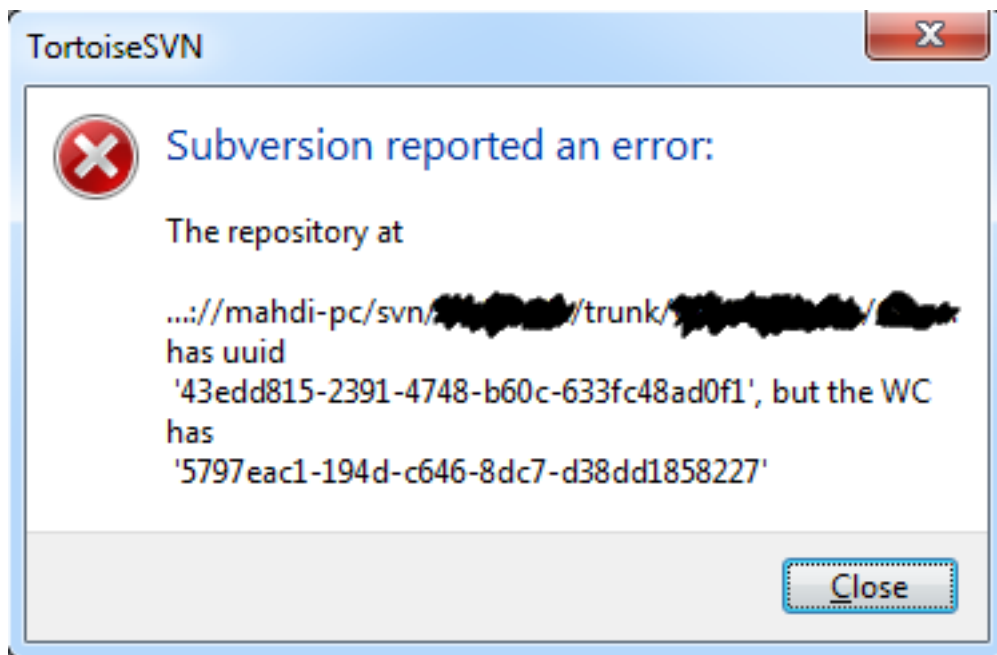
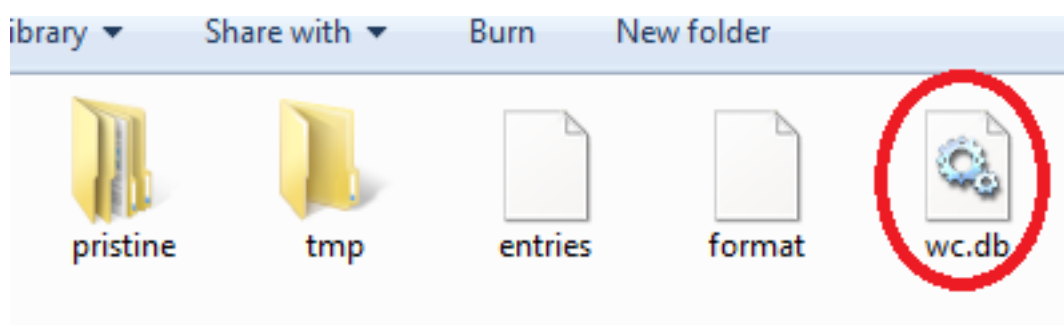


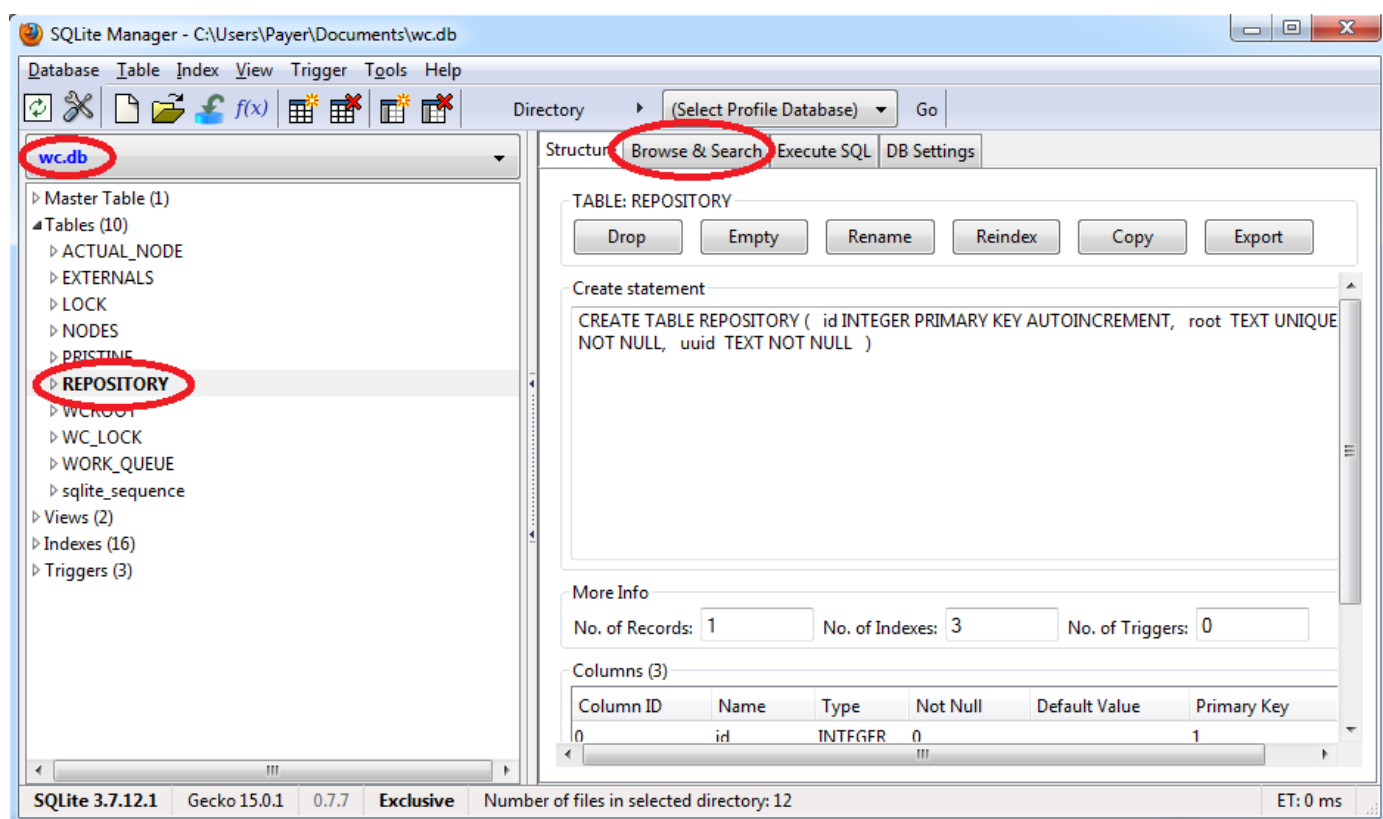
بنا به دلایلی، کدهای نگهداری شده توسط svn در سیستم خودم رو باید به سیستم دیگری وارد می‌کردم که شامل چند پروژه می‌شد و در قسمت مربوطه چند پروژه کار شده بود. تا این که مشکلی پیش آمد و جهت ادامه توسعه مجبور به برگرداندن پروژه‌ها به سیستم قبلی شدیم. حالا مشکل این بود که به repository قبلی هیچ گونه دسترسی نداشتیم و نمیخواستیم که کدهای در دست هم بصورت یک پروژه تازه وارد سرور svn شود. برای همین مجبور بودیم که از طریق relocate کردن، این کار را انجام دهیم، که به خطای زیر برخورد کردم:



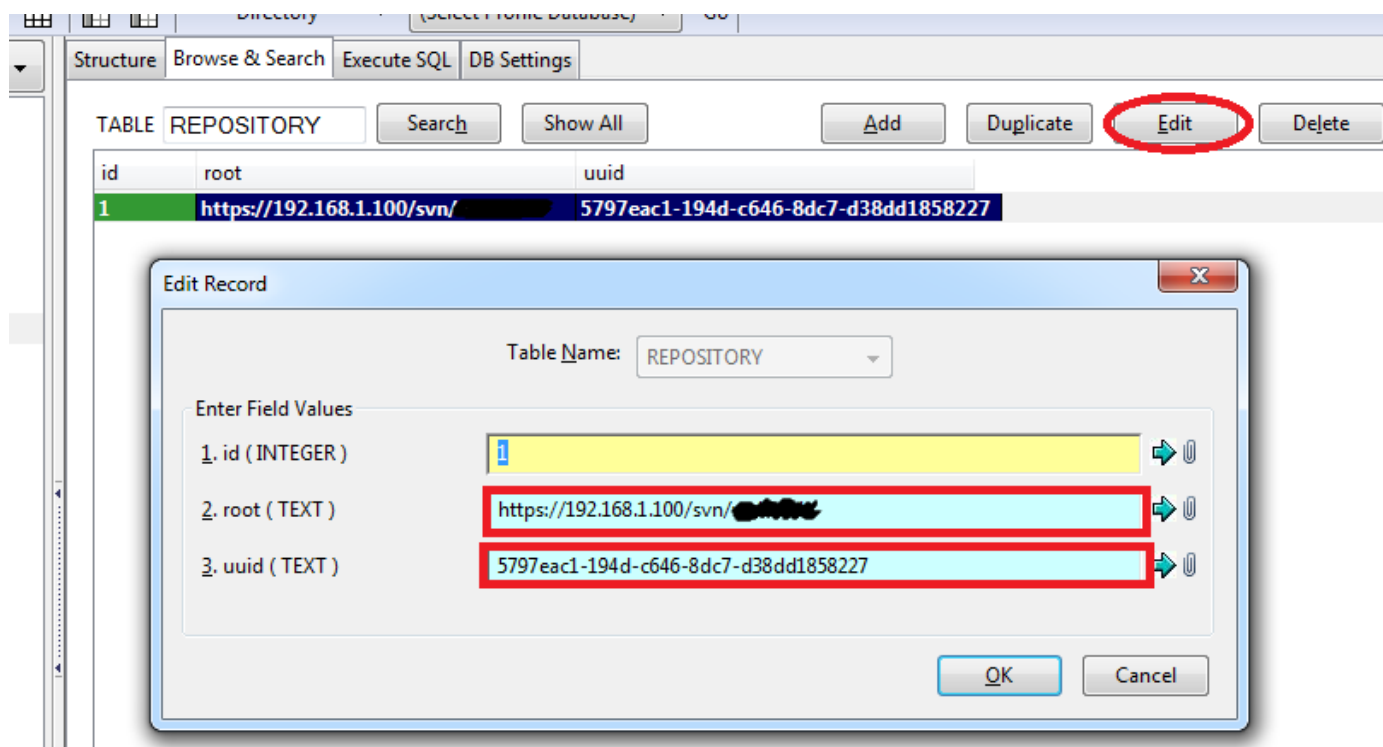
بله؛ مشکل از اینجا ناشی می‌شد که سرور دوم کدها را import نکرده بود و بلکه با یک کامیت ساده، کدها وارد سرور شده بودند.  
**راه حل:** در کل برای تغییر این uid که در تصویر مشخص شده، باید فایل SQLite مربوطه را یافت و به راحتی، دو مشخصه مربوط به آدرس مخزن و uid را در آن اصلاح کرد.  
 در پوشه svn کنار پروژه، فایلی با نام wc.db از نوع SQLite وجود دارد:



که به سادگی می‌توان آن را با برنامه‌ای مانند [افزونه ای در فایرفاکس](#) گشود و ویرایش کرد:



همانطور که مشخص است، در جدول repository، تک ردیفی وجود دارد که مشخصات مربوط به مخزن svn در آن ذخیره شده است:



**نکته:** در این نوع آدرس دهی، باید حتما به نوع قرار گیری پروژه ها در کنار هم نیز دقت داشت. مثلا به محل آخری که در آدرس وجود دارد باید دقت کرد تا در مسيردهی، اشتباهی صورت نگیرد و فایلها جابجا کامیت نشوند.

## نظرات خوانندگان

نویسنده: وحید نصیری  
تاریخ: ۱۳۹۱/۰۸/۲۳ ۱۳:۰

این مطلب امروز به درد من خورد!  
پروژه iTextSharp محل مخزنش رو عوض کرده یا شاید هم Source forge دست به تغییر و تحول زده. محل جدید برای checkout شده:

```
svn checkout http://svn.code.sf.net/p/itextsharp/code/trunk itextsharp-code
```

و با uuid جدید زیر

```
820d3149-562b-4f88-9aa4-a8e61a3485cf
```

برای relocate مخزن قبلی به محل جدید حتما نیاز است این uuid را در بانک اطلاعاتی فوق ویرایش کرد.

ما در شرکت برای Source Control از SVN استفاده می‌کنیم، مزایای سورس کنترل آنقدر واضح است که دیگه من اینجا چیزی ازش نمیگم

اما برای استفاده از سورس کنترل یک مشکلی وجود دارد، اگر شما تعدادی پروژه را به کاربران خاصی بدین و تعدادی رو ندین، اون کاربر وقتی پروژه‌ها را می‌گیره با مشکل ارجاعات پروژه‌ها مواجهه است. چرا که برخی از پروژه‌های ارجاعی، روی کامپیوتر برنامه نویس 1 وجود نداره. برعکسش هم همین طوره، چون اون کاربر، پروژه‌های ارجاعی رو نداره، باید به جاش به اسمبلی نهایی اون پروژه ارجاع بده. بنابراین وقتی مدیر پروژه‌ها رو می‌گیره، باز ارجاعات اشتباه هستند!

ما اینجا برای رفع این مشکل ابزاری درست کردیم، به اسم SolutionExplorer.

این ابزار فایل solution رو به همراه پوشه حاوی فایل‌های اسمبلی می‌گیره. اگر پروژه ای به اسمبلی ای ارجاع داده باشه که پروژه اش توی solution باشه، ارجاع به اسمبلی رو تبدیل میکنه به ارجاع به پروژه و برعکسش، اگر پروژه ای به پروژه دیگه ای ارجاع داده باشه که توی solution وجود نداشته باشه، توی پوشه اسمبلی ها، دنبال اسمبلی ای می‌گرده که اسمش شبیه اسم پروژه ارجاعی باشه و اگر پیدا کنه، ارجاع رو عوض می‌کنه

البته برای جلوگیری از به هم ریختگی، نرم افزار از فایل‌های پروژه ای که دستکاری می‌کنه، پشتیبان می‌گیره [دانلود پروژه](#)

توجه:

\* این برنامه از تمامی جهات تست نشده است، با ریسک خودتون ازش استفاده کنید (ما تو شرکت دیگه ریسکی نداریم: )

\* سیستم نامگذاری اسمبلی‌ها و پروژه‌های ما ممکنه فرق کنه

\* اگر به مشکلی برخوردید، لطفا زیر همین مطلب برام بنویسید

\* انتخاب پوشه اسمبلی ها، الزامی نیست

## نظرات خوانندگان

نویسنده: محسن خان  
تاریخ: ۱۱:۴۰ ۱۳۹۲/۰۲/۰۲

ممنون. ایده خوبی هست.

یک روش دیگر هم استفاده از نیوگت هست برای مدیریت لوکال وابستگی‌ها

[Creating and then using a NuGet local repository](#)

[How to access NuGet when NuGet.org is down](#)

نویسنده: بهزاد  
تاریخ: ۱۱:۴۲ ۱۳۹۲/۰۲/۰۲

در مورد وابستگی‌های نوگت کاری نکردیم، فقط در مورد پروژه‌ها و اسمبلی‌هاست

نویسنده: محسن خان  
تاریخ: ۱۲:۳۰ ۱۳۹۲/۰۲/۰۲

نیوگت لوکال روی شبکه می‌تونید تعریف کنید بر اساس وابستگی‌های داخلی خودتون. یعنی کاملاً مستقل از نیوگت روی اینترنت.

نویسنده: یوسف نژاد  
تاریخ: ۸:۲۹ ۱۳۹۲/۰۲/۰۳

شما میتونین خروجی تمام پروژه‌های ریفرنس داده شده در پروژه‌های دیگه رو به یک مسیر مشخص و مشترک تنظیم کنید. تمام پروژه‌ها هم ریفرنس خودشون رو از اون مسیر مشخص بگیرن. سپس فایل‌های dll یا exe موردنظر رو بصورت multi-check out تنظیم کنید. بعدش هرکسی که آخرین نسخه از اون کتابخونه رو داره توسعه میده هر روز چکین کنه و بقیه هم هر روز get latest کنن. کاری که ما داریم به راحتی در شرکت خودمون انجام میدیم.

نویسنده: سیروس  
تاریخ: ۱۳:۳۱ ۱۳۹۲/۰۲/۰۳

ما هم از این روش استفاده می‌کنیم.

نویسنده: منیژه محمدی  
تاریخ: ۲۱:۱۳ ۱۳۹۲/۰۸/۲۶

در مورد TFS چطور ؟ در مورد ان پیشنهادی ندارید؟

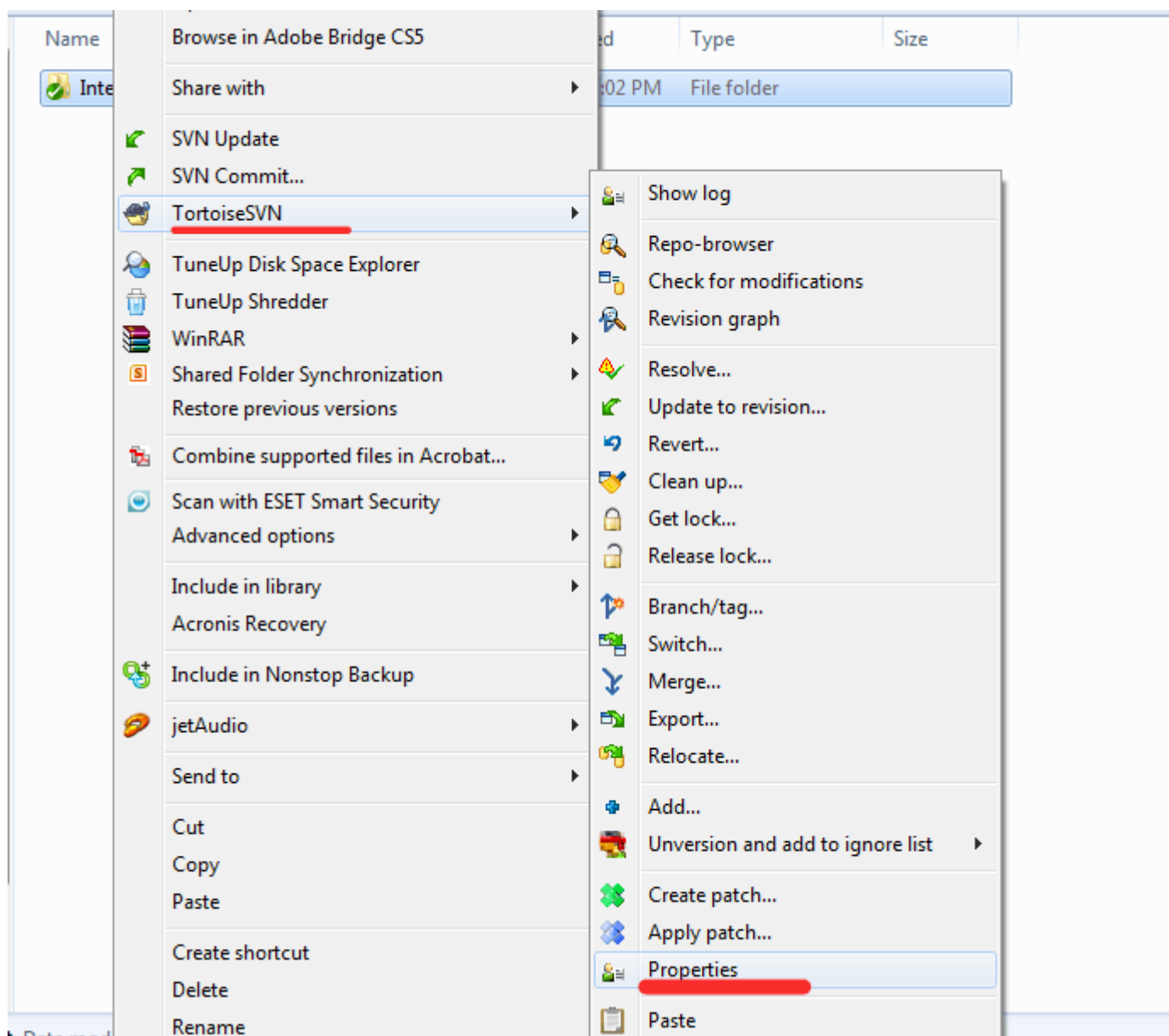
پیش نیاز اگر در مورد TortoiseSVN و سورس کنترل اطلاعات پایه ندارید، کتاب [مدیریت فایل‌های یک پروژه نرم افزاری با استفاده از Subversion](#) آقای نصیری را مطالعه کنید و همچنین سیستم پیگیری خطای [YouTrack](#) را نگاهی بیاندازید (البته اگر اطلاعی ندارید).

## مقدمه

هنگام کار روی یک پروژه، باگ‌ها، وظیفه‌ها و موضوعاتی به شما واگذار می‌شود که باید آنها را انجام دهید. هنگام commit کردن تغییرات، برای مشخص شدن اینکه تغییرات مربوط به کدام Bug-Id بوده است باید سیستم Bug/Issue Tracker رو با سورس کنترل یکپارچه کنیم.

## یکپارچه سازی TortoiseSVN و YouTrack

1- روی یک نسخه کاری پروژه راست کلیک، از منوی TortoiseSVN گزینه Properties را انتخاب کنید.



2- از پنجر باز شده دکمه New، گزینه Other را انتخاب کنید. در پنجره باز شده از منوی کشویی مربوط به Property Name، مقادیر

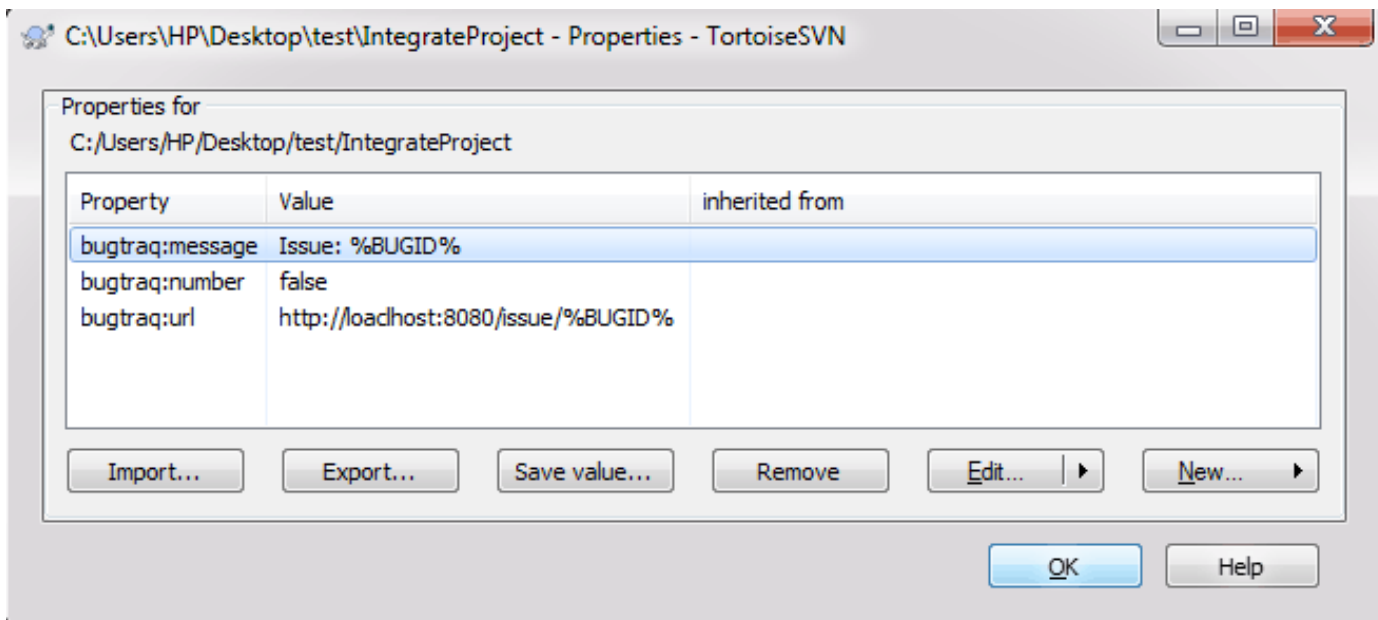
خصلت‌های زیر را تنظیم کنید: **bugtraq:url1** : آدرس YouTrack Sever که به این صورت وارد می‌شود: %

<http://localhost:8080/issue/%BUGID>

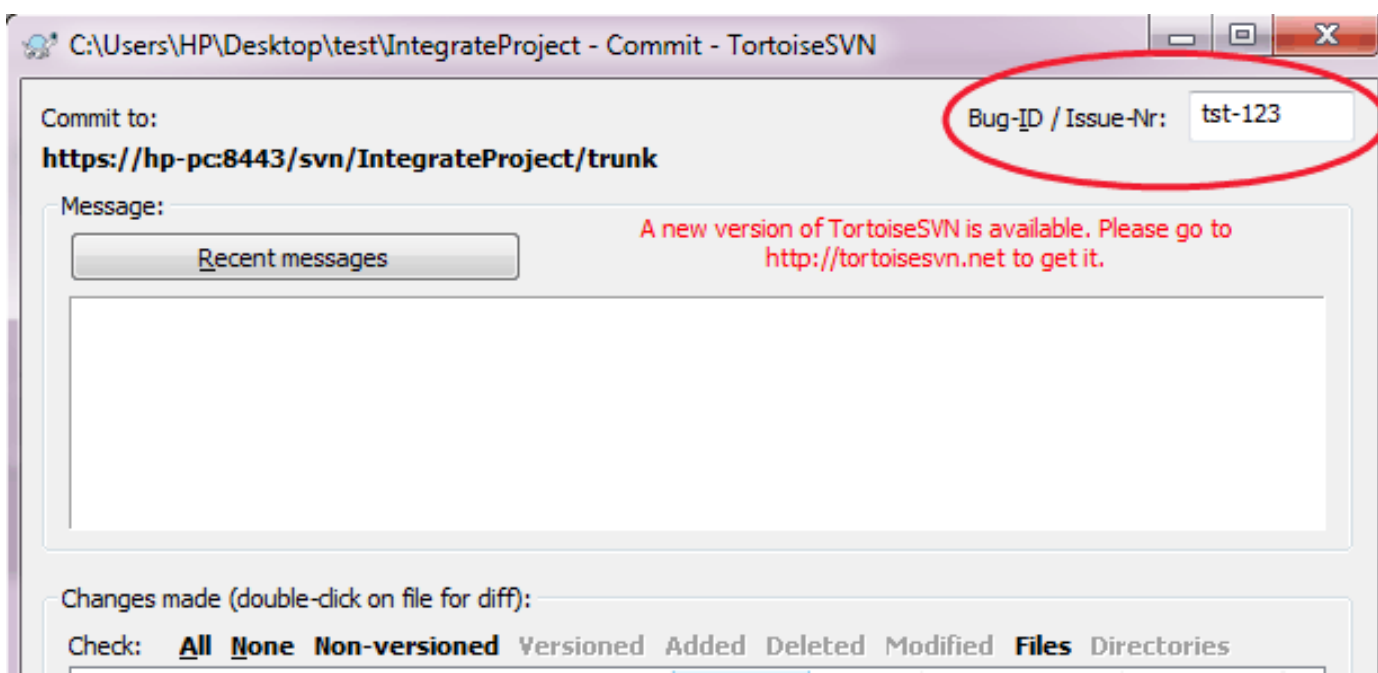
**bugtraq:message** : درو اقع الگویی پیامی هست که برای نگهداری Bug-Id استفاده می‌شود و باید شامل کلمه %BUGID% باشد.

مثلا: %BUGID%

**bugtraq:number** : مقدار این خصلت را false وارد کنید؛ چون Bug-Id های YouTrack می‌توانند شامل عدد و حروف باشند.

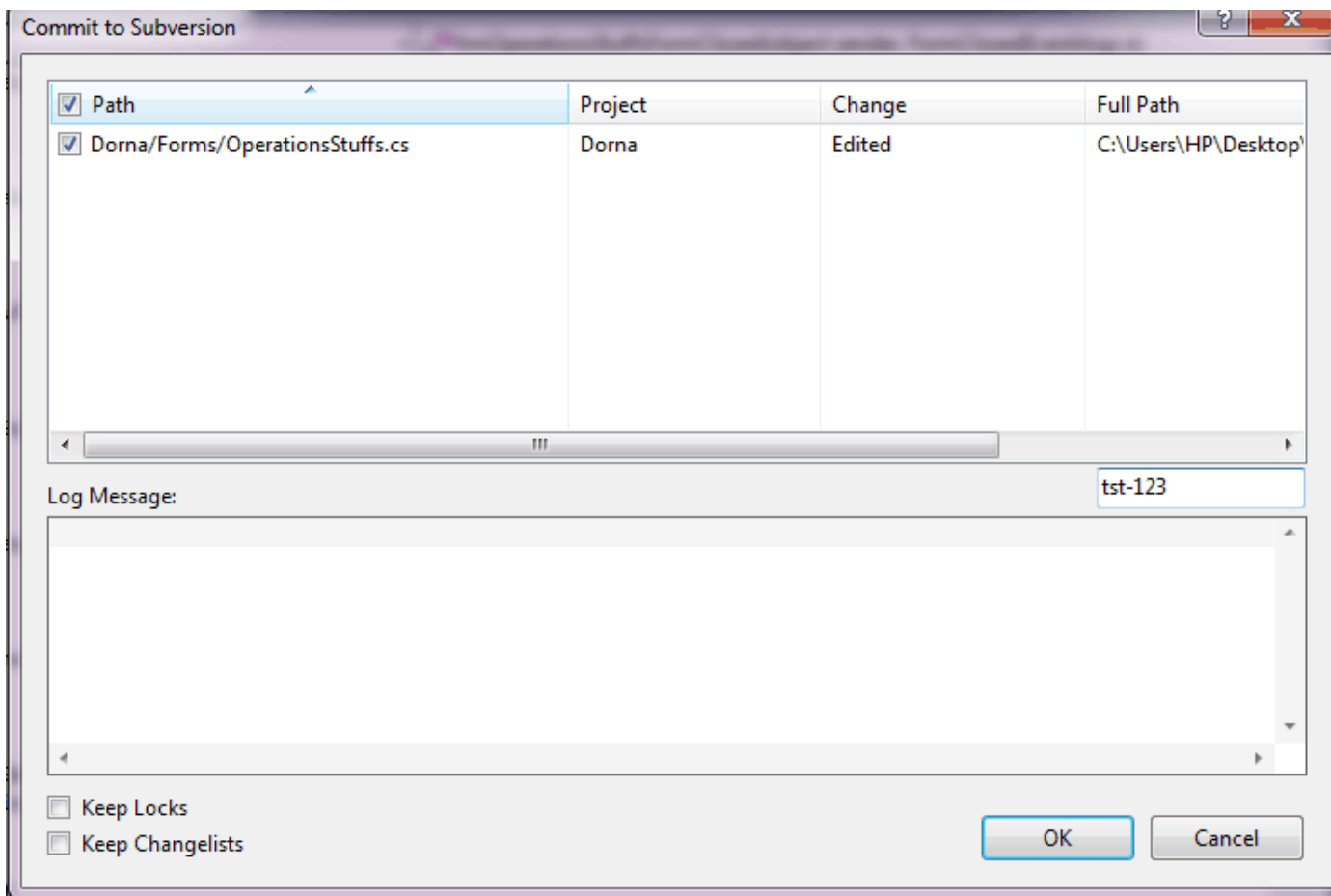


بعد از اینکه این سه خصلت را مقاردهی کردید، تغییرات را Commit کنید. همانطور که می‌بینید یک Textbox (بالا، سمت راست) اضافه شده که محل وارد کردن Bug-Id مربوط به تغییرات است. از این پس، می‌توانید Bug-Id یا Issue-Id های مربوط به هر تغییرات را در آن Textbox وارد کنید.

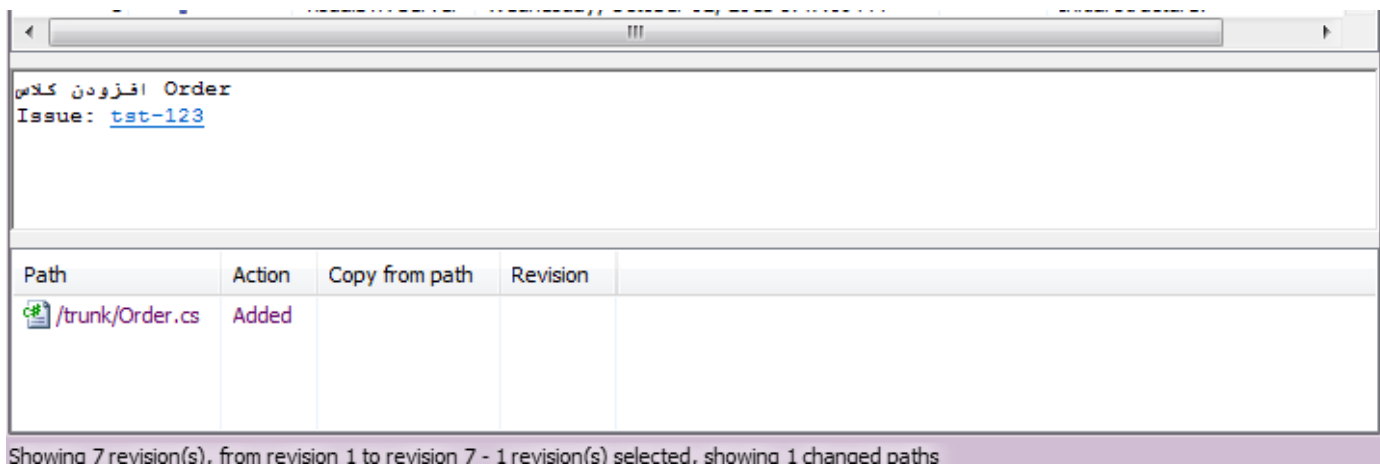




همچنین تغییرات در پلاگین AnkhSVN در ویژال استودیو نیز اعمال می‌شود:



اکنون، در متن commit ها شماره Bud-Id نیز ذکر شده است.



**نکته 1:** اگر YouTrack روی یک سرور نصب هست، بجای localhost نام کامپیوتر سرور یا آی پی آن را وارد کنید. پورت 8080

نیز بصورت پیش فرض است و اگر هنگام نصب آن را تغییر داده اید، اینجا نیز آنرا تغییر دهید.

**نکته 2:** خصلت bugtraq:message یک الگوی پیام از شما می‌گیرد؛ یعنی الگو را تحت هر شکلی می‌توان وارد کرد. بعنوان مثال الگو را به این شکل وارد کنید: "برای مشاهده جزئیات بیشتر به Bug-Id شماره %BUGID% مراجعه کنید." **نکته 3:** اگر خصلت bugtraq:number مقدارش true باشد، برای وارد کردن Bug-Id فقط از عدد می‌توانید استفاده کنید. بصورت پیش فرض مقدار این خصلت true است. **نکته 4:** می‌توانید این تنظیمات را در یک فایل Export کنید و در بقیه پروژه ها، با یک مرحله و بسادگی آنرا Import کنید.

خصلت‌های دیگری نیز می‌توان بر روی مخزن کد اعمال کرد که از حوزه این مقاله خارج است. همچنین تنظیمات اختیاری جانبی دیگری نیز برای یکپارچه سازی وجود دارند. برای دیدن این تنظیمات روی نسخه کاری راست کلیک، از منوی TortoiseSVN گزینه Properties را انتخاب کنید و از پنجره باز شده روی دکمه New و گزینه Bugtraq (Issue tracker integration) را انتخاب کنید.

**Issue tracker**  
Specify the URL to access the issue tracker. Use %BUGID% as a placeholder for the real issue number.

URL:

☐ Remind me to enter a bug-ID

**Message**  
Specify how the commit message should be built from the entered bug-ID. Use the placeholder %BUGID% for the real bug-ID. If you leave these settings empty, TortoiseSVN will use the regular expressions instead.

Message pattern:

Message label:

Bug-ID is: ☒ Arbitrary text ☐ Numeric

Insert message at: ☐ Top ☒ Bottom

**Regular Expression**  
Enter the regular expression patterns for filtering out the bug-ID from a commit message.

Message part expression:

Bug-ID expression:

**IBugTraqProvider**

Provider uuid win32:  uuid x64:

Provider parameters:

☐ Apply property recursively

OK Cancel Help

برای اطلاعات بیشتر در مورد این تنظیمات، داکيومنت [یکپارچه سازی با سیستم‌های Bug tracking / Issue Tracking](#) را مطالعه کنید.