

## Ajax : تحولی بزرگ در عرصه وب ( بخش پنجم )

آنچه تاکنون گفته شده است :

- [بخش اول](#) تاثیر متقابل وب و نرم افزار بر یکدیگر
- [بخش دوم](#) Ajax و فناوری های مرتبط با آن
- [بخش سوم](#) بررسی نمونه برنامه های مبتنی بر Ajax
- [بخش چهارم](#) برنامه نویسی غیرهمزمان در برنامه های وب

پس از آشنائی با اصول اولیه Ajax در طی چهار مقاله ، پیاده کنندگانی که علاقه مند به استفاده از این فناوری در برنامه های وب می باشند این پرسش را مطرح می نمایند که آیا برای استفاده از پتانسیل های Ajax می بایست از يك فریمورک و یا toolkit خاص استفاده کرد ؟ در ادامه ضمن پاسخ به این سوال ، به بررسی معماری ASP.NET Ajax خواهیم پرداخت .

### چرا به يك فریمورک Ajax نیاز داریم ؟

بدون بهره گیری از پتانسیل های يك toolkit و یا يك فریمورک ، پیاده سازی برنامه های مبتنی بر Ajax کار ساده ای نخواهد بود و پیاده کنندگان با مسائل متعددی مواجه خواهند شد :

- **سازگاری بین مرورگرها :** گذشته از پیاده سازی نسخه های متعدد و مختلف از شی XMLHttpRequest ، هر مرورگر يك نسخه متفاوت از DOM را پیاده سازی کرده است . به روز نگه داشتن تغییرات بین مرورگرها و مدیریت تشخیص مرورگرها می تواند يك فرآیند مشکل و خسته کننده را برای پیاده کنندگان به دنبال داشته باشد . یکی از اهداف مهم يك toolkit و یا فریمورک ، تفکیک پیچیدگی ها و تفاوت ها است تا پیاده کنندگان بتوانند با بکارگیری مجموعه ای از امکانات رابط برنامه نویسی ( API ) ، عملیات مشابه و یکسانی را انجام دهند .
- **ضرورت درک عمیق تر از جاوا اسکریپت :** جاوا اسکریپت ذاتاً يك زبان پیچیده نیست ، گرچه ، تعداد زیادی از پیاده کنندگان برنامه های وب این عقیده را ندارند . جاوا اسکریپت ویژگی هائی نظیر شی گراء و یا Type safe را که بتواند انتظار برنامه نویسان را تامین نماید ، ارائه نمی نماید . مفاهیمی نظیر توارث ، اینترفیس ها و رویدادها را می توان در جاوا اسکریپت شبیه سازی کرد ولی پیاده سازی آنها برای پیاده کنندگان کار مشکلی خواهد بود . اشکال زدائی و عدم حمایت از زبان های اسکریپت نویسی سمت سرویس گیرنده در محیط های IDE ( برگرفته شده از environments integrated development ) نیز به پیچیدگی کار می افزاید .

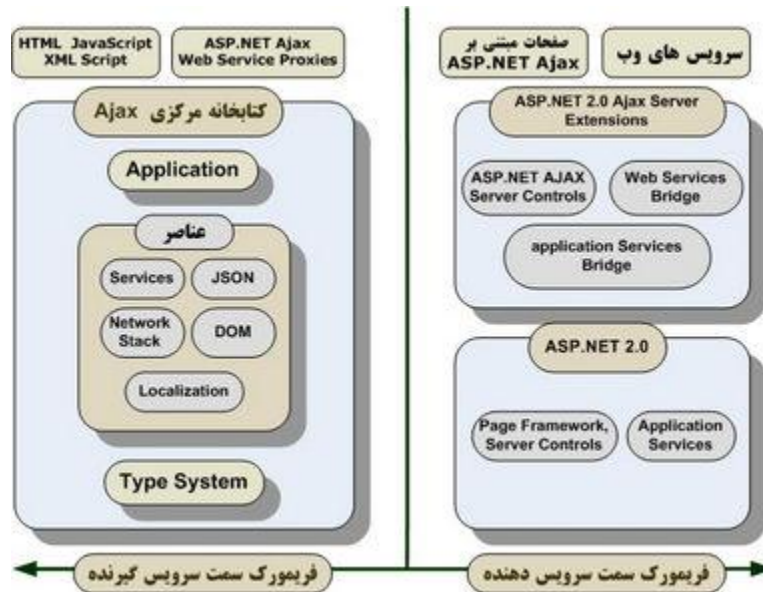
با توجه به موارد فوق ، توصیه می گردد در زمان پیاده سازی برنامه های وب مبتنی بر Ajax از يك فریمورک و یا toolkit استفاده شود و پیاده کنندگان خود را درگیر برخی کارهای پیچیده و تکراری ننمایند . ASP . NET Ajax یکی از فریمورک های ارائه شده در این خصوص است که در ادامه با معماری آن بیشتر آشنا می شویم .

## معماری فریمورک ASP. NET Ajax

فریمورک ASP. NET Ajax ، به پیاده کنندگان این امکان را می دهد که بتوانند برنامه های وب قدرتمندتر ، با توان پاسخگوئی بیشتر و سازگار با مرورگرهای مختلف را ایجاد نمایند . در نگاه اول ممکن است این گونه برداشت شود که فریمورک يك کتابخانه Ajax است . با این که استنباط فوق درست است ولی تصویری واقعی از آنچه وجود دارد را در ذهن ایجاد نمی کند.

با بررسی معماری فریمورک فوق و آشنائی با نحوه عملکرد هر يك از عناصر موجود در این ساختار ، با قابلیت های فریمورک ASP. NET Ajax بیشتر خواهیم شد .

شکل 1 ، ساختار معماری فریمورک ASP. NET Ajax را نشان می دهد . اولین چیزی که در شکل فوق مشهود است ، پوشش این فریمورک در دو سمت سرویس گیرنده و سرویس دهنده است . علاوه بر مجموعه ای از کتابخانه های سمت سرویس گیرنده و عناصر ، با مجموعه ای از امکانات در سمت سرویس دهنده مواجه هستیم که امکان بکارگیری آنها توسط کنترل های سرویس دهنده ASP. NET و سرویس ها وجود دارد .



شکل 1 : ساختار معماری فریمورک ASP. NET Ajax

## فریمورک سمت سرویس گیرنده

یکی از نکات جالب در ارتباط با فریمورک سمت سرویس گیرنده ، عدم وابستگی کتابخانه مرکزی به عناصر سرویس دهنده است . از کتابخانه فوق می توان جهت پیاده سازی برنامه های نوشته شده توسط PHP و یا ColdFusion و سایر زبان های برنامه نویسی و یا پلت فرم ها نیز استفاده کرد .

با توجه به انعطاف پذیری معماری فوق ، می توان آن را بطور منطقی به دو بخش تقسیم کرد : فریمورک سرویس دهنده و فریمورک سرویس گیرنده تقسیم نمود .

آگاهی از نحوه عملکرد معماری فوق در سمت سرویس گیرنده ، برای پیاده کنندگان سمت سرویس دهنده نیز ضروری است چراکه این بخش آغازگر حیات يك صفحه وب می باشد .

## Microsoft Ajax Library

در هسته فریمورک سمت سرویس دهنده Microsoft Ajax Library قرار دارد که از آن با نام کتابخانه مرکزی و یا هسته نام برده می شود. این کتابخانه شامل مجموعه ای از فایل های جاوا اسکریپت است که می توان از آنها صرفنظر از ویژگی های سرویس دهنده استفاده کرد. در ادامه به بررسی هر يك از اجزاء و یا لایه های کتابخانه فوق خواهیم پرداخت.

کار خود را با بررسی لایه Type system که زیربنای تمامی لایه های دیگر است، آغاز می نمایم. (در نسخه های قبلی ASP. NET Ajax با کد نام Atlas، به کتابخانه مرکزی Client Script Library گفته می شد).

- **System Type**: هدف از لایه فوق، معرفی مفاهیم برنامه نویسی شی گراء نظیر کلاس ها، توارث، اینترفیس ها و handling event برای جاوا اسکریپت است. این لایه همچنین نوع های موجود جاوا اسکریپت را توسعه می دهد. مثلاً نوع String و Array در جاوا اسکریپت توسعه یافته اند تا قابلیت هائی مشابه را برای پیاده کنندگان برنامه های وب ASP. NET ایجاد نمایند. type system اساس کار سایر اجزاء این کتابخانه را فراهم می نماید.

- **Component**: در لایه بعدی کتابخانه مرکزی، لایه component قرار دارد که بر روی زیرساخت type system ایجاد شده است. لایه فوق عملیات حیاتی و مهمی را برای کتابخانه مرکزی انجام می دهد. این لایه امکانات حمایتی لازم برای سریال سازی JSON (برگرفته شده از JavaScript Object Notation)، ارتباطات شبکه ای، محلی سازی، تعامل با DOM (برگرفته شده از Document Object Model) و سرویس هائی نظیر تائید و پروفایل را برای برنامه های ASP.NET ارائه می نماید. این لایه همچنین، امکان تولید و ایجاد ماژول هائی با قابلیت استفاده مجدد را که می توان آنها را در گروه هائی به عنوان کنترل ها و رفتارها تقسیم نمود، فراهم می نماید.

- **Application**: لایه فوق بالاترین لایه در کتابخانه مرکزی است که می توان برای آن از يك عنوان تشریحی و توصیفی بهتر استفاده کرد: Application model. همانند چرخه حیات يك صفحه در ASP.NET، این لایه يك مدل برنامه نویسی مبتنی بر رویداد را ارائه می نماید که شما می توانید از آن به منظور کار با عناصر DOM، عناصر نرم افزاری و چرخه حیات يك برنامه در مرورگر استفاده نمایید.

## HTML، جاوا اسکریپت و اسکریپت XML

صفحات وب مبتنی بر Ajax در ASP. NET با استفاده از HTML، جاوا اسکریپت و يك گرامر تعریفی مبتنی بر XML که به آن XML script گفته می شود، نوشته می گردند. بدین ترتیب پیاده کنندگان دارای گزینه های متعددی برای نوشتن کد صفحات وب در سمت سرویس گیرنده می باشند. می توان کد را به صورت اخباری و با استفاده از XML Script و یا به صورت دستوری با جاوا اسکریپت پیاده سازی کرد. عناصر تعریف شده در XML Script شامل يك تگ اسکریپت جدید به صورت زیر می باشند.

```
<script type="text/xml-script">
```

مرورگر قادر به تشخیص تگ Script است ولی دارای يك مکانیزم برای پردازش نوع xml-script نمی باشد . در مقابل ، فایل های جاوا اسکریپت از فریمورک ASP.NET Ajax می توانند اسکریپت را تفسیر و يك نمونه از عناصر و کنترل های موجود بر روی صفحه را ایجاد نمایند . کد زیر ، نحوه استفاده از XML Script به منظور نمایش يك پیام پس از استقرار صفحه در حافظه مرورگر را نشان می دهد .

XML-Script : گزینه ای برای پیاده سازی صفحات مبتنی بر Ajax

```
<script type="text/xml-script">
<page xmlns="http://schemas.microsoft.com/xml-script/2005">
  <components>
    <application load="page_load" />
  </components>
</page>
</script>
<script type="text/javascript">
  function page_load(sender, e) {
    alert("Hello from XML-Script!");
  }
</script>
```

در مثال فوق ، يك تابع نوشته شده به زبان جاوا اسکریپت با نام Page\_load به صورت تعریفی به رویداد Load در چرخه حیات صفحه نسبت داده شده است . با اجرای این صفحه ، تابع Page\_Load فراخوانده می شود تا يك پیام برای سرویس گیرنده نمایش داده شود .

شاید برای شما این سوال مطرح شده باشد که چرا در مقابل جاوا اسکریپت می بایست از XML Script استفاده کرد ؟ ( عکس این سوال نیز می تواند مطرح شود ) . در پاسخ می توان به تمایل پیاده کنندگان و توانمندی آنها اشاره کرد . برخی از پیاده کنندگان ترجیح می دهند که از يك زبان نشانه گذاری در مقابل اسکریپت استفاده نمایند . در مقابل تعداد دیگری از پیاده کنندگان استفاده از زبان جاوا اسکریپت را در مقابل يك زبان نشانه گذاری ، ترجیح می دهند . به هر حال از دو رویکرد فوق می توان استفاده کرد و هر يك دارای مزایا و محدودیت های مختص به خود می باشند.

### ASP.NET AJAX service proxies

فریمورک در سمت سرویس گیرنده قابلیت فراخوانی سرویس های وب را از طریق جاوا اسکریپت و به کمک مجموعه ای از پروکسی های سمت سرویس گیرنده که از طریق سرویس دهنده ایجاد شده اند ، فراهم می نماید . پروکسی ها را می توان به منزله يك مرجع وب در کد مدیریت یافته دات نت در نظر گرفت . يك پروکسی کلاسی است که به عنوان يك اینترفیس برای آیتم دیگر عمل می نماید . در این حالت ، يك سرویس وب .

## جمع بندی فریمورک سمت سرویس گیرنده

تا این جا يك دید کلی و اولیه نسبت به فریمورک سمت سرویس گیرنده پیدا کردیم . اجازه دهید سفری داشته باشیم به سمت سرویس دهنده و ببینیم که این فریمورک در سمت سرویس دهنده چه حرفی برای گفتن دارد و از چه نوع معماری تبعیت می کند .

## فریمورک در سمت سرویس دهنده

در بالاترین سطح ASP. NET 2.0 ، مجموعه ای از کنترل ها و سرویس های ارزشمند وجود دارد که فریمورک دات نت را جهت حمایت از Ajax تحت تاثیر قرار می دهد . این لایه از فریمورک سرویس دهنده ، ASP.NET AJAX server extensions نامیده شده و مشتمل بر سه بخش مجزا است :

- کنترل های سرویس دهنده Ajax
- bridge web services
- bridge services application

در ادامه هر يك از امکانات فوق را به اختصار توضیح می دهیم .

## کنترل های سرویس دهنده Ajax

مجموعه ای جدید از کنترل های سرویس دهنده می باشند که به جعبه ابزار ASP. NET اضافه می شوند . دو کنترل آنها شاخص تر از سایر کنترل ها می باشند . اولین کنترل ، ScriptManager است که به عنوان مغز يك صفحه وب مبتنی بر Ajax تلقی می گردد . یکی از چندین مسئولیت کنترل فوق ، مدیریت ناحیه ای از صفحه است که در حین Postback غیرهمزمان می بایست بطور پویا بهنگام گردد.

دومین کنترل ، UpdatePanel است که از آن به منظور تعریف ناحیه ای بر روی صفحه که می بایست به صورت جزئی بهنگام گردد ، استفاده می شود . با تلفیق توانمندی دو کنترل اشاره شده وضعیت بخش رابط کاربر يك برنامه وب بطرز محسوسی بهبود خواهد یافت ( در واقع Postback سنتی جای خود را به Postback غیرهمزمان می دهد ) . پیامد بکارگیری کنترل های فوق ، بهنگام سازی نواحی خاص و مشخص شده ای بر روی صفحه ، در مقابل refresh تمامی صفحه است .

سایر عناصر server extension ، سرویس هائی را شامل می شوند که به منزله يك پل ارتباطی بین سرویس گیرنده و سرویس دهنده رفتار می نمایند .

## web services bridge

برنامه های وب عموماً " ، محدود به منابع موجود بر روی سرویس دهنده محلی می باشند . گذشته از منابع خارجی اندکی نظیر فایل های تصویر و CSS ، برنامه ها مجاز به دستیابی به منابع در حوزه برنامه سرویس گیرنده نمی باشند . به منظور غلبه بر این موانع ، server extensions در فریمورک ASP. NET Ajax شامل يك web service bridge است که يك gateway برای فراخوانی سرویس های وب از طریق اسکریپت های سمت سرویس گیرنده را فراهم می نماید . تامین داده از سمت سرویس دهنده برای استفاده در سمت سرویس گیرنده یکی از موارد کاربرد این فناوری است .

## bridge Application Service

با توجه به ارتباط تنگاتنگ ASP. NET Ajax با ASP. NET ، دستیابی به برخی از سرویس های application نظیر تائیدیه و یا پروفایل را می توان به يك برنامه موجود اضافه نمود . این ویژگی باعث می شود بتوان عملیاتی نظیر بررسی اطلاعات حساس يك کاربر و دستیابی به اطلاعات پروفایل را که توسط اسکریپت های سمت سرویس گیرنده ارسال شده اند را انجام داد .

### خلاصه

در این مقاله ضمن اشاره به ضرورت استفاده از يك فریمورک در برنامه های وب مبتنی بر Ajax ، با عناصر موجود در سمت سرویس دهنده و سرویس گیرنده فریمورک ASP.NET AJAX آشنا شدیم . در بخش بعد با نحوه تعامل این عناصر در جهت تامین خواسته پیاده کنندگان برنامه های وب مبتنی بر Ajax آشنا خواهیم شد . بدین منظور به بررسی دو سناریوی مختلف خواهیم پرداخت : مدل پیاده سازی با محوریت سرویس گیرنده و مدل پیاده سازی با محوریت سرویس دهنده .