

آشنایی با Subversion بخش چهارم

تعامل کاربران با مخزن فایل و ساخت نسخه کاری

هم اکنون که سرور آماده است دیگر کاربران مختلف می‌توانند به آن متصل شده، فایلها را دریافت کرده و کار خود را بر روی پروژه آغاز نمایند. اجازه دهید خود را یک کاربر عادی فرض کنیم که توسط مدیر سیستم به پروژه دعوت شده و قرار است که بر روی آن کار کند. ما ابتدا باید نسخه ای از فایلهای پروژه را بر روی سیستم خود دانلود کنیم تا بتوانیم کار خود را آغاز نماییم. به عمل برداشت فایلها از مخزن فایل و تشکیل نسخه کاری شخصی اصطلاحاً checkout می‌گویند. فرض کنید این جناب مدیری که ما را به شرکت در پروژه دعوت کرده است، آدرس سرور subversion خود را به ما داده و از ما توقع دارد که در اسرع وقت کار خود را بر روی پروژه آغاز نماییم. خوب، کار خیلی ساده است، در واقع باید فایلها را از سرور مرکزی بر روی دستگاه خود منتقل کرده و کار خود را آغاز نماییم. خوب، قطعاً ما می‌خواهیم که فایلها را درون شاخه ای از دستگاه خود، مثلاً Calender ذخیره نماییم و آدرسی که مدیر پروژه به ما داده است svn.hoomand.com می‌باشد. ما با اجرای دستور زیر اولین برداشت خود را از مخزن فایل انجام می‌دهیم:

```
$ svn checkout http://svn.hoomand.com Calender
A firstFile.php
A someOtherFile.php
A someGraphic.png
A README
A template/gui.tpl
A template/body.html
...
Checked out revision 15
```

همانطور که می‌بینید با دستور svn checkout و مشخص کردن مکان سرور و شاخه‌ای از دستگاه خود که می‌خواهیم نسخه کاری (Working copy) خود را در آنجا ذخیره کنیم، تمام کارها براحتی انجام می‌شود. کاراکترهای A بدین معنا هستند که فایلها/شاخه‌ها در حال Add شدن می‌باشند. در انتها نیز پیغامی داده می‌شود مبنی بر اینکه نسخه ۱۵ مخزن فایل توسط شما دانلود شده است. با مفهوم نسخه و این شماره به زودی آشنا می‌شویم. اگر بصورت عادی و با دستور ls محتوای شاخه Calender را نگاه کنید قطعاً فایلهای خودتان را در آنجا می‌بینید. ولی اگر با دقت بیشتری نگاه کنید و از دستور ls -a استفاده کنید، می‌بینید که در شاخه Calender و تمام زیر شاخه‌های این شاخه، شاخه مخفی ای به نام svn وجود دارد. فکر می‌کنید محتوای این شاخه چه باشد؟ ببینید هنگامی که شما فایلهای خود را از مخزن فایل اصلی واقع در سرور بر می‌دارید، می‌خواهید که بر روی آنها کار کنید و یک سری تغییراتی در آنها انجام دهید. فرض کنید شما فایل someOtherFile.php را ویرایش کرده و چند خط به آن اضافه کنید. همینطور فرض کنید که فایل someGraphic.png را باز کرده و چندین شکل جدید به آن اضافه کنید. ممکن است احساس کنید به فایل index.php نیز نیاز دارید و آنرا از صفر ایجاد نمایید. خوب، اجازه دهید در همین میان شما را با دستور svn status آشنا کنم. وظیفه این دستور چیست؟ این دستور فایلهای فعلی شما را که در نسخه کاریتان وجود دارد با نسخه اصلی بدون تغییر که ابتدا دانلود کردید مقایسه کرده و فایلهای تغییر یافته را به شما نشان می‌دهد. در مورد مثال خودمان اگر این دستور را اجرا کنیم، نتایج زیر را مشاهده خواهیم کرد:

```
$ svn status
M someOtherFile.php

M someGraphic.png

? index.php
```

همانطور که می‌بینید M بدین معناست که فایلهای someOtherFile.php و someGraphic.png تغییر (Modify) یافته‌اند. ? هم به این معناست که subversion فایل index.php را نمی‌شناسد، خوب، حق هم دارد، چون این فایل را جدید اضافه کرده‌اید و اصلاً در ورژنی که از مخزن فایل گرفته‌اید وجود نداشته است. فکر می‌کنید subversion از کجا متوجه می‌شود که چه فایلهایی نسبت به نسخه‌ای که در ابتدا دانلود کرده‌اید تغییر یافته‌اند و یا اضافه شده‌اند؟ جواب ساده است: subversion همیشه یک نسخه از فایلهای اصلی دانلود شده را در جایی ذخیره می‌کند تا بعداً بتواند تغییرات شما را نسبت به آن مقایسه کند و بدین صورت این نتایج را ارائه کند. بنابراین svn status به هیچ عنوان برای فهم تغییرات انجام گرفته و داشتن ملاکی برای مقایسه دوباره به مخزن فایل سرور مراجعه نمی‌کند. به بیان دیگر شما می‌توانید هیچ ارتباطی با اینترنت نداشته باشید و این اطلاعات را دریافت کنید. حال فکر می‌کنید subversion این نسخه اولیه را کجا نگهداری می‌کند؟ اگر هنوز سوال اول ما به خاطرتان هست که پرسیدیم داخل شاخه‌های مخفی svn واقع در هر شاخه‌ای چه چیزی قرار دارد، این پاسخ همان سوال است، در داخل این شاخه‌ها نسخه‌های اصلی فایلهایی که در آن شاخه وجود دارند قرار دارد، نسخه‌هایی که هیچگاه تغییر پیدا نمی‌کنند. البته در این شاخه یک سری فایلهای کنترلی مربوط به subversion نیز قرار دارند که subversion برای کار به آنها نیاز دارد.

تغییرات قابل انجام در نسخه کاری

پس همانطور که دیدید شما می‌توانید روی نسخه کاری خود کار کرده و تغییراتی را در فایلها ایجاد نمایید. حال اجازه دهید ببینیم شما چه تغییراتی را و چگونه می‌توانید ایجاد نمایید:

تغییر فایل‌های موجود

شما می‌توانید فایل‌های موجود در نسخه کاری خود را ویرایش کرده و هر بلایی که خواستید به هنگام ویرایش بر سر آنها بیاورید! بعد از انجام ویرایش خود، هیچ نیازی نیست که به subversion بگویید فایل شما تغییر پیدا کرده است. خود subversion متوجه این قضیه می‌شود، برای اثبات این مدعا هم می‌توانید دستور `svn status` را اجرا کنید و ببینید که فایل ویرایش شده شما را به همراه یک M نشان می‌دهد که همانطور که گفته شد این M نشان می‌دهد فایل تغییر یافته است.

اضافه سازی فایل/شاخه جدید

شما می‌توانید هر فایل و یا شاخه‌ای را که خواستید به نسخه کاری خود اضافه کنید، ولی دیگر subversion آنرا بصورت خودکار برای اضافه شدن به مخزن فایل اصلی علامت گذاری نمی‌کند. اگر خاطرتان باشد در مثال بالا، ما فایل `index.php` را به نسخه کاری خود اضافه کردیم. قطعاً پس از اجرای دستور `svn status` می‌توانید فایل اضافه شده خود را با یک علامت سوال (?) مشاهده کنید، که این بدان معناست که subversion نمی‌داند با این فایل چکار باید بکند. اگر خواستار این باشید که این فایل در آینده پس از ارسال نسخه کاریتان به سرور به مخزن فایل اصلی اضافه شود، باید آنرا برای اضافه شدن انتخاب نمایید. این کار را با دستور `svn add` انجام می‌دهیم:

```
$ svn add index.php
```

```
A index.php
```

عینا همین کار را برای یک شاخه هم باید انجام داد. فرض کنید شاخه‌ای به اسم `includes` می‌سازید و چندین فایل درون آن اضافه می‌کنید، اگر بیرون از این شاخه دستور `svn status` را اجرا کنید، می‌بینید که نام آن شاخه به همراه تمام فایل‌های درون آن با ؟ آمده است. برای اضافه کردن آن شاخه و تمام فایل‌های درون آن تنها کافیست که همانند دستور بالا آن شاخه را `add` نمایید. در subversion بصورت پیش فرض عمل اضافه سازی بصورت پیمایش عمقی (Recursive) انجام می‌شود، یعنی با اضافه کردن یک شاخه تمام زیر شاخه‌ها و فایل‌های آنها نیز اضافه می‌شود. البته با اضافه سازی سوئیچ `-N` می‌توان این رفتار پیش فرض را تغییر داد. خوب، حال که فایل `index.php` را بوسیله دستور `svn add` به نسخه کاری خود اضافه کردیم، اگر مجدداً دستور `svn status` را اجرا کنیم خواهیم داشت:

```
$ svn status
```

```
M someOtherFile.php
```

```
M someGraphic.png
```

```
A index.php
```

همانطور که می‌بینید جای علامت سوال با یک کاراکتر `A` عوض شده است که این بدان معناست که تمام کارها بدرستی پیش رفته است. اگر دقت کنید می‌بینید که فایل‌های `someOtherFile.php` نیز دارای علامت `M` است و همانطور که قبلاً گفتیم این علامت بدین معناست که این فایل نسبت به دفعه آخری که از مخزن فایل اصلی موجود بر سرور دانلود شده است، توسط ما تغییر یافته است. حال فرض کنید بخواهیم بفهمیم که ما چه تغییراتی را بر روی این فایل انجام داده‌ایم، چاره چیست؟!

بیزن هومند

[1] <http://subversion.tigris.org>