

Gebruik van Subversion

Joris Borgdorff

February 23, 2006

Contents

1	Subversion verkrijgen	1
2	Linux installatie en gebruik	1
3	Windows installatie en gebruik	2
4	Algemeen	3
4.1	Revisions	3
4.2	De trunk, branch en tag	3
4.3	Conflict, lock, diff en merge	5
4.4	Verder...	6

1 Subversion verkrijgen

Subversion 1.3.0 is beschikbaar op <http://subversion.tigris.org/>, voor Linux en Windows.

Verder heb ik op <http://svn.cs.uu.nl/> een repository massanalyst aangemaakt. Als ik op de universiteit ben zal ik jullie daaraan toevoegen als gebruikers. Onze repository zelf staat op <https://svn.cs.uu.nl:12443/repoman/massanalyst/>, dit adres heb je ook nodig voor de subversion client.

2 Linux installatie en gebruik

Voor Linux kun je het op normale wijze compileren, behalve dat je `./configure --with-ssl` moet typen i.p.v. de gebruikelijke `./configure`.

De commando's worden dan

```
$ tar -xjf subversion-1.3.0.tar.bz2
$ cd subversion-1.3.0
$ ./configure --with-ssl
$ make
$ su
$ make install
```

Vervolgens zijn de belangrijkste commando's:

- `svn checkout https://svn.cs.uu.nl:12443/repos/massanalyst/`
Dit zorgt ervoor dat in de huidige map de SVN map massanalyst komt waar alle bestanden worden uit de repository in staan.
- `svn update`
Dit zorgt ervoor dat de huidige map ge-update wordt naar de laatste versie die in de repository staat.
- `svn revert`
Dit zorgt ervoor dat de in huidige map alle wijzigingen ongedaan worden gemaakt en je terug gaat naar de laatste update die je hebt gedaan.
- `svn commit`
Dit zorgt ervoor dat de gemaakte wijzigingen worden doorgevoerd in de repository.
- `svn add [bestand]`
Dit zorgt ervoor een nieuw bestand ook in subversion wordt opgenomen.
- `svn mv [van] [naar]`, `svn cp [van] [naar]`, `svn rm [bestand]`, `svn mkdir [map]`
Vrij intuïtieve commando's. Zorg er altijd voor dat je deze commando's gebruikt en niet de systeem commando's, dat bespaart je een hoop gedoe.

3 Windows installatie en gebruik

Hiervoor is de interface van TortoiseSVN erg handig, te verkrijgen op <http://tortoisesvn.tigris.org>. Vereiste is wel dat je Windows 2000 SP2 of Windows XP hebt. Op de universiteit is TortoiseSVN ook aanwezig.

De commando's werken hetzelfde als bij de command-line subversion alleen dan staan ze in het rechtermuisknop menu in Windows Verkenner, zie pagina's 3 en 4.

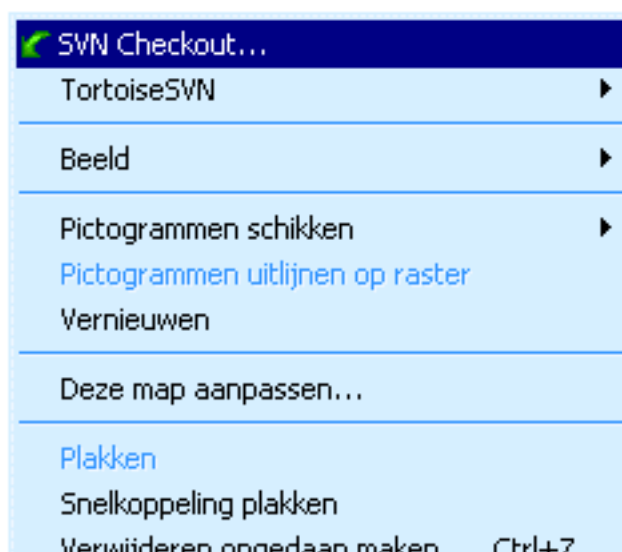


Figure 1: TortoiseSVN begin menu

4 Algemeen

4.1 Revisions

Dit is niet zo heel ingewikkeld: bij elke commit die wordt gedaan wordt het revision nummer een opgehoogd. Je kunt later dan teruggaan naar een oudere revision om een oudere versie te bekijken.

Wat belangrijk is, is dat je regelmatig een commit doet, en dat elke commit staat voor een specifiek doel dat bereikt is (functie toegevoegd, bug opgelost) en niet voor meerdere doelen tegelijk. Zo is het voor iedereen duidelijk wat er bij die revision is veranderd.

De Head revision is de laatste revision tot nu toe.

4.2 De trunk, branch en tag

In de trunk zal het grootste deel van het ontwikkelen gebeuren. Hier ben je dagelijks mee bezig. Zorg wel dat elke versie die je inlevert nog compileert, als het om code gaat. Anders doen andere mensen updates op die code en kan niemand nog compileren.

Een ingewikkeld concept wat toch nog wel van pas kan komen is tag. Stel je hebt een versie uitgebracht (versie 1.0 bijv.) en je bent alweer verder aan het werken aan de code. Er wordt een kritieke bug gemeld over versie 1.0,

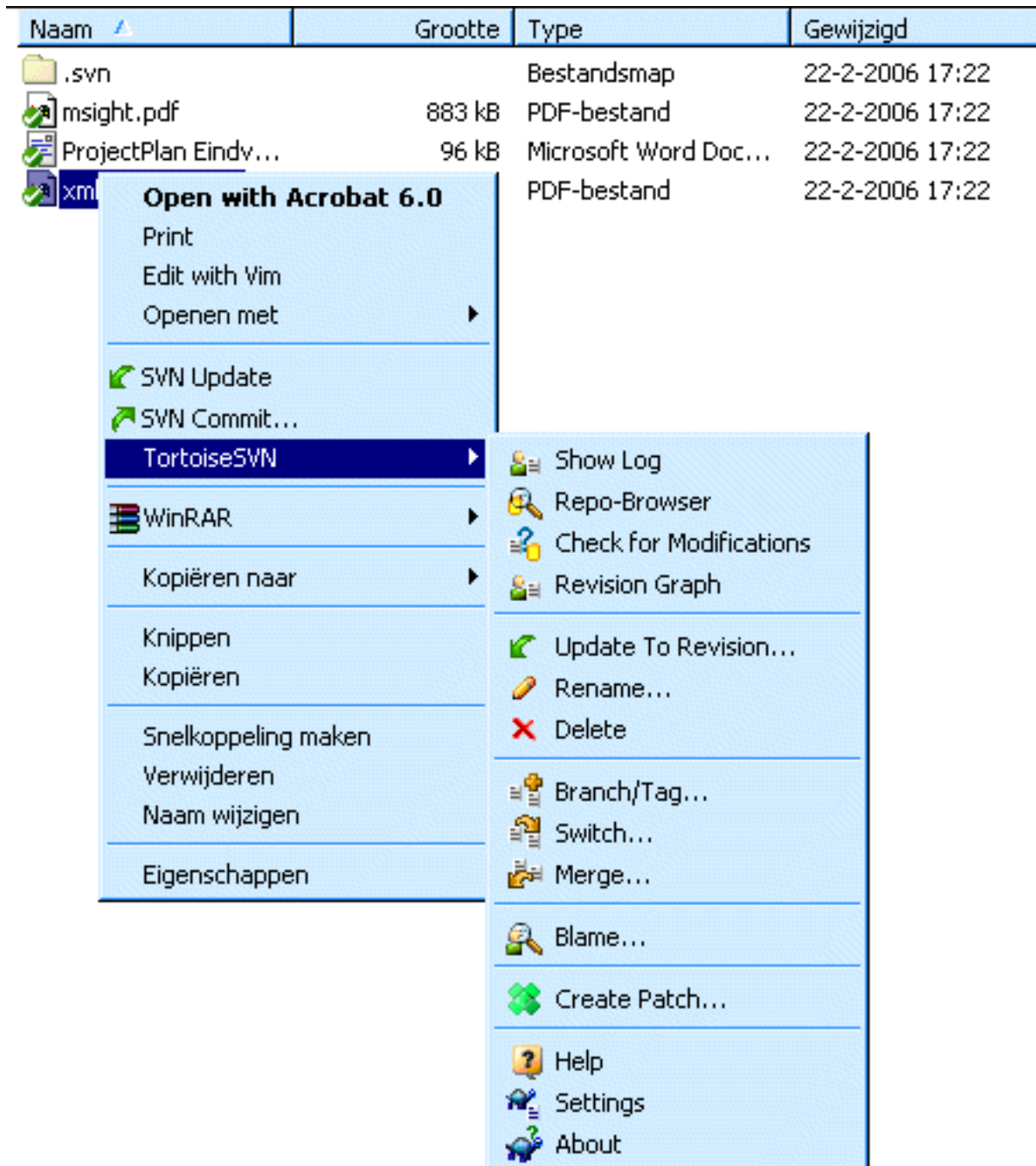


Figure 2: TortoiseSVN filemenu

maar de code die je nu aan het schrijven bent is nog niet stabiel genoeg om te gebruiken. Daarom kun je beter versie 1.0 opslaan. Dit kan in de tag directory, met het commando `tag`. Dus als je versie 1.0 uitbrengt dan copieer je de huidige code met het tag commando naar de map `tags/1.0`. Dan kun je in de trunk verder ontwikkelen en eventuele kritieke bugs in de map `tags/1.0` oplossen.

Een ander concept is dat van branchen. Daar zit het volgende idee achter: stel we besluiten een deel van de architectuur om te gooien (hopelijk niet) of we zijn een nieuwe module aan het ontwikkelen die geïntegreerd moet worden. De code in de trunk moet altijd kunnen compileren dus als je aan een module begint is het niet zo handig dat in de trunk te doen. Hiervoor kan je een branch maken, naar de map `branch/module_ofzo`. Dit is weer een kopie van de trunk waar je ongestoord verder aan kan werken en waar je ook regelmatig een commit op kan doen.

4.3 Conflict, diff en merge

Als er twee mensen een update hebben gedaan van revision 100 (bijvoorbeeld), en verder werken op dezelfde code en allebei proberen in te leveren dan heeft de eerste geen probleem, maar als de tweede dat probeert krijgt die een conflict. Als je een conflict krijgt moet je drie dingen doen:

1. een diff om te kijken wat het verschil is tussen de twee versies,
2. proberen de verschillen zoveel mogelijk weg te werken door bijvoorbeeld van verschillende functies alleen de nieuwste versie te gebruiken
3. bij onverenigbare verschillen de persoon die de andere versie heeft gemaakt opbellen om te overleggen hoe je verder gaat.

Maar een conflict kun je natuurlijk beter voorkomen door goede communicatie/taakverdeling!

Een andere manier waarop verschillen ontstaan is door de tags en branches. Het kan voorkomen dat er een bug zit in de core die op een van die plekken wordt opgelost maar in elke branch en tag zit. Daarom is het wijs om bij die branches en tags regelmatig een merge te doen. Dit kijkt naar alle verschillen tussen de tag/branch en de trunk. Zo kan je je tag/branch nog up-to-date houden. Bij een branch moet je ook een merge doen als je code geïntegreerd kan worden met de trunk. Als je dit gedaan hebt kan je branch weg.

4.4 Verder...

Een commando wat handig zou kunnen zijn is `blame`. Dan wordt er in de file die je blamet voor elke regel gezet van wie deze afkomstig is. Als je dan een raar stukje code tegenkomt kan je diegene aanspreken om het te verduidelijken en/of verbeteren.

Gebruik bij specifieke vragen het commando `svn help`. Lees (delen van) het online Subversion voor meer achtergrond-informatie op <http://svnbook.red-bean.com/>