

100% Free CCNA Exam File

Real Latest Exam Questions
from Current CCNA Exam Pool

Exam-Labs



TortoiseSVN

Info

[About](#)
[About TortoiseSVN](#)

[Screenshots](#)
[Screenshots of various dialogs](#)

[Testimonials](#)
[What users say about TortoiseSVN](#)

[News Archive](#)
[News archive](#)

Support

[FAQ](#)
[Frequently asked questions](#)

[Help files](#)
[The complete documentation](#)

[Useful tips](#)
[Tips about not well known features](#)

[Mailing lists](#)
[Where to find the mailing lists](#)

[Report bugs](#)
[How and where to report a bug](#)

[Translations](#)
[Help translate](#)

[Project status](#)
[Overview of releases](#)

Misc

[Issue tracker plugins](#)
[Plugins to connect with issue trackers](#)

[SVNProtocolHandler](#)
[A plugin for IE to browse svn:// repositories](#)

How to integrate TortoiseSVN into Visual Studio

Posted on December 10, 2012

Advertisement

Download Button

dreamstime.com



If you're using [Visual Studio](#), you can integrate TortoiseSVN commands to various context menus. Of course, there are Subversion plugins for Visual Studio like [AnkhSVN](#) or [VisualSVN](#), with the latter already using TortoiseSVN for many of its UI. Those plugins also have the advantage that when you do refactoring, they automatically record the renames and moves in SVN.

The first step is to add the TortoiseSVN commands as *external tools*, under the menu *TOOLS* > *External Tools*...

Misc posts

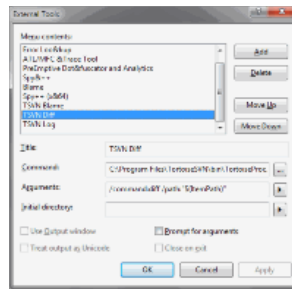
Posts that don't fit into any other category



Open Hub

3536
USERS

I USE IT



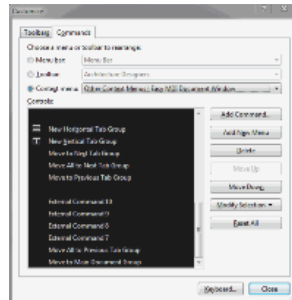
Add the name of the command, the path to *TortoiseProc.exe* and then the parameters for the command.

Use the VS variables wherever needed. Since I add my commands to the context menu of the open file tab, here's the parameters I used:

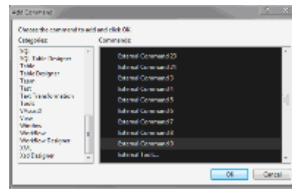
- `/command:blame /path:"$(ItemPath)" /line:$(CurLine) /startrev:1 /endrev:HEAD`
- `/command:diff /path:"$(ItemPath)"`
- `/command:log /path:"$(ItemPath)"`

Notice the `/line:` parameter: this will make TortoiseBlame automatically scroll to the same line the cursor is located in the opened file in Visual Studio. If you omit the `/startrev` and `/endrev` parameters for the blame command then a dialog is first shown where you can specify more options for the blame.

Now to add those new commands to the file tab context menu, go to *TOOLS->Customize...*, select the *Commands* tab, click the radio button *Context menu* and then select *Other Context Menus | Easy MDI Document Window*.



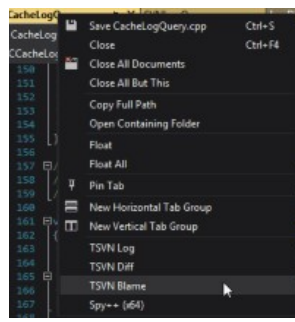
Then click the *Add command...* button:



Now you have to select the commands. Problem is that the custom commands are not shown with their title but only as *External Command X* with *X* being the number of the external command.

In my case, the commands were number 9-11, you might have to do some trial-and-error here. Just add the commands you think are the ones you added and then check if the right ones show up in the context menu.

And this is how the context menu of the open file tab looks like:



♥ 2004-2019 – THE TORTOISESVN TEAM. PLEASE SHARE.