی که فایل آخرین تغییرات را متحمل شده ، این اولین بازدید از صفحه باشد ،

کد ASP با استفاده از کامپایلر Common Language Runtime به زبان اسمبلی شده (به

) MSIL پس برای اجرا به کد دودویی ویژه ماشین کامپایل میشود.

4- کد دودویی یک dll. از کلاس NET. است که در یک محل موقتی ذخیره میشود.

5- دفعه بعدی که صفحه در خواست میشود ، سرور بررسی میکند که آیا کد تغییر یافته است یا نه .

اگر همان کد قبلی باشد آنگاه مرحله کامپایل حذف شده و کد کلاس کامپالی شده پیشین اجرا

میگردد ، در غیر این صورت کلاس حذف شده و منبع جدید دوباره کامپایل میشود.

6- کد کامپایل شده اجرا گردیده و مقادیر در خواستی مانند فیلدهای ورودی فرم یا پارامترهای

URL تفسیر میشوند.

7- اگر برنامه مویسی از فرمهای وب استفاده کرده باشد ، آنگاه سرور میتواند نرم افزار مورد استفاده

بازدید کننده را شناسایی نموده و صفحاتی را مطابق نیازمندی های بازدید کننده مناسب سازی شده

اند منتقل نماید . به عنوان نمونه می توان به کد ویژه Netscape یا کد زبان نشانه گذاری بی سیم

(WML) برای دستگاه های موبایل اشاره نمود.

8- هر به مرورگر وب بازدید کننده ارسال خواهند شد.

9- عناصر فرم به اسکریپت و نشانه های سمت کلاینت یعنی HTML و JavaScript برای

مرورگرهای وب و WMLScript و WMLScript برای دستگاه های موبایل تبدیل خواهند شد .

گاهی به JSP

Java Server Page (JSP) نوعی فن آوری مبتنی بر زبان JAVA می باشد و امکان تولید وب سایتهای پویا (فعال) فراهم می سازد JSP. توسط

شرکت SUN Microsystems برای برنامه نویسی سمت سرور تولید شده . فیلدهای JSP همان فایلهای HTML می باشند همراه قطعات ویژه شامل



کدهای Java که قابلیت پویای صفحات را ایجاد می کند.

شکل زیر یک وب نوعی را نشان می دهد ، سرویس گیرندگان مختلف از طریق اینترنت به وب سرور متصل می شوند. در این مثال وب سرور apache که

بسیار محبوب می باشد بروی سیستم عامل UNIX اجرا شده.

در ابتدا صفحات وب به صورت ایستا نمایش داده می شدند . معمولاً این اولین تجربه مردم در تولید صفحات وب بود ت سایتهای شخصی و اطلاعات

قروش شرکتها را ثبت کنند . بعد از مدتی Perl و C زبانهای بودند که بر روی سرور برای ایجاد ماهیت پویا قرار داده می شدند.

بعد از مدت کوتاهی اکثر زبانها از جمله VB , C++ , Delphi و Java این قابلیت را داشتند که برنامه هایی را با استفاده از خاصیت پویایی و استفاده

از داده های متنی و پایگاه داده تولید کنند.

15

این برنامه ها به نام برنامه های CGI سمت سرور شناخته شدند ASP. توسط شرکت ماکروسافت تولید شد تا به برنامه نویسان HTML این امکان را

بدهد که براحتی مندرجات پویا را که توسط سرور رایگان ماکروسافت (IIS) پشتیبانی می شد مورد استفاده قرار دهند .JSP. همتهای ASP است که در

این بخش ارائه خواهد شد.

شکل زیر وب سروری را نشان میدهد که از فایل‌های JSP پشتیبانی میکند توجه داشته باشید که وب سرور به پایگاه داده ها متصل میباشد.

کدهای JSP بر روی وب سرور در JSP Servlet Engine اجرا می شوند JSP Servlet Engine. به صورت پویا HTML تولید می کند و خروجی

HTML را به مرورگر سرویسگیرنده می فرستد.

چرا از JSP استفاده می کنیم؟

آموختن JSP آسان است و این امکان را به برنامه نویسان میدهد تا برنامه ها و وب سایتها را به روشی آشکار و استاندارد تولید کنند.

اساس JSP مبتنی بر زبان Java می باشد که زبانی شی گراست JSP. محیطی قدرتمند را برای گسترش وب عرضه می کند

دلایل اصلی استفاده از JSP

1- قابلیت اجرا در محیط های مختلف

2- اجزا توسط Javabeans and EJB قابلیت استفاده دوباره دارند.

3- امتیازات Java

شما میتونید فایل های JSP را از محیطی به محیط دیگر انتقال دهید (وب سرور و . Servlet Engine)

HTML و گرافیک که در مرورگر وب نمایش داده میشود در کلاس لایه نمایش (presentation layer) قرار میگیرند و کد JSP در سرور در کلاس

پیاده سازی (implementation) قرار میگیرند. با داشتن پیاده سازی و نمایش جداگانه طراحان تنها در بخش نمایش و برنامه نویسان JSP در پیاده

سازی تمرکز می کنند...

مقایسه JSP و ASP

ASP و JSP در عملیاتی که انجام میدهند بسیار مشابه هستند JSP. ممکن است دارای دوره یادگیری طولانیتری باشد. هر دو قابلیت گنجاندن کد در

HTML برای کار با پایگاه داده ها و مندرجات پویا را دارند.

از آنجایی که ASP معمولاً در محیط های ماکروسافت NT , IE یافت می شود و JSP در هر محیطی که با خصوصیات J2EE مطابقت داشته باشد کار

میکند.

JSP اجازه می دهد که اجزا دوباره به وسیله Javabeans , EJB's استفاده شوند ASP . امکان استفاده از COM , ActiveX را فراهم میسازد.

مقایسه JSP و Servlet

کی Servlet کلاسی از Java میباشد که امکان ویژه ای از سرویسهای سمت سرور را فراهم میسازد. نوشتن کد های HTML در Servlet کار بسیار

دشواری میباشد.

در Servlet شما برای تولید HTML باید تعداد زیادی دستور Println استفاده کنید.

معماری JSP

JSP بر روی فن آوری Servlet شرکت SUN پیاده شده است. فایل JSP صفحه ای HTML است که در آن تکه کدهای JSP گنجانده شده است.

این تکه کدها می توانند شامل کد های Java باشند. پسوند فایل های .jsp. JSP ندرتاً .htm , .html می باشد. موتور(کامپایلر)

JSP فایل .jsp را تجزیه کرده و فایل کد Servlet را تولید میکند. بعد فایل servlet را کامپایل کرده و فایل Class را تولید میکند. این کار تنها بار

اول انجام میگیرد و این دلیل کند کار کردن JSP برای بار اول میباشد

مراحل لازم برای پاسخ گویی به درخواست صفحه JSP

1- کاربر وارد سایت شده و مرورگر در خواست را از طریق اینترنت ارسال می کند.

2- در خواست به وب سرور ارسال می شود.

3- وب سرور تشخیص می دهد که فایل در خواستی .jsp بوده و فایل را به JSP Servlet Engine می فرستد.

4- اگر فایل JSP برای مرتبه اول صدا زده شده باشد فایل تجزیه می شود در غیر این صورت به مرحله ۷ می رود.

5- این مرحله مرحله تولید کد Servlet می باشد. همه HTML باید به دستورهای println تبدیل شود.

6- کد Servlet کامپایل شده و Class را تولید می کند.

7- کد HTML تولید می شود.

8- HTML از طریق اینترنت به مرورگر کاربر ارسال می شود. و نتیجه بر روی مرورگر کاربر نمایش داده می شود.

نتیجه گیری

این مقاله چکیده ای بود از ASP و JSP که ما را با خصوصیات ابتدایی و پایه و فرایند تولید صفحات وب فعال آشنا می ساخت.

با توجه به تلاش های گسترده ای که تا کنون در امر تولید صفحات وب شده باید پی به ارزش این فن آوری برده باشیم. استفاده از صفحات وب با توجه

به دنیای امروزی که دنیای ارتباطات و انتقال اطلاعات می باشد بسیار ضروری است و باید از این فن در اطلاع رسانی، آموزش، و تبلیغات برای پیشرفت

جامعه و رسیدن به اهداف عالی آن تلاش نماییم