

این سناریو رو در نظر بگیرید:

وب سرور ما در همان محلی قرار دارد که SVN Server نصب شده است. می‌خواهیم به ازای هربار Commit تیم به مخزن SVN ما، سایت ارائه شده توسط وب سرور نیز به صورت خودکار به روز شود. چه باید کرد؟!

احتمالا خیلی‌ها تصور می‌کنند که امکان پذیر نیست؛ چون مخزن SVN موجود در سرور، ساختار خودش را دارد و همانند فایل‌های یک پروژه معمولی نگهداری نمی‌شود.

برای انجام اینکار چندین روش موجود است، که تمام آن‌ها به مفهوم hooks در SVN گره خورده است. هرچند hook به معنای قلاب است، اما در اینجا معنای تریگر را دارد. شبیه به تریگرهای SQL Server: پیش یا پس از انجام کار یا رخداد مشخصی، فلان کار را انجام بده. (برای اطلاعات بیشتر می‌توانید به فصل hooks در این کتابچه مراجعه کنید: (+))

در میان این قلاب‌های موجود، می‌توان از قلاب post-commit جهت به روز رسانی یک سایت پس از هر هماهنگ سازی با مخزن SVN استفاده کرد. پیشنهاد من به تمام کسانی که می‌خواهند کار با SVN را شروع کنند استفاده از برنامه رایگان [Visual SVN Server](#) است. این برنامه سازگاری فوق العاده‌ای با محیط ویندوز دارد (از لحاظ تعریف سطح دسترسی‌ها). همچنین تعریف hooks را هم به شدت ساده کرده است. فقط کافی است روی یک مخزن کد تعریف شده در [Visual SVN Server](#) کلیک راست کرده و در برگه باز شده، تنظیمات سطوح دسترسی یا تعاریف Hooks را اضافه نمود (در اینجا اعمال سطوح دسترسی روی پوشه‌ها یا روی فایل‌ها نیز به همان شکل با کلیک راست و کم و زیاد کردن کاربران میسر است؛ همانند دادن دسترسی بر اساس امکانات NTFS و اکتیو دایرکتوری).

بنابراین به صورت خلاصه:

فرض بر این است که مخزن کد SVN ایی را بر روی سرور راه اندازی کرده‌اید. همچنین پوشه‌ای را که می‌خواهید ریشه سایت باشد، مثلا در مسیر دلخواه C:\path\www قرار دارد. برای شروع کار، باید صورت گیرد. یا می‌توان از TortoiseSVN استفاده کرد یا چون مخزن کد در همان سرور است، دستور زیر نیز کار می‌کند:

```
svn checkout file:///c:/svn/MyRepository/trunk C:\path\www
```

سپس یک فایل bat باید درست کنید با محتوای زیر:

```
svn update file:///c:/svn/MyRepository/trunk C:\path\www
```

این فایل bat باید در همان قسمت تعریف hook post-commit استفاده شود.

به این معنا که پس از هر commit، لطفا مسیر C:\path\www را بر اساس آخرین به روز رسانی‌های مخزن کد به صورت خودکار به روز کن. در این حالت اگر فایلی حذف شده باشد، به صورت خودکار از ریشه سایت شما حذف می‌شود و اگر فایل یا فایل‌هایی تغییر کرده باشند نیز سریعا به روز رسانی آن‌ها انجام خواهد شد.

در روش svn update، پوشه‌های مخفی svn نیز در ریشه سایت حضور خواهند داشت. وجود آن‌ها هم الزامی است زیرا update بر همین اساس کار می‌کند.

اگر می‌خواهید این پوشه‌های مخفی وجود نداشته باشند از دستور svn export استفاده کنید. فقط دقت کنید که در این حالت اگر فایلی از مخزن کد حذف شده باشد، باز هم در ریشه سایت وجود خواهد داشت. راه حلی هم که توصیه شده، این است که در

همان bat فایلی که درست می‌کنید ابتدا دستور حذف محتویات پوشه ریشه را صادر کنید و بعد svn export . البته بدیهی است این روش نسبت به svn update کندتر است و svn update به شدت بهینه و سریع می‌باشد.  
یا راه دیگر بجای حذف کردن پوشه موجود و بعد export به آن، استفاده از برنامه‌هایی مانند Robocopy است که می‌توانند عملیات همگام سازی را هم انجام دهند. در این حالت محتوای فایل bat شما شبیه به دستورات زیر خواهد شد:

```
svn checkout file:///c:/svn/MyRepository/trunk C:\temp\Site1 >> output.log
robocopy C:\temp\Site1 C:\path\www *.* /S /XF *.cs *.tmp *.sln *.csproj *.webinfo /XD .svn _svn /PURGE
>> output.log
```

به این معنا که پس از هر commit به مخزن کد (با توجه به تعریف قلاب ذکر شده)، ابتدا یک svn checkout در یک پوشه موقتی (خارج از ریشه اصلی سایت) انجام گردیده و سپس برنامه robocopy با موارد مشابه آن وارد عمل شده و تغییرات را با ریشه اصلی هماهنگ می‌کنند (در اینجا می‌توان مشخص کرد چه فایل‌هایی با پسوندهای مشخص، با ریشه سایت هماهنگ نشوند).

در کل همان روش svn update به نظر سریعتر و مقرون به صرفه‌تر است. اگر از IIS استفاده می‌کنید، به صورت پیش فرض کسی نمی‌تواند محتوای پوشه‌ای را با وارد کردن آدرس آن در مرورگر بررسی کند، همچنین IIS فایل‌هایی را که نمی‌شناسد (پسوند از پیش تعریف شده‌ای در بانک اطلاعاتی آن ندارند)، سرو نمی‌کند و در صورت درخواست آن‌ها، خطای 404 یا "پیدا نشد" به کاربر نهایی ارائه خواهد شد.

## نظرات خوانندگان

نویسنده: رضوی  
تاریخ: ۱۳۹۰/۰۲/۳۰ ۲۰:۵۷:۰۶

با عرض سلام استاد  
واقعا از راهنمایی و کمکتون ممنونم

نویسنده: میلاد  
تاریخ: ۱۳۹۰/۰۲/۳۱ ۱۹:۴۵:۲۵

بخ نظرتون Web Deploy جواب بهتری به این مساله نیست؟  
لطفا راجع به Setup کردنش روی سرور اگر مطلبی دارید به اشتراک بزارید. ممنون می شم.

نویسنده: وحید نصیری  
تاریخ: ۱۳۹۰/۰۲/۳۱ ۲۰:۵۵:۵۴

بحث اینجا کار تیمی است نه تک نفره و deploy به سرور. به همین جهت صحبت از مخزن کد شد و SVN.  
اینها مشکلات web deploy است:  
- با IIS 6 آنچنان سازگار نیست و IIS 7 را طلب می کند. (با روش فوق سرور شما آپاچی هم باشد کار می کند)  
- بحث rollback با webdeploy اصلا معنی ندارد. اما با سورس کنترل به سادگی انجام می شود. فرض کنید الان به اشتباه یک سری کار به مخزن SVN ارسال شده. بلافاصله هم با روش فوق در ریشه سایت قرار گرفته. اصطلاحا revert به نگارش پایدار قبلی در SVN بسیار ساده است.  
+ تمام مزیت های ورژن کنترل را هم لحاظ کنید. با روش فوق دقیقا مشخص است چه کسانی روی فایل ها کار کرده اند و چه زمانی. تاریخچه ارسال ها موجود است. امکان حرکت بین نگارش های مختلف و سوئیچ بین آنها معنا پیدا می کند و ...  
- روش فوق نیاز آنچنانی به داشتن دسترسی بالا روی سرور ندارد.

به همین جهت برای کار حرفه ای با web deploy از برنامه team city و یک سری مخلفات دیگر استفاده می کنند. یک سری 5 قسمتی رو اینجا می تونید پیدا کنید: [\(+\)](#)  
و پس از مطالعه به این نتیجه خواهید رسید که روش فوق پایدارتر و دردسر کمتری دارد. حتی روی یک سرور لینوکسی هم قابل پیاده سازی است

نویسنده: میلاد کرباسی زاده  
تاریخ: ۱۳۹۰/۰۳/۱۶ ۰۸:۵۴:۳۳

ممنون از راهنمایی :-)