

Ajax : تحولی بزرگ در عرصه وب (بخش دوم)

در [بخش اول](#) ضمن بررسی تاثیر متقابل وب بر نرم افزار ، مروری داشتیم به وب ایستا و وب پویا و این که برای انجام پردازش های مورد نیاز در برنامه های وب می توان از امکانات و فناوری های متعددی در سمت سرویس گیرنده و سرویس دهنده استفاده کرد . هدف از بیان مطالب فوق ، پاسخ به این سوال بود که چرا به وجود يك فناوری دیگر نظیر Ajax نیاز است (گرچه Ajax يك فناوری نیست و مجموعه ای از فناوری ها را شامل می شود) . با این که ایده فناوریهای نظیر Ajax در گذشته نیز وجود داشته است و پیاده کنندگان برنامه های وب در زمان خود با بکارگیری برخی فناوری ها و ترفندها توانسته بودند مشکل خود را تا حدود زیادی برطرف نمایند ولی گذشت زمان و گسترش باور نگرانی اینترنت ، باعث شد تا استفاده از بستر فوق برای ارائه سرویس های online در دستور کار اکثر سازمان ها و شرکت ها در اقصی نقاط جهان قرار بگیرد . همین موضوع باعث شد که دگر باره جامعه بزرگ طراحان و پیاده کنندگان برنامه های وب این نیاز را احساس کنند که به يك راه حل جامع و سیستماتيك برای برخورد با مسائل خود (خصوصا " تعامل با کاربر) در دنیای نوین برنامه نویسی وب نیاز دارند . دنیایی که در آن دقت و سرعت در ارائه خدمات online با بکارگیری کمترین منابع و تولید بیشترین بهره وری ، رمز ماندگاری يك سازمان در مدار رقابت و پیشرفت است . همین امر باعث شد که شرکت های بزرگ تولید کننده نرم افزار و خدمات online نیز دست به کار شوند و چیزی را خلق کنند که ما امروزه از آن با نام Ajax یاد می کنیم .

در این بخش به بررسی موارد زیر خواهیم پرداخت :

- تولد Ajax و ریشه یابی علت انتخاب این نام
- Ajax و فناوری های مرتبط با آن
- اصول حاکم بر فناوری Ajax .
- نحوه کارکرد Ajax در برنامه های وب

تولد Ajax

در فوریه سال 2005 ، Jesse James Garrett از Adaptive Path مقاله ای را با عنوان " [يك رویکرد جدید به برنامه های وب](#) " بر روی وب سایت خود منتشر کرد . وی در این مقاله ضمن اشاره به این موضوع که فاصله بین برنامه های وب و desktop در حال از بین رفتن است ، به بررسی دو نمونه از برنامه های وب جدید با نام Google Suggest و Google Maps پرداخته بود . Garrett پس از تشریح نحوه عملکرد برنامه های فوق ، برای اولین مرتبه از واژه Ajax در مقاله خود استفاده کرد .

وی در مقاله خود دنیایی را پیش بینی کرده بود که در آن برنامه ه ای وب از لحاظ قابلیت دستیابی ، پاسخ گوئی و سادگی قابل مقایسه با برنامه های desktop خواهند شد .

Ajax چیست ؟

- Ajax برگرفته شده از XMLHttpRequest Asynchronous است و مشتمل بر مجموعه ای از فناوری ها است ، نه صرفا " يك فناوری " .

- Ajax ، اساس و پایه چیزی است كه امروزه از آن با نام وب 2 نام برده می شود (نسل جدید وب) . ایده اصلی ، ایجاد برنامه های وبی است كه دارای شكل ظاهری و توانمندی هایی مشابه با برنامه های desktop باشند .
 - Ajax چنڱی بیشتر از يك نگرش جدید به برنامه های وب نیست . نگرشی كه در آن صرفاً " يك حجم اندك از اطلاعات بین سرویس گیرنده و سرویس دهنده با هدف افزایش تعامل كاربر با برنامه ، مبادله می گردد .
 - Ajax ، مشتمل بر مجموعه ای از فن آوری ها است كه هر يك دارای جایگاه مختص به خود می باشند :
- فن آوری های استاندارد presentation نظیر XHTML و CSS
- بهنگام سازی پویای يك صفحه مستقر شده در حافظه با استفاده از DOM (برگرفته شده از Document Object Model)
- مبادله و پردازش داده با استفاده از XML (فرمت مبادله داده) و XSLT (تبدیل XML به XHTML)
- بازیابی داده غیرهمزمان با استفاده از XMLHttpRequest (كارگزار اولیه مبادله اطلاعات)
- و جاوا اسکریپت كه همه چیز را به یكدیگر مرتبط می نماید و از آن برای برنامه نویسی Ajax engine استفاده می گردد .
- در دنیای واقعی ، از تمامی فناوری های فوق در Ajax استفاده می گردد ولی به وجود HTML/XHTML ، DOM و Javascript بیش از همه نیاز است چون :
- از XHTML برای نمایش اطلاعات استفاده می گردد .
- از DOM برای تغییر بخش هایی از يك صفحه XHTML بدون نیاز به load مجدد صفحه استفاده می گردد.
- از جاوا اسکریپت ، برای مقداردهی اولیه ارتباط بین سرویس گیرنده و سرویس دهنده و بهنگام سازی صفحات وب به كمك DOM استفاده می گردد .
- يك عنصر اساسی و مهم دیگر كه در مقاله Garrett به آن اشاره نشده است ، ضرورت انجام پردازش های سمت سرویس دهنده است . تمامی فناوری های اشاره شده (در پاراگراف قبل) ، در ارتباط مستقیم با Ajax engine سمت سرویس گیرنده می باشند . بدون وجود يك سرویس دهنده مطمئن و پاسخگو به منظور ارسال محتویات برای Ajax engine ، در عمل Ajax فاقد كارآئی لازم خواهد بود . صرفنظر از فناوری استفاده شده در سمت سرویس دهنده (نظیر PHP و یا ASP.NET) ، می بایست این اطمینان حاصل گردد كه داده با فرمت صحیح برای Ajax engine ارسال می گردد .
 - از شی XMLHttpRequest درون جاوا اسکریپت برای ارتباط با سرویس دهنده استفاده می گردد تا در ادامه بتوان داده برگردانده شده از سرویس دهنده (با فرمت Xml و یا متن معمولی) را پردازش كرد .
 - از DHTML و CSS برای نمایش نتایج در مرورگو استفاده می گردد .
 - هدف از كنارهم قرار گرفتن تمامی فناوری های اشاره شده ، این است كه شكل ظاهری برنامه های وب و نحوه پاسخ گوئی آنها به خواسته كاربران مشابه برنامه های desktop گردد .

همانگونه كه در نام این فناوری مشخص است ، Ajax مشتمل بر مجموعه ای از فناوری ها است : Asynchronous JavaScript XML

- **غیرهمزمان و یا asynchronous** بدین معنی است که مرورگر لازم نیست منتظر بماند تا داده از سرورس دهنده برگردانده شود و می تواند به محض دریافت داده آن را پردازش نماید . به عبارت دیگر ، انتقال داده در پس زمینه انجام شده و مرورگر مجبور به توقف و انتظار برای تحقق اتفاقی نخواهد بود . رویکرد فوق نشاندهنده یکی از ویژگی های مهم و حیاتی Ajax است . در چنین مواردی ، می توان داده را پس از ارسال توسط سرورس دهنده ، پردازش کرد . لازم نیست تمامی برنامه را معطل رسیدن داده از سمت سرورس دهنده کرد .
- در صورتی که لازم است پردازش های سمت سرورس گیرنده منتظر دریافت داده بمانند ، مکانیزم دریافت و پردازش داده می بایست بطور همزمان انجام شود . در صورتی که يك ارتباط ضعیف بین سرورس گیرنده و سرورس دهنده برقرار شده باشد این موضوع می تواند عملکرد و موفقیت يك برنامه وب را با مشکل مواجه نماید .
- **بخش جاوا اسکریپت واژه Ajax** نیز بسیار مهم است چراکه بالفعل شدن پتانسیل های Ajax در مرورگر توسط آن محقق می گردد . Ajax به کمک جاوا اسکریپت با سرورس دهنده ارتباط خود را برقرار می نماید و در ادامه نیز داده برگردانده شده از سرورس دهنده را پردازش می نماید .
- **بخش xml واژه Ajax دارای چه مسئولیتی است ؟** XML به عنوان يك استاندارد جهانی در عرصه وب مطرح است و به کمک آن می توان از يك روش مبتنی بر متن برای مبادله داده در عرصه اینترنت استفاده کرد . یکی از دلایل مهم گسترش xml ، ماهیت مبتنی بر متن آن است . با توجه به این که طراحی اینترنت بگونه ای انجام شده بود تا بتواند مستندات مبتنی بر متن نظیر اسناد HTML را نمایش دهد ، وجود xml به منزله قلابی است که می توان آن را دور اینترنت انداخت و به صید داده پرداخت . داده ئی که به همراه ساختار خود می تواند امکان پردازش را در سایر سیستم ها فراهم نماید . به همین دلیل است که برنامه های Ajax بگونه ای نوشته می گردند تا بتوانند بر روی داده برگردانده شده از سرورس دهنده با فرمت xml ، عملیات مورد نظر را انجام دهند . به عبارت دیگر ، پس از برقراری ارتباط با سرورس دهنده ، داده با فرمت xml برگردانده خواهد شد .
- xml ، صرفاً یکی از روش های برگرداندن داده است و در صورت لزوم می توان از فرمت هائی دیگر نظیر متن معمولی نیز استفاده کرد .
- در کنار جاوا اسکریپت و XML ، فناوری Ajax با فناوری های دیگری نظیر DHTML و CSS نیز کار می کند . به کمک فناوری های فوق می توان داده موجود در يك صفحه وب را بدون نیاز به load تمامی صفحه بهنگام کرد (صرفاً load بخشی که ضرورت آن احساس می شود) . پتانسیل فوق یکی از اهداف و ویژگی های مهم برنامه های وب مبتنی بر فناوری Ajax محسوب می گردد .
- بخشی از جاوا اسکریپت که امکان Ajax را میسر می سازد ، شی XMLHttpRequest است . شی فوق تقریباً در تمامی مرورگرهای مدرن و پیشرفته از قبل تعبیه شده است . به کمک این شی امکان اتصال به سرورس دهنده و مدیریت داده برگردانده شده از آن در پس زمینه فراهم می گردد .
- اقتدار Ajax نه تنها وابسته به جاوا اسکریپت بلکه مدیون شی XMLHttpRequest است .

اصول Ajax

Ajax هنوز در ابتدای راه است و بسیاری از پیاده کنندگان با این پرسش مواجه هستند که از این فناوری در چه زمانی و در چه نوع برنامه هائی می توان استفاده کرد . عدم شناخت مناسب از این فناوری باعث می شود تا از آن

در برنامه هایی استفاده گردد که نه تنها قابلیت و یا ویژگی جدیدی را به برنامه اضافه نمی نماید بلکه در بسیاری از موارد عدم موفقیت يك برنامه و نارضایتی کاربران آن را به دنبال خواهد داشت .
زمانی می توان این ادعا را داشت که يك برنامه وب مبتنی بر Ajax در اهداف خود موفق بوده است که در آن اصول زیر رعایت شده باشد .

- **به حداقل رساندن ترافیک :** برنامه های Ajax می بایست حتی المقدور حجم اندکی از اطلاعات را برای سرویس دهنده ارسال و یا از آن دریافت نمایند . به عبارت دیگر ، با بکارگیری Ajax می بایست حجم ترافیک بین سرویس گیرنده و سرویس دهنده کاهش یابد .
- **عدم سردرگمی کاربر :** برنامه های وب مبتنی بر فناوری Ajax مدل تعامل با لاربر مختلفی را نسبت به برنامه های وب سنتی معرفی کرده اند . در مقابل وب استاندارد (کلیک کن و منتظر باش) ، برخی برنامه های Ajax از يك رویکرد دیگر برای پیاده سازی بخش رابط کاربر خود نظیر drag-and-drop و یا double-clicking استفاده می نمایند . صرفنظر از نوع مدل انتخاب شده برای پیاده سازی تعامل با کاربر ، مهم ارائه امکانات بخش رابط کاربر بگونه ای است که همواره کاربر در سریع ترین زمان بتواند در خصوص انجام حرکت بعدی خود تصمیم گیری نماید .
- **استفاده منطقی از تجارب مثبت گذشته :** وقت خود را بی خودی به ابداع مدل های تعامل با کاربر جدید که کاربران شما با آنها آشنا نیستند ، هدر ندهید . استفاده از تجارب برنامه های وب سنتی و desktop می تواند پیاده کنندگان را در طراحی يك مدل مناسب جهت تعامل با کاربر کمک نماید .
- **اجتناب از بکارگیری عناصر غیرضروری :** از بکارگیری عناصر غیر ضروری نظیر تکرار انیمیشن و بخش هایی نظیر blinking page اجتناب کنید. بکارگیری این چنین عناصری باعث می شود تا کاربران در زمان انجام فعالیت های مورد نظر دچار سردرگمی شوند .
- **قابلیت دسترسی و رضایت کاربران :** برنامه های Ajax را با در نظر گرفتن رفتار و نوع خواسته کاربران طراحی و پیاده سازی نمائید . برای طراحی و پیاده سازی خود را در يك مکان بسته محبوس ننمائید. در غیر اینصورت همواره این احتمال وجود خواهد داشت که خواسته برخی از کاربران نادیده گرفته شود . قابلیت دسترسی و رضایت کاربران یکی از نکات بسیار مهم در موفقیت يك برنامه کامپیوتری محسوب می گردد .
- **پیشگیری از دانلود تمامی صفحه :** تمامی ارتباط با سرویس دهنده پس از مقدار دهی اولیه دانلود صفحه ، می بایست توسط Ajax engine مدیریت گردد . عدم مدیریت صحیح بر روی فرآیند فوق می تواند بخش رابط کاربر يك برنامه وب را با مشکل اساسی مواجه سازد (نظیر دانلود حجم اندکی از داده در يك مکان و دانلود تمامی صفحه در بخش دیگر)
- **توجه به نیاز کاربران قبل از هر چیز :** برنامه های Ajax را قبل از هر چیز با کاربران خیالی در ذهن طراحی نمائید . سهولت در انجام عملیات متداول می بایست در دستور کار قرار بگیرد . کمتر در اندیشه انجام کارهای تبلیغاتی و یا افکت های غیرضروری باشید .

نقطه مشترك بين تمامی اصول اشاره شده ، قابلیت استفاده از يك برنامه با حداکثر ظرفیت و رضایتمندی کاربران است . هدف اولیه Ajax ، بهبود و ارتقاء بخش رابط کاربر برنامه ها با هدف رضایتمندی بیشتر کاربران است .

نحوه کار Ajax

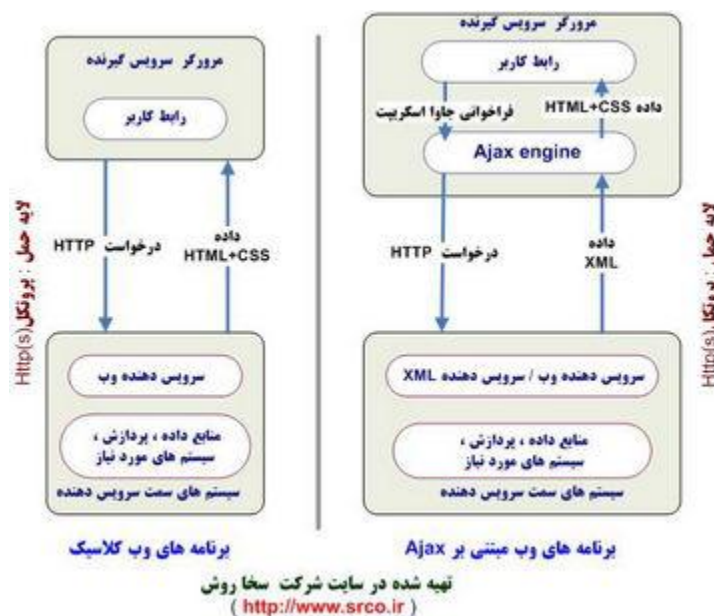
در برنامه های وب سنتی مرورگر مسئولیت مقداردهی اولیه درخواست ها و پردازش آنها جهت ارسال به سرور دهنده وب را برعهده دارد . در مقابل ، در Ajax يك لایه میانی (که Garrett آن را Ajax engine نامید) مسئولیت مدیریت ارتباط برقرار شده را برعهده می گیرد . Ajax engine در واقع يك شی جاوا اسکریپت و یا تابع است و زمانی که به اطلاعاتی از سرور دهنده نیاز باشد ، از آن استفاده خواهد شد .

در مقابل برنامه های وب سنتی که يك لینک به يك منبع دیگر ارائه می گردد (نظیر يك صفحه وب دیگر) ، در برنامه های وب مبتنی بر Ajax ، هر لینک باعث فراخوانی Ajax engine می گردد که وظیفه آن زمانبندی و انجام درخواست بطور غیرهمزمان است .

سرور دهنده (که از لحاظ سنتی اسناد HTML ، تصاویر ، CSS و جاوا اسکریپت را ارائه می نماید) ، بگونه ای پیکربندی می گردد تا داده مورد نیاز را در اختیار Ajax Engine قرار دهد. این داده می تواند متن معمولی ، XML و یا سایر فرمت های مورد نیاز داده باشد. بدیهی است در چنین مواردی ، Ajax engine می بایست قادر به خواندن و تفسیر داده باشد .

پس از دریافت پاسخ لازم از سرور دهنده ، Ajax engine عملیات خود را که اغلب بررسی داده و ایجاد تغییرات لازم در بخش رابط کاربر است ، آغاز می نماید . با توجه به این که در فرآیند فوق به اطلاعات کمتری نسبت به برنامه های وب سنتی نیاز است ، بخش رابط کاربر با سرعت بیشتری بهنگام می گردد و کاربر قادر به انجام کارهای خود با سرعت بیشتری است .

در شکل 1 ، عملکرد برنامه های وب سنتی نسبت به برنامه های وب مبتنی بر Ajax نشان داده شده است .



شکل 1 : عملکرد برنامه های وب سنتی نسبت به برنامه های وب مبتنی بر Ajax

منبع : Adaptive Path

خلاصه

فناوری های مرتبط با Ajax در طی سالیان گذشته مطرح و از آنها در مجموعه ای از نرم افزارها استفاده می گردید . Jesse James Garrett در مقاله خود تمامی فناوری های اشاره شده را در زیر يك سقف و با نام Ajax کنار هم قرار داد . با بکارگیری فناوری Ajax در نرم افزارهای برجسته ای نظیر Suggest Google در سال 2005 ، Ajax در کانون توجه جهانی قرار گرفت . وب 2 ، برای نیل به اهداف خود به Ajax دل بسته است و Ajax می تواند تصویری جدید از برنامه های وب را در عرصه اینترنت به نمایش بگذارد .

وجود برخی خصایص خاص در بخش رابط کاربر برنامه های desktop و نحوه تعامل کاربران با اینگونه برنامه ها ، حلقه گمشده ای در برنامه های وب است. حلقه ای که Ajax سعی دارد خلاء آن را پر کند . قطعاً در آینده ای نه چندان دور شاهد حضور بیشتر برنامه های وبی خواهیم بود که دارای شکل ظاهری و عملکردی مشابه (خصوصاً) تعامل با کاربر) با برنامه های desktop می باشند .

شاید روزی فرا رسد که یکی از واژه های برنامه های desktop و برنامه های وب به نفع دیگری کنار برود و یا هر دو بر يك نام مشترك دیگر به توافق برسند .

در بخش سوم به بررسی نمونه برنامه هائی خواهیم پرداخت که در آنها از فناوری Ajax استفاده می گردد .