

TortoiseSVN

Een Subversion client voor Windows

Version 1.6.16

**Stefan Küng
Lübbe Onken
Simon Large**

TortoiseSVN: Een Subversion client voor Windows: Version 1.6.16

door Stefan Küng, Lübbe Onken, en Simon Large

Vertaler: Jeroen van Amstel, Eddo Harkema, Ewout Muller, Christan Peper, Guus Raaphorst, Christophe de Wagter, Johannes Zwart

Uitgegeven 2011/01/21 21:21:17 (r20750)

Inhoudsopgave

Voorwoord	xi
1. Publiek	xi
2. Leesgids	xi
3. TortoiseSVN is gratis!	xii
4. Gemeenschap	xii
5. Dankwoord	xii
6. Gebruikte terminologie in dit document	xii
1. Introductie	1
1.1. Wat is TortoiseSVN?	1
1.2. TortoiseSVN's Historie	1
1.3. TortoiseSVN's mogelijkheden	1
1.4. Installeren van TortoiseSVN	2
1.4.1. Systeemvereisten	2
1.4.2. Installatie	3
1.4.3. Taalpakketten	3
1.4.4. Spellingscontrole	3
2. Basis Versie-Beheer Concepten	5
2.1. Het Archief	5
2.2. Versioneermodellen	6
2.2.1. Het Probleem van Bestand Delen	6
2.2.2. De Vergrendel-Wijzig-Ontgrendel Oplossing	6
2.2.3. De Kopieer-Wijzig-Samenvoeg Oplossing	8
2.2.4. Wat doet Subversion?	10
2.3. Subversion in Actie	10
2.3.1. Werkkopieën	10
2.3.2. Archief URLs	12
2.3.3. Revisies	12
2.3.4. Hoe Werkkopieën Het Archief Bijhouden	14
2.4. Samenvatting	15
3. Het Archief	16
3.1. Archief creatie	16
3.1.1. Een archief aanmaken via de opdrachtregel applicatie.	16
3.1.2. Het Archief Aanmaken Met TortoiseSVN	16
3.1.3. Lokale Toegang tot het Archief	17
3.1.4. Een archief benaderen op een Netwerk schijf	17
3.1.5. Archief indeling	18
3.2. Archief Reservekopie	19
3.3. Server gesitueerde hook scripts	20
3.4. Links voor Ophalen	20
3.5. Toegang hebben tot het Archief	21
3.6. Svnservice Based Server	21
3.6.1. Introductie	21
3.6.2. Installatie svnservice	21
3.6.3. Svnservice opstarten	22
3.6.4. Standaard Autorisatie met svnservice	24
3.6.5. Verbeterde beveiliging met SASL	25
3.6.6. Authenticatie met svn+ssh	26
3.6.7. Pad-gebaseerde Autorisatie met svnservice	26
3.7. Apache Based Server	26
3.7.1. Introductie	26
3.7.2. Installatie van Apache	27
3.7.3. Installatie van Subversion	28
3.7.4. Configuratie	28
3.7.5. Meerdere Archieven	30
3.7.6. Pad-gebaseerde autorisatie	31

3.7.7. Authenticatie door een Windows Domein	31
3.7.8. Meerdere authenticatie sources	33
3.7.9. De server beveiligen met SSL	34
3.7.10. Werken met virtuele SSL hosts en client certificaten	36
4. Dagelijkse gebruikersgids	38
4.1. Aan de slag	38
4.1.1. Overlappende Pictogrammen	38
4.1.2. Contekstmenus	38
4.1.3. Slepen en Plaatsen	40
4.1.4. Algemene snelkoppelingen	40
4.1.5. Authenticatie	40
4.1.6. Maximaliseren van Windows	41
4.2. Data Importeren In Een Archief	41
4.2.1. Importeren	42
4.2.2. Invoeren op zijn Plaats	43
4.2.3. Speciale Bestanden	43
4.3. Een lokale werkkopie (WC) ophalen	44
4.3.1. Ophaaldiepte	44
4.4. Je Wijzigingen Vastleggen In Het Archief	46
4.4.1. De Commit Dialoog	46
4.4.2. Werksets	49
4.4.3. Objecten van de vastleggingslijst uitsluiten	49
4.4.4. Vastleg Log Meldingen	49
4.4.5. Vastlegging voortgang	51
4.5. Je Werkkopie Bijwerken Met De Wijzigingen Van Anderen	52
4.6. Conflicten oplossen	54
4.6.1. Bestandsconflicten	54
4.6.2. Mapstructuur conflicten	55
4.7. Statusinformatie ophalen	58
4.7.1. Overlappende Pictogrammen	58
4.7.2. TortoiseSVN Kolommen In Windows Verkenner	60
4.7.3. Lokale en Server Status	60
4.7.4. Diffs bekijken	62
4.8. Werksets	62
4.9. Logboekboodschappen Venster	65
4.9.1. Het Logboek Boodschappen Venster Opstarten	66
4.9.2. Revisie Logboek Acties	66
4.9.3. Extra Informatie Opvragen	67
4.9.4. Ophalen van uitgebreide logboek boodschappen	72
4.9.5. Huidige Werkkopie Revisie	73
4.9.6. Samenvoeg Traceer Functies	73
4.9.7. Logboekboodschap en Auteur Wijzigen	74
4.9.8. Filter op Logmeldingen	75
4.9.9. Statistische informatie	76
4.9.10. Offline Modus	80
4.9.11. De weergave verversen	80
4.10. Tonen van Verschillen	80
4.10.1. Bestandsverschillen	81
4.10.2. Opties voor regeleinden, spaties en witregels	82
4.10.3. Vergelijken van mappen	82
4.10.4. Vergelijk afbeeldingen met TortoiseIDiff	84
4.10.5. Externe Verschil/Samenvoeg gereedschappen	85
4.11. Nieuwe Bestanden En Mappen Toevoegen	85
4.12. Kopiëren/Verplaatsen/Hernoemen Bestanden en Mappen	86
4.13. Negeren bestanden en mappen	88
4.13.1. Patroonherkenning in negerlijsten	89
4.14. Verwijderen, Verplaatsen en Hernoemen	89
4.14.1. Verwijderen van bestanden en mappen	90

4.14.2. Verplaatsen van bestanden en mappen	91
4.14.3. Wijzig hoofdlettergebruik in een bestandsnaam	92
4.14.4. Omgaan met conflicten in hoofdlettergebruik in bestandsnamen	92
4.14.5. Herstellen van Bestandhernoemingen	92
4.14.6. Verwijderen van Bestanden Zonder Versiebeheer	93
4.15. Wijzigingen ongedaan maken	93
4.16. Schonen	94
4.17. Project Instellingen	94
4.17.1. Subversion Eigenschappen	95
4.17.2. TortoiseSVN Project Eigenschappen	99
4.18. Externe Objecten	101
4.18.1. Externe Mappen	101
4.18.2. Externe Bestanden	103
4.19. Vertakken / labelen	104
4.19.1. Aanmaken van Tak of Label	104
4.19.2. Ophalen of Wisselen... ..	106
4.20. Samenvoegen	107
4.20.1. Een Revisie-Bereik Samenvoegen	108
4.20.2. Een tak reïntegreren	110
4.20.3. Twee Verschillende Takken Samenvoegen	111
4.20.4. Samenvoegopties	112
4.20.5. De Resultaten van het Samenvoegen Bekijken	113
4.20.6. Samenvoegen traceren	114
4.20.7. Conflicten Oplossen tijdens Samenvoegen	114
4.20.8. Samenvoegen van een Afgehandelde Tak	115
4.20.9. Onderhoud aan een Tak met Nieuwe Functies	116
4.21. Blokkeren	116
4.21.1. Hoe Het Blokkeren Werkt In Subversion	117
4.21.2. Blokkade verkrijgen	117
4.21.3. Een Blokkade Vrijgeven	118
4.21.4. Blokkadestatus Controleren	119
4.21.5. Niet-geblokkeerde bestanden als Alleen-lezen instellen	119
4.21.6. Hook Scripts Voor Blokkeringen	120
4.22. Een Patch Maken en Toepassen	120
4.22.1. Maken van een Patchbestand	120
4.22.2. Een Patchbestand Toepassen	121
4.23. Wie Heeft Welke Regel Gewijzigd?	122
4.23.1. Verklaar Bestanden	122
4.23.2. Verklaar Verschillen	124
4.24. De Archief-verkenner	125
4.25. Revisie schema's	127
4.25.1. Revisie Schema Knooppunten	128
4.25.2. Wijzigen van de Weergave	129
4.25.3. Gebruik van de grafiek	129
4.25.4. De weergave verversen	129
4.25.5. Boomstructuren inkorten	130
4.26. Exporteren van een Subversion Werkkopie	130
4.26.1. Versiebeheer van een werkkopie verwijderen	132
4.27. Verhuizen van een werkkopie	132
4.28. Integratie met Bug Tracking Systemen / Issue Trackers	133
4.28.1. Issue-nummers aan Logberichten toevoegen	133
4.28.2. Informatie Verkrijgen vanuit de Issue Tracker	136
4.29. Integratie met Web-gebaseerde Archief Viewers	137
4.30. TortoiseSVN Instellingen	138
4.30.1. Algemene instellingen	138
4.30.2. Revisie Grafiek Instellingen	147
4.30.3. Overlappende Pictogram Instellingen	149
4.30.4. Netwerk Instellingen	152

4.30.5. Externe Programma Instellingen	154
4.30.6. Bewaarde Gegevens Instellingen	158
4.30.7. Log Caching	159
4.30.8. Hook Scripts op de Client	162
4.30.9. TortoiseBlame Instellingen	166
4.30.10. Register Instellingen	166
4.30.11. Subversion Werk Mappen	168
4.31. Laatste stap	168
5. Het SubWCRev Programma	169
5.1. De SubWCRev Command Line	169
5.2. Trefwoordvervanging	170
5.3. Voorbeeld trefwoord	171
5.4. COM interface	172
6. IBUGtraqProvider interface	175
6.1. De IBUGtraqProvider interface	175
6.2. De IBUGtraqProvider2 interface	176
A. Veelgestelde Vragen (FAQ)	179
B. Hoe kan ik...	180
B.1. Verplaats een hoop bestanden in één keer	180
B.2. Forceer gebruikers een log-bericht in te typen	180
B.2.1. Hook-script op de server	180
B.2.2. Project eigenschappen	181
B.3. Update Geselecteerde Bestanden in het Archief	181
B.4. Draai Revisies Terug in het Archief (ongedaan maken)	181
B.4.1. Gebruik het logboek boodschappen scherm	181
B.4.2. Gebruik het samenvoeg scherm	181
B.4.3. Gebruik svndumpfilter	182
B.5. Vergelijk Twee Revisies van een Bestand of Map	182
B.6. Een veelgebruikt subproject gebruiken	182
B.6.1. svn:externals Gebruiken	183
B.6.2. Een geneste werkkopie gebruiken	183
B.6.3. Gebruik een relatieve lokatie	183
B.7. Maak een snelkoppeling naar een archief	184
B.8. Negeer bestanden die al opgenomen zijn in het versiebeheer	184
B.9. Versiebeheer van een werkkopie verwijderen	184
B.10. Een werkkopie verwijderen	184
C. Handige tips voor beheerders	185
C.1. Verspreid TortoiseSVN via groepsbeleid	185
C.2. De update controle omleiden	185
C.3. Zetten van de SVN_ASP_DOT_NET_HACK omgevingsvariable	186
C.4. Opties in het contextmenu uitschakelen	186
D. Automatisering van TortoiseSVN	188
D.1. TortoiseSVN commando's	188
D.2. TortoiseIDiff Commando's	192
E. Opdrachtprompt Interface Cross Reference	193
E.1. Conventies en Basis Regels	193
E.2. TortoiseSVN commando's	193
E.2.1. Ophalen	193
E.2.2. Verversen	193
E.2.3. Verversen naar Revisie	194
E.2.4. Vastleggen	194
E.2.5. Verschil	194
E.2.6. Toon logboek	195
E.2.7. Controleer op Wijzigingen	195
E.2.8. Revisie schema	195
E.2.9. Archief Verkenner	195
E.2.10. Bewerk conflicten	196
E.2.11. Opgelost	196

E.2.12. Hernoemen	196
E.2.13. Verwijderen	196
E.2.14. Ongedaan maken	196
E.2.15. Schonen	196
E.2.16. Plaats Blokkade	196
E.2.17. Blokkade Opheffen	196
E.2.18. Tak/Label	197
E.2.19. Optie	197
E.2.20. Samenvoegen	197
E.2.21. Exporteren	197
E.2.22. Locatie wijzigen	197
E.2.23. Maak Archief Hier	198
E.2.24. Toevoegen	198
E.2.25. Importeren	198
E.2.26. Verklaren	198
E.2.27. Voeg aan Negeerlijst toe	198
E.2.28. Maak patch	198
E.2.29. Patch toepassen	198
F. Implementatie details	199
F.1. Overlappende Pictogrammen	199
G. Svnserve beveiligen met SSH	201
G.1. Een Linux server opzetten	201
G.2. Een Windows server opzetten	201
G.3. SSH Client Tools voor gebruik met TortoiseSVN	202
G.4. Maken van OpenSSH Certificaten	202
G.4.1. Sleutels aanmaken met ssh-keygen	202
G.4.2. Sleutels aanmaken met PuTTYgen	202
G.5. Testen met PuTTY	202
G.6. SSH testen met TortoiseSVN	203
G.7. SSH Configuration Varianten	204
Verklarende woordenlijst	205
Register	209

Lijst van figuren

2.1. Een typisch Client/Server Systeem	5
2.2. Het probleem om te ontwijken	6
2.3. De Vergrendel-Wijzig-Ontgrendel Oplossing	7
2.4. De Kopieer-Wijzig-Samenvoeg Oplossing	8
2.5. ...Kopieren-Wijzigen-Samenvoegen Vervolg	9
2.6. Het archief bestandssysteem	11
2.7. Het Archief	13
3.1. Het TortoiseSVN menu voor bestanden zonder versiebeheer	16
4.1. Overlappende pictogrammen weergegeven in de verkenner	38
4.2. Kontekstmenu van een directory reeds onder versiebeheer	39
4.3. Verkenner bestandsmenu van een snelkoppeling in een versiebeheerde map	40
4.4. Authenticatie dialoog	41
4.5. Het invoer venster	42
4.6. Het Ophalen venster	44
4.7. De Commit dialoog	47
4.8. De Vastleggen Dialoog Spellingscontrole	50
4.9. Het voortgangsvenster geeft de voortgang van een vastlegging weer	51
4.10. Voorgangsvenster van een afgeronde verversing	52
4.11. Overlappende pictogrammen weergegeven in de verkenner	58
4.12. Controleer op Wijzigingen	60
4.13. Vastleggen venster met werkset	64
4.14. Het Logboek Boodschappen Venster	66
4.15. Het Context Menu Van Het Bovenste Gedeelte In Het Logboek Boodschappen Venster	67
4.16. Context menu van het bovenste deelvenster voor 2 geselecteerde revisies	70
4.17. Het context menu van het onderste deelvenster van het Logboekboodschappen venster	71
4.18. Het logboekboodschappen venster toont de revisies van samenvoegingen	74
4.19. Vastleggingen per Auteur Histogram	77
4.20. Vastleggingen-per-Auteur taartdiagram	78
4.21. Vastleggingen-per-datum Grafiek	79
4.22. Schakel naar Offline werken dialoog	80
4.23. De Vergelijk Revisies Dialoog	83
4.24. De beeld-verschil viewer	84
4.25. Verkenner context menu voor bestanden zonder versiebeheer	86
4.26. Rechts-slepen-menu voor een map met versiebeheer	87
4.27. Verkenner context menu voor bestanden zonder versiebeheer	88
4.28. Verkenner context menu voor bestanden met versiebeheer	90
4.29. Herstel dialoog	
4.30. Verkenner eigenschappen pagina, Subversion tab	95
4.31. Subversion kenmerken pagina	96
4.32. Toevoegen van eigenschappen	97
4.33. De Tak/Label Dialoog	105
4.34. De Wissel Dialoog	107
4.35. Het Samenvoeg Hulpprogramma - Selecteer Revisie Bereik	109
4.36. Het Samenvoeg Hulpprogramma - Herintegreer Samenvoegen	111
4.37. Het Samenvoeg Hulpprogramma - Samenvoegen van Boomstructuren	112
4.38. Het Samenvoeg Conflict Onderzoeksscherm	115
4.39. De Samenvoeg Re-integratie Dialoog	116
4.40. Het blokkeringsscherm	118
4.41. Het Kijk of er updates zijn scherm	119
4.42. Het Maak Patch Venster	121
4.43. Het Verklar/Annoteren Scherm	122
4.44. TortoiseBlame	123
4.45. De Archief-verkenner	125
4.46. Een Revisie Schema	127
4.47. Het Exporteren-vanaf-een-URL scherm	131

4.48. De Verhuis Dialoog	132
4.49. Voorbeeld issue tracker zoek scherm	137
4.50. Het Instellingen Scherm, Algemene Pagina	139
4.51. Het Instellingen Scherm, Context Menu Pagina	141
4.52. Het Instellingen Scherm, Schermen 1 Pagina	142
4.53. Het Instellingen Scherm, Schermen 2 Pagina	144
4.54. Het Instellingen Scherm, Kleuren Pagina	145
4.55. Het instellingen scherm, Revisie schema pagina	147
4.56. Het instellingen scherm, Revisieschema kleuren pagina	148
4.57. Het instellingen scherm, Overlappend pictogram pagina	149
4.58. Het instellingen scherm, Pictogramset pagina	152
4.59. Het instellingen scherm, Netwerk pagina	153
4.60. Het instellingen scherm, Diff viewer pagina	154
4.61. Het instellingen scherm, Verschil/Samenvoeg uitgebreid scherm	157
4.62. Het instellingen scherm, Opgeslagen gegevens pagina	158
4.63. Het Instellingen Scherm, Log Cache Pagina	159
4.64. Het Instellingen Scherm, Log Cache Statistieken	161
4.65. Het Instellingen Scherm, Hook Scripts Pagina	162
4.66. Het Instellingen Scherm, Hook Scripts Configureren	163
4.67. Het Instellingen Scherm, Issue Tracker Integratie Pagina	165
4.68. Het Instellingen Scherm, TortoiseBlame pagina	166
C.1. De upgrade-dialoog	185

Lijst van tabellen

2.1. Archief Toegang URLs	12
3.1. Apache <code>httpd.conf</code> Instellingen	29
5.1. Lijst met beschikbare opdrachtregel opties	169
5.2. Lijst met beschikbare opdrachtregel opties	170
5.3. COM/automatiseringsmethoden worden ondersteund	172
C.1. Menuopties en hun waarden	186
D.1. Lijst van alle beschikbare commando's en opties	189
D.2. Lijst van beschikbare opties.	192

Voorwoord



TortoiseSVN

- Werk je in een team?
- Is het weleens gebeurd dat je aan een bestand werkte, terwijl tegelijkertijd iemand anders aan hetzelfde bestand werkte? Zijn jouw wijzigingen in dat bestand daardoor verloren gegaan?
- Heb je weleens een bestand opgeslagen en wilde je daarna de wijzigingen terugdraaien? Wilde je weleens bekijken hoe een bestand er een tijdje geleden uitzag?
- Heb je weleens een defect in je project gevonden en wilde je weten wanneer dat defect in je bestanden is gekomen?

Als je een van deze vragen met “ja” hebt beantwoord, dan is TortoiseSVN voor jou geschikt! Lees verder en ontdek hoe TortoiseSVN je kan helpen bij je werk. Het is echt niet moeilijk.

1. Publiek

Dit boek is geschreven voor computer intellectuelen, die Subversion willen gebruiken om hun data te beheren, maar niet graag opdrachtprompts gebruiken. Aangezien TortoiseSVN een windows shell-extensie is, wordt verondersteld dat de gebruiker bekend is met de windows verkenner en weet hoe deze te gebruiken.

2. Leesgids

Dit **Voorwoord** geeft een beknopte toelichting over het TortoiseSVN project, de gemeenschap die hieraan werkt, alsmede de licentievoorwaarden voor gebruik en verspreiding.

Hoofdstuk 1, *Introductie* legt uit wat TortoiseSVN is, wat het doet, waar het vandaan komt en wat de basisinstructies zijn om het op je PC te installeren.

In **Hoofdstuk 2, *Basis Versie-Beheer Concepten*** geven we een korte introductie van het *Subversion* revisie beheer systeem, dat ten grondslag ligt aan TortoiseSVN. Dit is geleend van de documentatie van het Subversion project en licht de verschillende benaderingen van versie beheer toe, alsook hoe Subversion werkt.

Hoofdstuk 3, *Het Archief* legt uit hoe een lokaal archief op te zetten, wat bruikbaar kan zijn om Subversion en TortoiseSVN te testen op een enkele PC. Het gaat ook over het beheer van een archief, wat eveneens relevant is voor archieven op een server. Er is ook in een stuk beschreven hoe een server opgezet moet worden als je er een nodig hebt.

Hoofdstuk 4, *Dagelijkse gebruikersgids* is de belangrijkste sectie, omdat hierin de voornaamste functies van TortoiseSVN alsmede hoe ze te gebruiken, worden uitgelegd. Het is in de vorm van een handleiding, die begint met het ophalen van een werkkopie, het wijzigen ervan, het vastleggen van je wijzigingen, enz. Het vervolgt dan met meer geavanceerde thema's.

Hoofdstuk 5, *Het SubWCRev Programma* is een afzonderlijk programma samengevoegd met TortoiseSVN, welke informatie uit je werkkopie kan halen en in een bestand kan schrijven. Het is bruikbaar om bouw informatie in je project te voegen.

De **Bijlage B, *Hoe kan ik...*** sectie beantwoordt algemene vragen over het uitvoeren van taken, die niet expliciet op andere plekken worden behandeld.

De *Bijlage D, Automatisering van TortoiseSVN* toont hoe de TortoiseSVN GUI dialogen vanaf de opdrachtprompt kunnen worden aangeroepen. Dit is bruikbaar bij scripten waarbij wel gebruikersinteractie nodig is.

De *Bijlage E, Opdrachtprompt Interface Cross Reference* correleren de TortoiseSVN commando's met hun equivalenten in de Subversion commando regel client `svn.exe`.

3. TortoiseSVN is gratis!

TortoiseSVN is gratis. Je hoeft niet te betalen om het te gebruiken en je kunt het gebruiken zoals je zelf wilt. Het is ontwikkeld onder de GNU “General Public License (GPL)”.

TortoiseSVN is een “Open Source” project. Dit betekent dat je volledige leestoegang hebt tot de bronbestanden van dit programma. Je kunt de bronbestanden via de URL <http://code.google.com/p/tortoisesvn/source/browse/> bekijken. Hierbij word je naar een gebruikersnaam en wachtwoord gevraagd. De gebruikersnaam is `guest` en het wachtwoord moet je leeg laten. De meest recente versie (waaraan we op dit moment werken) is te vinden in de `/trunk/` en de opgeleverde versies zijn te vinden onder `/tags/`.

4. Gemeenschap

Zowel TortoiseSVN als Subversion zijn ontwikkeld door een groep mensen, die aan deze projecten werkt. Ze komen uit verschillende landen van de hele wereld en ontwikkelen samen deze mooie programma's.

5. Dankwoord

Tim Kemp
voor het oprichten van het TortoiseSVN project

Stefan Küng
voor het harde werk om TortoiseSVN te maken zoals het nu is

Lübbe Onken
voor de mooie icons, het logo, het zoeken naar defecten, het vertalen en het beheer van de vertalingen

Simon Large
voor het helpen met de documentatie en het opsporen van bugs

Het Subversion Boek
voor de geweldige introductie in Subversion en diens hoofdstuk 2 dat we hier gekopieerd hebben

Het Tigris Stijl project
voor sommige stijlen die hergebruikt zijn in dit document

Onze medewerkers
voor de patches, defect rapportages en nieuwe ideeën, en voor het helpen van anderen door vragen in onze mailinglist te beantwoorden.

Onze donateurs
voor de vele uren plezier met de muziek die zij ons stuurden

6. Gebruikte terminologie in dit document

Om het lezen van de documentatie te vergemakkelijken worden de namen van de schermen en menu's van TortoiseSVN opgemaakt in een ander font. Bijvoorbeeld de Log Dialog.

Een menu keuze wordt weergegeven door een pijl. TortoiseSVN → Show Log betekent: selecteer *Show Log* van het *TortoiseSVN* context menu.

Als een lokaal context menu in een van de TortoiseSVN dialogs verschijnt, wordt het als volgt weergegeven: Context Menu → Save As ...

Gebruikers interface knoppen worden als volgt weergegeven: Druk OK om door te gaan.

Gebruikersacties worden met een vetgedrukt font weergegeven. **Alt+A**: druk de **Alt-Key** op je toetsenbord en terwijl je deze ingedrukt houdt, druk je de **A-Key** ook in. Rechts-slepen: druk de rechtermuis knop in en terwijl je deze ingedrukt houdt, *sleep* je de items naar de nieuwe lokatie.

Systeemitvoer en toetsenbordinput wordt eveneens weergegeven met een ander font.



Belangrijk

Belangrijke opmerkingen zijn gemarkeerd met een pictogram.



Tip

Tips die je leven gemakkelijker maken



Let op

Locaties waar je voorzichtig moet zijn bij wat je doet



Waarschuwing

Daar waar extreme voorzorg moet worden genomen, is gevaar voor datacorruptie of andere vervelende dingen, die kunnen gebeuren als deze waarschuwingen worden genegeerd.



Hoofdstuk 1. Introductie

Versie beheer is de kunst van het beheren van wijzigingen in informatie. Als sinds lange tijd is dit een cruciaal gereedschap voor programmeurs, die meestal kleine wijzigingen in de software maken en dan de volgende dag deze wijzigingen ongedaan maken of controleren. Stel je een team voor met zulke ontwikkelaars die tegelijkertijd werken - en misschien zelfs wel simultaan aan dezelfde bestanden! - en je kunt je voorstellen waarom een goed systeem nodig is om *de mogelijke chaos te beheersen*.

1.1. Wat is TortoiseSVN?

TortoiseSVN is een gratis open-source client voor het *Subversion* versie beheer systeem. Dat wil zeggen dat TortoiseSVN bestanden en folders beheert in de tijd. Bestanden worden opgeslagen in een centraal *archief*. Dit archief lijkt veel op een gewone fileservers, behalve dat het elke wijziging die ooit in de bestanden en folders gemaakt is, onthoudt. Hierdoor ben je in staat om oudere versies van je bestanden te herstellen en de geschiedenis van hoe, wanneer en door wie je data is gewijzigd, te onderzoeken. Dit is de reden dat veel mensen Subversion en versie beheer systemen over het algemeen als een soort van "tijd machine" beschouwen.

Sommige versie beheer systemen zijn ook software configuratie management (SCM) systemen. Deze systemen zijn speciaal gemaakt om source code bomen te beheren en hebben vele kenmerken specifiek voor software ontwikkeling - zoals het automatisch begrijpen van programmeertalen, of het voorzien in gereedschappen voor het bouwen van software. Subversion echter, is niet een van deze systemen; het is een generiek systeem, dat gebruikt kan worden voor het beheren van *elke* verzameling bestanden, inclusief source code.

1.2. TortoiseSVN's Historie

Tim Kemp vond in 2002 dat Subversion een zeer goed versie beheer systeem was, maar dat het tekort schoot op gebied van de GUI. Het idee om een Subversion client met Windows shell te integreren, kwam voort uit de soortgelijke integratie voor CVS genaamd TortoiseCVS.

Tim bestudeerde de source code van TortoiseCVS en gebruikte deze als basis voor TortoiseSVN. Hij startte vervolgens het project, registreerde het domein `tortoisesvn.org` en plaatste de source code online. Gedurende diezelfde tijd zocht Stefan Küng een goed en gratis versie beheer systeem en vond zowel Subversion als de source code van TortoiseSVN. Aangezien TortoiseSVN nog niet klaar voor gebruik was, sloot hij zich bij het project aan en begon te programmeren. Al snel herschreef hij het meeste van de bestaande code en begon met het toevoegen van bestaande commando's en eigenschappen tot aan het punt waar niets van de originele code nog bestond.

Hoe stabiel Subversion werd, hoe meer gebruikers het aantrok; gebruikers die ook TortoiseSVN als Subversion client gingen gebruiken. De gebruikersgroep groeide snel en groeit nog steeds iedere dag. Op dat moment bood Lübke Onken aan om te helpen met leuke icons en een logo voor TortoiseSVN. Hij zorgt ook voor de website en beheert de vertalingen.

1.3. TortoiseSVN's mogelijkheden

Wat maakt TortoiseSVN zo'n goede Subversion client? Zie hier een korte lijst van eigenschappen.

Shell integratie

TortoiseSVN integreert naadloos in de Windows shell (i.e. de Verkenner). Dit betekent dat je kan blijven werken met de gereedschappen waar je al bekend mee bent. En je hoeft niet steeds een andere applicatie op te starten als je de versie beheer functionaliteit nodig hebt!

En daarbij komt nog dat je de Windows Verkenner niet perse hoeft te gebruiken. TortoiseSVN's context menus werken in vele andere bestandbeheer programma's, alsook in de Bestand/Open dialoog, welke in de meeste standaard Windows applicaties aanwezig is. Je moet hierbij wel

bedenken dat TortoiseSVN in eerste instantie ontwikkeld is als uitbreiding op de Windows Verkenner. Daardoor is het mogelijk dat in andere applicaties de integratie niet zo compleet is en bijvoorbeeld de overlappende pictogrammen niet zichtbaar zijn.

Overlappende pictogrammen

De status van elk bestand en folder onder versiebeheer wordt weergegeven met een klein overlappend pictogram. Op die manier kun je direct zien wat de status is van je werkkopie.

Gemakkelijke toegang tot Subversion commando's

Alle Subversion commando's zijn beschikbaar via het context menu van de Verkenner. TortoiseSVN voegt zijn eigen submenu hierin.

Aangezien TortoiseSVN een Subversion client is, willen we je ook een aantal kenmerken van Subversion zelf laten zien.

Map versiebeheer

CVS houdt alleen de geschiedenis van individuele bestanden bij, maar Subversion implementeert een “virtueel” geversioneerd filesysteem, dat wijzigingen in de gehele folderstructuur bijhoudt. Bestanden *en* mappen staan beide onder versiebeheer. Het resultaat is dat er echte commando's zijn om bestanden en mappen te **verplaatsen** en te **kopiëren**.

Atomische commits

Een enkele vastleggen-actie gaat of helemaal, of helemaal niet in het archief. Hierdoor kunnen ontwikkelaars wijzigingen maken en vastleggen als logische brokken.

geversioneerde metadata

Elk bestand en elke map hebben een onzichtbare set van “properties”. Je kunt elk willekeurig sleutel/waarde paar bedenken dat maar je wilt, en opslaan. Eigenschappen worden net als bestandinhoud in de tijd geversioneerd.

Keuze van netwerkklagen

Subversion heeft een abstracte notie wat betreft toegang tot het archief, wat het gemakkelijk maakt voor mensen om een nieuw netwerkmechanisme te implementeren. Subversions “geavanceerde” netwerk server is een module voor een Apache webserver. die een afgeleide van HTTP genaamd WebDAV/DeltaV spreekt. Dit biedt Subversion een groot voordeel als het gaat om stabiliteit en interoperabiliteit, en voorziet in verschillende belangrijke eigenschappen, die ook nog eens gratis zijn: zoals onder andere authenticatie, autorisatie, netwerkdatacompressie, en bladeren in het archief. Een kleinere op zichzelfstaand Subversion serverproces is ook beschikbaar. Deze server gebruikt een eigen netwerkprotocol, die gemakkelijk over ssh getunneld kan worden.

Consistente data-afhandeling

Subversion drukt bestandsverschillen uit door middel van een binair verschilalgoritme, dat identiek werkt voor tekstbestanden (mens-leesbaar) en binaire bestanden (mens-onleesbaar). Beide typen bestanden worden op dezelfde gecompriëerde manier in het archief opgeslagen en verschillen worden over het netwerk in beide richtingen gezonden.

Efficiënt aftakken en labelen

De kosten van aftakken en labelen

Kraakbestandheid

Subversion heeft geen historische bagage; het is geïmplementeerd als een verzameling van shared C libraries met een goed gedefiniëerde API. Dit maakt Subversion zeer onderhoudbaar en bruikbaar door andere applicaties en talen.

1.4. Installeren van TortoiseSVN

1.4.1. Systeemvereisten

TortoiseSVN draait op Windows 2000 SP2, Windows XP of hoger. Windows 98, Windows ME en Windows NT4 worden vanaf TortoiseSVN 1.2.0 niet langer ondersteund, maar je kan nog steeds de oudere versies ophalen als je die echt nodig hebt.

Als je een probleem tijdens of na installatie van TortoiseSVN tegenkomt, raadpleeg dan eerst de [Bijlage A, Veelgestelde Vragen \(FAQ\)](#).

1.4.2. Installatie

TortoiseSVN wordt geleverd met een gemakkelijk te gebruiken installatieprogramma. Dubbelklik op het installatiebestand en volg de instructies op. Het installatieprogramma zorgt voor de rest.



Belangrijk

Je hebt Beheerdersrechten nodig om TortoiseSVN te mogen installeren.

1.4.3. Taalpakketten

De TortoiseSVN gebruikers interface is vertaald in veel verschillende talen, je kunt dus waarschijnlijk je eigen gewenste taal kiezen om te installeren. Je kunt de beschikbare talenpakketten vinden op onze [translation status page](http://tortoisesvn.net/translation_status) [http://tortoisesvn.net/translation_status]. En als jouw gewenste talenpakket nog niet beschikbaar is, ben je welkom om je bij ons team aan te sluiten en je eigen vertaling bij te voegen ;-)

Elk talenpakket is als een .exe installatieprogramma samengesteld. Je hoeft enkel het installatieprogramma te starten en de instructies op te volgen. De volgende keer dat je herstart, zal de vertaling beschikbaar zijn.

1.4.4. Spellingscontrole

TortoiseSVN bevat een spellingscontrole, waarmee je je vastleg logmeldingen kunt controleren. Dit is vooral nuttig als de projecttaal niet jouw moedertaal is. De spellingcontrole gebruikt dezelfde woordenboeken als [OpenOffice](http://openoffice.org) [http://openoffice.org] en [Mozilla](http://mozilla.org) [http://mozilla.org].

Het installatieprogramma voegt automatisch de VS en VK Engelse woordenboeken toe. Als je andere talen wilt, is de simpelste optie om een van de TortoiseSVN talenpakketten te installeren. Dit installeert het gewenste woordenboeken als ook de TortoiseSVN lokale gebruikers interface. De volgende keer dat je herstart, zal het woordenboek beschikbaar zijn.

Of je kunt de woordenboeken zelf installeren. Als je OpenOffice of Mozilla hebt geïnstalleerd, dan kun je de woordenboeken, die in de installatiemappen van die applicaties zijn te vinden, kopiëren. Anders moet je de gewenste woordenboeken van <http://wiki.services.openoffice.org/wiki/Dictionaries> kopiëren.

Als je de woordenboekbestanden hebt, moet je ze waarschijnlijk hernoemen zodat de bestandsnaam alleen lokale karakters bevat. Bijvoorbeeld:

- en_US.aff
- en_US.dic

Daarna kopiëer je ze naar de bin submap van de TortoiseSVN installatiemap. Normaliter is dit C:\Program Files\TortoiseSVN\bin. Als je de bin submap niet wilt vervuilen, dan kun je in plaats daarvan de spellingscheckerbestanden in C:\Program Files\TortoiseSVN\Languages plaatsen. Als die map niet bestaat, moet je hem eerst aanmaken. De volgende keer dat je TortoiseSVN start, zal de spellingschecker beschikbaar zijn.

Als je meerdere woordenboeken installeert, gebruikt TortoiseSVN deze regels om te selecteren welke te gebruiken.

1. Controleer de `tsvn:projectlanguage` instelling. Refereer naar [Paragraaf 4.17, "Project Instellingen"](#) voor informatie over het instellen van projecteigenschappen.

2. Als er geen projecttaal is ingesteld, of als die taal niet is geïnstalleerd, probeer dan de taal die hoort bij de Windows landinstelling.
3. Als de exacte Windows landinstelling niet werkt, probeer dan de “Basis” taal, bv. de_CH (Zwitsers-Duits) wordt de_DE (Duits).
4. Als bovenstaande oplossingen niet werken, dan is de standaardtaal Engels, welke is opgenomen in de standaardinstallatie.

Hoofdstuk 2. Basis Versie-Beheer Concepten

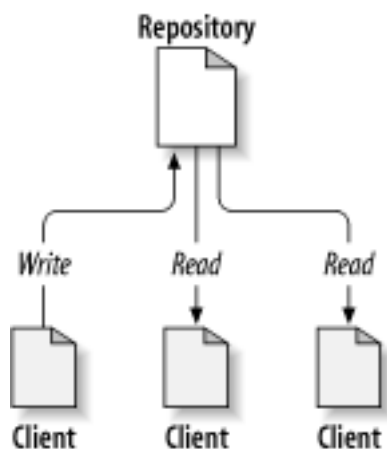
Dit hoofdstuk is een iets gewijzigde versie van hetzelfde hoofdstuk uit het Subversion boek. Een online versie van het Subversion boek is beschikbaar op: <http://svnbook.red-bean.com/>.

Dit hoofdstuk is een korte en informele introductie in Subversion. Als je nieuw bent op gebied van versiebeheer, dan is dit hoofdstuk echt iets voor jou. We beginnen met een uiteenzetting over algemene versiebeheer concepten, verdiepen ons in de specifieke gedachten achter Subversion, en tonen een aantal simpele voorbeelden van toepassingen van Subversion.

Ook al gaan de voorbeelden in dit hoofdstuk over mensen die een verzameling programmabroncode delen, vergeet niet dat Subversion elke verzameling bestanden kan beheren - het is niet beperkt tot assisteren van computerprogrammeurs.

2.1. Het Archief

Subversion is een centraal systeem voor het delen van informatie. De kern is een *archief*, dat een centrale opslagruimte van data is. Het archief slaat informatie op in de vorm van een *bestandssysteemboom* - een typische hiërarchie van bestanden en mappen. Een ongelimiteerd aantal *clients* kan op het archief aansluiten en deze bestanden lezen of schrijven. Door data te schrijven maakt een client informatie beschikbaar voor anderen; door het lezen van data ontvangt de client informatie van anderen.



Figuur 2.1. Een typisch Client/Server Systeem

Waarom is dit nu zo interessant? Tot zoverre lijkt dit op de definitie van een typische bestandsserver. En inderdaad *is* het archief een soort van bestandsserver, maar niet de normale soort. Wat het Subversion archief zo bijzonder maakt, is dat *het elke wijziging onthoudt*, die ooit weggeschreven is: elke wijziging in elk bestand en zelfs wijzigingen in de mappenstructuur, zoals toevoegen, verwijderen en herschikken van bestanden en mappen.

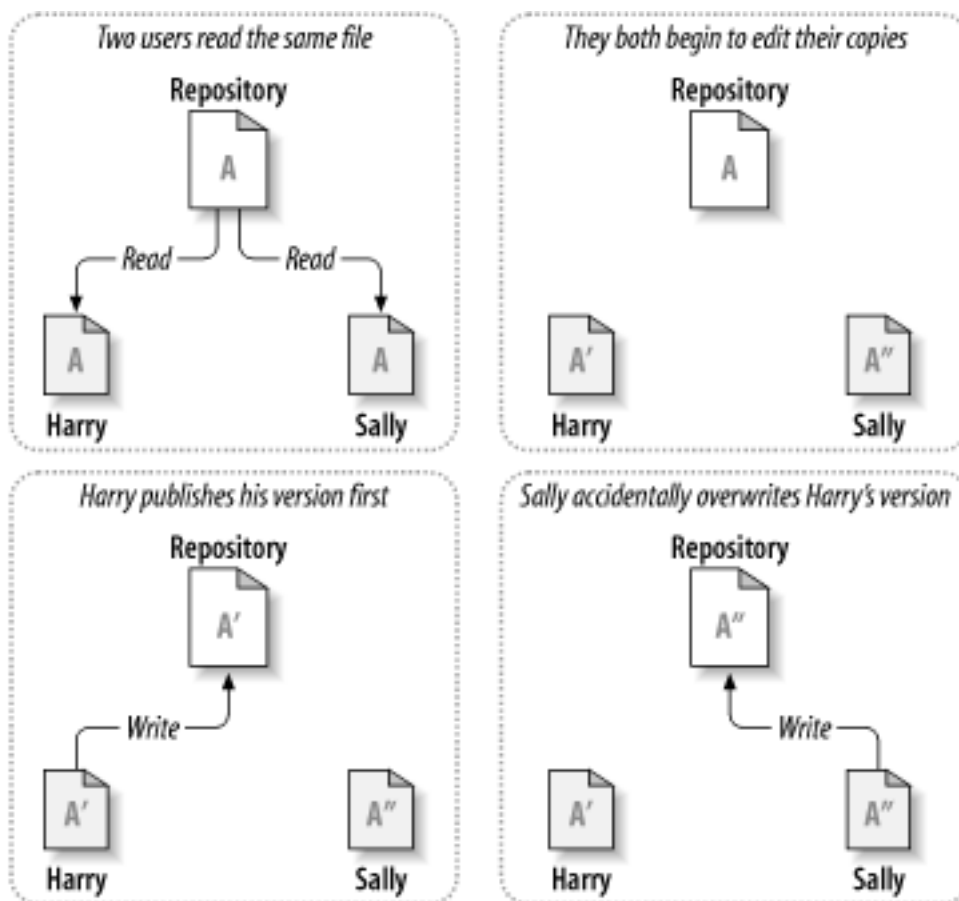
Wanneer een client data van het archief leest, ziet deze standaard alleen de laatste versie van het filesysteem. De client kan echter ook *vorige* versies van het filesysteem bekijken. Een client kan bijvoorbeeld historische vragen stellen, zoals “wat zat er in deze map afgelopen woensdag?”, of “wie was de laatste persoon die dit bestand heeft gewijzigd en welke wijzigingen heeft hij gemaakt?” Dit zijn het soort vragen, dat in het hart zit van elk *versiebeheersysteem*: systemen die zijn ontworpen om wijzigingen in de data vast te leggen en te traceren in de tijd.

2.2. Versioneermodellen

Alle versiebeheersystemen moeten hetzelfde fundamentele probleem oplossen: hoe zorgt een systeem ervoor dat gebruikers informatie kunnen delen, maar voorkomt dat ze per ongeluk op elkaars voeten gaan staan? Het is maar al te gemakkelijk voor gebruikers om per ongeluk elkaars wijzigingen te overschrijven in het archief.

2.2.1. Het Probleem van Bestand Delen

Beschouw het volgende scenario: stel we hebben twee collega's, Harry en Sally. Ze besluiten beide om hetzelfde archief te wijzigen op hetzelfde moment. Als Harry de wijzigingen als eerste in het archief opslaat, dan is het mogelijk dat Sally (enkele ogenblikken later) per ongeluk zijn wijzigingen overschrijft met haar nieuwe versie van het bestand. Alhoewel Harry's wijzigingen niet voor goed verloren zijn (omdat het systeem elke wijziging onthoudt), zijn de wijzigingen die Harry aanbracht *niet* zichtbaar in Sally's nieuwe versie van het bestand, omdat zij Harry's wijzigingen helemaal niet eens gezien heeft. Harry's werk is dus in feite verloren - of ontbreekt in elk geval in de laatste versie van het bestand - en waarschijnlijk ook nog per ongeluk. Dit is precies de situatie die we willen voorkomen!

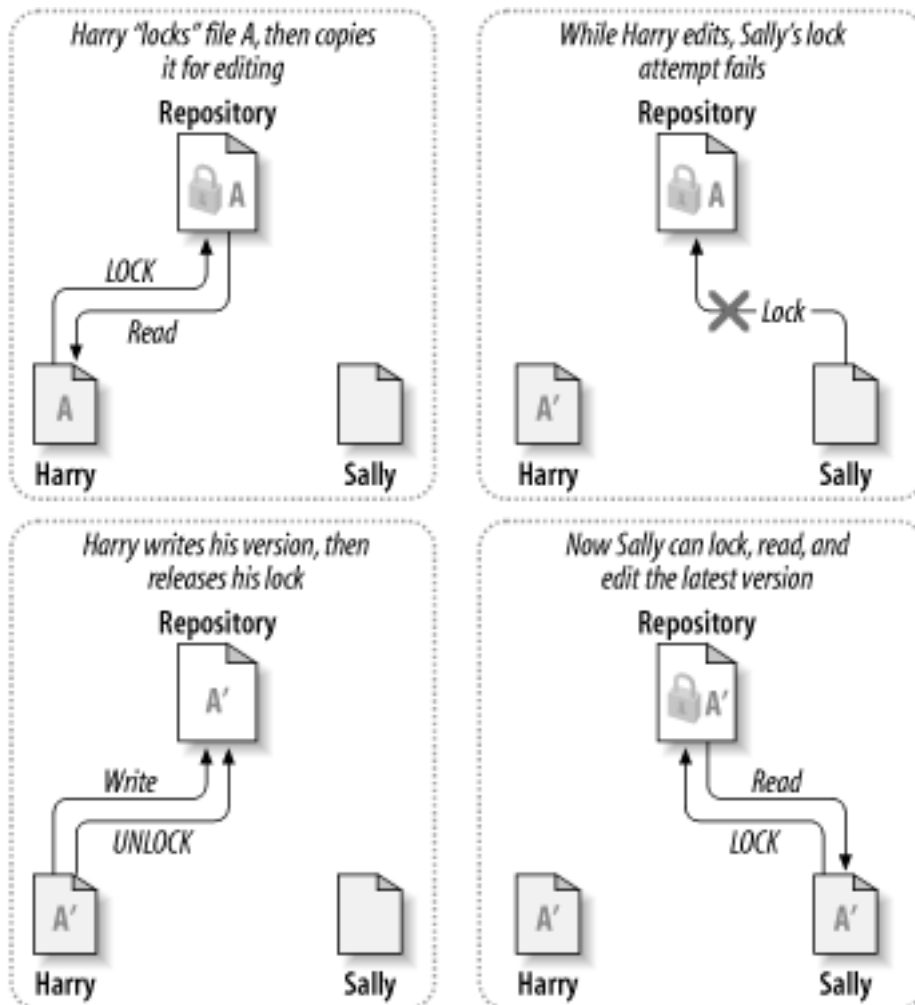


Figuur 2.2. Het probleem om te ontwijken

2.2.2. De Vergrendel-Wijzig-Ontgrendel Oplossing

Veel versiebeheersystemen gebruiken een *blokkeren-wijzigen-blokkade opheffen* model om dit probleem te adresseren, hetgeen een zeer simpele oplossing is. In zo'n systeem staat het archief maar aan één persoon tegelijkertijd toe om een bestand te wijzigen op één bepaald moment. Eerst moet Harry het bestand *blokkeren*, voordat hij kan beginnen om het bestand te wijzigen. Blokkeren van een bestand lijkt op het lenen van een boek bij de bibliotheek; als Harry het bestand heeft geblokkeerd, kan Sally geen wijzigingen

hierin aanbrengen. Als zij probeert het bestand te blokkeren, zal het archief dit verzoek weigeren. Ze kan alleen het bestand bekijken en wachten tot Harry klaar is met zijn wijzigingen en de blokkade opheft. Nadat Harry de blokkade op het bestand heeft opgeheven, is zijn beurt voorbij en is Sally aan de beurt om het bestand te blokkeren en te wijzigen.



Figuur 2.3. De Vergrendel-Wijzig-Ontgrendel Oplossing

Het probleem met "blokkeren-wijzigen-blokkade opheffen" is dat het erg beperkend is en vaak voor gebruikers een wegversperring wordt.

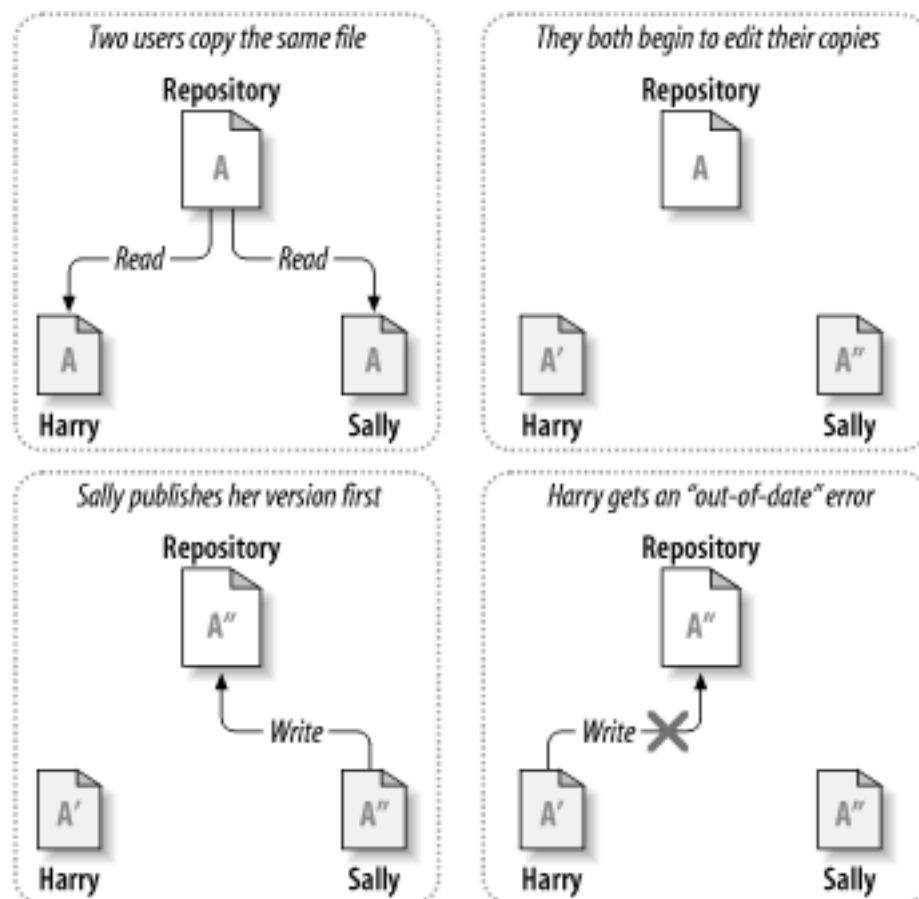
- *Blokkeren kan administratieve problemen veroorzaken.* Soms zal het gebeuren dat Harry een bestand blokkeert en dat daarna vergeet. In de tussentijd kan Sally niets doen, omdat ze aan het wachten is om het bestand te mogen wijzigen. En als Harry dan ook nog op vakantie gaat, moet Sally een beheerder vragen om Harry's blokkade op te heffen. Deze situatie veroorzaakt veel onnodige vertraging en verloren tijd.
- *Blokkeren kan onnodige serializatie veroorzaken.* Stel dat Harry aan het begin van een bestand iets wil wijzigen en Sally aan het einde van datzelfde bestand. Deze wijzigingen overlappen elkaar niet. Ze zouden gemakkelijk tegelijkertijd het bestand kunnen wijzigen, waarbij geen grote schade zal ontstaan, ervanuitgaande dat de wijzigingen netjes samengevoegd worden. Er is geen reden om op elkaars beurt te wachten.
- *Blokkeren kan een vals gevoel van veiligheid geven.* Stel dat Harry bestand A blokkeert en wijzigt, terwijl Sally tegelijkertijd bestand B blokkeert en wijzigt. Maar wat als A en B van elkaar afhankelijk zijn, en de wijzigingen die gemaakt zijn, niet compatibel zijn? A en B zullen ineens niet meer samen

kunnen werken. Het blokkeringssysteem was niet in staat om dit probleem te voorkomen - maar op de een of andere manier gaf het wel een vals gevoel van veiligheid. Het is voor de hand liggend dat Harry en Sally bedenken dat ze door het blokkeren van de bestanden, een veilige, geïsoleerde taak beginnen, wat ze ervan weerhoudt om hun incompatibele wijzigingen te bespreken.

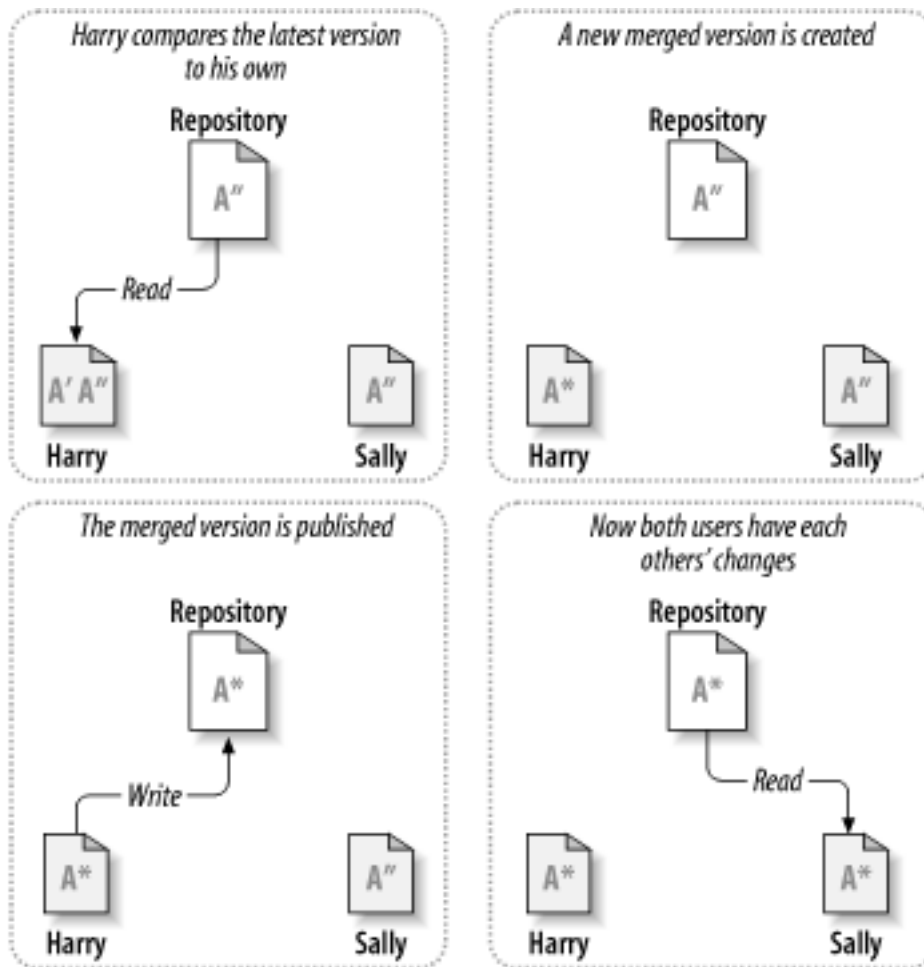
2.2.3. De Kopieer-Wijzig-Samenvoeg Oplossing

Subversion, CVS, en andere versiebeheersystemen gebruiken een *kopieer-wijzig-samenvoeg* model als alternatief voor het blokkeren. In dit model leest elke gebruiker het archief en maakt een persoonlijke *werkkopie* van het bestand of project. Gebruikers werken dan parallel en wijzigen hun persoonlijke kopieën. Uiteindelijk worden alle persoonlijke kopieën samengevoegd in een nieuwe finale versie. Het versiebeheersysteem ondersteunt meestal dit samenvoegen, maar uiteindelijk is een mens verantwoordelijk voor het correct uitvoeren daarvan.

Zie hier een voorbeeld. Stel dat Harry en Sally elk een werkkopie hebben gemaakt vanaf het archief van hetzelfde project. Ze werken tegelijkertijd en wijzigen hetzelfde bestand A in hun werkkopie. Sally slaat haar wijzigingen als eerste in het archief op. Als Harry later probeert zijn wijzigingen op te slaan, informeert het archief hem dat het bestand A *verouderd* is. Met andere woorden, bestand A in het archief is op de een of andere manier in het archief gewijzigd sinds hij het bestand voor het laatst gekopieerd heeft. Harry zal nu zijn client vragen om de wijzigingen vanuit het archief met zijn werkkopie van bestand A *samen te voegen*. De kans is groot dat wijzigingen die Sally heeft gemaakt, niet overlappen met zijn eigen wijzigingen; dus zodra hij beide sets heeft geïntegreerd, kan hij zijn werkkopie in het archief opslaan.



Figuur 2.4. De Kopieer-Wijzig-Samenvoeg Oplossing



Figuur 2.5. ...Kopieren-Wijzigen-Samenvoegen Vervolg

Maar wat gebeurt er als Sally's wijzigingen *wel* overlappen met Harry's wijzigingen? Wat dan? Deze situatie wordt een *conflict* genoemd, en is meestal helemaal niet zo'n probleem. Als Harry zijn client vraagt om de wijzigingen in het archief samen te voegen met zijn werkkopie, wordt zijn kopie van bestand A gemarkeerd als zijnde in staat van conflict: hij zal beide sets van conflicterende wijzigingen kunnen bekijken en handmatig kiezen tussen de sets. Software kan niet automatisch conflicten oplossen; alleen mensen zijn in staat om te begrijpen en de juiste intelligente keuzes te maken. Zodra Harry handmatig de conflicten heeft opgelost (misschien wel door te overleggen met Sally), kan hij de wijzigingen in het samengevoegde bestand opslaan in het archief.

Het kopiëren-wijzigen-samenvoegen model lijkt misschien wat chaotisch, maar in de praktijk werkt het zeer probleemloos. Gebruikers kunnen parallel werken, en hoeven nooit te wachten op elkaar. Als ze aan dezelfde bestanden werken, zal het in de meeste gevallen geen overlappende wijzigingen betreffen; Conflicten treden niet vaak op. En de tijd die het kost om een conflict op te lossen is veel minder dan de tijd die verloren gaat door het blokkeersysteem.

Uiteindelijk komt het allemaal aan op één kritieke factor: communicatie tussen de gebruikers. Als een gebruiker slecht communiceert, zullen zowel syntactische als semantische conflicten meer gaan optreden. Geen enkel systeem kan gebruikers dwingen om perfect te communiceren, en geen enkel systeem kan semantische conflicten detecteren. Het heeft dus geen enkele zin om in slaap gesust te worden door een blokkeersysteem dat zogenaamd conflicten kan voorkomen; in de praktijk lijkt het of blokkeren de productiviteit meer dan wat dan ook remt.

Er is één voorkomende situatie waar het blokkeren-wijzigen-vrijgeven model beter werkt, en dat is als je bestanden hebt, waarvan wijzigingen niet samenvoegbaar zijn. Als je archief bijvoorbeeld grafische

bestanden bevat en twee mensen wijzigen een grafische bestand op hetzelfde moment, is er geen manier om deze wijzigingen samen te voegen. Ofwel de wijzigingen van Harry of die van Sally zullen verloren gaan.

2.2.4. Wat doet Subversion?

Subversion gebruikt standaard de kopieer-wijzigen-samenvoegen methode, en in veel gevallen is dat alles wat je nodig hebt. Echter, vanaf versie 1.2, ondersteund Subversion ook het vergrendelen van bestanden. Als je niet samenvoegbare bestanden hebt, of als het management simpelweg het gebruik van de vergrendelwerkwijze oplegt, levert Subversion nog steeds de functionaliteit die je nodig hebt.

2.3. Subversion in Actie

2.3.1. Werkkopieën

Eerder heb je al wat gelezen over werkkopieën; nu zullen we demonstreren hoe de Subversion client deze creëert en gebruikt.

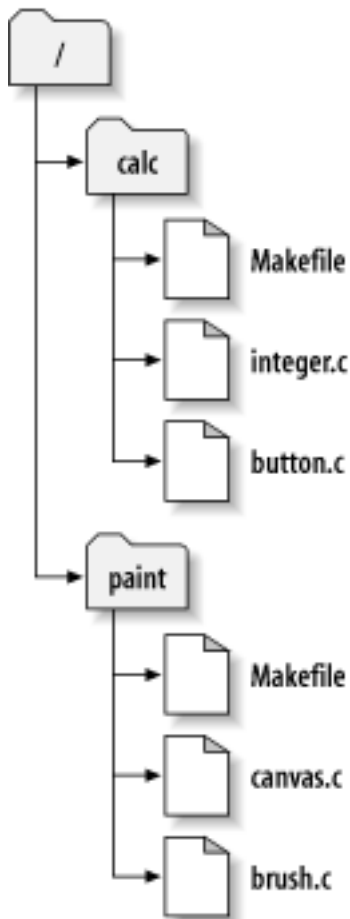
Een Subversion werkkopie is eenvoudigweg een mapstructuur met een verzameling van bestanden op je lokale systeem. Je kunt de bestanden bewerken zoals je wilt en als het broncode bestanden zijn, is het mogelijk om ze vanaf die locatie te compileren op de gebruikelijke manier. Je werkkopie is je eigen privé werkomgeving: Subversion zal zelf geen wijzigingen van anderen in je lokale kopie verwerken, of jouw werk aan anderen beschikbaar stellen, zonder dat je daar zelf opdracht voor hebt gegeven.

Nadat je wijzigingen in je bestanden hebt aangebracht en hebt geverifieerd of deze goed werken, biedt Subversion commando's om je wijzigingen te *publiceren* voor andere mensen die met jou aan het project werken (door middel van het opslaan in het archief). Als andere mensen hun eigen werk publiceren, biedt Subversion je opdrachten om die wijzigingen samen te voegen in je werkkopie (door te lezen vanuit het archief).

In een werkkopie zitten ook bestanden, die door Subversion gemaakt en bijgehouden worden, die ondersteuning bieden bij het uitvoeren van deze opdrachten. Elke map in je werkkopie heeft een submap met de naam `.svn`, beter bekend als de *administratie map* van de werk kopie. De bestanden in de administratie map helpt Subversion om bestanden te detecteren die nog ongepubliceerde wijzigingen hebben en welke bestanden achterhaald zijn ten opzicht van het werk van anderen.

A standaard Subversion archief bevat vaak de (bron) bestanden voor meerdere projecten; vaak is er voor elk project een submap in de boomstructuur van het archief bestandssysteem. Bij deze indeling correspondeert de werkkopie van een gebruiker meestal met een bepaalde submap van het archief.

Bijvoorbeeld, stel je hebt een archief met twee software projecten.



Figuur 2.6. Het archief bestandssysteem

Met andere woorden, de hoofdmap van het archief heeft twee submappen: `paint` en `calc`.

Om een werkkopie te maken, moet je een submapstructuur uit het archief *ophalen*. (De term ophalen ook uitgelegd worden als een manier van vergrendelen, maar in feite wordt er voor jou eenvoudigweg een eigen kopie gemaakt van het project).

Stel je brengt wijzigingen aan in `button.c`. Omdat in de `.svn` map de laatste wijzigingsdatum en originele inhoud wordt bijgehouden, kan Subversion detecteren en aangeven dat het bestand gewijzigd is. Echter maakt Subversion de wijzigingen nog niet bekend aan anderen, totdat jij daarvoor de expliciete opdracht geeft. De handeling van het publiceren van je wijzigingen staat bekend als het *vastleggen* (or *inchecken*) van wijzigingen in het archief.

Om je wijzigingen te publiceren naar anderen, gebruik je het Subversion commando **Vastleggen**.

Met dit commando worden je wijzigingen in `button.c` vastgelegd in het archief; als een andere gebruiker een werkkopie ophaalt van `/calc`, dan zal deze jouw wijzigingen in de laatste versie zien.

Stel je hebt een medewerker, Sally, die op hetzelfde moment als jij een werkkopie ophaalt van `/calc`. Als jij je wijzigingen aan `button.c` vastlegt, blijft Sally's werk ongewijzigd; Subversion wijzigt alleen werkkopieën op aanvraag van de gebruiker.

Om haar project bij te werken, kan Sally Subversion vragen om haar werkkopie *bij te werken*, door het Subversion commando **Verversen** te gebruiken. Hiermee worden jouw wijzigingen in haar kopie verwerkt, net zoals andere wijzigingen die vastgelegd zijn sinds zij haar werkkopie ophaalde.

Merk op dat Sally niet aan hoefde te geven welke bestanden bijgewerkt moesten worden; Subversion gebruikt hiervoor de informatie in de `.svn` map and informatie uit het archief, om te bepalen welke bestanden bijgewerkt moeten worden.

2.3.2. Archief URLs

Subversion archieven kunnen op verschillende methodes benaderd worden – vanaf de lokale harde schijf, of met verschillende netwerk protocollen. Een archief locatie is echter altijd een URL. De vorm van de URL bepaald de toegangsmethode:

Schema	Toegangsmethode
file://	Directe archief toegang op een lokale of netwerk schijf.
http://	Toegang via WebDAV protocol naar een met Subversion-bekende Apache server.
https://	Gelijk aan http://, maar met SSL versleuteling.
svn://	Geautoriseerde TCP/IP toegang met een eigen protocol naar een svnservice server.
svn+ssh://	Geautoriseerd, versleutelde TCP/IP toegang met een eigen protocol naar een svnservice server.

Tabel 2.1. Archief Toegang URLs

Over het algemeen wordt voor de Subversion URLs de standaard syntax gebruikt, waardoor het mogelijk is om server namen en poort nummers als onderdeel van de URL te specificeren. De `file://` toegangsmethode wordt normaal gesproken alleen gebruikt voor lokale toegang, alhoewel het ook gebruikt kan worden voor UNC paden naar een netwerk host. De URL krijgt in die gevallen de schrijfwijze `file://hostname/path/to/repos`. Voor een lokale machine is de `hostname` niet nodig of `localhost`. Hierdoor verschijnen lokale paden vaak met drie schuine strepen, `file:///path/to/repos`.

Ook de gebruikers van het `file://` schema op een Windows systeem moet een onofficiële “standaard” syntax gebruiken voor het benaderen van archieven die op dezelfde machine staan, maar op een andere schijf dan de schijf waar op gewerkt wordt. Eén van de twee volgende URL pad vormen kunnen gebruikt worden, waarbij X de schijfletter aangeeft waarop het archief zich bevindt.

```
file:///X:/path/to/repos
...
file:///X|/path/to/repos
...
```

Merk op dat de URL gewone schuine strepen gebruikt, terwijl de standaard (niet URL) methode voor een pad op een Windows machine een schuine streep naar links gebruikt.

Een FSFS archief kan zonder risico's via een netwerk schijf benaderd worden, maar het is *niet* mogelijk om een BDB archief op deze manier te benaderen.



Waarschuwing

Creëer of bewerk een Berkeley DB archief niet op een netwerk schijf. Een dergelijk archief bestaat *niet* op een niet-lokaal bestandssysteem. Ook niet als je de netwerk schijf koppelt aan een schijfletter. De resultaten zijn onvoorspelbaar als je een Berkeley DB probeert te gebruiken via een netwerk schijf. Misschien zie je direct mysterieuze foutmeldingen, maar het kan ook maanden duren voordat je er achter komt dat het archief subtiel corrupt is geraakt.

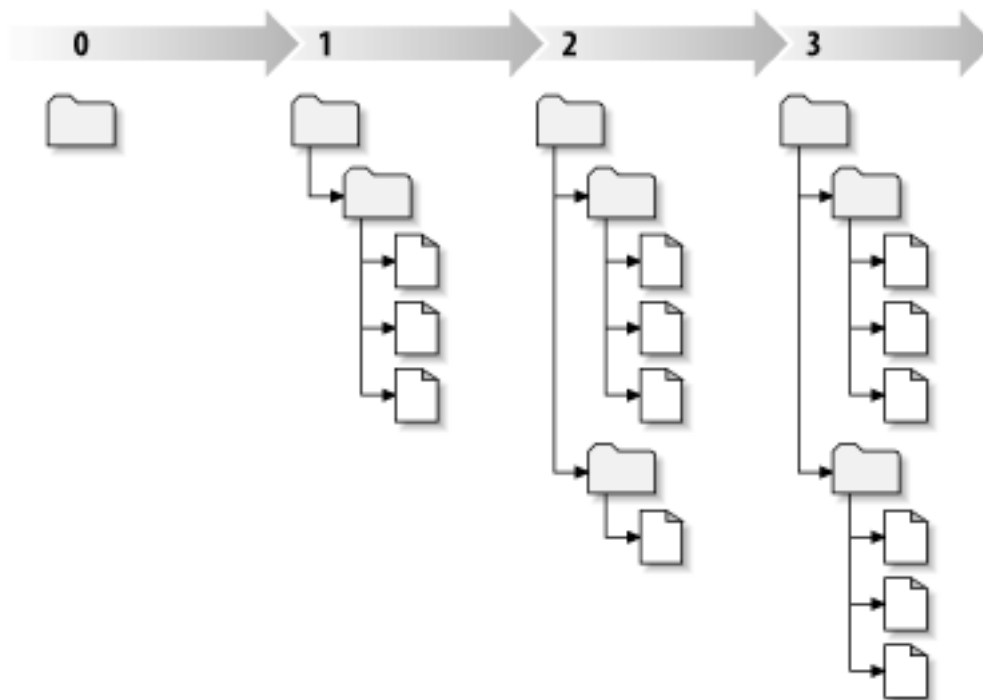
2.3.3. Revisies

Een **SVN Vastleggen** actie worden wijzigingen aan meerdere bestanden en mappen als één enkele actie vastgelegd in het archief. In je werkkopie kun je mappen, bestanden en de inhoud daarvan wijzigen, creëren, verwijderen, hernoemen en kopiëren en vervolgens de hele set van wijzigingen in één keer vastleggen.

In het archief wordt ieder vastlegging behandeld als behandeld als een complete transactie in zijn geheel: dat houdt in dat of alle wijzigingen worden vastgelegd of geen enkele wijziging wordt vastgelegd. Subversion houdt vast aan deze werkwijze met het oog op programma crashes, netwerk problemen en andere gebruikersacties.

Elke keer als het archief een Vastlegging accepteert, creëert het een nieuwe status van de bestandssysteem boom, dit wordt een *revisie* genoemd. Elke revisie krijgt een uniek geheel getal, één nummer hoger dan het nummer van de vorige revisie. De initiële revisie van een vers archief heeft nummer nul en heeft alleen een lege hoofdmap.

Een mooie manier om een archief te visualiseren, is het gebruik van serie van bomen. Stel je een verzameling van revisienummers voor, aan de linker kant beginnend bij 0 en oplopend van links naar rechts. Elke revisie heeft een boom van het bestandssysteem onder zich hangen en elke boom is een "momentopnamen" van het archief zoals het er uit zag na het vastleggen.



Figuur 2.7. Het Archief

Globaal Revisie Nummer

In tegenstelling tot veel andere versie beheersystemen, past Subversion revisie nummers toe op een *complete boom*, niet op individuele bestanden. Elk revisienummer selecteert een gehele boom, een specifieke status waarin het archief zich bevond nadat iemand zijn wijzigingen had vastgelegd. Anders gezegd, revisie N geeft de status van het archief weer naar de Nde vastlegging. Als een Subversion gebruiker praat over "revisie 5 van foo.c", dan bedoeld diegene echt "foo.c zoals het in revisie 5 aanwezig is." Merk op dat over het algemeen een bestand aanwezig in revisie N en M *niet* perse verschillend zijn.

Het is belangrijk om op te merken dat werkkopieën niet altijd overeen komen met één enkele revisie in het archief; er kunnen bestanden in zitten die uit verschillende revisies komen. Bijvoorbeeld, je haalt een werkkopie op uit een archief, waarvan de meest recente revisie nummer 4 is:

```
calc/Makefile:4
    integer.c:4
    button.c:4
```

Op dit moment komt de werk map precies overeen met revisie 4 in het archief. Echter, stel je wijzigt nu `button.c` en legt de wijziging vast. Er van uitgaande dat niemand anders wijzigingen heeft vastgelegd, zal jouw toepassing revisie 5 van het archief creëren, en zal je werkkopie er als volgt uit zien.

```
calc/Makefile:4
    integer.c:4
    button.c:5
```

Stel dat op dit moment Sally een wijziging in `integer.c` vastlegt, dan wordt revisie 6 aangemaakt. Als jij zelf dan **svn verversen** uitvoert om je werkkopie bij te werken, dan zal deze er zo uitzien:

```
calc/Makefile:6
    integer.c:6
    button.c:6
```

Sally's wijziging aan `integer.c` zal in je werkkopie zichtbaar worden en je eigen wijziging zal nog steeds in `button.c` zitten. In dit voorbeeld zal de tekst van `Makefile` gelijk zijn voor revisie 4, 5 en 6, maar zal Subversion je werkkopie van `Makefile` markeren met revisie 6. Dit om aan te geven dat dit nog de actuele versie is. Kortweg, als je een schone versie ophaalt van je werkkopie, dan komt die kopie overeen met één enkele revisie in het archief.

2.3.4. Hoe Werkkopieën Het Archief Bijhouden

Voor elk bestand in een werk map houdt Subversion twee essentiële stukjes informatie bij in de `.svn/` administratie map:

- op welke revisie je werkkopie gebaseerd is (bekend als de *werk revisie* van een bestand) en
- een tijdstempel met het tijdstip van de laatste keer dat je lokale kopie is ververst vanuit het archief.

Met deze informatie en communicerend met het archief, kan Subversion bepalen in welke van de volgende vier fases een bestand verkeert:

Niet gewijzigd en actueel

Het bestand is ongewijzigd in de werk map en er zijn geen wijzigingen in het archief vastgelegd sinds de werk revisie is aangemaakt. Het **vastleggen** en het commando **verversen** op zo'n bestand zal geen actie tot gevolg hebben.

Lokaal gewijzigd en actueel

Het bestand is in de werk map gewijzigd en er zijn nog geen wijzigingen aan het bestand vastgelegd in het archief na de basis revisie. De lokale wijzigingen zijn nog niet vastgelegd in het archief, dus het **vastleggen** commando zal je wijzigingen succesvol publiceren en het **verversen** commando zal niets doen.

Ongewijzigd en achterhaald

Dit bestand is niet gewijzigd in de werk map, maar er is wel een wijziging in het archief. Dit bestand moet uiteindelijk bijgewerkt worden om overeen te komen met de gepubliceerde revisie. Het **commit** commando zal niets doen, het **verversen** zal de laatste wijzigingen in je werkkopie beschikbaar stellen.

Lokaal gewijzigd en achterhaald

Het bestand is lokaal en in het archief gewijzigd. Het **vastleggen** commando zal een *bestand achterhaald* foutmelding geven. Het bestand moet eerst ververst worden; het **verversen** zal proberen

de gepubliceerde wijzigingen samen te voegen met de lokale wijzigingen. Als het Subversion niet lukt om de bestanden samen te voegen op een voor hand liggende manier, dan laat Subversion het aan de gebruiker over om dit probleem op te lossen.

2.4. Samenvatting

In dit hoofdstuk is een aantal fundamentele Subversion concepten behandeld:

- De begrippen van het centrale archief, de werkkopie van de gebruiker en de verzameling van archief revisie bomen zijn geïntroduceerd.
- Enkele eenvoudige voorbeelden zijn gegeven over hoe twee medewerkers Subversion kunnen gebruiken om wijzigingen naar elkaar te publiceren en van elkaar te ontvangen, door gebruik te maken van het 'kopieer-wijzig-samenvoeg' model.
- Er is kort gesproken over hoe Subversion informatie en wijzigingen bijhoudt in een werkkopie.

Hoofdstuk 3. Het Archief

Onafhankelijk van welk protocol er gebruikt wordt om een archief te benaderen, tenminste één keer moet het archief aangemaakt worden. Dit kan gedaan worden met via een Subversion opdracht scherm of met TortoiseSVN.

Als je nog geen Subversion archief aangemaakt hebt, dan wordt het nu tijd om dat te doen.

3.1. Archief creatie

Het is mogelijk om een archief aan te maken met een FSFS backend of met het oudere Berkeley Database (BDB) formaat. Het FSFS formaat is over het algemeen sneller en makkelijker de administreren, bovendien werkt het zonder problemen om netwerk schijven en Windows 98. Het is lang verondersteld dat het BDB formaat stabiel was, eenvoudigweg omdat het langer gebruikt is. Echter aangezien FSFS nu ook al meerdere jaren in het veld in gebruik is, is dat argument niet zo sterk meer. Lees ook eens *Choosing a Data Store* [<http://svnbook.red-bean.com/en/1.5/svn.reposadmin.planning.html#svn.reposadmin.basics.backends>] in het Subversion boek voor meer informatie.

3.1.1. Een archief aanmaken via de opdrachtregel applicatie.

1. Maak een lege map aan met de naam SVN (bijv. D:\SVN\), welke gebruikt zal worden als basis map voor alle archieven.
2. Maak nog een map aan MyNewRepository in D:\SVN\.
3. Open de opdrachtprompt (of DOS-scherf), ga naar D:\SVN\ and type

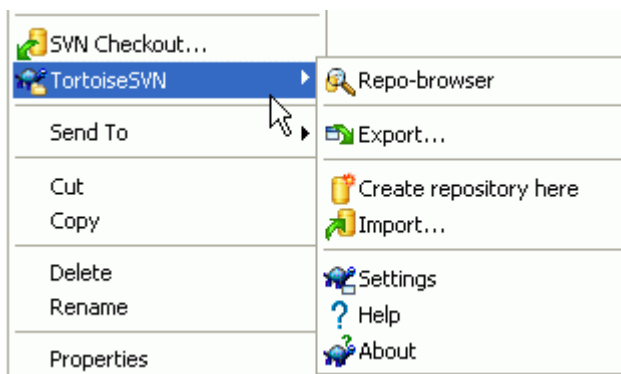
```
svnadmin create --fs-type bdb MyNewRepository
```

of

```
svnadmin create --fs-type fsfs MyNewRepository
```

Nu is er een nieuw archief gemaakt in D:\SVN\MyNewRepository.

3.1.2. Het Archief Aanmaken Met TortoiseSVN



Figuur 3.1. Het TortoiseSVN menu voor bestanden zonder versiebeheer

1. Open de Windows Verkenner
2. Maak een nieuwe map aan en noem het bijvoorbeeld `SVNRepository`
3. Klik met de rechter muisknop op de nieuwe map en selecteer TortoiseSVN → Maak hier een archief....

In de nieuwe map wordt dan een archief gemaakt. *Bewerk of wijzig deze bestanden niet zelf!!!*. Als je foutmeldingen krijgt, controleer dan of de map leeg is en of je schrijfrechten hebt.



Tip

TortoiseSVN biedt niet langer de mogelijkheid om BDB archieven aan te maken, maar het is wel mogelijk om deze via de opdrachtprompt aan te maken. FSFS archieven zijn normaalgesproken makkelijker te beheren en het maakt het ons ook makkelijker om TortoiseSVN te onderhouden, vanwege compatibiliteitsverschillen tussen verschillende BDB versies.

In toekomstige versies van TortoiseSVN wordt de `file://` toegangsmethode naar BDB archieven door deze compatibiliteitsproblemen niet meer ondersteund. Natuurlijk blijft er wel de mogelijkheid om zo'n archief te benaderen met het `svn://`, `http://` of `https://` protocol. Daarom adviseren we met klem nieuwe archieven die d.m.v. het `file://` protocol altijd aan te maken als FSFS.

Ons advies is om de `file://` methode alleen te gebruiken voor lokale testdoeleinden. Het gebruik van een server is veiliger en betrouwbaarder in alle situaties waarbij er meer dan één ontwikkelaar gebruik maakt van het archief.

3.1.3. Lokale Toegang tot het Archief

Om een lokaal archief te benaderen, moet je het pad naar de map weten. Onthoudt dat Subversion alle paden naar het archief verwacht in het formaat: `file:///C:/SVNRepository/`. Merk op dat er normale schuine strepen worden gebruikt.

Om een archief op een netwerk te benaderen, kan een schijfletter toegewezen worden of door gebruik te maken van het UNC pad. Voor UNC paden moet het formaat `file://ServerName/path/to/repos/` gebruikt worden. Merk op dat er maar 2 schuinen strepen voor staan.

Voordat SVN 1.2 er was, moesten UNC paden op een onduidelijkere manier opgegeven worden `file:///\\ServerName/path/to/repos`. Dit formaat wordt nog wel ondersteund, maar het gebruik wordt niet aanbevolen.



Waarschuwing

Creëer of bewerk een Berkeley DB archief niet op een netwerk schijf. Een dergelijk archief bestaat *niet* op een niet-lokaal bestandssysteem. Ook niet als je de netwerk schijf koppelt aan een schijfletter. De resultaten zijn onvoorspelbaar als je een Berkeley DB probeert te gebruiken via een netwerk schijf. Misschien zie je direct mysterieuze foutmeldingen, maar het kan ook maanden duren voordat je er achter komt dat het archief subtiel corrupt is geraakt.

3.1.4. Een archief benaderen op een Netwerk schijf

In theorie is het mogelijk om een FSFS archief op een netwerk schijf met meerdere gebruikers te benaderen met het `file://` protocol, dit is echter *niet* aanbevolen. We raden dit *sterk* af en ondersteunen gebruik op die manier niet.

Ten eerste geef je elke gebruiker directe schrijf rechten op het archief, zodat iedereen per ongeluk het hele archief weg kan gooien of het op een andere manier onbruikbaar kan maken.

Ten tweede ondersteund niet elke netwerk bestandsgebruik protocol de manier van vergrendelen die Subversion nodig heeft, waardoor je archief mogelijk onbruikbaar wordt. Dit hoeft niet direct te gebeuren, maar bijvoorbeeld op een moment dat twee gebruikers het archief tegelijkertijd benaderen.

Ten derde moeten de permissies exact ingesteld worden, wat misschien nog net lukt met Windows, maar met SAMBA is dit erg lastig.

`file://` gebruik is bedoeld voor lokaal gebruik met een enkele gebruiker, in het bijzonder voor testen en debuggen. Als je het archief wilt delen, dan moet je *echt* een server opbouwen, wat niet zo moeilijk is als je denkt. Lees [Paragraaf 3.5, “Toegang hebben tot het Archief”](#) voor richtlijnen bij het kiezen en opzetten van een server.

3.1.5. Archief indeling

Voordat je je data in het archief opneemt, moet je nadenken hoe je de indeling wilt hebben. Als je één van de aanbevolen indelingen kiest, zul je dat later veel profijt van hebben.

Er zijn een paar standaard aanbevolen indelingen voor een archief. Vaak wordt er een `trunk` map aangemaakt voor de “basislijn” van de ontwikkeling, een `branches` map voor vertakkingen en een `tags` map voor labels (vrijgaves). Als een archief maar één project bevat, worden van deze mappen op hoofdniveau gemaakt:

```
/trunk
/branches
/tags
```

Als een archief meerdere projecten bevat, wordt er vaak voor een indeling per vertakking gekozen:

```
/trunk/paint
/trunk/calc
/branches/paint
/branches/calc
/tags/paint
/tags/calc
```

...of per project:

```
/paint/trunk
/paint/branches
/paint/tags
/calc/trunk
/calc/branches
/calc/tags
```

Indexeren per project heeft zin als de projecten niet veel met elkaar te maken hebben en elk project afzonderlijk opgehaald wordt uit het archief. Voor projecten die je in één keer wilt ophalen of die met elkaar verbonden zijn in een enkele distributie is het vaak om te indexeren per vertakking. Hiermee hoeft je alleen de basislijn (`trunk`) op te halen en de relaties tussen de projecten zijn eenvoudiger te zien.

Als de `/trunk` `/tags` `/branches` aanpak op hoofdniveau toepast, moet je de hele boomstructuur kopiëren voor elke vertakking en label, en in sommige andere gevallen biedt deze structuur juist de meeste flexibiliteit.

Voor projecten zonder onderlinge relatie heb je mogelijk de voorkeur voor afzonderlijkearchieven. Als je wijzigingen vastlegt, wordt het revisienummer van het gehele archief verhoogd, niet het nummer van

een project. Het hebben van twee ongerelateerde projecten in één archief betekend dat er grote sprongen kunnen zitten in de revisienummers. De Subversion en TortoiseSVN projecten bevinden zich op hetzelfde hostadres, maar zijn twee aparte archieven waarmee onafhankelijk ontwikkeld kan worden, en dus geen verwarring over build nummers.

Natuurlijk ben je vrij om deze standaard indelingen te negeren. Je kunt elke gewenste variatie creëren, net welke jou en je team het beste past. Onthoudt dat wat je ook kiest, je er niet voor eeuwig aan vast zit. Het is mogelijk om je archief te reorganiseren op elke gewenst moment. Omdat vertakkingen en labels normale mappen zijn, kan TortoiseSVN deze voor je verplaatsen of hernoemen naar hartenlust.

Het schakelen van de ene naar de andere indeling is simpel een door de server uitgevoerde verplaatsactie; Mocht je de indeling van het archief niet handig vinden, verplaats de mappen naar hartenlust.

Als je nog geen basis mapstructuur structuur in het archief hebt aangemaakt, doe dat dan nu. Er zijn twee methodes om dit te doen. Als je de /trunk /tags /branches structuur wilt aanmaken, dan kun je de archief-verkenner hiervoor gebruiken (door drie keer apart vastleg acties). Als je een dieper hiërarchische structuur wilt maken, dan is het makkelijker om dat eerst op de schijf te doen en deze structuur dan in één keer te importeren, door:

1. creëer een nieuwe lege map op je harde schijf
2. Maak je gewenste mapstructuur in die map aan – sla er nog geen bestanden in op!
3. Importeer deze structuur in je archief, klik met rechter muisknop op de map en kies TortoiseSVN → Importeren.... Hiermee wordt je tijdelijke map in de hoofdmap van het archief gezet waarmee de basis van het archief gecreëerd is.

Merk op dat de naam van de map die je importeert, niet verschijnt in het archief, alleen de inhoud daarvan. Bijvoorbeeld, creëer de volgende structuur:

```
C:\Temp\New\trunk
C:\Temp\New\branches
C:\Temp\New\tags
```

Importeren van C:\Temp\New in de hoofdmap van het archief heeft het volgende resultaat:

```
/trunk
/branches
/tags
```

3.2. Archief Reservekopie

Onafhankelijk van welke type archief je gebruikt is het van vitaal belang dat je regelmatig een reservekopie maakt en deze controleert. Als er een fout optreedt op de server, kun je misschien nog wel bij een recente versie van je bestanden komen, maar zonder het archief zal alle historie verloren zijn.

De meest eenvoudige (maar niet aanbevolen) methode is simpelweg een kopie te maken van de archief map op een apart medium. Echter moet je dan zeker weten dat geen enkel proces de data op *geen enkele wijze* dan ook gebruikt op dat moment. Een BDB archief wordt beschreven ook al lijkt het er op dat er alleen gelezen wordt, zoals bijvoorbeeld bij het opvragen van de status. Als je archief benaderd wordt tijdens het maken van een reservekopie, (web browser geopend, WebSVN, etc.) dan zal deze waardeloos zijn.

De aanbevolen methode hiervoor is gebruik te maken van


```
svnadmin hotcopy path/to/repository path/to/backup --clean-logs
```

voor het veilig kopiëren van je archief. Daarna kan een reservekopie gemaakt worden dan die kopie. De `--clean-logs` parameter is niet noodzakelijk, maar verwijdert overbodige log bestanden als je een reservekopie maakt van een BDB archief, waarmee mogelijk ruimte wordt bespaard.

De `svnadmin` tool wordt automatisch geïnstalleerd als je de Subversion opdrachtprompt client installeert. De beste manier om de opdrachtprompt programma's op een Windows PC te installeren is door de Windows installatie versie te downloaden. Deze is beter gecomprimeerd dan de `.zip` versie en dus kleiner, en de paden worden tijdens de installatie meteen goed ingesteld. De meest recente versie van de Subversion opdrachtprompt client kun je downloaden via <http://subversion.apache.org/getting.html>.

3.3. Server gesitueerde hook scripts

Een hook script is een programma dat gestart wordt door een gebeurtenis in het archief, bijvoorbeeld de creatie van een nieuwe revisie of de modificatie van een niet versiebeheerde eigenschap. Elke hook krijgt genoeg informatie om te weten welke gebeurtenis er heeft plaatsgevonden, om welk systeem het gaat en de naam van de gebruiker die de gebeurtenis veroorzaakte. Afhankelijk van de uitkomst van de hook, kan het hook programma de action continueren, stoppen, of op een manier opschorten. Voor meer details over de geïmplementeerde hooks, lees het hoofdstuk *Hook Scripts* [<http://svnbook.red-bean.com/en/1.5/svn.reposadmin.create.html#svn.reposadmin.create.hooks>] in het Subversion boek.

Deze hook scripts worden uitgevoerd door de server waarop het archief zich bevindt. Met TortoiseSVN is het mogelijk om eigenlijk hook scripts te maken, die lokaal uitgevoerd worden bij bepaalde gebeurtenissen. Lees [Paragraaf 4.30.8, "Hook Scripts op de Client"](#) voor meer informatie.

Voorbeelden van hook scripts staan in de `hooks` map van het archief. Deze voorbeelden zijn geschikt voor Unix/Linux servers en kunnen aangepast worden om te gebruiken op een Windows server. De hook kan een batch bestand zijn of een executable. De onderstaande voorbeeld batch kan gebruikt worden voor een revisieeigenschap-wijziging.

```
rem Only allow log messages to be changed.
if "%4" == "svn:log" exit 0
echo Property '%4' cannot be changed >&2
exit 1
```

Merk op dat alles wat verstuurd wordt met `stdout` genegeerd wordt. Als je een bericht wilt tonen in het Vastleggen geweigerd dialoogvenster, dan moet je dit sturen naar `stderr`. In een batch bestand kun je dit doen met `>&2`

3.4. Links voor Ophalen

Als je je Subversion archief beschikbaar wilt stellen aan anderen, dan wil je mogelijk een link naar je archief op je website zetten. Je kunt hiervoor bijvoorbeeld een *Ophalen-link* voor andere TortoiseSVN gebruikers gebruiken.

Bij de installatie van TortoiseSVN, registreert deze een nieuw protocol, `tsvn:`. Als een TortoiseSVN gebruiker op zo'n link klikt, zal automatisch het Ophalen dialoogvenster geopend worden met de URL ingevuld.

Om een dergelijke link in je eigen HTML pagina op te nemen, moet je de code toevoegen die er bijvoorbeeld zo uitziet:

```
<a href="tsvn:http://project.domain.org/svn/trunk">
</a>
```

Natuurlijk ziet het er nog beter uit als je een passend plaatje gebruikt, bijvoorbeeld het *TortoiseSVN logo* [<http://tortoisesvn.tigris.org/images/TortoiseCheckout.png>] of een eigen figuur.

```
<a href="tsvn:http://project.domain.org/svn/trunk">
<img src=TortoiseCheckout.png></a>
```

Het is ook mogelijk om een link naar een specifieke revisie te maken, bijvoorbeeld:

```
<a href="tsvn:http://project.domain.org/svn/trunk?100">
</a>
```

3.5. Toegang hebben tot het Archief

Voor het gebruik van TortoiseSVN (of een andere Subversion client) moet je een plaats hebben waar het archief opgeslagen wordt. Je kunt het archief lokaal opslaan en benaderen met het `file://` protocol, of je kunt het archief op een serverplaatsen en het benaderen met het `http://` of `svn://` protocol. De beide server protocollen kunnen ook met een versleuteling werken. Je gebruikt dan `https://`, `svn+ssh://` of `svn://` met SASL.

Als je een openbaar hosting service gebruikt, zoals *Google Code* [<http://code.google.com/hosting/>] of als je server al geconfigureerd is door iemand anders, dan hoef je niets anders te doen. Lees dan verder bij [Hoofdstuk 4, Dagelijkse gebruikersgids](#).

Indien je geen server hebt, alleen werkt of Subversion en TortoiseSVN evalueert, dan is een lokaal archief waarschijnlijk de beste keuze. Maak een archief aan op je PC zoals omschreven in [Hoofdstuk 3, Het Archief](#). De rest van dit hoofdstuk kan overgeslagen worden en lees verder vanaf [Hoofdstuk 4, Dagelijkse gebruikersgids](#).

Als je een archief op een netwerk schijf wilt maken voor meerdere gebruikers, denk daar dan nog eens over na. Lees [Paragraaf 3.1.4, "Een archief benaderen op een Netwerk schijf"](#) om te begrijpen waarom wij dat een slecht idee vinden. Het opzetten van een server is niet zo moeilijk als het lijkt en een server geeft je een betere betrouwbaarheid en mogelijk snelheid.

De volgende paragrafen beschrijven stapsgewijs hoe een server op een Windows machine opgezet moet worden. Natuurlijk is het ook mogelijk om de server op een Linux machine te bouwen, maar dat ligt buiten de scope van dit document. Meer informatie over de Subversion opties en hoe de beste architectuur te kiezen voor jouw situatie is beschreven in het Subversion boek onder [Server Configuration](#) [<http://svnbook.red-bean.com/en/1.5/svn.serverconfig.html>].

3.6. Svnserve Based Server

3.6.1. Introductie

Bij Subversion zit Svnserve, een lichtgewicht stand-alone server, die gebruikt maakt van een standaard TCP/IP verbinding. Deze is ideaal voor kleine installaties of daar waar een volledige Apache server niet gebruikt kan worden.

In de meeste gevallen is svnserve makkelijker om op te zetten en werkt sneller dan een Apache server, maar ontbeert enkele geavanceerde functies. Nu SASL ook ondersteund wordt is het nog veilig ook.

3.6.2. Installatie svnserve

1. Haal de meest recente versie van Subversion op via <http://subversion.apache.org/getting.html>. Een voorbereid installatie programma kan opgehaald worden vanaf CollabNet via <http://www.collab.net/downloads/subversion>. Dit installatie programma stelt svnserve in als een Windows service en installeert ook de onderdelen die nodig zijn voor SASL beveiliging.

2. Als je al een versie van Subversion geïnstalleerd hebt en svnservice opgestart is, dan moet je deze eerst stoppen.
3. Start de Subversion installatie. Als je de installatie op je server uitvoert (aanbevolen), dan kun stap 4 overslaan.
4. Open de windows verkener, ga naar de installatie map van Subversion (meestal C:\Program Files\Subversion) en zoek in de bin map de bestanden svnservice.exe, intl3_svn.dll, libapr.dll, libapriconv.dll, libapriutil.dll, libdb*.dll, libeay32.dll en sslsleay32.dll. Kopieer deze bestanden, of kopieer alle bestanden in de bin map, naar een map op de server, bijvoorbeeld c:\svnservice.

3.6.3. Svnservice opstarten

Nu svnservice geïnstalleerd is, moet deze nog op je server opgestart worden. Dit kan het makkelijkst in een opdrachtprompt of met een snelkoppeling en het volgende commando te gebruiken: :

```
svnservice.exe --daemon
```

. Svnservice gaat nu wachten op binnenkomende aanvragen op poort 3690. De --daemon optie geeft aan dat svnservice als daemon process moet werken, zodat het altijd loopt totdat het handmatig wordt gestopt.

Als je nog geen archief hebt aangemaakt, volg dan de instructies van de Apache server installatie configuratie [Paragraaf 3.7.4, "Configuratie"](#).

Om te testen of svnservice werkt, kies TortoiseSVN → Archief-verkener om een archief te bekijken.

Stel dat je archief opgeslagen is op c:\repos\TestRepo en je server heeft de naam localhost, voer dan:

```
svn://localhost/repos/TestRepo
```

in als de archief-verkener daar om vraagt.

Het is ook mogelijk in de veiligheid te verhogen en tijd te besparen bij het invoeren van URLs met svnservice door gebruik te maken van de --root optie voor het instellen van de hoofd map en het beperken van toegang tot een bepaalde map op de server:

```
svnservice.exe --daemon --root drive:\path\to\repository\root
```

Met de vorige test kan svnservice opstarten als:

```
svnservice.exe --daemon --root c:\repos
```

En in TortoiseSVN is de archief-verkener URL verkort tot: :

```
svn://localhost/TestRepo
```

Merk op dat de --root optie ook nodig als het archief op een andere partitie of schijf dan svnservice is opgeslagen.

Svnservice kan elk willekeurig aantalarchieven bedienen. Sla dearchieven op in de hoofd of een onderliggende map zoals juist gedefinieerd en benader deze door gebruik te maken van de relatieve URL vanaf de hoofdmap.



Waarschuwing

Creëer of bewerk een Berkeley DB archief niet op een netwerk schijf. Een dergelijk archief bestaat *niet* op een niet-lokaal bestandssysteem. Ook niet als je de netwerk schijf koppelt aan een schijfletter. De resultaten zijn onvoorspelbaar als je een Berkeley DB probeert te gebruiken via een netwerk schijf. Misschien zie je direct mysterieuze foutmeldingen, maar het kan ook maanden duren voordat je er achter komt dat het archief subtiel corrupt is geraakt.

3.6.3.1. Svnserve als service laten draaien

Svnserve de een gebruiker laten opstarten is niet de beste methode. Er moet dan altijd een gebruiker ingelogd zijn en onthouden moet worden dat svnserve gestart moet worden na het herstarten van het systeem. Beter is het om svnserve als een Windows service te laten lopen. Vanaf Subversion 1.4 kan svnserve als native Windows service geïnstalleerd worden.

Om svnserve als Windows service te installeren, zodat deze automatisch start bij het opstarten van Windows, moet het volgende commando in zijn geheel gegeven worden:

```
sc create svnserve binpath= "c:\svnserve\svnserve.exe --service
--root c:\repos" displayname= "Subversion" depend= tcpip
start= auto
```

Als één van de paden spaties bevat, dan moet je quotes gebruiken, zoals volgt:

```
sc create svnserve binpath= "
\"C:\Program Files\Subversion\bin\svnserve.exe\"
--service --root c:\repos" displayname= "Subversion"
depend= tcpip start= auto
```

Het is ook mogelijk om een beschrijving achter het commando op te geven. Dit wordt dan weergegeven in de Windows Service Manager.

```
sc description svnserve "Subversion server (svnserve)"
```

Merk de ongebruikelijke opmaak op van `sc`. Bij een `optie=` waarde paar staat er een spatie na de `=` voor de waarde, niet tussen de optie en de `=`.



Tip

Microsoft beveelt aan om services te laten draaien onder het Local Service of Network Service account. Lees voor meer informatie *The Services and Service Accounts Security Planning Guide* [<http://www.microsoft.com/technet/security/topics/serversecurity/serviceaccount/default.mspx>]. Om een service aan te maken onder het Local Service account, voeg het volgende toe aan het voorbeeld van hierboven.

```
obj= "NT AUTHORITY\LocalService"
```

Merk op dat je de Local Service rechten moet toekennen aan Subversion en je archieven, als ook aan elke applicatie die gebruikt wordt door een hook script. De gedefinieerde groep hiervoor heet "LOCAL SERVICE".

Zodra je de service geïnstalleerd hebt, moet je de services manager starten om de gedefinieerde service te starten (dit is alleen de eerste keer nodig, daarna start het elke keer als je het systeem opnieuw start)

Lees voor meer informatie [Windows Service Support for Svnserve](http://svn.collab.net/repos/svn/trunk/notes/windows-service.txt) [http://svn.collab.net/repos/svn/trunk/notes/windows-service.txt].

Als je een eerdere versie van svnservice geïnstalleerd hebt, waarbij de SVNService wrapper is gebruikt, moet je de wrapper eerst als service stoppen en dan de-installeren als service. Gebruik het commando

```
svnservice -remove
```

om de register sleutel te verwijderen.

3.6.4. Standaard Autorisatie met svnservice

De standaard svnservice instellingen geven anoniemen alleen-lezen rechten. Dit houdt in dat je een `svn://` URL kunt gebruiken om te verversen of op te halen, of de archief-verkenner in TortoiseSVN om het archief te bekijken, maar het vastleggen van aanpassingen is niet mogelijk.

Om schijf-rechten toe te kennen op een archief, moet je het `conf/svnservice.conf` bestand in je archief map bewerken. Dit bestand bestuurd de instellingen van de svnservice daemon en bevat ook nog nuttige documentatie.

Het is mogelijk om anoniemen schrijf-rechten te geven door het volgende in te stellen:

```
[general]
anon-access = write
```

Je weet dan echter niet wie er wijzigingen in het archief heeft vastgelegd, omdat de `svn:author` optie dan leeg blijft. Je hebt dan ook geen controle over wie er wijzigingen vastlegt in het archief. Dit is best wel een riskante configuratie!

Om dit te voorkomen kun je een database beveiligen met een wachtwoord:

```
[general]
anon-access = none
auth-access = write
password-db = userfile
```

Waarbij `userfile` een bestand is die opgeslagen is in dezelfde map als `svnservice.conf`. Dit bestand kan ergens anders in je bestandssysteem opgeslagen worden (handig als je meerdere archieven hebt die dezelfde toegangsrechten nodig hebben) en er kan naar verwezen worden met een absoluut pad, of een pad relatief ten opzichte van de `conf` map. Als je een pad gebruikt, dan moet dit met `/de/unix/` methode. Het gebruik van `\` of schijfletters zal niet werken. De `userfile` moet er uit zien als

```
[users]
username = password
...
```

Dit voorbeeld verbiedt alle toegang voor ongeautoriseerde (anonieme) gebruikers en geeft lees- en schrijfrechten aan de gebruikers in de `userfile`.



Tip

Als je meerdere archieven beheert met dezelfde wachtwoord database, dan kan het makkelijker worden door gebruik te maken van een zogenaamd authenticatie gebied. TortoiseSVN kan de gebruikerslidmaatschappen tijdelijk opslaan, zodat je deze maar één keer in hoeft te vullen. Meer informatie hierover staat in het

Subversion boek, *Create a 'users' file and realm* [<http://svnbook.red-bean.com/en/1.5/svn.serverconfig.svnservice.html#svn.serverconfig.svnservice.auth.users>] en *Client Credentials* *Caching* [<http://svnbook.red-bean.com/en/1.5/svn.serverconfig.netmodel.html#svn.serverconfig.netmodel.credcache>]

3.6.5. Verbeterde beveiliging met SASL

3.6.5.1. Wat is SALS?

De Cyrus Simple Authentication en Security Layer is openbron software geschreven door de Carnegie Mellon Universiteit. Het voegt generieke authenticatie en encryptie mogelijkheden toe aan een netwerk protocol. Sinds versie 1.5 kunnen Subversion's svnservice server en de TortoiseSVN applicatie omgaan met deze bibliotheek.

Voor een completere beschrijving van de mogelijkheden, moet je het Subversion boek lezen *Using svnservice with SASL* [<http://svnbook.red-bean.com/en/1.5/svn.serverconfig.svnservice.html#svn.serverconfig.svnservice.sasl>]. Als je alleen op eenvoudige manier veilige authenticatie en encryptie op een Windows server wilt gebruiken, zodat je archief veilig gebruikt kan worden via het grote gemene Internet, lees dan verder...

3.6.5.2. SASL Authenticatie

Om specifieke SASL mechanismes op de server aan te zetten moet je eerst drie dingen doen. Ten eerste, maak een [sasl] sectie in het svnservice.conf bestand van je archief, met de volgende parameter:

```
use-sasl = true
```

Vervolgens maak je een bestand met de naam svn.conf op een makkelijk plaats – normaal gesproken in de map waar subversion is geïnstalleerd.

Daarna maak je twee nieuwe register sleutels om SASL aan te geven waar de gegevens gevonden kunnen worden. Maak een sleutel aan met de naam [HKEY_LOCAL_MACHINE\SOFTWARE\Carnegie Mellon\Project Cyrus\SASL Library] en creëer daar twee nieuwe waarden in: SearchPath wijzend naar de map met de sasl*.dll plug-ins (meestal in de Subversion map) en ConfFile wijzend naar de map met svn.conf. Deze sleutels zijn al aangemaakt als je de CollabNet installatie gebruikt hebt.

Bewerk svn.conf zodat het volgende er in staat:

```
pwcheck_method: auxprop
auxprop_plugin: sasldb
mech_list: DIGEST-MD5
sasldb_path: C:\TortoiseSVN\sasldb
```

De laatste regel toont de locatie van de authenticatie database, een bestand met de naam sasldb. Deze kan overal staan, maar een logische plaats is één map hoger dan het archief pad. Zorg er voor dat de svnservice service leesrechten heeft voor dit bestand.

Als svnservice al draaide, dan moet je deze opnieuw starten zodat het de bijgewerkte configuratie inleest.

Nu alles ingesteld is, moet je alleen nog gebruikers en wachtwoorden aanmaken. Hiervoor heb je saslpaswd2 nodig. Als je de CollabNet installatie hebt gebruikt, staat dat programma in de installatie map. Gebruik het commando ongeveer zoals volgt:

```
saslpaswd2 -c -f C:\TortoiseSVN\sasldb -u realm username
```

De `-f` optie bepaalt de database locatie, `realm` moet gelijk zijn aan de waarde van het `svnserve.conf` bestand en `username` is de gebruikersnaam. Merk op dat de `realm` geen spaties mag bevatten.

Met `sasldblistusers2` kun je een lijst maken van de gebruikersnamen opgeslagen in de database.

3.6.5.3. SASL Encryptie

Met twee waardes in de `svnserve.conf` van het archief kun je verschillende encryptie niveaus instellen:

```
[sasl]
use-sasl = true
min-encryption = 128
max-encryption = 256
```

De `min-encryption` en `max-encryption` variabelen stellen het encryptieniveau in dat nodig is voor de server. Zet beide waardes op 0 om de encryptie helemaal uit te schakelen. Stel beide waardes in op 1 om gebruik te maken van cheksums (ter voorkoming van geknoei met data en het garanderen van de integriteit zonder encryptie te gebruiken). Als je encryptie toe wilt staan, maar niet wilt verplichten, stel dan de minimumwaarde in op 0 en de maximumwaarde op een bitlengte. Voor het onvoorwaardelijk eisen van encryptie stel je beide waardes in op een getal groter dan 1. In ons vorige voorbeeld, verlangen we het gebruik van 128-bit encryptie, maar niet meer dan 256-bit.

3.6.6. Authenticatie met svn+ssh

Een andere manier om gebruikers te authenticeren met een `svnserve` server, is door gebruik te maken van een Secure Shell (SSH). Dit is niet zo eenvoudig in te stellen als SASL, maar kan nuttig zijn in bepaalde gevallen.

Bij deze methode wordt `svnserve` niet als een daemon proces gebruikt, want de secure Shell start `svnserve` voor je, zodat het draait als een SSH geauthenticeerde gebruiker. Om dit te kunnen gebruiken, heb je een `secure shell daemon` nodig op je server.

Een basismethode voor het installeren van je server wordt beschreven in [Bijlage G, Svnserve beveiligen met SSH](#). Andere informatie over SSH vind je in de FAQ als je zoekt op “SSH”.

Meer detailinformatie over `svnserve` kun je vinden in [Version Control with Subversion](http://svnbook.red-bean.com) [http://svnbook.red-bean.com].

3.6.7. Pad-gebaseerde Autorisatie met svnserve

Sinds Subversion 1.3, ondersteunt `svnserve` ook het `mod_authz_svn` padgebaseerde autorisatie model dat beschikbaar is op een Apache server. Hiervoor moet je in het `conf/svnserve.conf` bestand in je archief map een regel toevoegen die refereert naar je autorisatiebestand.

```
[general]
authz-db = authz
```

Hierbij is `authz` een bestand dat je aan moet maken om de toegangsrechten in te stellen. Je kunt voor elk archief een apart bestand gebruiken of één bestand voor meerdere archieven. Lees [Paragraaf 3.7.6, “Pad-gebaseerde autorisatie”](#) voor een beschrijving van het bestandsformaat.

3.7. Apache Based Server

3.7.1. Introductie

De op Apache gebaseerde Subversion installatie is de meest flexibele installatie. Ondanks het feit dat deze installatie iets ingewikkelder is, levert het voordelen op die andere servers niet hebben.

WebDAV

De op Apache gebaseerde Subversion server gebruikt het WebDAV protocol, wat door veel andere programma's ook ondersteund wordt. Het is bijvoorbeeld mogelijk om een archief te koppelen als een "Web map" in de Windows verkenner en het te gebruiken als een normale map in je bestandssysteem.

Het Archief Verkennen

Je kunt met je browser naar de URL van je archief gaan en door de inhoud bladeren, zonder dat je daarvoor een Subversion client nodig hebt. Hierdoor kunnen veel meer gebruikers bij je gegevens komen.

Authenticatie

Je kunt elke authenticatiemethode gebruiken die Apache ondersteund, inclusief SSPI en LDAP.

Beveiliging

Aangezien Apache zeer stabiel en veilig is, krijg je vanzelf dezelfde beveiliging voor je archief. Hierbij inbegrepen zit SSL encryptie.

3.7.2. Installatie van Apache

Het eerste wat je nodig hebt voor het installeren van Apache is een computer met Windows 2000, Windows XP+SP1, Windows 2003, Vista of Server 2008.



Waarschuwing

Merk op dat Windows XP zonder service pack 1 valse netwerk data kan genereren en daarmee het archief kan beschadigen!

1. Haal de laatste versie van de Apache web server op via <http://httpd.apache.org/download.cgi>. Zorg ervoor dat je versie 2.2.x ophaalt – versie 1.3.xx zal niet werken!

De msi installatie van Apache wordt gevonden door de klikken op `other files`, blader dan naar `binaries/win32`. Waarschijnlijk is het het beste om te kiezen voor de msi installatie `apache-2.2.x-win32-x86-openssl-0.9.x.msi` (deze bevat OpenSSL).

2. Als je het Apache2 installatieprogramma op je computer hebt staan, kun je er dubbel op klikken. Het programma zal je dan door het installatieproces leiden. Zorg er voor dat je de server-URL goed ingeeft (als je geen DNS naam voor je server hebt, gebruik dan het IP-adres). Het is aanbevolen om Apache te installeren *voor alle gebruikers, op poort 80 en als een Service*. Let op: de installatie kan fout gaan als je IIS of een ander programma hebt dat luistert op poort 80. Als dat gebeurt, ga dan naar de programs map, `\Apache Group\Apache2\conf` en zoek het bestand `httpd.conf`. Bewerk dat zo dat `Listen 80` gewijzigd wordt naar een vrije poort, bijvoorbeeld `Listen 81`. Herstart de installatie – nu zal deze zonder fouten moeten kunnen lopen.
3. Test nu of de Apache web server goed draait door met je web browser naar `http://localhost/` te surfen – een vooraf ingesteld website zal dan weergegeven worden.



Let op

Als je er voor kiest om Apache als service te installeren, hou er dan rekening mee dat deze dan standaard onder het lokaal systeem account zal draaien. Het is verstandiger en veiliger om voor Apache een apart account aan te maken.

Zorg er voor dat het Account, waaronder Apache draait, een expliciete vermeld is in de toegangscontrolelijst van van de map met het archief (rechter muisknop | eigenschappen)

| beveiliging) en alle rechten daarin heeft. Gebruikers zijn anders niet in staat om hun wijzigingen vast te leggen.

Zelfs als Apache onder het lokaal systeem account draait, heb je deze vermelding nodig (wat in dit geval dan het SYSTEEM account is).

Als Apache niet deze rechten heft, zullen je gebruikers “Toegang geweigerd” foutmeldingen krijgen, welke in de Apache foutlog als error 500 zullen staan.

3.7.3. Installatie van Subversion

1. Download de laatste versie van de Subversion Win32 binaries voor Apache. Let er op dat je de juiste versie hebt voor integratie met jouw versie van Apache, anders krijg je een vreemde foutmelding bij het herstarten. Als je Apache 2.2.x hebt, ga dan naar <http://subversion.tigris.org/servlets/ProjectDocumentList?folderID=8100>.
2. Start de Subversion installatie en volg de instructies. Als de Subversion installatie detecteert dat je Apache hebt geïnstalleerd, dan ben je al bijna klaar. Als de installatie geen Apache server heeft gevonden, dan moet je enkele extra acties uitvoeren.
- 3.

Gebruik de windows verkenner en ga naar de installatie map van Subversion (meestal c:\program files\Subversion) en zoek de bestanden /httpd/mod_dav_svn.so en mod_authz_svn.so. Kopieer deze bestanden naar de Apache modules map (meestal c:\program files\apache group\apache2\modules).

4. Kopieer de bestanden /bin/libdb*.dll en /bin/intl3_svn.dll vanuit de installatie map van Subversion naar de Apache bin map.
5. Bewerk het configuratie bestand van Apache (meestal C:\Program Files\Apache Group\Apache2\conf\httpd.conf) met een text bewerkingsprogramma zoals Notepad en voer de volgende wijzigingen door:

Verwijder het commentaar teken (het '#' teken) bij de volgende regels:

```
#LoadModule dav_fs_module modules/mod_dav_fs.so
#LoadModule dav_module modules/mod_dav.so
```

Voeg de volgende twee regels toe aan het eind van de LoadModule sectie.

```
LoadModule dav_svn_module modules/mod_dav_svn.so
LoadModule authz_svn_module modules/mod_authz_svn.so
```

3.7.4. Configuratie

Nu is alles voor Apache en Subversion ingesteld, maar Apache weet nog niet hoe er omgegaan moet worden moet Subversion programma's zoals TortoiseSVN. Om Apache te laten weten welke URL gebruikt wordt voor Subversion archieven, moet je het Apache configuratiebestand bewerken (meestal te vinden in c:\program files\apache group\apache2\conf\httpd.conf) met een tekst bewerkingsprogramma (bijv. Notepad):

1. Voeg de volgende regels toe aan het eind van het configuratiebestand:

```
<Location /svn>
  DAV svn
  SVNListParentPath on
  SVNParentPath D:\SVN
```

```
#SVNIndexXSLT "/svnindex.xsl"
AuthType Basic
AuthName "Subversion repositories"
AuthUserFile passwd
#AuthzSVNAccessFile svnaccessfile
Require valid-user
</Location>
```

Hiermee stel je Apache zo in dat alle Subversion archieven zich bevinden in D:\SVN. De archieven worden naar de buitenwereld bedoeld vanaf URL `http://MyServer/svn/`. Toegang wordt alleen verleend aan gebruikers die opgenomen zijn in het passwd bestand.

2. Voor het aanmaken van een passwd bestand open je een (DOS) opdrachtprompt, ga naar de apache2 map (meestal `c:\program files\apache group\apache2`) en maak het bestand aan met

```
bin\htpasswd -c passwd <username>
```

. Hiermee wordt het bestand passwd aangemaakt, die gebruikt wordt voor authenticatie. Gebruikers kunnen toegevoegd worden met

```
bin\htpasswd passwd <username>
```

3. Herstart de Apache service nogmaals.
4. Ga met je browser naar `http://MyServer/svn/MyNewRepository` (waarin MyNewRepository de naam is van het Subversion archief dat je eerder hebt aangemaakt). Als alles goed is gegaan, word je gevraagd om je gebruikersnaam en wachtwoord in te voeren, waarna je de inhoud van je archief kunt bekijken.

Een korte uitleg van wat je zojuist hebt ingevoerd:

Instelling	Uitleg
<Location /svn>	geeft aan dat Subversion archieven beschikbaar zijn via URL <code>http://MyServer/svn/</code>
DAV svn	geeft aan welke module verantwoordelijk is voor het afhandelen van die URL – in dit geval de Subversion module.
SVNListParentPath on	Voor Subversion versie 1.3 en hoger, kan met deze optie een lijst van alle beschikbare archieven onder SVNParentPath weergegeven worden.
SVNParentPath D:\SVN	geeft aan Subversion door dat er ook naar archieven onder D:\SVN gezocht moet worden.
SVNIndexXSLT "/svnindex.xsl"	Wordt gebruikt om het bladeren met een web browser mooier te maken.
AuthType Basic	wordt gebruikt om basis authenticatie, dus Username/password, aan te schakelen.
AuthName "Subversion repositories"	wordt gebruikt om aan de gebruiker aan te geven waarvoor authenticatie nodig is als er een authenticatie dialoog venster getoond wordt.
AuthUserFile passwd	specificeert welk bestand met wachtwoorden gebruikt moet worden voor authenticatie. specificeert welk bestand met wachtwoorden gebruikt moet worden voor authenticatie.

Instelling	Uitleg
AuthzSVNAccessFile	specificeert de lokatie van het toegangsbestand voor paden binnen een Subversion archief.
Require valid-user	specificeert dat alleen gebruikers die een geldig gebruikersnaam en wachtwoord hebben ingevoerd, de URL kunnen laden.

Tabel 3.1. Apache httpd.conf Instellingen

Dat zomaar een voorbeeld. Er zijn veel en veel meer dingen die je met de Apache web server kunt doen.

- Als je wilt dat je archief voor iedereen te lezen is, terwijl schrijfrechten alleen toegekend worden aan specifieke gebruikers, dan kun je de volgende regel wijzigen:

```
Require valid-user
```

```
to
```

```
<LimitExcept GET PROPFIND OPTIONS REPORT>
Require valid-user
</LimitExcept>
```

- Met een passwd bestand beperk en geef je toegang tot alle archieven in één geheel. Als je controle wilt hebben over wie toegang heeft tot elke map in een archief, verwijder dan de commentaartekens bij de regel

```
#AuthzSVNAccessFile svnaccessfile
```

en maak een Subversion toegangsbestand aan. Apache zorgt er voor dat alleen gerechtigde gebruikers toegang hebben tot de /svn locatie en geeft dan direct ook de gebruikersnaam door aan Subversion's AuthzSVNAccessFile module, waardoor het de toegangsregels kan afdwingen zoals opgeslagen in het Subversion toegangsbestand. Merk op dat paden gespecificeerd worden als `repos:path` of `path`. Als je geen bepaald archief opgeeft, zal de regel toegepast worden op alle archieven onder `SVNParentPath`. Het formaat van het autorisatie-policy bestand die gebruikt wordt door `mod_authz_svn` is beschreven in [Paragraaf 3.7.6, "Pad-gebaseerde autorisatie"](#).

- Om het bladeren door een archief met een browser 'mooier' te maken, verwijder je de commentaartekens bij de regel

```
#SVNIndexXSLT "/svnindex.xsl"
```

en sla de bestanden `svnindex.xsl`, `svnindex.css` en `menucheckout.ico` in de hoofdmap (meestal `C:/Program Files/Apache Group/Apache2/htdocs`). De map wordt ingesteld met de `DocumentRoot` optie in je Apache configuratie bestand.

Deze drie bestanden kun je direct uit ons archief halen via <http://tortoisesvn.googlecode.com/svn/trunk/contrib/svnindex>. ([Paragraaf 3, "TortoiseSVN is gratis!"](#) geeft een beschrijving hoe je het TortoiseSVN bronbestanden archief moet benaderen).

Het XSL bestand van het TortoiseSVN archief heeft een gimmick: als je met je web browser door je archief bladert, dan heeft elke map in je archief een icoontje aan de rechter kant. Als je op dat icoontje klikt, wordt het TortoiseSVN Ophalen venster gestart met die URL.

3.7.5. Meerdere Archieven

Als je de `SVNParentPath` optie hebt gebruikt, hoef je niet het Apache configuratiebestand te veranderen als je een nieuw Subversion archief hebt aangemaakt. Maak eenvoudigweg het nieuwe archief aan op de zelfde locatie als de eerste en je bent klaar! In mijn bedrijf heb ik direct toegang tot die specifieke map op de server via SMB (via normale Windows bestandstoegang). Ik kan hier dus gewoon een nieuwe map aanmaken, het TortoiseSVN commando `TortoiseSVN → Maak hier een archief` en het nieuwe project heeft een thuis...

Als je Subversion 1.3 of later gebruikt, kun je de `SVNListParentPath` on optie gebruiken om Apache toe te staan een lijst te maken van alle beschikbare projecten als je met je browser naar het pad van de hoofdmap gaat in plaats van een specifiek archief.

3.7.6. Pad-gebaseerde autorisatie

De `mod_authz_svn` module maakt het mogelijk om nauwkeurig afgestemde toegangscontrole gebaseerd op gebruikers namen en archiefpaden te organiseren. Deze methode is beschikbaar in Apache en, vanaf Subversion 1.3, ook in `svnserve`.

Een voorbeeld bestand kan er zo uit zien:

```
[groups]
admin = john, kate
devteam1 = john, rachel, sally
devteam2 = kate, peter, mark
docs = bob, jane, mike
training = zak
# Standaard toegangsregel voor allearchieven
# Iedereen can lezen, admins kunnen schrijven, Dan German is uitgesloten.
[/]
* = r
@admin = rw
dangerman =
# Geef ontwikkelaars complete toegang tot hun projectarchieven
[proj1:/]
@devteam1 = rw
[proj2:/]
@devteam2 = rw
[bigproj:/]
@devteam1 = rw
@devteam2 = rw
trevor = rw
# Geef de documentatie mensen schrijfrechten op alle document mappen
[/trunk/doc]
@docs = rw
# Geeft stagiaires alleen schrijfrechten in trainingarchieven
[TrainingRepos:/]
@training = rw
```

Merk op dat het controleren van elk pad tijdrovend is, vooral bij een revisie log. De server controleert elk gewijzigd pad in elke revisie and controleert of er gelezen kan worden, hetgeen erg lang kan duren voor revisies die veel bestanden betreffen.

Authenticatie en autorisatie zijn losstaande processen. Als een gebruiker toegang wil tot een archief, dan moet *zowel* aan de authenticatie voorwaarden *als ook* aan de autorisatie voorwaarden voldaan worden.

3.7.7. Authenticatie door een Windows Domein

Zoals je mogelijk al opgemerkt hebt, moet je voor elke gebruiker een aparte gebruikersnaam/wachtwoord combinatie opnemen in het `passwd` bestand. Als je (vanwege veiligheidsredenen) wilt dat gebruikers periodiek hun wachtwoord wijzigen, dan moet je dat handmatig doen.

Maar er is een oplossing voor dat probleem – tenminste als je het archief benaderd vanbinnen een LAN met een windows domein controller: `mod_auth_sspi`!

De originele SSPI module is aangeleverd door Syneapps, inclusief broncode. Maar de ontwikkeling daarvan is gestopt. Maar wanhoop niet, de community heeft het opgepakt en verbeterd. Het heeft een nieuw thuis op [SourceForge](http://sourceforge.net/projects/mod-auth-sspi/) [http://sourceforge.net/projects/mod-auth-sspi/].

- Haal de module die bij jouw apache versie hoort op en kopieer het bestand `mod_auth_sspi.so` naar de Apache module map.
- Bewerk het Apache configuratie bestand: voeg de regel

```
LoadModule sspi_auth_module modules/mod_auth_sspi.so
```

toen aan de `LoadModule` sectie. Let op dat je deze regel toevoegt *voor* de regel

```
LoadModule auth_module modules/mod_auth.so
```

- Om ervoor te zorgen dat op deze Subversion locatie deze methode van authenticatie gebruikt, moet je de regel

```
AuthType Basic
```

wijzigen in

```
AuthType SSPI
```

en moet je ook

```
SSPIAuth On
SSPIAuthoritative On
SSPIDomain <domaincontroller>
SSPIOmitDomain on
SSPIUsernameCase lower
SSPIPerRequestAuth on
SSPIOfferBasic On
```

toevoegen binnen het `<Location /svn>` blok. Als je geen domein controller hebt, laat dan de naam van de domein controle staan op `<domaincontroller>`.

Merk op dat als je SSPI gebruikt voor authenticatie, dat je dan de regel `AuthUserFile` niet meer nodig hebt om een wachtwoord bestand te definiëren. Apache authenticceert je gebruikersnaam en wachtwoord dan via je windows domein. Je moet dan je gebruikerslijst in je `svnaccessfile` naar `DOMAIN \username` laten verwijzen.



Belangrijk

De SSPI authenticatie is alleen mogelijk voor met SSL beveiligde verbindingen (https). Als je alleen normale http verbindingen gebruikt naar je server, dan zal het niet werken.

Om SSL op je server aan te zetten, lees hoofdstuk: [Paragraaf 3.7.9, "De server beveiligen met SSL"](#)



Tip

Subversion `AuthzSVNAccessFile` bestanden zijn hoofdlettergevoelig in relatie tot gebruikersnamen (`JUser` is anders dan `juser`).

In Microsoft's wereld, Windows domeinen en gebruikersnamen zijn niet hoofdlettergevoelig. Sommige netwerkbeheerders vinden het leuk om voor gebruikersnamen CamelCase te gebruiken (bijv. `JUser`).

Door dit verschil kun je gebeten worden als je SSPI authenticatie gebruikt onder windows en de gebruikersnamen op dezelfde manier als de gebruiker deze heeft ingevoerd worden doorgegeven aan Subversion, omdat `JUser` en `juser` voor Subversion verschillende gebruikers zijn. Internet Explorer geeft vaak de gebruikersnaam automatisch door aan Apache onafhankelijk van de manier waarop het geschreven is.

Het eindresultaat is dat je voor elke gebruiker soms op zijn minst twee vermeldingen nodig hebt in je `AuthzSVNAccessFile` voor elke gebruiker -- één met kleine letters en één met de zelfde indeling zoals Internet Explorer deze doorgeeft aan Apache. Je zult je gebruikers ook moeten aanleren om voor hun inloggegevens altijd kleine letters te gebruiken als zij de archieven benaderen via TortoiseSVN.

De fout and toegang logbestanden van Apache zijn je beste vriend bij het ontrafelen van dit soort problemen, omdat deze je helpen de gebruikersnaam te bepalen die doorgegeven zijn aan Subversion's `AuthzSVNAccessFile` module. Om alles werkend te krijgen zul je soms wat moeten experimenteren met het exacte formaat van de gebruikersregel in de `svnaccessfile` (bijv. `DOMAIN\user` vs. `DOMAIN//user`).

3.7.8. Meerdere authenticatie sources

Het is mogelijk om meer dan één authenticatiebron te gebruiken voor je Subversion archief. Om dit te gebruiken, moet je elke authenticatie type instellen als niet-gezaghebbend (non-authoritative), zodat Apache bij meerdere bronnen de gebruikersnaam/wachtwoordcombinatie zal controleren.

Een gebruikelijke oplossing is om de Windows domein authenticatie en een `passwd` bestand te gebruiken, zodat je gebruikers zonder een Windows domein account toch toegang kunt geven tot het SVN archief.

- Om Windows domein en `passwd` bestand authenticatie te gebruiken, moet je de volgende opties opnemen binnen het `<Location>` blok van je Apache configuratiebestand:

```
AuthBasicAuthoritative Off
SSPIAuthoritative Off
```

Dit is een voorbeeld van de complete Apache configuratie met de gecombineerd Windows domein en `passwd` bestand authenticatie:

```
<Location /svn>
  DAV svn
  SVNListParentPath on
  SVNParentPath D:\SVN

  AuthName "Subversion repositories"
```

```
AuthzSVNAccessFile svnaccessfile.txt

# NT Domain Logins.
AuthType SSPI
SSPIAuth On
SSPIAuthoritative Off
SSPIDomain <domaincontroller>
SSPIOfferBasic On

# Htpasswd Logins.
AuthType Basic
AuthBasicAuthoritative Off
AuthUserFile passwd

Require valid-user
</Location>
```

3.7.9. De server beveiligen met SSL

Apache 2.2.x ondersteunt OpenSSL, maar dit is niet standaard ingeschakeld. Je moet dit handmatig aanzetten.

1. Verwijder in het apache configuratiebestand de commentaar tekens op de regels:

```
#LoadModule ssl_module modules/mod_ssl.so

en onderaan

#include conf/extra/httpd-ssl.conf

verander dan de regel (op een regel)

SSLMutex "file:C:/Program Files/Apache Software Foundation/\
Apache2.2/logs/ssl_mutex"

in

SSLMutex default
```

2. Vervolgens moet je een SSL certificaat aanmaken. Open hiervoor een opdrachtprompt (DOS-scherm) en ga naar de Apache map (bijv. C:\program files\apache group\apache2) en geef het volgende commando:

```
bin\openssl req -config conf\openssl.cnf -new -out my-server.csr
```

Je zult gevraagd worden om een beveiligingszin op te geven. Gebruik hiervoor niet alleen korte eenvoudige woorden, maar gebruik een hele zin (bijv. een deel van een gedicht). Hoe langer de zin is, hoe beter. Je moet ook de URL van je server opgeven. Alle andere vragen zijn optioneel, maar we bevelen aan om die ook in te vullen.

Normaalgesproken zal het `privkey.pem` automatisch gegenereerd worden, als dit niet gedaan wordt, moet je dit commando geven om het alsnog te genereren:

```
bin\openssl genrsa -out conf\privkey.pem 2048
```

Voer vervolgens de volgende commando's in

```
bin\openssl rsa -in conf\privkey.pem -out conf\server.key
```

en (op één regel)

```
bin\openssl req -new -key conf\server.key -out conf\server.csr \  
-config conf\openssl.cnf
```

en dan (op één regel)

```
bin\openssl x509 -in conf\server.csr -out conf\server.crt  
-req -signkey conf\server.key -days 4000
```

Hiermee wordt een certificaat aangemaakt die over 4000 dagen verloopt. Geef als laatste commando (op één regel):

```
bin\openssl x509 -in conf\server.cert -out conf\server.der.crt  
-outform DER
```

Deze commando's genereren enkele bestanden in de Apache conf map (server.der.crt, server.csr, server.key, .rnd, privkey.pem, server.cert).

3. Herstart de Apache service.

4. Ga met je browser naar `https://servername/svn/project ...`



SSL en Internet Explorer

Als je je server met SSL beveiligd en je gebruikt authenticatie met een Windows domein, dan zul je merken dat het verkennen van je archief met Internet Explorer niet meer werkt. Geen zorgen – dit komt alleen doordat Internet Explorer niet kan authenticeren. Andere browsers hebben dit probleem niet en TortoiseSVN en andere Subversion applicaties kunnen nog steeds authenticeren.

Als je IE nog steeds wilt kunnen gebruiken om het archief te verkennen, kun je:

- een aparte `<Location /path>` richtlijn in het Apache configuratiebestand definiëren en de `SSPIBasicPreferred` optie toevoegen. Dit zorgt er voor dat IE weer kan authenticeren, maar andere browsers en Subversion kunnen niet op deze locatie authenticeren.
- Verkennen met unencrypted authenticatie (zonder SSL). Vreemd genoeg heeft IE geen problemen met authenticeren als een verbinding niet beveiligd is met SSL.
- Bij de “standaard” SSL installatie staat vaak het volgende statement in Apache's virtuele SSL host:

```
SetEnvIf User-Agent ".*MSIE.*" \  
nokeepalive ssl-unclean-shutdown \  
downgrade-1.0 force-response-1.0
```

Er zijn (waren?) goede redenen voor deze configuratie, lees http://www.modssl.org/docs/2.8/ssl_faq.html#ToC49. Als je NTLM authenticatie wilt gebruiken, dan moet je

keepalive gebruiken. Als de commentaartekens weghaalt bij `SetEnvIf` dan moet het mogelijk zijn om IE met Windows authenticatie over SSL te laten authenticeren met Apache op Win32 inclusief `mod_auth_sspi`.



SSL toegang forceren

Als je SSL ingesteld hebt om je archief veiliger te maken, dan wil je waarschijnlijk de normale toegang zonder SSL (http) uitschakelen en dus alleen https toegang toestaan. Om dit te doen moet je nog een extra richtlijn toevoegen aan het Subversion `<Location>` blok: `SSLRequireSSL`.

Een voorbeeld `<Location>` blok zal er dan zo uit zien:

```
<Location /svn>
  DAV svn
  SVNParentPath D:\SVN
  SSLRequireSSL
  AuthType Basic
  AuthName "Subversion repositories"
  AuthUserFile passwd
  #AuthzSVNAccessFile svnaccessfile
  Require valid-user
</Location>
```

3.7.10. Werken met virtuele SSL hosts en client certificaten

Door Nigel Green naar de TortoiseSVN mailinglijst gestuurd. Bedankt!

Bij bepaalde server configuraties moet je een enkele server opzetten met 2 virtuele SSL hosts: De eerste voor publieke toegang zonder eisen voor een client certificaat. De tweede als beveiligde, die een client certificaat vereist, waarop de Subversion server draait.

Door een `SSLVerifyClient Optional` richtlijn aan de *per-server* sectie van de Apache configuratie toe te voegen (d.w.z. buiten de `VirtualHost` en `Directory` blokken), moet Apache een client certificaat opvragen voor de initiële SSL handdruk. Door een fout in de `mod_ssl` is het van essentieel belang dat dit certificaat op dit moment opgevraagd wordt, omdat het niet werkt als de SSL verbinding opnieuw uitonderhandeld wordt.

Dit kan opgelost worden door de volgende richtlijn toe te voegen aan de Virtuele host map die je wilt afsluiten voor Subversion:

```
SSLRequire %{SSL_CLIENT_VERIFY} eq "SUCCESS"
```

Deze richtlijn geeft alleen toegang tot de map als een client certificaat ontvangen en succesvol geverifieerd is.

Samenvattend zijn dit de relevante regels in de Apache configuratie:

```
SSLVerifyClient Optional

### Virtual host configuration for the PUBLIC host
### (not requiring a certificate)

<VirtualHost 127.0.0.1:443>
  <Directory "pathtopublicfileroot">
    </Directory>
```

```
</VirtualHost>

### Virtual host configuration for SUBVERSION
### (requiring a client certificate)
<VirtualHost 127.0.0.1:443>
  <Directory "subversion host root path">
    SSLRequire %{SSL_CLIENT_VERIFY} eq "SUCCESS"
  </Directory>

  <Location /svn>
    DAV svn
    SVNParentPath /pathtorepository
  </Location>
</VirtualHost>
```

Hoofdstuk 4. Dagelijkse gebruikersgids

Dit document beschrijft het dagelijkse gebruik van de TortoiseSVN client. Het is *geen* introductie tot versiebeheersystemen en *geen* introductie tot Subversion (SVN). Het is meer een document wat je opent als je ongeveer weet wat je wilt doen, maar niet meer weet hoe je het moet doen.

Als je een introductie tot versiebeheer met Subversion zoekt, dan bevelen we het fantastische boek *Version Control with Subversion* [<http://svnbook.red-bean.com/>] aan.

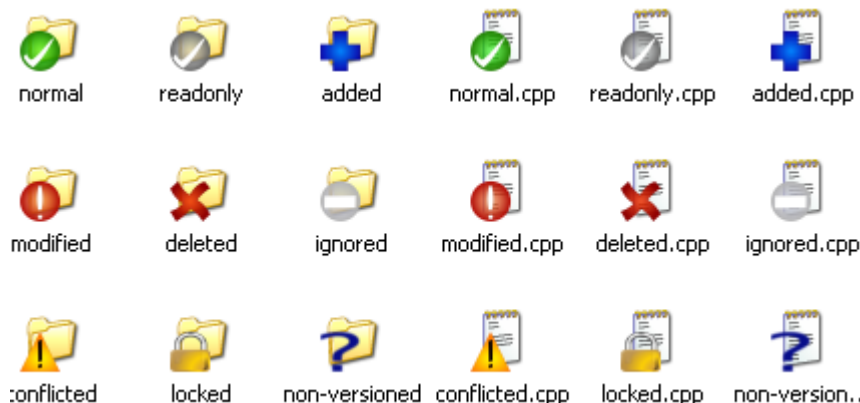
Dit document wordt continue verbeterd, net zoals TortoiseSVN en Subversion. Als je een fout vindt, meld deze dan aan via de maillijst, zodat we de documentatie kunnen bijwerken. Enkele screenshots in dit document geven mogelijk niet de actuele status van de software weer. Vergeef ons alsjeblieft. We werken aan TortoiseSVN in onze vrije tijd.

Om het meeste effect te bereiken met de Daily User Guide, moet je:

- TortoiseSVN al geïnstalleerd hebben.
- bekend zijn met versiebeheersystemen.
- de basis van Subversion kennen.
- een server geïnstalleerd hebben en/of toegang hebben tot een Subversion archief.

4.1. Aan de slag

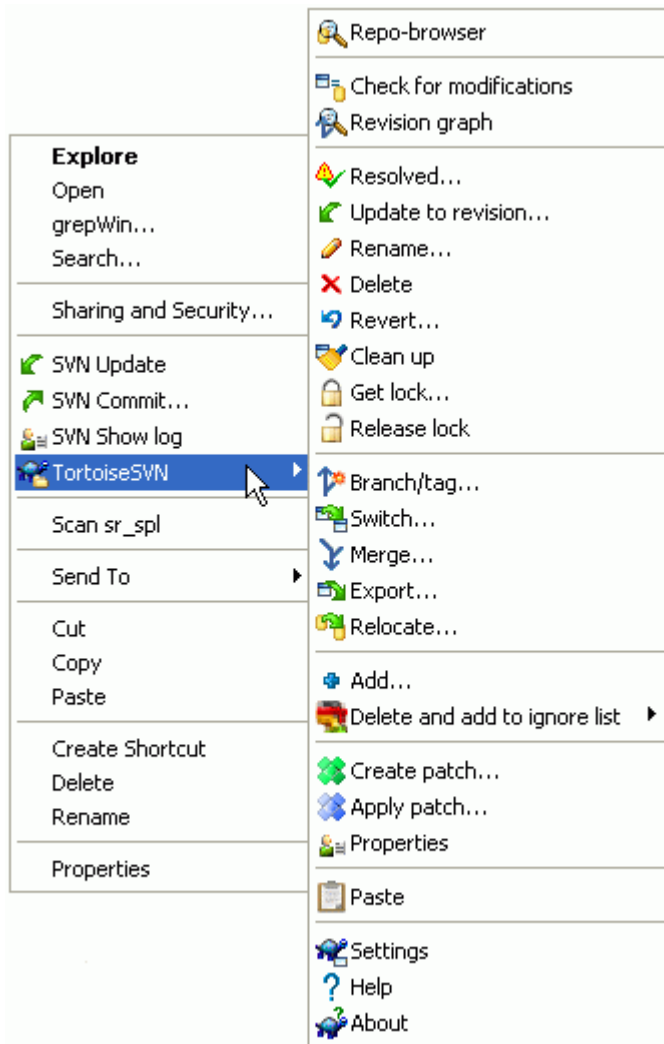
4.1.1. Overlappende Pictogrammen



Figuur 4.1. Overlappende pictogrammen weergegeven in de verkennen

Eén van de meest zichtbare functies van TortoiseSVN zijn de overlappende pictogrammen (ook wel icon overlays genoemd), welke weergegeven worden op bestanden in werkkopie. Deze laten in één oogopslag zien welke bestanden gewijzigd zijn. Lees [Paragraaf 4.7.1, “Overlappende Pictogrammen”](#) om te leren wat de verschillende overlappende pictogrammen betekenen.

4.1.2. Contextmenu's



Figuur 4.2. Contextmenu van een directory reeds onder versiebeheer

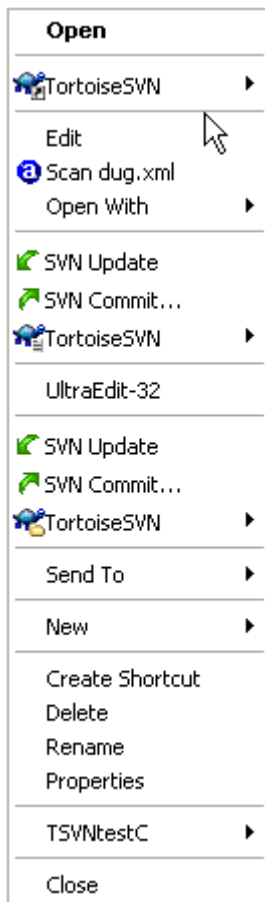
Alle TortoiseSVN commando's worden vanuit het context menu van een verkenner gegeven. De meeste zijn direct zichtbaar als je met rechts klikt op een bestand of map. Welke commando's beschikbaar zijn, hangt af van het feit of het bestand, de map of de bovenliggende map beheerd wordt met versiebeheer. Het TortoiseSVN menu is ook te zien als onderdeel van het Verkenner bestandsmenu.



Tip

Commando's die minder vaak worden gebruikt, zijn alleen beschikbaar in het uitgebreide context menu. Om het uitgebreidere menu aan te roepen, hou dan de **Shift** toets ingedrukt als je met rechts klikt.

In enkele gevallen kun je meerdere TortoiseSVN instanties zien. Dit is geen bug!



Figuur 4.3. Verkenner bestandsmenu van een snelkoppeling in een versiebeheerde map

Dit is een voorbeeld van een snelkoppeling zonder versiebeheer in een map met versiebeheer, waarbij in het verkenner bestandsmenu *drie* instanties van TortoiseSVN aanwezig zijn. Een is voor de map, een voor de snelkoppeling zelf en een derde voor het object waar de snelkoppeling naar wijst. Om deze instanties te onderscheiden, hebben de iconen een rechtsonder indicator waarmee aangegeven wordt of het een menu is voor een bestand, een map, een snelkoppeling of voor meerdere geselecteerde items.

Als je Windows 200 gebruikt zul je zien dat de context menu's in platte tekst worden weergegeven, zonder menu iconen erboven. We zijn ons bewust dat dit in eerdere versies wel werkte, maar Microsoft heeft de manier waarop de iconen gebruikt kunnen worden veranderd voor Vista. Wat tot gevolg had dat wij een andere manier van weergeven moesten gaan gebruiken, welke helaas niet werkt onder Windows 2000.

4.1.3. Slepen en Plaatsen

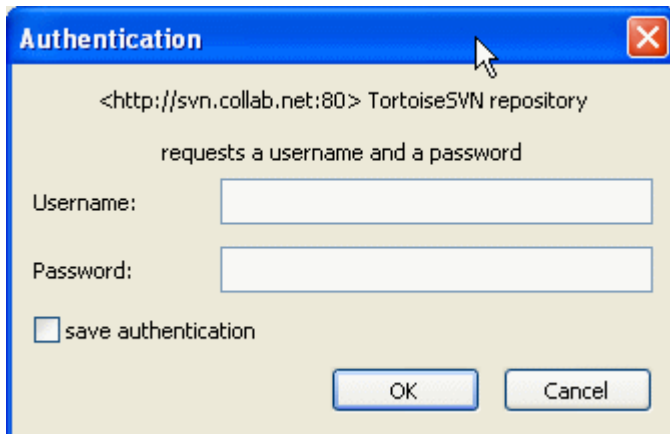
Er zijn andere commando's beschikbaar bij Slepen en plaatsen. Als je bestanden of mappen met rechts sleept naar een nieuwe locatie binnen je werkkopie of als je een bestand of map zonder versiebeheer naar een map met versiebeheer sleept.

4.1.4. Algemene snelkoppelingen

Enkele veel gebruikte acties hebben de welke bekende Windows sneltoets combinaties, hoewel deze niet zijn aangegeven bij de knoppen of in menu's. Als je niet weet hoe je de voor de hand liggende functies moet gebruiken, bijvoorbeeld verversen, lees dan deze paragraaf.

4.1.5. Authenticatie

Als je een archief met wachtwoordbeveiliging probeert te bereiken, dan krijg je een authenticatie scherm te zien.



Figuur 4.4. Authenticatie dialoog

Voer je gebruikersnaam en wachtwoord in. Het selectievakje zorgt ervoor dat TortoiseSVN je logingegevens opslaat in Subversion's standaard map: %APPDATA%\Subversion\auth in drie submappen.

- `svn.username` bevat alleen gegevens voor basis authenticatie (gebruikersnamen, wachtwoord).
- `svn.ssl.server` bevat SSL server certificaten.
- `svn.username` bevat alleen gebruikersnamen, voor authenticatie zonder wachtwoord.

Als je de authenticatie cache voor alle servers wilt wissen, ga dan naar de Opgeslagen Gegevens sectie van het Instellingen venster van TortoiseSVN. Met de "Wissen" knop bij de optie "Authenticatie gegevens" kun je alle opgeslagen gegevens uit de `auth` mappen wissen, alsook de gegevens opgeslagen in het register door oudere TortoiseSVN versies. Lees ook [Paragraaf 4.30.6, "Bewaarde Gegevens Instellingen"](#).

Sommige mensen hebben graag dat de authenticatie gegevens worden verwijderd als ze uitloggen uit Windows of de pc uitschakelen. Dat kan gedaan worden met een script dat gestart wordt bij het afsluiten. Dit script kan de welke %APPDATA%\Subversion\auth map wissen, bijvoorbeeld

```
@echo off
rmdir /s /q "%APPDATA%\Subversion\auth"
```

Je kunt een beschrijving vinden over hoe je zo'n script moet maken op [windows-help-central.com](http://www.windows-help-central.com/windows-shutdown-script.html) [http://www.windows-help-central.com/windows-shutdown-script.html].

Lees [Paragraaf 3.5, "Toegang hebben tot het Archief"](#) voor meer informatie over hoe je authenticatie en toegangsbeveiliging installeert op je server.

4.1.6. Maximaliseren van Windows

In verschillende vensters van TortoiseSVN wordt er veel informatie weergegeven. Het is vaak alleen maar nodig om verticaal of horizontaal de volledige scherm breedte te gebruiken, in plaats van te maximaliseren zodat het hele scherm gevuld is. Voor het gemak zijn er snelkoppelingen toegevoegd aan de Maximaliseren knop. Gebruik de middelste muisknop om verticaal te maximaliseren en de rechter muisknop om horizontaal te maximaliseren.

4.2. Data Importeren In Een Archief

4.2.1. Importeren

Als je data importeert in een bestaand archief waarin al enkele projecten aanwezig zijn, dan is de structuur van het archief al vastgesteld. Als je gegevens importeert in een nieuw archief, dan is het de moeite waard om even goed na te denken over de structuur van het archief. Lees [Paragraaf 3.1.5, “Archief indeling”](#) voor advies hierover.

Deze sectie beschrijft het Subversion Importeren commando, welke ontwikkeld is voor het importeren van een complete mapstructuur in een archief in één keer. De functie heeft echter wel wat tekortkomingen:

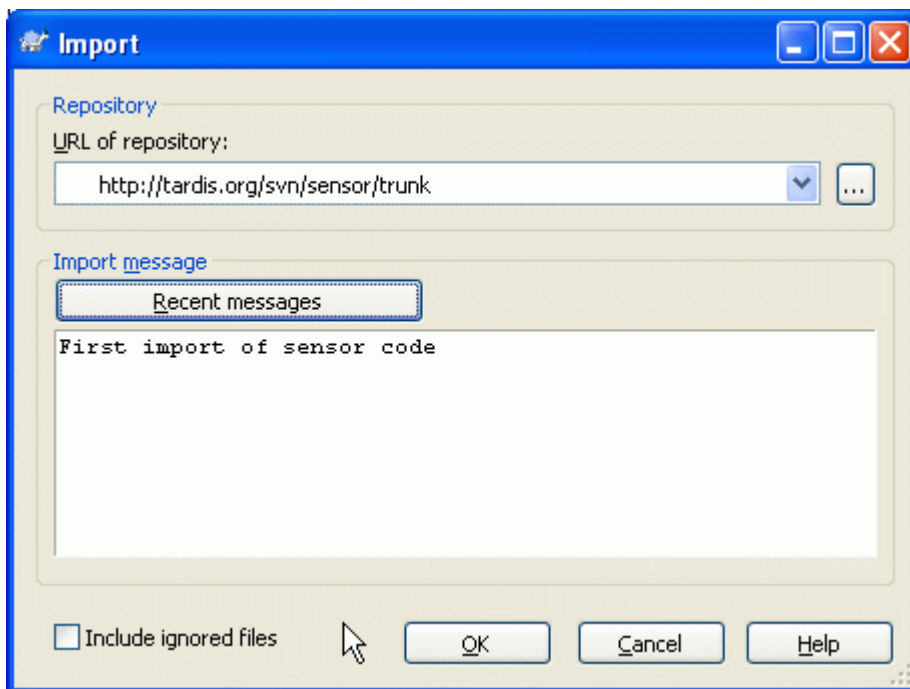
- Het is niet mogelijk om aan te geven welke bestanden en mappen geïmporteerd moeten worden, behalve dan door het gebruik van globale negeer instellingen.
- De geïmporteerde map wordt geen werkkopie. Je moet het Ophalen commando gebruiken om weer een kopie van de bestanden van de server te halen.
- Het importeren vanaf een verkeerd mapniveau gaat erg makkelijk.

Vanwege deze redenen adviseren wij het importeer commando niet te gebruiken. In plaats daarvan is het beter om een twee-staps methode te gebruiken die beschreven is in [Paragraaf 4.2.2, “Invoeren op zijn Plaats”](#). Maar omdat je dit toch leest, volgt hier een korte beschrijving hoe de importeer functie werkt...

Voordat je je project in het archief importeert, moet je:

1. Alle bestanden verwijderen die niet nodig zijn om het project te bouwen (tijdelijke bestanden, bestanden die door een compiler gemaakt worden bijv. *.obj, gecompileerde binaire bestanden,...)
2. De bestanden in mappen en sub-mappen sorteren. Ondanks dat het mogelijk is om bestanden te hernoemen of verplaatsen, is het verstandig om je projectstructuur in één keer goed te hebben voordat je importeert.

Selecteer nu in de Windows verkennen de hoofdmap van je project mapstructuur en klik met rechts om het context menu te openen. Selecteer TortoiseSVN → Importeren... waardoor dit venster wordt geopend:



Figuur 4.5. Het invoer venster

In dit venster moet je de URL van het archief en de locatie in het archief opgeven waar je je project wilt importeren. Het is belangrijk je te realiseren dat je de lokale map zelf niet in het archief zal verschijnen, alleen maar de inhoud daarvan. Bijvoorbeeld als je een structuur hebt als:

```
C:\Projects\Widget\source
C:\Projects\Widget\doc
C:\Projects\Widget\images
```

en je importeert C:\Projects\Widget in `http://mydomain.com/svn/trunk`, dan zul je misschien verrast zijn dat je de submappen direct in de trunk vindt in plaats van in de Widget submap. Je moet de submap ook in de URL opnemen. `http://mydomain.com/svn/trunk/Widget-X`. Merk op dat met het importeren commando automatisch de benodigde mappen in het archief aanmaakt als deze niet bestaan.

Het import bericht wordt gebruikt als bericht in het logboek.

Bestanden en mappen die het overeen komen met de globale-negeren patronen worden *niet* geïmporteerd. Om dit gedrag te omzeilen kun je de optie **Genegeerde bestanden meenemen** selecteren. Lees [Paragraaf 4.30.1, "Algemene instellingen"](#) voor meer informatie voor het instellen van een globaal negeren patroon.

Zodra je op OK klikt zal TortoiseSVN de complete mapstructuur inclusief alle bestanden in het archief importeren. Het archief is nu in het archief met versiebeheer opgeslagen. Merk op dat de map die je geïmporteerd hebt *NIET* nog geen versiebeheer heeft. Om een *werkkopie* met versiebeheer te krijgen moet je eerst nog het project nog Ophalen uit het archief. Of lees verder om te weten te komen hoe je een map direct in versiebeheer opneemt.

4.2.2. Invoeren op zijn Plaats

Als je al een archief hebt en je daar een nieuwe mapstructuur aan wilt toevoegen, volg dan deze stappen:

1. Gebruik de archief-verkenner om een nieuwe project map direct in het archief te maken.
2. Haal de nieuwe map op en sla deze op dezelfde locatie als de map die je wilt importeren. Het lijkt dus er op dat je de lokale map overschijft. Geen zorgen. Je zult een waarschuwing krijgen dat de lokale map niet leeg is. Dat is niet erg. Je hebt uiteindelijk dan een hoofdmap met versiebeheer met inhoud zonder versiebeheer.
3. Gebruik de optie TortoiseSVN → Toevoegen... op deze map met versiebeheer om bepaalde of alle inhoud toe te voegen. Je kunt bestanden toevoegen en verwijderen, `svn:ignore` eigenschappen instellen op mappen en andere wijzigingen die je nodig hebt.
4. Leg de hoofmap vast (SVN Vastleggen) en je hebt een nieuwe mapstructuur met versiebeheer en een lokale werkkopie gemaakt van je bestaande map.

4.2.3. Speciale Bestanden

Soms wil je een bestand van versiebeheer voorzien, waarin gebruiker specifieke data staat. Dat houdt dat je een bestand hebt die elke ontwikkelaar/gebruiker aan zijn/haar lokale instellingen wil aanpassen. Het is lastig om zo'n bestand van versiebeheer te voorzien, want elke gebruiker zou zijn/haar wijzigingen telkens opslaan in het archief.

In zulke situaties bevelen we het gebruik van sjablonen aan. Maak een bestand aan met alle gegevens die je gebruikers nodig hebben. Sla het bestand op in het archief en laat de ontwikkelaars het bestand ophalen. Vervolgens moet elke ontwikkelaar een *kopie maken* en deze hernoemen. Daarna levert het wijzigen van het bestand geen problemen meer op.

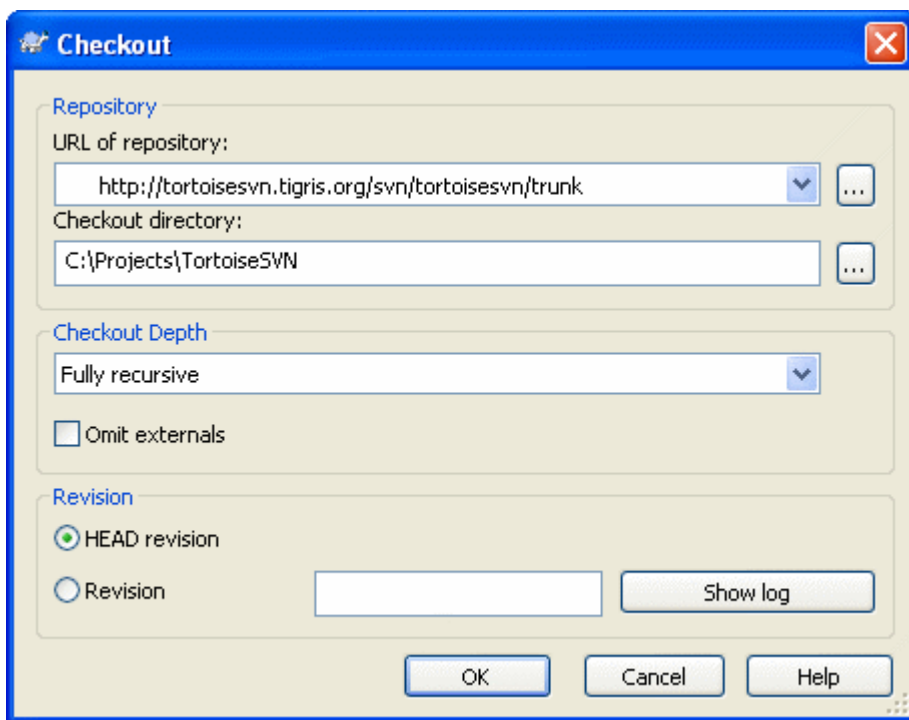
Het TortoiseSVN bouwscript kun je als voorbeeld bekijken. Deze roept het bestand `TortoiseVars.bat` aan, welke niet in het archief bestaat. Het bestand `TortoiseVars.tpl` is wel aanwezig. `TortoiseVars.tpl` is de sjabloon waarvan elke ontwikkelaar een kopie moet maken en deze moet hernoemen naar `TortoiseVars.bat`. In dat bestand is commentaar toegevoegd, zodat elke ontwikkelaar weet welke regels hij/zij in zijn/haar gewenste lokale instellingen moet wijzigen.

Om de gebruikers niet te verstoren, hebben ook het bestand `TortoiseVars.bat` op de negerlijst van de bovenliggende map gezet, oftewel we hebben de bestandsnaam in de Subversion eigenschap `svn:ignore` opgenomen. Hierdoor wordt hij niet telkens als bestand zonder versie beheer aangegeven bij een vastlegging.

4.3. Een lokale werkkopie (WC) ophalen

Om een werkkopie te verkrijgen, moet je de bestanden *ophalen* uit het archief.

Selecteer in windows verkenner een map waarin je je werkkopie wilt hebben. Klik rechts om het context menu te openen en selecteer TortoiseSVN → SVN Ophalen..., waarna het volgende venster wordt geopend:



Figuur 4.6. Het Ophalen venster

Als je een mapnaam invoert die nog niet bestaat, dan zal deze automatisch worden aangemaakt.

4.3.1. Ophaaldiepte

Je kunt de *diepte* van de recursiviteit instellen van de mappen en gegevens die je wilt ophalen. Als je maar een bepaald deel wilt hebben van een grote mapstructuur, dan kun je alleen de hoofdmap ophalen en vervolgens de mappen selecteren die je recursief wilt ophalen.

Volledig recursief

Gehele boom uitchecken, inclusief alle knopen en subfolders

Enkel niveau, inclusief folders

Haal de gespecificeerd map op, inclusief alle bestanden en onderliggende mappen, maar niet de inhoud van de onderliggende mappen.

Alleen bestanden

Haal de gespecificeerde map op, inclusief alle bestanden, maar haal niet de mappen op.

Alleen dit object

Haal alleen de map op. Haal de bestanden en onderliggende mappen niet op.

Werkkopie

Behoudt de mapdiepte zoals in de werkkopie gespecificeerd is. Deze optie is niet gebruikt in het Ophalen venster, maar wordt standaard gebruikt bij vensters die mapdiepte gebruiken.

Exclusief

Wordt gebruikt om de werkkopie mapdiepte te beperken, nadat een map al met de inhoud is gevuld. Deze optie is alleen beschikbaar in het Verversen naar revisie venster.

Als je een uitgedunde werkkopie ophaalt door iets anders te kiezen dan volledig recursief voor de ophaal diepte, dan kun je additionele submappen selecteren met de archief-verkenner (browser (Paragraaf 4.24, “De Archief-verkenner”) of met het “kijk of er updates zijn” venster (Paragraaf 4.7.3, “Lokale en Server Status”).

Klik met rechts in de windows verkenner op de opgehaalde map en selecteer TortoiseSVN → Archief-verkenner om de archief-verkenner te openen. Ga naar de submap die je aan je werkkopie wilt toevoegen en kies Context menu → Verversen naar Revisie.... Deze optie wordt alleen zichtbaar als het geselecteerde object nog niet in je werkkopie bestaat, maar de bovenliggende map wel.

In het kijk of er updates zijn venster, klik op de knop Controleer archief. Het venster zal alle bestanden en mappen laten zien die in het archief zijn, maar die je niet opgehaald hebt als Op afstand toegevoegd. Klik rechts op de map(pen) die je aan je werkkopie wilt toevoegen en kies dan Context menu → Verversen.

Deze functie is handig als je maar een klein deel wilt ophalen van een grote mapstructuur, maar ook het gemak wilt hebben dat je maar één werkkopie hoeft bij te werken. Stel je heb een grote mapstructuur met de submappen Project01 tot Project99, waarvan je alleen Project03, Project25 en Project76/SubProj wilt ophalen. Doorloop dan de volgende stappen:

1. Haal de hoofdmap op met diepte “Alleen dit object”. Je krijgt dan alleen een lege hoofdmap.
2. Selecteer de nieuwe map en kies TortoiseSVN → Archief-verkenner om de inhoud van het archief te bekijken.
3. Klik met rechts op Project03 en Context menu → Verversen naar revisie.... Laat de standaard instellingen zo staan en klik op OK. Die map is nu volledig gevuld.

Herhaal dezelfde stappen voor Project25.

4. Ga naar literal>Project76/SubProj



Veranderen van de werkkopie mapdiepte

Als je een werkkopie met een bepaalde mapdiepte hebt opgehaald, dan kun je de mapdiepte later veranderen met behulp van Context menu → Verversen naar revisie....



Een oudere server gebruiken

Servers met oudere versies dan 1.5 begrijpen de werkkopie mapdiepte aanvragen niet. Deze kunnen dus niet altijd efficiënt met zulke aanvragen omgaan. Het commando werkt wel,

maar een oudere server verzend waarschijnlijk alle data. De gebruiker moet dan eruit filteren wat hij/zij er niet bij wil hebben. Dit kan betekenen dat er meer netwerkverkeer optreedt dan gewenst. Als het mogelijk is, zou je de server moeten bijwerken naar versie 1.5.

Als het project referenties heeft naar externe projecten die je *niet* direct wilt ophalen, gebruik dan het selectievak Externen uitsluiten.



Belangrijk

Als Externen uitslaten aangevinkt is of je wilt de mapdiepte vergroten, dan moet je hiervoor gebruikmaken van TortoiseSVN → Verversen naar revisie... in plaats van TortoiseSVN → Verversen. De normale Verversen functie zal alle externe verbindingen verversen en de bestaande mapdiepte blijven gebruiken.

Het is aanbevolen om alleen de basislijn (trunk) of dieper gelegen mappen van je mapstructuur op te halen. Als je de hoofdmap van het hele archief in je URL opneemt, dan zal je harddisk waarschijnlijk vollopen. Je krijgt dan namelijk een complete kopie van het hele archief, inclusief alle takken en labels van je project.



Exporteren

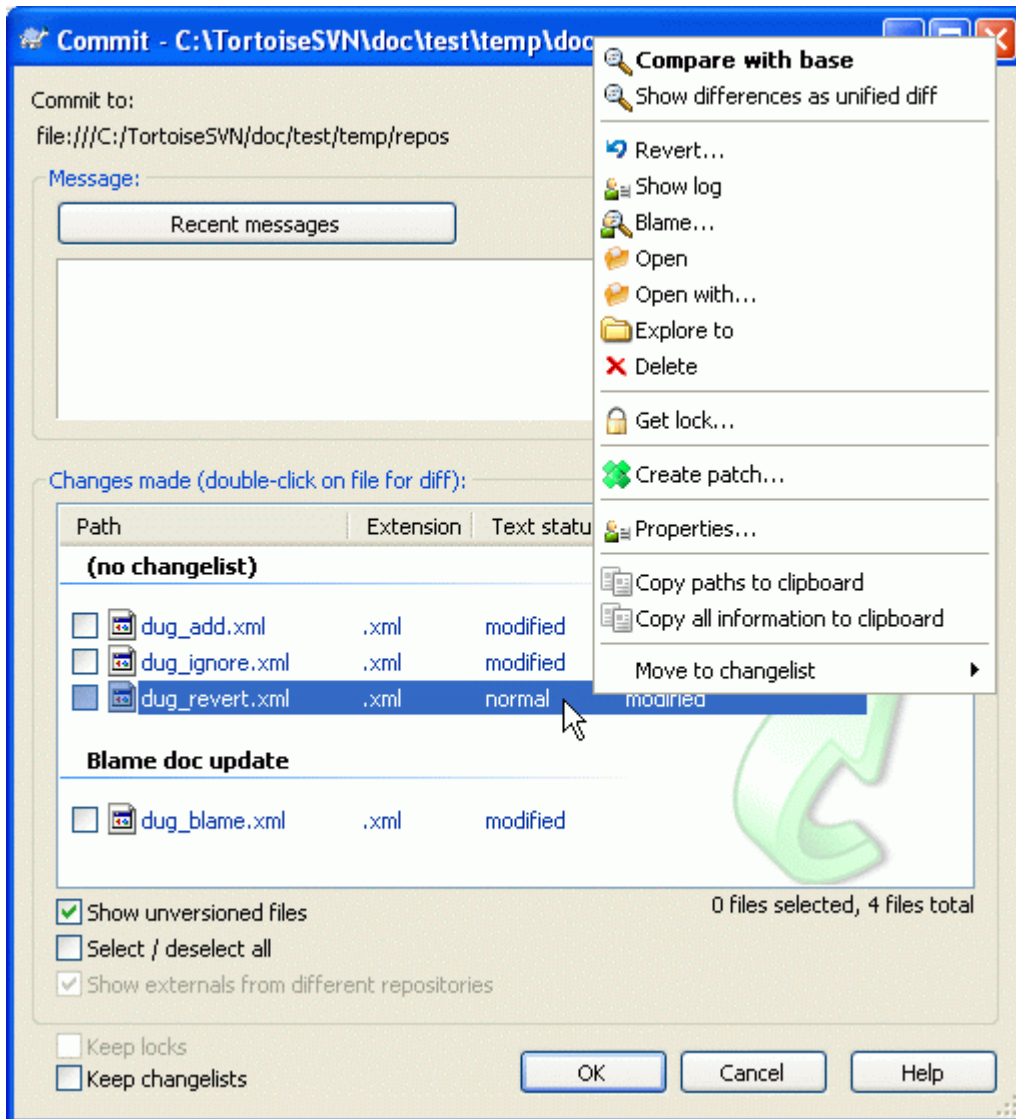
Soms wil je een lokale kopie maken zonder de .svn mappen, bijvoorbeeld om een zip-bestand te maken van je bronbestanden. Lees [Paragraaf 4.26, "Exporteren van een Subversion Werkkopie"](#) om te weten te komen hoe dit moet.

4.4. Je Wijzigingen Vastleggen In Het Archief

Het versturen van de wijzigingen die jij hebt gedaan, staat bekend als *vastleggen* van de wijzigingen. Voordat je je wijzigingen vastlegt, moet je zekerstellen dat je werkkopie bijgewerkt is naar de laatste stand van zaken. Dit kun je doen door TortoiseSVN → Verversen te kiezen. Je kunt ook eerst TortoiseSVN → Kijk of er updates zijn gebruiken, om te bekijken welke bestanden er lokaal en op de server gewijzigd zijn.

4.4.1. De Commit Dialoog

Als je werkkopie helemaal bij is en er zijn geen conflicten, dan ben je klaar om je wijzigingen vast te leggen. Selecteer de bestanden en mappen die je wilt vastleggen en kies TortoiseSVN → Vastleggen....



Figuur 4.7. De Commit dialoog

De vastleggen dialoog geeft elk gewijzigd bestand weer, inclusief toegevoegde, verwijderde en bestanden zonder versiebeheer. Deselecteer een gewijzigd bestand, als je deze niet wilt vastleggen. Als je een bestand zonder versiebeheer in het archief wilt opnemen, dan kun je deze direct aanvinken zodat deze ook vastgelegd wordt.

Objecten die verplaatst zijn naar een andere archieflocatie worden aangeduid met een (s) teken. Het kan zijn dat je omgeschakeld bent terwijl je werkte aan een tak en vergeten was om terug te schakelen naar de basislijn. Dit is je waarschuwing!



Bestanden of mappen vastleggen?

Als je bestanden vastlegt, dan zal de vastleggen dialoog alleen de bestanden laten zien die je geselecteerd hebt. Als je een map vastlegt, dan zal de vastleggen dialoog de gewijzigde bestanden automatisch selecteren. Als je een nieuw aangemaakt bestand vergeten bent, dan wordt deze automatisch gevonden als je een map vastlegt. Het vastleggen van een map betekent *niet* dat elk bestand als gewijzigd wordt aangeduid; het wordt alleen maar eenvoudiger voor doordat het je werk uit handen neemt.

Als je bestanden hebt gewijzigd die vanuit een ander archief komen door het gebruik van `svn:externals`, dan kunnen die wijzigingen niet als één enkele vastlegging worden vastgelegd. Een waarschuwingsteken onderaan de lijst geeft aan of dit gebeurd is. De tooltip geeft aan dat die externe bestanden apart vastgelegd moeten worden.



Veel bestanden zonder versiebeheer in de vastleggen dialoog.

Als je vindt dat het vastleggen venster te veel bestanden zonder versiebeheer weergeeft (bijv. reservekopie bestanden van de compiler of tekstverwerker), dan zijn daar verschillende oplossingen voor. Je kunt:

- het bestand (of een wildcard extensie) toevoegen aan de lijst met de te negeren bestanden op de instellingen pagina. Dit zal op elke werkkopie die je hebt worden toegepast.
- het bestand toevoegen aan de literal `>svn:ignore` met TortoiseSVN Verwijder en voeg toe aan negerlijst. Dit zal alleen de mappen raken waarvoor de `svn:ignore` eigenschap is ingesteld. Met het SVN Eigenschappen venster kun je de `svn:ignore` eigenschap van een map aanpassen.

Dubbelklikken op een gewijzigd bestand in het vastleggen venster zal het externe vergelijkingsprogramma opstarten om jouw wijzigingen te laten zien. Het context menu zal je meer opties geven, zoals weergegeven in de screenshot. Je kunt ook bestanden hier vandaan verslepen naar andere programma's zoals een tekstverwerker of IDE.

Je kunt objecten selecteren en de-selecteren door op het selectievak links ervan te klikken. Bij mappen kun je **Shift**-selecteren gebruiken om de actie recursief te maken.

De kolommen in het onderste venster kunnen aangepast worden. Als je met rechts klikt op de koptekst van een kolom, dan wordt er een context menu weergegeven waarmee je kunt selecteren welke kolommen er getoond moeten worden. Je kunt ook de breedte van de kolommen aanpassen door de overgang tussen twee kolommen in het koptekst van de kolommen te verslepen. De aanpassingen worden opgeslagen, waardoor je de volgende keer dezelfde weergave weer terugkrijgt.

Als je een vastlegging opstart, wordt standaard alle blokkades die jij gezet hebt op bestanden verwijderd nadat de vastlegging succesvol is afgerond. Als je deze blokkades wilt behouden, zorg er dan voor dat het **Blokkades behouden** selectievak is aangekruist. De standaard status van deze optie wordt gelezen uit de `no_unlock` optie in het Subversion configuratiebestand. Lees [Paragraaf 4.30.1, "Algemene instellingen"](#) voor informatie over hoe je het configuratiebestand kunt aanpassen.



Slepen en Plaatsen

Je kunt bestanden van een andere locatie naar het vastleggen venster slepen, als de werkkopieën maar in het zelfde archief zitten. Bijvoorbeeld, je hebt een enorm grote werkkopie met meerdere windows verkenners open om verschillende mappen in de hiërarchie te bekijken. Als je niet vanaf je hoofdmap wilt vastleggen (met een langdurende zoektocht naar alle wijzigingen), dan kun je het vastleggen venster openen op één map en bestanden vanaf een andere locatie in het venster slepen, zodat het in één actie meegenomen wordt bij het vastleggen.

Je kunt bestanden zonder versiebeheer in de werkkopie ook naar het vastleggen venster slepen, dan worden deze automatisch in SVN opgeslagen.



Externe Hernoemingen Repareren

Soms worden bestanden hernoemt buiten Subversion, waardoor deze bestanden in de lijst weergegeven worden als missend of zonder versiebeheer. Om te voorkomen dat je de

geschiedenis verliest, moet je Subversion vertellen dat er een verbinding is. Selecteer de oude (missende) naam en de nieuwe nieuwe en gebruik Context Menu → Repareer Verplaatsing om de bestanden aan elkaar te koppelen als hernoemden.

4.4.2. Werksets

Het vastleggen venster ondersteunt de werkset functie van Subversion om te helpen bij het groeperen van gerelateerde bestanden. Lees meer over deze functie in [Paragraaf 4.8, “Werksets”](#).

4.4.3. Objecten van de vastleggingslijst uitsluiten

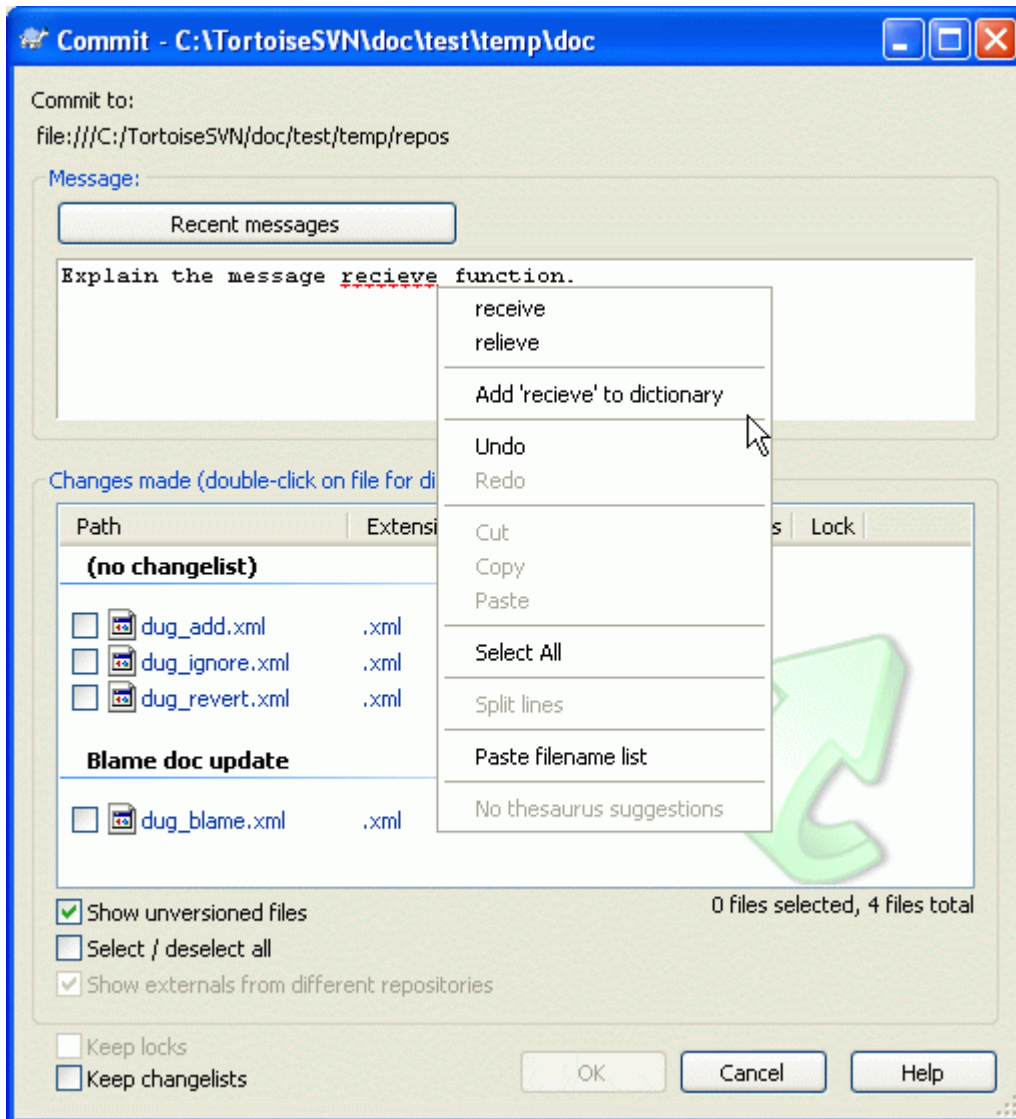
Soms heb je bestanden met versiebeheer die regelmatig wijzigen, maar die je eigenlijk niet wilt vastleggen. Dit kan er op duiden dat je een zwak punt heb in je bouwproces – waarom zitten die bestanden in versiebeheer? Zou je sjablonen moeten gebruiken? Soms is het gewoon onoverkomelijk. Een klassieke reden is dat je IDE de tijdstempel wijzigt elke keer als je het bouwproces hebt doorlopen. Het projectbestand moet wel in subversion komen, omdat het je instellingen bevat, maar het is niet nodig om deze elke keer vast te leggen als de alleen de tijdstempel is gewijzigd.

Om je uit dit soort vervelende situaties te helpen, hebben we een werkset gemaakt met de naam `ignore-on-commit` (negeren-bij-vastleggen). Elk bestand dat op deze lijst staat, zal automatisch gedeselecteerd worden in het vastleggen venster. Je kunt wijzigingen nog steeds wel vastleggen, maar je moet de bestanden dan handmatig selecteren in het vastleggen venster.

4.4.4. Vastleg Log Meldingen

Zorg er voor dat je een log bericht invult, welke de wijzigingen beschrijft die je doorgevoerd hebt. Dit helpt om te bepalen wat er wanneer gewijzigd is als je op een later moment door de log berichten bladert. Het bericht kan zo kort of lang zijn als je wilt; bij veel project is gedefinieerd wat je er in moet zetten, welke taal je moet gebruiken en soms zelfs een strakke formattering.

Je kunt je tekst eenvoudig opmaken door gebruik te maken van stijlregels die lijken op de regels van emails. Om stijlen toe te passen op tekst, gebruikt `*tekst*` voor vet, `_tekst_` voor onderstrepen en `^tekst^` voor cursief.



Figuur 4.8. De Vastleggen Dialoog Spellingscontrole

TortoiseSVN bevat spellingscontrole om je te helpen berichten te maken zonder typefouten. Verkeerd gespelde woorden worden gemarkeerd. Gebruik het context menu om de suggesties te bekijken. Natuurlijk is niet *technische* technische term die je kent bekend, waardoor correct gespelde woorden soms worden aangeduid als verkeerd gespeld. Maak je geen zorgen. Je kunt deze toevoegen aan je persoonlijke woordenboek via het context menu.

Het log venster heeft ook een automatisch aanvullingsfunctie voor bestandsnamen en functies. Hiervoor worden gebruikelijke expressies en functienamen uit en de namen van de (tekst)bestanden, die je wilt vastleggen, gelezen. Als een woord dat je aan het typen bent overeen komt met een woord uit de lijst (als je tenminste 3 karakters hebt getypt of **Ctrl+Spatie** hebt ingedrukt), verschijnt er een lijst met complete namen waaruit je kunt selecteren. De gebruikelijke expressies die meegeleverd worden met TortoiseSVN worden opgeslagen in de TortoiseSVN installatie bin map. Je kunt ook je eigen gebruikelijke expressies definiëren en deze opslaan in %APPDATA%\TortoiseSVN\autolist.txt. Uiteraard wordt je eigen lijst niet overschrijven als je je TortoiseSVN installatie update. Als je niet bekend bent met reguliere expressies, bekijk dan eens http://en.wikipedia.org/wiki/Regular_expression en de online documentatie en tutorial: <http://www.regular-expressions.info/>.

Je kunt eerder gebruikte log berichten hergebruiken. Klik hiervoor op **Recente boodschappen**. Je kunt dan de laatste paar berichten die je gebruikt hebt bij deze werkkopie zien. Het aantal opgeslagen berichten kan aangepast worden in het TortoiseSVN instellingen venster.

Je kunt alle opgeslagen vastleggingsberichten wissen bij de **Opgeslagen gegevens** pagina van TortoiseSVN's instellingen. Je kunt ook individuele berichten wissen vanuit het **Recente boodschappen** venster door op de **Delete** knop te drukken.

Met het commando **Context Menu** → **Plak lijst bestandsnamen** kun je de gemarkeerde paden in het bericht invoegen.

Een andere methode voor het invoegen van de paden is door de bestanden te slepen van de bestandslijst naar het bewerk venster.



Speciale Map Instellingen

Er zijn verschillende speciale mapeigenschappen, waarmee er meer controle gekregen wordt over de formattering van vastleggingsberichten en de taal van de spellingscontrole. Lees voor meer informatie [Paragraaf 4.17, "Project Instellingen"](#).

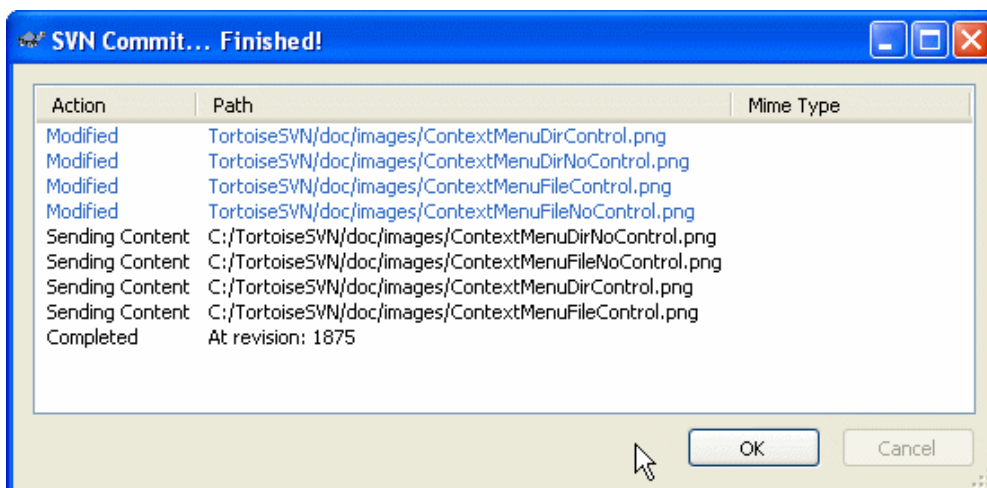


Integratie met Bug Tracking Programma's

Als het bug tracking systeem hebt geactiveerd, dan kun je één of meerdere Problemen in het Bug-ID / Issue-Nr: tekstveld invullen. Meerdere problemen moeten gescheiden worden door komma's. Als je gebruik maakt van gebruikelijke expressies bij Bug tracking, voeg dan je referenties toe als onderdeel van het log bericht. Voor meer informatie: [Paragraaf 4.28, "Integratie met Bug Tracking Systemen / Issue Trackers"](#).

4.4.5. Vastlegging voortgang

Na het drukken op de OK knop zal een venster verschijnen met de voortgang van de vastlegging.



Figuur 4.9. Het voortgangsvenster geeft de voortgang van een vastlegging weer

Het voortgangsvenster gebruikt kleurcodering om de verschillende vastleggingsacties weer te geven.

Blauw

Vastleggen van een wijziging.

Paars

Een nieuwe toevoeging committen.

Donker rood

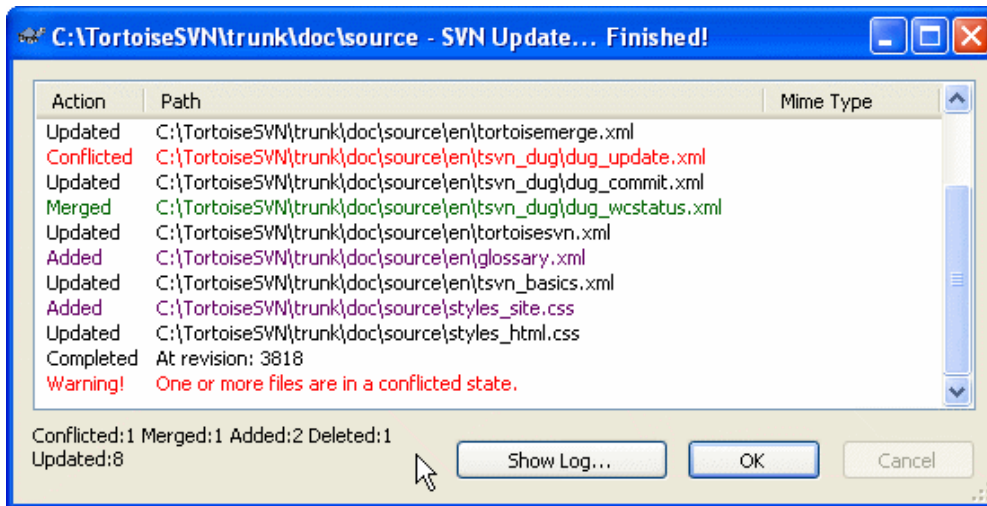
Vastleggen van een verwijdering of vervanging.

Zwart

Alle andere items.

Dit is het standaard kleurenschema, maar je kunt deze naar je eigen wensen aanpassen in het instellingen venster. Lees voor meer informatie [Paragraaf 4.30.1.4, "TortoiseSVN kleurinstellingen"](#).

4.5. Je Werkkopie Bijwerken Met De Wijzigingen Van Anderen



Figuur 4.10. Voortgangsvenster van een afgeronde verversing

Je moet er regelmatig voor zorgen dat wijzigingen die door anderen gemaakt zijn, in jouw werkkopie meegenomen worden. Het proces van het ophalen van de wijzigingen vanaf de server naar je lokale werkkopie staat bekend als *verversen*. Verversen kan op een enkel bestand, een aantal bestanden of recursief op een hele mapstructuur. Om te verversen, selecteer de bestanden en mappen. Klik rechts en selecteer TortoiseSVN → Verversen in het verkennen context menu. Er zal een venster getoond worden die de voortgang van de verversing weergeeft. Wijzigingen van anderen worden in jouw bestanden verwerkt, zonder dat jouw wijzigingen in (mogelijk) dezelfde bestanden daarbij verloren gaan. Het archief wordt niet door het verversen bijgewerkt of veranderd.

Het voortgangsvenster gebruikt kleurcodering om verschillende acties te markeren.

Paars

Nieuw object toegevoegd aan je werkkopie.

Donker rood

Overtollig object verwijdert uit je werkkopie, of missend object vervangen in je werkkopie.

Groen

Wijzigingen uit het archief succesvol samengevoegd met je lokale wijzigingen.

Fel rood

Het samenvoegen van wijzigingen uit het archief met de lokale wijzigingen heeft geresulteerd in conflicten die opgelost moeten worden.

Zwart

Ongewijzigd object in je werkkopie bijgewerkt met de nieuwe versie uit het archief.

Dit is het standaard kleurenschema, maar je kunt deze naar je eigen wensen aanpassen in het instellingen venster. Lees voor meer informatie [Paragraaf 4.30.1.4, "TortoiseSVN kleurinstellingen"](#).

Als er tijdens het samenvoegen *conflicten* zijn opgetreden (als bijvoorbeeld anderen dezelfde regel op een andere manier hebben gewijzigd als jijzelf hebt gedaan), dan zal het venster deze met rood weergeven. Je kunt dubbel klikken op deze regels om het externe samenvoegprogramma te starten.

Als het verversen afgerond is, geeft het voortgangsvenster onder de bestandslijst een samenvatting weer van het aantal vernieuwingen, toevoegingen, verwijderingen, conflicten, etc.. Deze samenvatting kan naar het klembord gekopieerd worden met **Ctrl+C**.

Het normale Verversen commando heeft geen opties en zorgt er alleen voor dat je werkkopie ververst wordt naar de HEAD revisie van het archief. Dat is wat je eigenlijk normaal gesproken altijd wilt doen. Als je meer controle wilt hebben over het verversen proces, dan moet je de functie TortoiseSVN → Verversen naar revisie... gebruiken. Hiermee kun je je werkkopie bijwerken naar een specifieke revisie, niet alleen de meest recente. Stel, je werkkopie bevat revisie 100 en je wilt de status van revisie 50 bekijken. Je kunt dan eenvoudig verversen naar revisie 50. In hetzelfde venster kun je ook de *mapdiepte* instellen van de huidige map. De termen die gebruikt worden, staan beschreven in [Paragraaf 4.3.1, "Ophaal diepte"](#). De standaard diepte is *Werkkopie*, waarmee de huidige mapdiepte behouden wordt. Je kunt ook aangeven of externe referenties genegeerd moeten worden bij de verversing (oftewel projecten die gekoppeld zijn met `svn:externals`).



Let op

Als je een bestand of map naar een bepaalde revisie ververst, dan moet je geen wijzigingen in die bestanden aanbrengen. Je krijgt dan een foutmelding als je je wijzigingen probeert vast te leggen dat je niet aan de laatste versie hebt gewerkt. Als je wijzigingen ongedaan wilt maken en opnieuw wilt starten vanaf een eerdere revisie, dan kun je dat doen via het revisie log venster. Lees [Paragraaf B.4, "Draai Revisies Terug in het Archief \(ongedaan maken\)"](#) voor meer instructies en alternatieve methodes.

Verversen naar Revisie kan soms handig om te bekijken hoe je project er een tijd geleden uitzag. Het verversen van individuele bestanden naar een eerdere revisie is zeker geen goed idee, omdat je werkkopie dan inconsistent wordt. Als de naam van een bestand gewijzigd is, dan zal dat bestand gewoon uit je werkkopie verdwijnen. Dit komt omdat er geen bestand met die naam in een eerdere revisie aanwezig was. Je moet je ook realiseren dat zo'n object (uit een eerdere revisie) een normaal overlappend pictogram heeft, waardoor het niet te onderscheiden is welke bestanden van de laatste revisie zijn.

Als je een lokale kopie wilt hebben van een oude versie van een bestand, dan kun je beter het Context Menu → Bewaar revisie in... commando gebruiken vanuit het log venster van het betreffende bestand.



Meerdere Bestanden/Mappen

Als je in de verkenner meerdere bestanden selecteert en vervolgens Verversen kiest, dan worden al die bestanden en mappen stuk voor stuk ververst. TortoiseSVN zorgt er voor dat bestanden en mappen uit hetzelfde archief bijgewerkt worden naar dezelfde revisie, zelfs als er gedurende het verversen een andere vastlegging heeft plaatsgevonden.



Lokale bestandsnaam bestaat al.

Het komt soms voor dat het verversen een foutmelding geeft dat er al een lokaal bestand bestaat met dezelfde naam. Dit gebeurt als Subversion een bestand, dat recent in versiebeheer is opgeslagen, probeert op te halen en er al een lokaal bestand in de werkkopie is zonder versiebeheer, maar met dezelfde naam. Subversion zal nooit een bestand zonder versiebeheer overschrijven – het bestand bevat misschien gegevens waarmee je aan het werk bent, terwijl het per toeval dezelfde bestandsnaam heeft als een nieuw bestand die net door een andere ontwikkelaar is vastgelegd.

Als je deze foutmelding krijgt, kun je dit probleem eenvoudig oplossen door het lokale bestand zonder versiebeheer te hernoemen. Nadat het verversen is afgerond, kun je kijken of het hernoemde bestand nog nodig is.

Als je foutmeldingen blijft krijgen, gebruik dan TortoiseSVN → Kijk of er updates zijn om een lijst te laten maken van alle bestanden met problemen. Op die manier kun je ze allemaal in één keer afhandelen.

4.6. Conflicten oplossen

Zo af en toe zul je tegen een *conflict* aanlopen als je bestanden in het archief ververst of samenvoegt, of als je je werkkopie omschakelt naar een andere URL. Er zijn twee soorten conflicten:

bestandsconflicten

Een bestandsconflict treedt op als er twee of meer ontwikkelaars dezelfde regel in één bestand hebben aangepast.

mapstructuur conflicten

Een mapstructuur conflict treedt op als een ontwikkelaar een bestand of map verplaatst, hernoemd of verwijderd heeft, welke ook door een andere ontwikkelaar is verplaatst, hernoemd, verwijderd of gewijzigd is.

4.6.1. Bestandsconflicten

Een bestandsconflict treedt op als twee of meer ontwikkelaars dezelfde regels in een bestand hebben gewijzigd. Omdat Subversion niets van de projecten weer, laat het het oplossen van conflicten over aan de gebruikers. Als er een conflict gemeld wordt, dan moet je het bestand in kwestie openen en de regels zoeken die beginnen met <<<<<<. Het conflicterende gedeelte wordt als volgt gemarkeerd:

```
<<<<<< filename
  your changes
=====
  code merged from repository
>>>>>> revision
```

Daarnaast plaatst Subversion drie additionele bestanden in je map:

filename.ext.mine

Dit is het bestand zoals het aanwezig was in je werkkopie, voordat je je werkkopie ging verversen. Oftewel de versie zonder conflictmarkeringen. Dit bestand bevat jouw laatste wijzigingen en niets anders.

filename.ext.rOLDREV

Dit de BASE revisie van het is bestand voordat je je werkkopie ging verversen. Oftewel het bestand die je opgehaald hebt voordat je jouw laatste wijzigingen er in ging zetten.

filename.ext.rNEWREV

Dit is het bestand die zojuist door je Subversion applicatie is ontvangen van de server toen je jouw werkkopie ging verversen. Dit bestand komt overeen met de HEAD revisie in het archief.

Je kunt of een extern samenvoeg/conflict bewerkingsprogramma openen met TortoiseSVN → **Bewerk Conflicten** of je kunt elke tekstverwerker gebruiken om handmatig de conflicten op te lossen. Je moet beslissen hoe de code er uit zou moeten zien, de noodzakelijke wijzigingen doorvoeren en het bestand opslaan.

Daarna moet je het commando TortoiseSVN → **Opgelost** kiezen en je wijzigingen vastleggen in het archief. Merk op dat het Opgelost commando het conflict zelf niet oplost. Het zorgt er alleen voor

dat de bestanden `filename.ext.mine` en `filename.ext.r*` worden verwijderd, zodat je je wijzigingen kunt vastleggen.

Subversion zal niet proberen om conflicten met binaire bestanden zelf op te lossen. Het lokale bestand blijft ongewijzigd (precies zoals je het als laatste had opgeslagen) en je krijgt `filename.ext.r*` bestanden ernaast. Als je je eigen wijzigingen wilt laten vervallen en de versie in het archief wilt behouden, gebruik dan het `Terugzetten` commando. Als je je eigen versie wilt gebruiken en de versie in het archief wilt overschrijven, gebruik dan het `Opgelost` commando en leg vervolgens je wijzigingen vast.

Het is mogelijk om het `Opgelost` commando op meerdere bestanden toe te passen. Klik met rechts op de bovenliggende map en selecteer `TortoiseSVN` → `Opgelost...` Er wordt dan een venster geopend met alle conflicterende bestanden in die map. Je kunt dan selecteren welke bestanden gemarkeerd moeten worden als `Opgelost`.

4.6.2. Mapstructuur conflicten

Een conflict in mapstructuur ontstaat als een ontwikkelaar bestanden of mappen verplaatst, hernoemd of verwijderd heeft, die een andere ontwikkelaar toevallig ook verplaatst, hernoemd, verwijderd of gewijzigd heeft. Er zijn nog veel meer situaties die tot een mapstructuur conflict kunnen leiden en elke situatie heeft weer een andere oplossing nodig.

Als een bestand lokaal uit Subversion verwijderd is, dan wordt deze ook van het lokale bestandssysteem verwijderd. Dat houdt in dat er geen overlappend pictogram weergegeven kan worden als dit bestand onderdeel is van een mapstructuurconflict en je er dus ook niet met rechts op kunt klikken om het probleem op te lossen. Gebruik dan het `Kijk of er updates zijn` venster om de `Conflicten oplossen` optie te kunnen benaderen.

TortoiseSVN kan helpen om de beste plaatst voor het samenvoegen van wijzigingen te vinden, maar voor het oplossen van conflicten is er toch meer handwerk nodig. Onthoudt dat na een verversing de BASE altijd de revisie van bestanden uit het archief bevat van het moment van verversen. Als je een wijziging terugzet nadat je verversst hebt, dan kom je terug op de stand van het archief, niet naar de stand die je toen je begon met het wijzigen.

4.6.2.1. Lokale verwijdering, inkomende wijziging bij verversing

1. Ontwikkelaar A wijzigt `Foo.c` en legt deze vast in het archief
2. Ontwikkelaar B heeft tegelijkertijd `Foo.c` naar `Bar.c` verplaatst in zijn werkkopie of heeft `Foo.c` of de bovenliggende map verwijderd.

Een verversing van de werkkopie van ontwikkelaar B resulteert in een mapstructuur conflict:

- `Foo.c` is verwijderd uit de werkkopie, maar deze is gemarkeerd als een mapstructuur conflict.
- Als het conflict veroorzaakt is door een hernoeming in plaats van verwijdering, dan zal `Bar.c` gemarkeerd worden als toegevoegd, maar deze zal dan niet de wijzigingen van ontwikkelaar A bevatten.

Ontwikkelaar B moet nu kiezen of hij de wijzigingen van ontwikkelaar A wil behouden. Als het om een naamswijziging gaat, dan kan hij de wijzigingen van `Foo.c` samenvoegen in het naar `Bar.c` hernoemde bestand. Bij eenvoudige bestands- of mapverwijderingen kan hij er voor kiezen om het object met de wijzigingen van ontwikkelaar A te behouden en de verwijdering te negeren. Of, door middel van het als `Opgelost` markeren van de conflicten zonder iets gewijzigd te hebben, worden de wijzigingen van ontwikkelaar A volledig genegeerd.

Het conflicten bewerkingsvenster biedt de mogelijkheid om wijzigingen samen te voegen, als het originele bestand van de hernoemde `Bar.c` aangewezen kan worden. Afhankelijk van waar de verversing was aangeropen, is het misschien niet mogelijk om het bronbestand te vinden.

4.6.2.2. Lokale wijziging, inkomende verwijdering bij verversing

1. Ontwikkelaar A hernoemt `Foo.c` naar `Bar.c` en legt de wijziging vast in het archief.

2. Ontwikkelaar B wijzigt `Foo.c` in zijn werkkopie.

Of bij een verplaatsing van een map ...

1. Ontwikkelaar A hernoemt hoofdmap `FooFolder` in `BarFolder` en legt de wijziging vast in het archief.

2. Ontwikkelaar B wijzigt `Foo.c` in zijn werkkopie.

Een verversing van de werkkopie van ontwikkelaar B leidt tot een mapstructuur conflict. Een eenvoudig bestandsconflict:

- `Bar.c` wordt toegevoegd aan de werkkopie als een normaal bestand.
- `Foo.c` wordt als toegevoegd (inclusief geschiedenis) gemarkeerd en heeft een mapstructuur conflict.

Een mapstructuur conflict:

- `BarFolder` wordt aan de werkkopie toegevoegd als een normale map.
- `FooFolder` wordt gemarkeerd als toegevoegd (inclusief geschiedenis) en heeft een mapstructuurconflict.

`Foo.c` wordt gemarkeerd als gewijzigd.

Ontwikkelaar B moet nu de keuze maken of deze meegaat met de herindeling van ontwikkelaar A. Als ontwikkelaar B daarin meegaat, kan hij zijn wijzigingen in het nieuwe bestand samenvoegen. Zo niet, dan kan ontwikkelaar B de wijzigingen van A terugdraaien en zijn eigen lokale versie behouden.

Om de eigen wijzigingen in de herindeling te verwerken, moet ontwikkelaar B uitvinden welke naam `Foo.c` heeft gekregen in het archief. Dit kan met het log venster. De wijzigingen moeten vervolgens met de hand verwerkt worden, omdat er nu nog geen mogelijkheid is dit te automatiseren of te vereenvoudigen. Als de wijzigingen doorgevoerd zijn, is het conflicterende pad overbodig en kan dan verwijderd worden. Gebruik hiervoor de **Verwijderen** knop in de conflict bewerkingsvenster om het conflict als opgelost te markeren.

Als ontwikkelaar B tot de conclusie komt dat de wijzigingen van A verkeerd zijn, dan moet dat aangegeven worden met de **Behouden** knop in het conflict bewerkingsvenster. Hiermee wordt het conflicterende bestand of de map gemarkeerd als opgelost. De wijzigingen van ontwikkelaar A moeten wel nog met de hand verwijderd worden. Ook hier helpt het log venster met te bepalen wat er gewijzigd/hernoemd is.

4.6.2.3. Lokale verwijdering, inkomende verwijdering bij verversing

1. Ontwikkelaar A hernoemt `Foo.c` in `Bar.c` en legt dit vast in het archief.

2. Ontwikkelaar B hernoemt `Foo.c` in `Bix.c`

Een verversing van de werkkopie van ontwikkelaar B resulteert in een mapstructuur conflict:

- `Bix.c` wordt gemarkeerd als toegevoegd inclusief geschiedenis.
- `Bar.c` is toegevoegd aan de werkkopie met 'normale' status.
- `Foo.c` wordt gemarkeerd als verwijderd en leidt tot een mapstructuur conflict.

Om dit conflict op te lossen, moet ontwikkelaar B uitzoeken welke bestandsnaam de conflicterend `Foo.c` heeft gekregen in het archief. Het log venster kan hiervoor gebruikt worden.

Ontwikkelaar B moet dan bepalen welke bestandsnaam van `Foo.c` behouden moet worden, degene van ontwikkelaar A of de zelf gekozen nieuwe bestandsnaam.

Wanneer ontwikkelaar B het conflict handmatig heeft opgelost, dan moet het conflict als opgelost worden gemarkeerd in het conflict bewerkingsvenster.

4.6.2.4. Lokaal missend, binnenkomende wijziging bij het samenvoegen

1. Ontwikkelaar A werkt in de basislijn en wijzigt `Foo.c` en legt de wijzigingen vast in het archief.
2. Ontwikkelaar B werkt aan een branch en hernoemd `Foo.c` in `Bar.c` en legt zijn wijzigingen vast in het archief.

Het samenvoegen van de wijzigingen van A op de basislijn naar de tak-werkkopie van ontwikkelaar B leidt tot een mapstructuur conflict:

- `Bar.c` bevindt zich al in de werkkopie met status 'normaal'.
- `Foo.c` wordt gemarkeerd als missend met een mapstructuur conflict.

Ontwikkelaar B moet het bestand als opgelost markeren in het conflict bewerkingsvenster, om ervoor te zorgen dat het bestand van de conflict lijst wordt gehaald. Vervolgens moet besloten worden of het missende bestand `Foo.c` vanuit het archief naar de werkkopie gekopieerd moet worden, of dat de wijzigingen van ontwikkelaar A in `Foo.c` samengevoegd moeten worden in `Bar.c`, of de wijzigingen te negeren door het conflict als opgelost te markeren zonder iets gewijzigd te hebben.

Merk op dat als je het missende bestand eerst uit het archief kopieert en het dan als opgelost markeert, dan je kopie meteen weer verwijderd wordt. Je moet eerst het conflict oplossen en daarna pas kopiëren.

4.6.2.5. Lokale wijziging, inkomende verwijdering bij samenvoegen

1. Ontwikkelaar A werkt aan de basislijn en hernoemt `Foo.c` in `Bar.c` en legt de wijziging vast in het archief.
2. Ontwikkelaar B werkt aan een vertakking (branch), wijzigt `Foo.c` en legt de wijziging vast in het archief.

Er is een vergelijkbare situatie voor verplaatste mappen, maar dit wordt nog niet gedetecteerd in Subversion 1.6 ...

1. Ontwikkelaar A werkt aan de basislijn en hernoemt de bovenliggende map `FooFolder` in `BarFolder` en legt de wijziging vast in het archief.
2. Ontwikkelaar B werkt aan de vertakking (branch) en wijzigt `Foo.c` in de werkkopie.

Het samenvoegen van de wijzigingen van A op de basislijn naar de tak-werkkopie van ontwikkelaar B leidt tot een mapstructuur conflict:

- `Bar.c` is gemarkeerd als toegevoegd.
- `Foo.c` is gemarkeerd als gewijzigd en met een mapstructuur conflict.

Ontwikkelaar B moet nu de keuze maken of deze meegaat met de herindeling van ontwikkelaar A. Als ontwikkelaar B daarin meegaat, kan hij zijn wijzigingen in het nieuwe bestand samenvoegen. Zo niet, dan kan ontwikkelaar B de wijzigingen van A terugdraaien en zijn eigen lokale versie behouden.

Om de eigen wijzigingen in de herindeling te verwerken, moet ontwikkelaar B uitvinden welke naam `Foo.c` heeft gekregen in het archief. Dit kan met het log venster van het samen te voegen bronbestand. De conflict verwerker toont alleen het log van de werkkopie, omdat niet bekend is welk pad gebruikt is bij het samenvoegen. Dit moet je zelf handmatig uitzoeken. De wijzigingen moeten vervolgens met de hand verwerkt worden, omdat er nu nog geen mogelijkheid is dit te automatiseren of te vereenvoudigen. Als de wijzigingen doorgevoerd zijn, is het conflicterende pad overbodig en kan dan verwijderd worden. Gebruik hiervoor de **Verwijderen** knop in de conflict bewerkingsvenster om het conflict als opgelost te markeren.

Als ontwikkelaar B tot de conclusie komt dat de wijzigingen van A verkeerd zijn, dan moet dat aangegeven worden met de **Behouden** knop in het conflict bewerkingsvenster. Hiermee wordt het conflicterende bestand of de map gemarkeerd als opgelost. De wijzigingen van ontwikkelaar A moeten wel nog met de hand verwijderd worden. Ook hier helpt het log venster met te bepalen wat er gewijzigd/hernoemd is.

4.6.2.6. Lokale verwijdering, inkomende verwijdering bij samenvoegen

1. Ontwikkelaar A werkt aan de basislijn en hernoemt `Foo.c` in `Bar.c` en legt de wijziging vast in het archief.
2. Ontwikkelaar B werkt aan een vertakking (branch) en hernoemt `Foo.c` naar `Bix.c` en legt de wijziging vast in het archief.

Het samenvoegen van de wijzigingen van A op de basislijn naar de tak-werkkopie van ontwikkelaar B leidt tot een mapstructuur conflict:

- `Bix.c` is gemarkeerd met de normale (ongewijzigde) status.
- `Bar.c` is gemarkeerd als toegevoegd inclusief geschiedenis.
- `Foo.c` is gemarkeerd als missend en heeft een mapstructuur conflict.

Om dit conflict op te lossen, moet ontwikkelaar B uitzoeken welke bestandsnaam de conflicterend `Foo.c` heeft gekregen in het archief. Het log venster van het samen te voegen bronbestand kan hiervoor gebruikt worden. Het conflict verwerker toont alleen het log van de werkkopie, omdat niet bekend is welk pad gebruikt is bij het samenvoegen. Je moet dat pad zelf handmatig opzoeken.

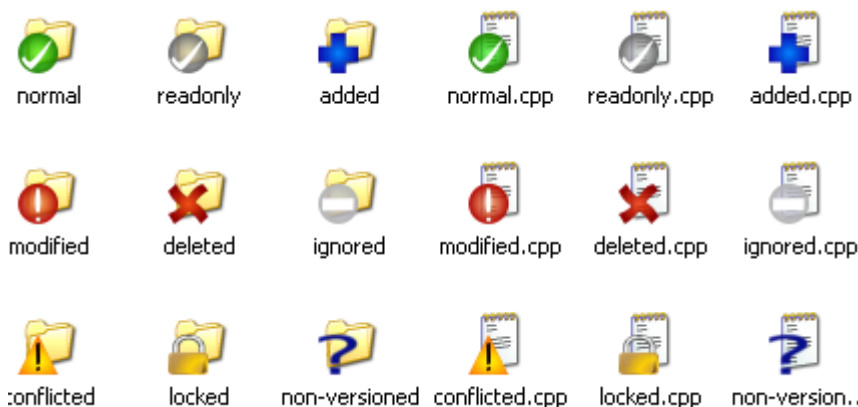
Ontwikkelaar B moet dan bepalen welke bestandsnaam van `Foo.c` behouden moet worden, degene van ontwikkelaar A of de zelf gekozen nieuwe bestandsnaam.

Wanneer ontwikkelaar B het conflict handmatig heeft opgelost, dan moet het conflict als opgelost worden gemarkeerd in het conflict bewerkingsvenster.

4.7. Statusinformatie ophalen

Terwijl je aan je werkkopie werkt, wil je vaak weten welke bestanden je gewijzigd, toegevoegd, verwijderd of hernoemt hebt, maar ook welke bestanden gewijzigd en vastgelegd zijn door anderen.

4.7.1. Overlappende Pictogrammen



Figuur 4.11. Overlappende pictogrammen weergegeven in de verkener

Nu je een werkkopie hebt opgehaald uit het Subversion archief, kun je de bestanden zien in de Windows verkener met de gewijzigde pictogrammen/iconen. Dit is één van de redenen waarom TortoiseSVN zo populair is. TortoiseSVN voegt zogenaamde overlay iconen, dat zijn iconen die over de normale iconen geprojecteerd worden, toe aan de originele iconen. Afhankelijk van de status in Subversion wordt telkens een ander pictogram weergegeven.



Een verse werkkopie opgehaald uit het archief heeft een groen vinkje als overlappend pictogram. Dit betekent dat dit object de Subversion status *normaal* heeft.



Zodra je een bestand gewijzigd en opgeslagen hebt, wordt de status gewijzigd in *gewijzigd* en zal het overlappende pictogram wijzigen in een rood pictogram met een uitroepteken. Op deze manier kun je makkelijk zien welke bestanden gewijzigd zijn sinds de laatste keer dat je je werkkopie ververst hebt.



Als bij het verversen zich een *conflict* voordoet, dan zal er een geel pictogram met een uitroepteken te zien zijn.



Als je de `svn:needs-lock` eigenschap ingesteld hebt op een bestand, dan zal Subversion dat bestand als alleen-lezen instellen totdat het bestand voor jou vergrendeld is. Dergelijke bestanden krijgen deze pictogram om aan te geven dat je het bestand eerst moet vergrendelen, voordat je het kunt wijzigen.



Als je een bestand vergrendeld hebt en de Subversion status is *normaal*, dan herinnert dit overlappende pictogram je eraan dat je het bestand nog vrij moet geven als je anderen de mogelijkheid wilt geven om daarin wijzigingen vast te laten leggen.



Dit pictogram geeft aan dat bestanden of mappen in de huidige map gemarkeerd zijn om te *verwijderen* uit het archief of dat een bestand in het archief mist in de folder van je werkkopie.



Het plusteken geeft aan dat het bestand of de map van versiebeheer voorzien moet worden en dus aan het archief *toegevoegd* moet worden.



Het balk-teken geeft aan dat een bestand of map *genegeerd* wordt voor versiebeheer. Dit pictogram is optioneel.



Dit pictogram geeft de bestanden en mappen aan die niet van versiebeheer voorzien zijn, maar ook niet genegeerd worden. Dit pictogram is optioneel.

In de praktijk zul je zien dat niet alle pictogrammen gebruikt zullen worden op jouw systeem. Dit komt doordat Windows maar een beperkt aantal overlappende pictogrammen toestaat. Als je ook nog een oude versie van TortoiseCVS gebruikt, dan zijn er niet genoeg sloten voor overlappende pictogrammen beschikbaar. TortoiseSVN probeert een goede burger (“Good Citizen (TM)”) te zijn en beperkt het gebruik van overlappende pictogrammen om andere applicaties ook een kans te geven.

Tegenwoordig zijn er meerdere Tortoise applicaties beschikbaar (TortoiseCVS, TortoiseHG, ...), waardoor de beperking van het aantal pictogrammen echt een probleem begint te worden. Om dit probleem te omzeilen is er in het TortoiseSVN project een standaard gedeelde set van pictogrammen geïntroduceerd. Deze set kan geladen worden als een DLL, welke door alle Tortoise applicaties gebruikt kan worden. Ga bij jouw applicatie leverancier na of daar deze set al geïntegreerd is. :-)

Een beschrijving van de overlappende pictogrammen en hoe deze zich met de Subversion status en andere technische details verhouden kun je vinden in [Paragraaf F.1, “Overlappende Pictogrammen”](#).

4.7.2. TortoiseSVN Kolommen In Windows Verkenner

De informatie die door de overlappende pictogrammen weergegeven (en nog veel meer), kan ook in een extra kolom weergegeven worden in de Gedetailleerde Weergave van Windows Verkenner.

Klik met rechts op de koptekst van een kolom, selecteer Meer ... in het weergegeven contextmenu. Er verschijnt dan een venster waarin je de gewenste kolommen en volgorde kunt aangeven van de weer te geven kolommen in de “Gedetailleerde Weergave”. Scroll naar beneden totdat je de opties beginnend met SVN in beeld komen. Selecteer degene die je wilt zien en sluit af door op OK te klikken. De kolommen zullen dan aan de rechter zijde worden toegevoegd. Je kunt de volgorde veranderen door de kolommen te verslepen en de breedte aanpassen, zodat het past bij jouw wensen.



Belangrijk

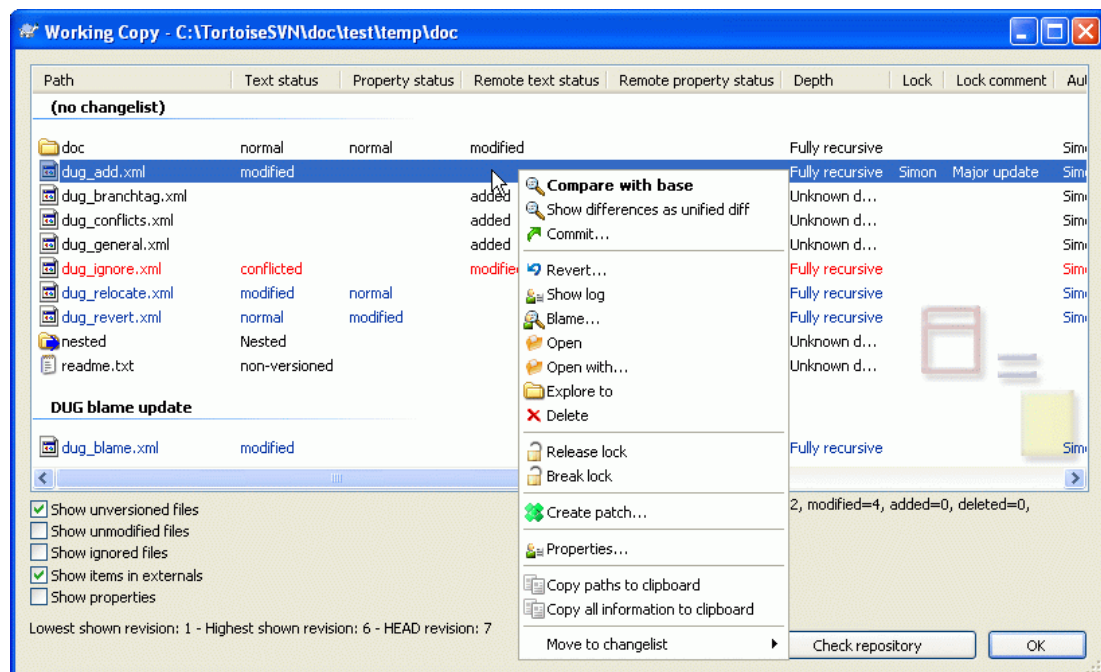
In Windows Vista zijn de extra kolommen niet beschikbaar. Microsoft heeft besloten extra kolommen niet meer voor elk type bestand toe te staan, alleen voor bepaalde bestanden kan dat nog.



Tip

Indien je wilt dat de huidige indeling in al jouw werkkopieën wordt getoond, kun je deze als standaard weergave instellen.

4.7.3. Lokale en Server Status



Figuur 4.12. Controleer op Wijzigingen

Het is vaak handig om te weten welke bestanden anderen gewijzigd en vastgelegd hebben. Hiervoor is het commando TortoiseSVN → Kijk of er updates zijn... erg handig. Het venster zal elk bestand weergeven dat op welke manier dan ook gewijzigd is in je werkkopie en ook de bestanden zonder versiebeheer.

Als je op de **Controleer Archief** klikt, dan kun je ook de wijzigingen in het archief bekijken. Hiermee kun je, voordat je je wijzigingen vastlegt, nagaan of er conflicten zullen ontstaan. Je kunt ook alleen geselecteerde bestanden verversen, zonder dat de hele map ververst hoeft te worden. De **Controleer Archief** haalt standaard de status op van de werkkopie met de instelling van mapdiepte van de werkkopie. Als je alle bestanden in het archief wilt zien, ook degene die je niet opgehaald hebt, dan moet je de **Shift** ingedrukt houden als je op de knop **Controleer Archief** drukt.

In dit venster worden kleurcoderingen toegepast om de status weer te geven.

Blauw

Lokaal gewijzigde objecten.

Paars

Toegevoegde objecten. Objecten die toegevoegd zijn met geschiedenisinformatie, hebben een + teken in de **Tekst status** kolom en in tooltip wordt aangegeven waar het object vandaan gekopieerd is.

Donker rood

Verwijderde of missende objecten.

Groen

Objecten die lokaal en in het archief gewijzigd zijn. De wijzigingen zullen samengevoegd worden bij het verversen, waarbij conflicten *kunnen* ontstaan.

Fel rood

Objecten die lokaal gewijzigd zijn en verwijderd uit het archief, of die gewijzigd zijn in het archief en lokaal verwijderd. Hierdoor *zal* een conflict ontstaan bij het verversen.

Zwart

Ongewijzigde objecten en objecten zonder versiebeheer.

Dit is het standaard kleurenschema, maar je kunt deze naar je eigen wensen aanpassen in het instellingen venster. Lees voor meer informatie [Paragraaf 4.30.1.4, "TortoiseSVN kleurinstellingen"](#).

Objecten die verplaatst zijn naar een andere archieflocatie worden aangeduid met een (s) teken. Het kan zijn dat je omgeschakeld bent terwijl je werkte aan een tak en vergeten was om terug te schakelen naar de basislijn. Dit is je waarschuwing!

Vanuit het context menu van het venster kun je de verschillen van de wijzigingen opvragen. Controleer de wijzigingen die *jij* lokaal gemaakt hebt door te kiezen voor **Context Menu** → **Vergelijk met basis**. Om de wijzigingen te bekijken die anderen in het archief hebben opgeslagen kies je **Context Menu** → **Toon verschillen als unified diff**.

Het is ook mogelijk om wijzigingen in individuele bestanden terug te draaien. Als je per ongeluk een bestand verwijderd hebt, dan zal deze weergegeven worden als *Ontbrekend*. Je kunt deze dan terughalen met *Maak ongedaan*.

Bestanden zonder versiebeheer of bestanden die op de negeren lijst staan, kunnen verwijderd worden (naar de prullenbak) van de lokale kopie door de optie **Context Menu** → **Verwijderen** te gebruiken. Als je bestanden permanent wilt verwijderen (zonder ze naar de prullenbak te doen), dan moet je **Shift** ingedrukt houden als je op **Verwijderen** klikt.

Als je de details van een bestand wilt bekijken, dan kun je het van hieruit naar een andere applicatie verlopen. Bijvoorbeeld naar een tekstverwerker of IDE.

Je kunt de kolommen aanpassen. Als je met rechts klikt op de kopetekst van een kolom, dan krijg je een context menu te zien waarmee je kunt aangeven welke kolommen je wilt zien. De breedte van de kolommen kun je aanpassen door de kolomovergangen te verslepen. De aanpassingen worden opgeslagen, zodat deze er een volgende keer net zo uit zien.

Je kunt groepen bestanden in een wijzigingenlijst opnemen. Dit is handig als je aan meerdere taken, die niets met elkaar te maken hebben, tegelijk werkt. Lees hiervoor [Paragraaf 4.4.2, "Werksets"](#).

Onderaan het scherm kun je zien welke reeks van archiefrevisies er in je werkkopie gebruikt worden. Dit zijn de revisies van de *vastlegging*, niet van de laatste *verversing*; dit geeft de reeks van revisies weer van de laatste vastleggingen, niet van de laatste verversing. Merk op dat de reeks van revisies alleen slaat op de objecten die weergegeven zijn in het venster, niet op je hele werkkopie. Als je deze informatie van je hele werkkopie wilt zien, dan moet je het selectievak **Niet gewijzigde bestanden weergeven** selecteren.



Tip

Als je een platte weergave wilt hebben van je werkkopie, oftewel alle bestanden van alle niveau's uit je mapstructuur, dan kun je daarvoor het beste de **Kijk of er updates zijn...** functie gebruiken. Zet de optie **Ongewijzigde bestanden weergeven** aan om alle bestanden in je werkkopie te zien.



Externe Hernoemingen Repareren

Soms worden bestanden hernoemt buiten Subversion, waardoor deze bestanden in de lijst weergegeven worden als missend of zonder versiebeheer. Om te voorkomen dat je de geschiedenis verliest, moet je Subversion vertellen dat er een verbinding is. Selecteer de oude (missende) naam en de nieuwe naam en gebruik **Context Menu** → **Repareer Verplaatsing** om de bestanden aan elkaar te koppelen als hernoemden.

4.7.4. Diffs bekijken

Vaak wil je de inhoud van bestanden even bekijken, om te bekijken wat je precies gewijzigd hebt. Dit kun je doen door een gewijzigd bestand te selecteren en **Vergelijk** in TortoiseSVN's context menu te selecteren. Hiermee wordt het externe vergelijk-programma gestart, waarmee het huidige bestand vergeleken zal worden met de ongewijzigde kopie (BASE revisie) die opgeslagen was na de laatste verversing.



Tip

Zelfs bestanden die niet in je werkkopie zitten of als je verschillende versies verspreid opgeslagen hebt, kun je verschillen bekijken:

Selecteer in de verkenner de twee bestanden die je wilt vergelijken (bijvoorbeeld met **Ctrl** en de muis) en selecteer vanuit TortoiseSVN's context menu **Vershil**

4.8. Werksets

In de ideale wereld werk je maar aan één ding tegelijk en zal je werkkopie alleen maar logische wijzigingen bevatten. Ok, dan terug naar de werkelijkheid. Het komt vaak voor dat je aan verschillende taken, die niets met elkaar te maken hebben, tegelijk moet werken. Als je dan naar het vastleggen venster kijkt, staan alle wijzigingen door elkaar. De *werkset* functie helpt je met het groeperen van bestanden, waardoor je een beter overzicht krijgt waar je mee bezig bent. Dit werkt natuurlijk alleen als de wijzigingen elkaar niet overlappen. Als bij twee verschillende taken hetzelfde bestand gewijzigd wordt, dan is er geen methode om die wijzigingen van elkaar te scheiden.



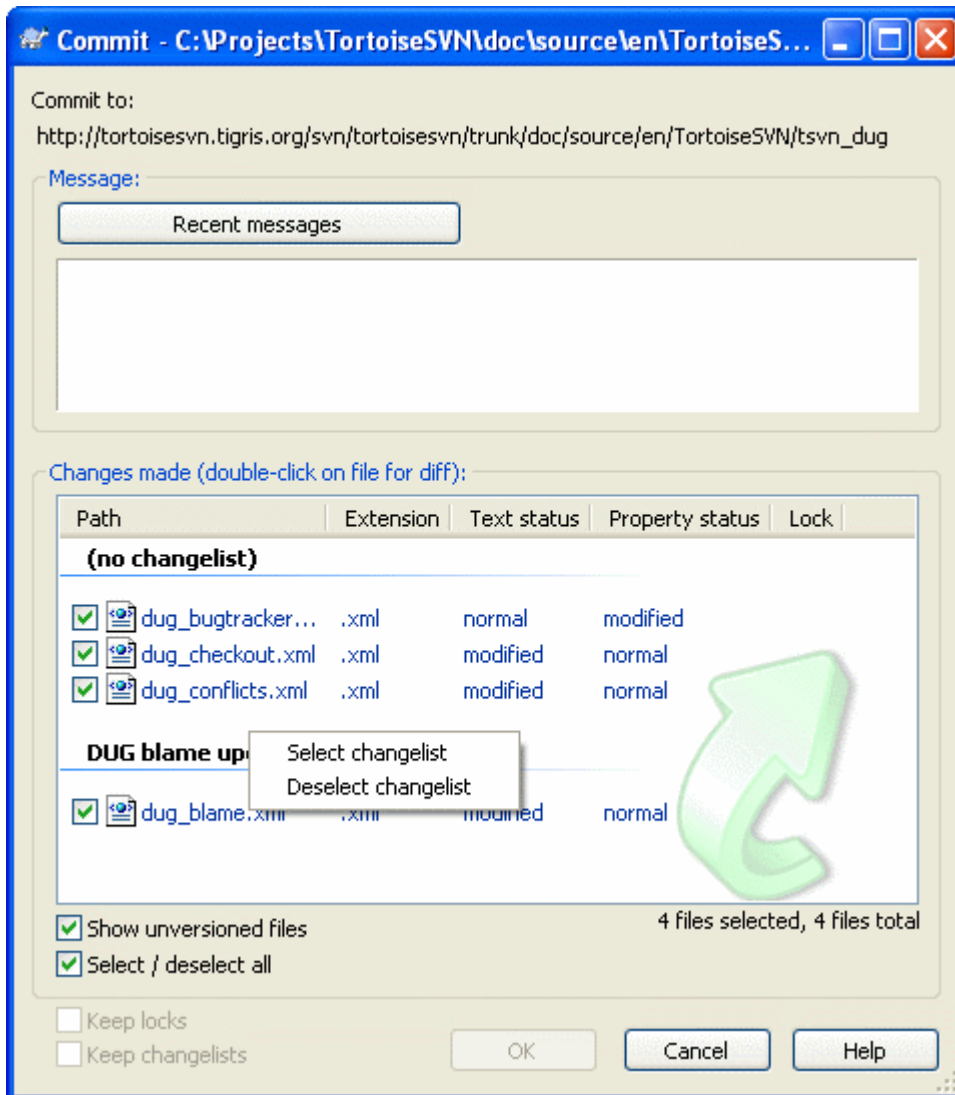
Belangrijk

De werkset functie is alleen in Windows XP en latere versies beschikbaar, omdat hiervoor bepaalde mogelijkheden van de shell nodig zijn die niet in Windows 2000 zitten. Sorry, maar Win2k is nu toch echt wel verouderd, dus ga alsjeblieft niet klagen.

Je kunt werksets vinden op verschillende plaatsen, maar de belangrijkste zijn die in het vastleggen venster en kijk-of-er-updates-zijn venster. Laten we beginnen met het kijk-of-er-updates-zijn venster nadat je aan verschillende functies en meerdere bestanden hebt gewerkt. Als je het venster voor het eerst opent, dan worden alle bestanden in de lijst weergegeven. Stel voor dat je deze lijst nu wilt organiseren en de bestanden wilt groeperen.

Selecteer één of meer bestanden en gebruik de functie **Context Menu** → **Verplaats naar werkset** om een object aan de werkset toe te voegen. In het begin zullen er geen werksets bestaan, waardoor er bij de eerste keer dat je dit doet een nieuwe werkset aangemaakt zal worden. Geef de werkset een naam die de lading dekt en klik op **OK**. Vervolgens zal de groepen met objecten weergegeven worden.

Als je eenmaal een werkset hebt aangemaakt, dan kun je er objecten naartoe slepen. Dit kan vanuit een andere werkset of vanuit de Windows Verkenner. Het verslepen vanuit de Verkenner is handig, omdat je objecten al aan de werkset kunt toevoegen nog voordat deze gewijzigd zijn. Dit kun je ook wel doen vanuit het kijk-of-er-updates-zijn venster, maar dan moet je alle bestanden zonder versiebeheer laten weergeven.



Figuur 4.13. Vastleggen venster met werkset

In het vastleggen venster kun je de bestanden dan zien, maar dan gegroepeerd per werkset. Buiten dat je een directe visuele groepering van de bestanden ziet, kun je ook nog de groepen van een kop voorzien waarmee je de set van bestanden voor het vastleggen kunt selecteren.

In XP is er een context menu beschikbaar als je met rechts op de kop van een groep klikt. In dat menu kun je met een optie alle groepsobjecten selecteren of deselecteren. Bij Vista is dat menu niet nodig. Daar kun je op de kop van de groep klikken en vervolgens op één van de objecten, waarna alle objecten dezelfde selectiestatus krijgen.

TortoiseSVN heeft één werkset gereserveerd voor eigen gebruik, dat is de negeren-bijvastleggen set. Deze wordt gebruikt voor het markeren van bestanden met versiebeheer, die je bijna nooit wilt vastleggen ook al zijn er lokale wijzigingen. Deze functie wordt beschreven in [Paragraaf 4.4.3, “Objecten van de vastleggingslijst uitsluiten”](#).

Als je bestanden vastlegt die opgenomen zijn in een werkset, dan zou je verwachten dat de koppeling aan de werkset niet langer nodig is. Normaal gesproken worden bestanden automatisch uit een werkset verwijderd als je je wijzigingen vastlegt. Als je het bestand in de werkset wilt behouden, gebruik dan de optie Werksets bewaren onderaan het vastleggen venster.



Tip

Werksets is een lokale functie. Het aanmaken en verwijderen van werksets zal het archief niet veranderen, noch de werkkopie van iemand anders. Het is een handig manier voor je om je bestanden te organiseren.

4.9. Logboekboodschappen Venster

Bij elke wijziging die je doorvoert en vastlegt, moet je een logboekbericht maken. Dat helpt je om later te achterhalen waarom je welke wijzigingen hebt gedaan. Daarbij heb je dan ook meteen een gedetailleerd rapport van je ontwikkelproces.

Het Logboek boodschappen venster haalt alle logboekboodschappen op en toont deze. Het venster is verdeeld in 3 gedeeltes.

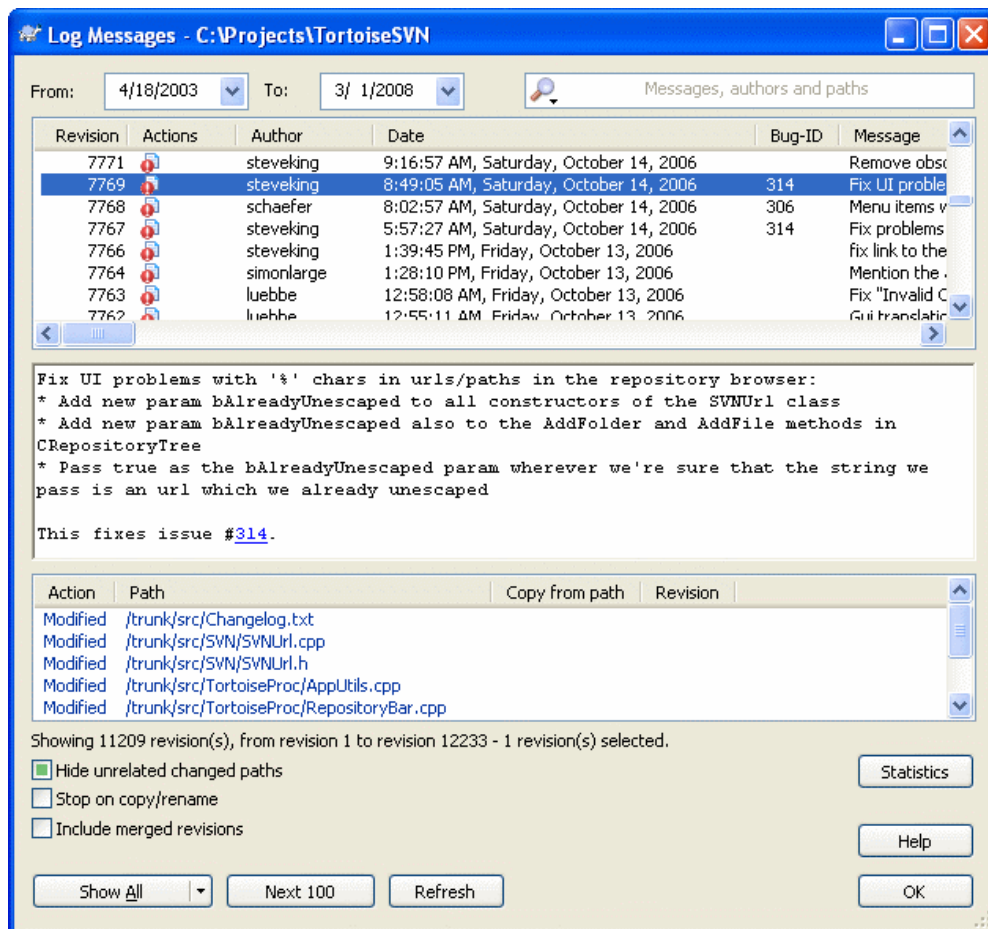
- Het bovenste gedeelte geeft een lijst van revisies weer, waarin wijzigingen aan bestanden en mappen zijn vastgelegd. Deze samenvatting geeft ook de datum, de tijd, degene die de revisie heeft vastgelegd en het begin van het logboekbericht weer.

Blauwe regels geven aan dat er iets naar deze revisie is gekopieerd, verplaatst of hernoemt is (mogelijk vanuit een vertakking/branch).

- Het middelste gedeelte geeft het complete logboekbericht weer van de geselecteerd revisie.
- Het onderste gedeelte geeft de bestanden en mappen weer, die in deze revisie gewijzigd zijn.

Maar het doet nog veel meer dan dat alleen - er zijn ook functies beschikbaar in het context menu, waarmee je nog veel meer informatie over de geschiedenis van het project kunt opvragen.

4.9.1. Het Logboek Boodschappen Venster Opstarten



Figuur 4.14. Het Logboek Boodschappen Venster

Er zijn verschillende plaatsen van waaruit je het Logboek boodschappen venster kunt openen:

- Vanuit het TortoiseSVN context submenu
- Vanuit de eigenschappen pagina
- Vanuit het voortgangsvenster nadat een verversing is afgerond. Dan zal het Logboek venster alleen de revisies weergeven van wijzigingen die doorgevoerd zijn, nadat je voor het laatst verversst hebt.

Als het archief niet beschikbaar is, zal er een venster krijgen met de vraag *Wilt u offline werken?*. Dit wordt verder beschreven in [Paragraaf 4.9.10, "Offline Modus"](#).

4.9.2. Revisie Logboek Acties

Het bovenste gedeelte van het venster heeft een **Acties** kolom, waarin met pictogrammen wordt aangegeven wat er in die revisie gedaan is. Er zijn vier verschillende pictogrammen, die elke in een aparte kolom worden weergegeven.



Als in een revisie bestanden of mappen gewijzigd zijn, dan wordt het *gewijzigd* pictogram weergegeven in de eerste kolom.



Als in een revisie een bestand of map is toegevoegd, dan wordt het *toegevoegd* pictogram weergegeven in de tweede kolom.

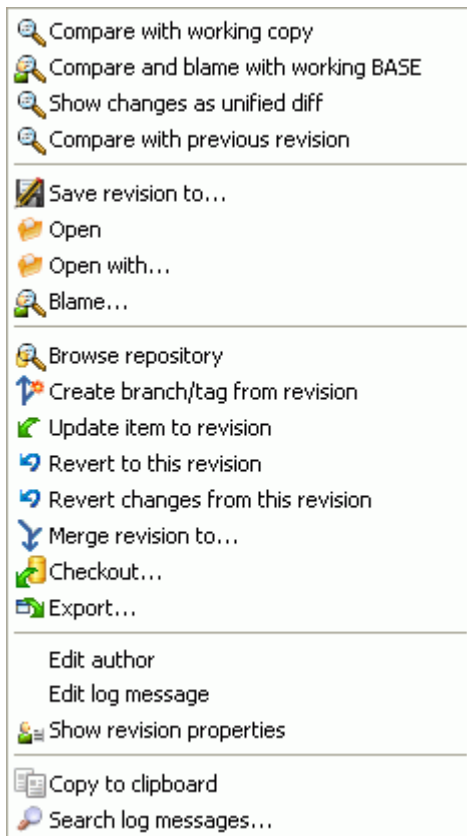


Als in een revisie bestanden of mappen verwijderd zijn, dan wordt het *verwijderd* pictogram weergegeven in de derde kolom.



Als in een revisie bestanden of mappen vervangen zijn, dan wordt het *vervangen* pictogram weergegeven in de vierde kolom.

4.9.3. Extra Informatie Opvragen



Figuur 4.15. Het Context Menu Van Het Bovenste Gedeelte In Het Logboek Boodschappen Venster

Het bovenste gedeelte van het Logboek boodschappen venster heeft een context menu, waarmee je nog veel meer informatie kunt opvragen. Enkele van deze functies worden alleen getoond als het logboek van een bestand wordt weergegeven, andere functies weer alleen bij een map.

Vergelijk met werkkopie

Vergelijk de geselecteerde revisie met je werkkopie. Het standaard Diff-programma is TortoiseMerge, welke meegeleverd wordt met TortoiseSVN. Als het logboek venster wordt weergegeven voor een map, dan zal in de lijst de gewijzigde bestanden weergegeven worden en heb je de mogelijkheid om de wijzigingen van individuele bestanden te bekijken.

Vergelijk en verklaar met werkkopie (BASE)

Voor de geselecteerde revisie: de geselecteerde revisie en de BASE versie vergelijken met het Diff-programma bekijken. Lees voor meer informatie [Paragraaf 4.23.2, "Verklaar Verschillen"](#). Deze functie is alleen voor bestanden beschikbaar.

Toon wijzigingen als unified diff

Bekijk de wijzigingen van de geselecteerde revisie als Unified-Diff bestand (GNU patch formaat). Hiermee worden alleen de verschillen getoond met enkele regels voor de context. Het is lastiger om dit te lezen dan een visuele vergelijking, maar geeft wel alle wijzigingen compact weer.

Vergelijk met vorige revisie

Vergelijk de geselecteerde revisie met de vorige revisie. Deze functie werkt ongeveer gelijk als het vergelijken met je werkkopie. Als je deze functie gebruikt op mappen, dan zal er eerst een venster weergegeven worden met gewijzigde bestanden. Je kunt dan selecteren welke bestanden je wilt vergelijken.

Vergelijk en verklaar met vorige revisie

Toont het venster met de gewijzigde bestanden, waarin je de te vergelijken bestanden kunt selecteren. Het geselecteerde bestand en de vorige revisie vergelijken en de verschillen verklaren met een visueel diff programma. (Alleen voor mappen).

Bewaar revisie in...

De geselecteerde revisie opslaan in een bestand, zodat je de oude versie van het bestand hebt. (Alleen voor bestanden)

Openen / Openen met...

Open het geselecteerd bestand met het standaard ingestelde programma voor dat bestandstype, of met een door jou geselecteerd programma. (Alleen voor bestanden)

Verklaren...

Het geselecteerde bestand tot aan de geselecteerde revisie verklaren. (Alleen voor bestanden)

Archief bekijken

Open de archief-verkenner om de geselecteerde bestanden en mappen te onderzoeken zoals deze waren op het moment van de geselecteerde revisie.

Maak tak/label van revisie

Maak een tak of label van de geselecteerde revisie. Deze functie is handig als je bijvoorbeeld vergeten bent een label aan te maken terwijl je wijzigingen al vastgelegd hebt, die eigenlijk niet in die versie zouden moeten zitten.

Ververs onderdeel naar revisie

Ververs je werkkopie met de gegevens van de geselecteerd revisie. Deze functie is handig als je de status van een project wilt bekijken uit het verleden. Ook is deze handig als er verschillende nieuwe revisies zijn aangemaakt en je je eigen werkkopie stapsgewijs wilt bijwerken in plaats van in één keer de hele sprong te nemen. Het is verstandig om een hele map te verversen in plaats van losse bestanden. Hiermee voorkom je dat je werkkopie inconsistenties zal bevatten.

Als je een eerdere wijziging permanent ongedaan wilt maken, gebruik dan **Ongedaan maken tot deze revisie**.

Ongedaan maken tot deze revisie

Maak de wijzigingen ongedaan en ga terug naar de geselecteerd revisie. Mocht je wijzigingen gemaakt hebben en tot de conclusie komen dat je eigenlijk terug wilt naar de stand van zaken bij revisie N, dan is dit het commando dat je moet gebruiken. Met deze functie maak je de wijzigingen die je doorgevoerd heb in je werkkopie ongedaan. Deze functie verandert dus *niets* in het archief, totdat je zelf wijzigingen vastlegt. Merk op dat deze functie *alle* wijzigingen die gedaan zijn na de geselecteerde revisie ongedaan maakt en dus het bestand of de map vervangt door de oudere versie.

Als je werkkopie voordat je deze actie uitvoert ongewijzigd is, dan zal na deze actie je werkkopie als gewijzigd aangegeven worden. Als je al lokale wijzigingen had aangebracht, dan zal met dit commando de *ongedaan-maken* wijzigingen worden samengevoegd in je werkkopie.

Op de achtergrond voert Subversion eigenlijk een samenvoeging van aangebrachte wijzigingen tot aan de geselecteerde revisie achterstevoren uit, waardoor de acties van eerdere vastleggingen ongedaan worden gemaakt.

Als je na het uitvoeren van deze actie bedenkt dat je eigenlijk de *ongedaanmaking ongedaan wilt maken*, zodat je je werkkopie terugkrijgt in de ongewijzigde situatie, dan moet je de functie TortoiseSVN → Ongedaan maken gebruiken vanuit de Windows Verkenner. Hiermee worden de lokale wijzigingen die door deze achterstevoren-samenvoeging waren gemaakt, teruggedraaid.

Als je alleen maar wilt zien hoe een bestand of map er uit zag bij een eerdere revisie, gebruik dan Verversen naar revisie of Revisie opslaan als....

Revisie wijzigingen ongedaan maken

Wijzigingen van de geselecteerde revisie ongedaan maken. De wijzigingen worden ongedaan gemaakt in je werkkopie, oftewel in het archief zal er *niets* veranderd worden. Merk op dat hiermee alleen de wijzigingen van die revisie ongedaan worden gemaakt in je lokale bestand; het bestand in je werkkopie wordt niet compleet vervangen door die van de eerdere revisie. Deze functie is erg handig als je een eerdere wijziging ongedaan wilt maken, terwijl er in de tussentijd andere ongerelateerde wijzigingen zijn doorgevoerd.

Als je werkkopie voordat je deze actie uitvoert ongewijzigd is, dan zal na deze actie je werkkopie als gewijzigd aangegeven worden. Als je al lokale wijzigingen had aangebracht, dan zal met dit commando de *ongedaan-maken* wijzigingen worden samengevoegd in je werkkopie.

Op de achtergrond voert Subversion eigenlijk een samenvoeging van die specifieke revisie achterstevoren uit, waardoor het resultaat van een vorige vastlegging ongedaan wordt gemaakt.

Je kunt de *ongedaanmaking weer ongedaan maken* zoals eerder beschreven is bij **Ongedaan maken tot deze revisie**.

Voeg revisie samen met...

Voeg de geselecteerde revisie(s) samen in een andere werkkopie. In venster kun de map selecteren van de werkkopie waarin je de samenvoeging wilt opslaan. Daarna komt er geen bevestigingsvenster meer en is er ook geen optie om de samenvoeging te testen. Het is daarom verstanding om de samenvoeging uit te voeren in een ongewijzigde werkkopie, zodat je de wijzigingen ongedaan kunt maken als het uiteindelijk niet blijkt te worden wat je had willen hebben. Deze functie komt van pas als je revisies van één tak in een andere wilt samenvoegen.

Ophalen...

Haal een verse kopie op van de geselecteerde map uit de betreffende revisie. Er wordt dan een venster weergegeven waarin je de URL en de revisie moet bevestigen en de locatie waar de map naartoe opgehaald moet worden.

Exporteren...

Exporteer het geselecteerde bestand of de map van de geselecteerde revisie. Er wordt dan een venster weergegeven waarin je de URL en de revisie moet bevestigen en de locatie op moet geven waar naartoe de map geëxporteerd moet worden.

Auteur wijzigen / Logboek boodschap wijzigen

Bewerk de logboekboodschap of auteur van een vorige vastlegging. Lees **Paragraaf 4.9.7, "Logboekboodschap en Auteur Wijzigen"** om hier meer over te weten te komen.

Toon revisie-eigenschappen

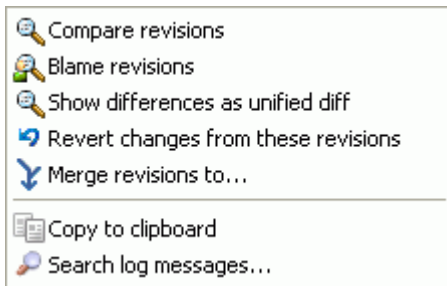
Bekijk en bewerk een eigenschap van een revisie, niet beperkt tot de logboekboodschap of auteur. Lees **Paragraaf 4.9.7, "Logboekboodschap en Auteur Wijzigen"**.

Naar klembord kopiëren

Kopieer details uit het logboek van de geselecteerde revisie naar het klembord. Hierbij worden het revisienummer, auteur, datum, logboekboodschap en de lijst met gewijzigde objecten voor elke revisie gekopieerd.

Doorzoek logboekberichten...

Doorzoek de logboekboodschappen op de ingevoerde tekst. Hiermee wordt er gezocht in de logboekberichten en in de samenvattingen van Subversion acties (die weergegeven worden in het onderste deel van het venster). De zoekopdrachten zijn niet hoofdlettergevoelig.



Figuur 4.16. Context menu van het bovenste deelvenster voor 2 geselecteerde revisies

Als je twee revisies selecteert (als je klikt en **Ctrl** ingedrukt houdt), dan zullen er in het context menu minder opties worden weergegeven:

Vergelijk revisies

Vergelijk de twee geselecteerde revisies met een visueel vergelijkingsprogramma. Het standaard Diff-programma is TortoiseMerge, welke meegeleverd wordt met TortoiseSVN.

Een nieuw venster wordt er geopend, als je deze optie selecteert voor een map. Daarin worden de gewijzigde bestanden weergegeven en worden er meer opties voor het vergelijken gegeven. Lees meer over het Vergelijk revisies venster in [Paragraaf 4.10.3, "Vergelijken van mappen"](#).

Verklaar revisies

Verklaar de twee revisies en vergelijk de verklaarrapportages in een visueel vergelijk programma. Lees voor meer informatie [Paragraaf 4.23.2, "Verklaar Verschillen"](#).

Toon verschillen als unified diff

Bekijk de verschillen van twee verschillende revisies opgeslagen in een unified-diff bestand.

Naar klembord kopiëren

Kopieer de logboekboodschappen naar het klembord.

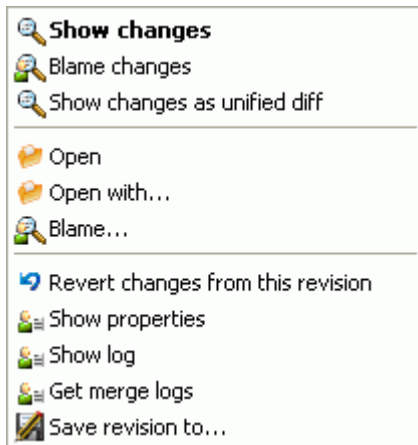
Doorzoek logboekberichten...

Doorzoek de logboekboodschappen.

Als je twee of meer revisies selecteert (als je klikt en **Ctrl** of **Shift** ingedrukt houdt), dan zal in het context menu een optie weergegeven worden waarmee alle wijzigingen van die revisies teruggedraaid kunnen worden. Dit is de makkelijkste methode om een verzameling van wijzigingen ineens ongedaan te maken.

Je kunt er ook voor kiezen om de geselecteerde revisies samen te voegen naar een andere werkkopie, zoals eerder beschreven is.

Als alle revisies van dezelfde auteur zijn, dan kun je in één keer de naam van de auteur wijzigingen voor die revisies.



Figuur 4.17. Het context menu van het onderste deelvenster van het Logboekboodschappen venster

Het onderste deelvenster van het logboek boodschappenvenster heeft ook een context menu met de volgende opties:

Toon wijzigingen

Toon bestandswijzigingen in het geselecteerde bestand. Dit context menu is alleen beschikbaar bij als *gewijzigd* gemarkeerde bestanden.

Verklaar wijzigingen

Verklaar de geselecteerde en vorige revisie van het geselecteerde bestand en vergelijk de verklarenrapportages met een visueel diff programma. Lees ook [Paragraaf 4.23.2, “Verklaar Verschillen”](#).

Toon wijzigingen als unified diff

Toon bestandswijzigingen in unified diff indeling. Dit context menu is alleen beschikbaar voor als *gewijzigd* gemarkeerde bestanden.

Openen / Openen met...

Open het geselecteerde bestand met het standaard programma voor dit bestandstype of met een zelf gekozen programma.

Verklaren...

Opent het verklaren venster, waarmee de geselecteerde revisie verklaard kan worden.

Revisie wijzigingen ongedaan maken

Make de wijzigingen van het geselecteerde bestand in die revisie ongedaan.

Weergeef eigenschappen

Bekijk de Subversion eigenschappen van het geselecteerde object.

Toon logboek

Toon het revisie logboek voor een enkel geselecteerd bestand.

Haal samenvoegboodschappen

Toon het revisie logboek voor een enkel geselecteerd bestand met de samengevoegde wijzigingen. Lees meer in [Paragraaf 4.9.6, “Samenvoeg Traceer Functies”](#).

Bewaar revisie in...

Sla de geselecteerde revisie op in een bestand, zodat je een oudere versie van dat bestand hebt.



Tip

Misschien is het je opgevallen dat we soms refereren naar wijzigingen en soms naar verschillen. Wat is het verschil?

Subversion gebruikt de revisienummers voor 2 doeleinden. Een revisie geeft in principe de staat van het archief weer op een zeker moment. Een revisie kan ook een set van wijzigingen aanduiden die zijn doorgevoerd op een vorige revisie waardoor de nieuwe revisie is gecreëerd. Bijvoorbeeld “Doorgevoerd in r1234” betekent dat met de wijzigingen die met r1234 zijn vastgelegd functie X gerealiseerd is. Om duidelijk te maken welke betekenis we bedoelen, maken we gebruik van twee termen.

Als je twee revisies N en M selecteert, dan geeft het context menu de mogelijkheid om de *verschillen* tussen de twee revisies te laten zien. In Subversion termen wordt dit aangegeven door `diff -r M:N`.

Als je alleen revisie N selecteert, dan geeft het context menu de mogelijkheid om de *wijzigingen* van die revisie te laten zien. In Subversion termen wordt dit aangegeven met `diff -r N-1:N` of `diff -c N`.

Het onderste deelvenster laat de bestanden zien die gewijzigd zijn in alle geselecteerde revisies, oftewel het context menu zal altijd de mogelijkheid geven om *wijzigingen* te laten zien.

4.9.4. Ophalen van uitgebreide logboek boodschappen

Het Logboek boodschappen venster geeft niet altijd alle in het verleden doorgevoerde wijzigingen weer met verschillende redenen:

- Bij een groot archief zijn er waarschijnlijk honderden of misschien wel duizenden wijzigingen. Binnenhalen van al die wijzigingen neemt meestal veel tijd in beslag, terwijl je waarschijnlijk alleen geïnteresseerd bent in de meest recente wijzigingen. Standaard worden er maximaal 100 logboek boodschappen opgehaald, maar je kunt deze waarde wijzigen bij TortoiseSVN → Instellingen (Paragraaf 4.30.1.2, “TortoiseSVN Instellingen Venster 1”).
- Als de **Stoppen bij kopie/hernoemd bestand** optie geselecteerd is, dan zal Toon Logboek stoppen op het punt waarop het geselecteerde bestand of de map vanaf een ander punt uit het archief gekopieerd is. Dit is handig bij het bekijken van takken (of labels), omdat er dan gestopt wordt bij de bron van de tak en er meteen een snel overzicht is van de wijzigingen in de tak.

Normaal gesproken zal je deze optie uit laten staan. TortoiseSVN onthoudt deze instelling, zodat jouw voorkeur bewaard blijft.

Als het Toon Logboek venster wordt aangeroepen vanuit het Samenvoegen venster, dan zal deze optie standaard geselecteerd zijn. Dit komt doordat bij het samenvoegen van takken er meestal gekeken wordt naar de wijzigingen van die specifieke tak. Verder terug kijken dan de bron van de tak heeft dan vaak geen nut.

Merk op dat Subversion hernoemen uitvoert als een kopieer/verwijder combinatie. Een hernoemde map of bestand zal er ook voor zorgen dat het logboek venster stopt als de Stoppen bij kopie/hernoemd bestand optie geselecteerd is.

Als je meer logboek boodschappen wilt zien, klik dan op de **Volgende 100** knop om de volgende 100 berichten op te halen. Je kunt dit zo vaak herhalen als je wilt.

Naast deze knop is er een multi-functionele knop. Hierbij wordt de laatst gebruikte optie onthouden, zodat je die weer kunt gebruiken. Klik op het pijltje om de andere opties te zien.

Gebruik **Toon Bereik** ... als je een bepaalde reeks van revisies wilt bekijken. Er zal dan een venster getoond worden, waarin je de start en eind revisie moet ingeven.

Gebruik de **Toon Alles** knop als je *alle* logboek boodschappen vanaf de HEAD revisie teruglopend tot aan revisie1 wilt zin.

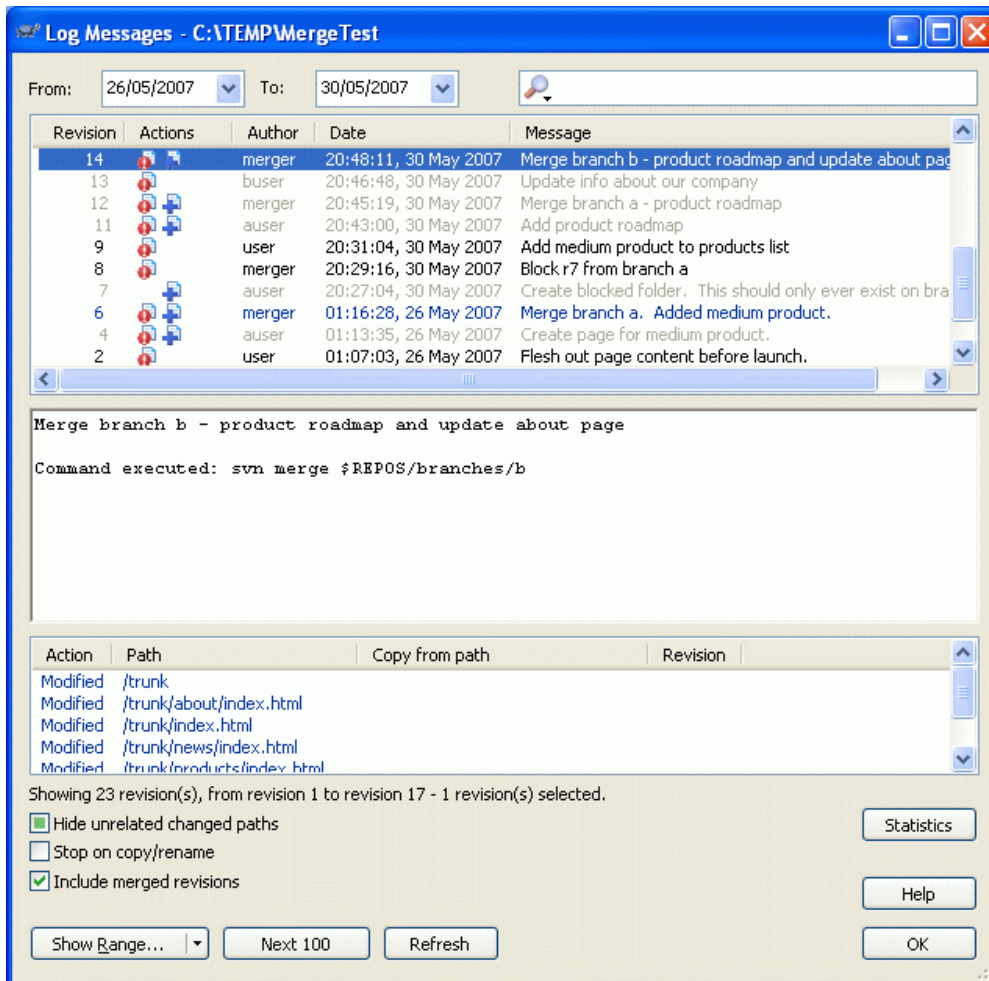
4.9.5. Huidige Werkkopie Revisie

Het logboek boodschappen venster toont de berichten van de HEAD revisie, niet van de huidige werkkopie revisie. Hierdoor komt het voor dat er berichten getoond worden voor bestanden en mappen die nog niet bijgewerkt zijn in je werkkopie. Om hier een duidelijker onderscheid in te maken, wordt het bericht van de vastlegging van jouw werkkopie revisie vetgedrukt weergegeven.

Als het logboek voor een map wordt weergegeven, dan is de gemarkeerde revisie de hoogste revisie die gevonden is in die map. Hiervoor wordt de hele werkkopie doorzocht. Dit gebeurt parallel in een aparte thread, zodat het weergeven van het revisie logboek geen vertraging oploopt. Als gevolg hiervan verschijnen de markering van de mappen niet direct.

4.9.6. Samenvoeg Traceer Functies

Subversion 1.5 en later slaat met behulp van eigenschappen de geschiedenis van Samenvoeg acties op. Hierdoor kun je een gedetailleerd overzicht krijgen van samengevoegde wijzigingen. Bijvoorbeeld, je ontwikkelt een nieuwe functie in een tak en voegt die later samen in de basislijn. In het logboek van de basislijn is de ontwikkeling van deze functie te zien als een enkele vastlegging, terwijl er misschien 1000 vastleggingen in de tak aan vooraf zijn gegaan.



Figuur 4.18. Het logboekboodschappen venster toont de revisies van samenvoegingen

Als je de details wilt zien welke revisies samengevoegd zijn bij die vastlegging, selecteer dan het **Samengevoegde revisies meenemen** selectievak. Hiermee zullen de logboekboodschappen opnieuw opgehaald worden en zullen de logboekboodschappen van de samengevoegde revisies tussen de andere logboekboodschappen ingevoegd worden. De samengevoegde revisies worden in grijs weergegeven, omdat deze wijzigingen aangegeven van een andere boomstructuur.

Samenvoegen is natuurlijk nooit eenvoudig. Gedurende de ontwikkelingen aan een tak zullen er soms samenvoegingen vanuit de basislijn gedaan moeten worden om ervoor te zorgen dat de tak bij blijft met de hoofdbroncode. Dat houdt in dat de samenvoeggeschiedenis van een tak ook een extra laag van samenvoeggeschiedenis zal hebben. Deze verschillende lagen worden met verschillende inspringingen weergegeven.

4.9.7. Logboekboodschap en Auteur Wijzigen

Revisie eigenschappen zijn compleet anders dan de eigenschappen van Subversion objecten. Revprops zijn beschrijvende objecten die gerelateerd zijn aan een specifiek revisienummer in het archief, zoals logboekboodschap, vastleggingsdatum en naam van de vastlegger (auteur).

Soms wil je een logboekboodschap wijzigen, omdat er bijvoorbeeld een tikfout in staat, de beschrijving wilt verbeteren of een ander soort wijziging wilt doorvoeren. Misschien wil je wel de auteur van de vastlegging wijzigen of omdat je authenticatie bent vergeten in te stellen or...

Subversion staat wijzigingen aan revisie eigenschappen in principe altijd toe. De mogelijkheid hiervoor is standaard uitgeschakeld, omdat wijzigingen niet ongedaan gemaakt kunnen worden (die wijzigingen worden niet met versies beheerd). Om deze wijzigingen mogelijk te maken, moet je een pre-revprop-change hook instellen. Lees het hoofdstuk over *Hook Scripts* [<http://svnbook.red-bean.com/en/1.5/svn.reposadmin.create.html#svn.reposadmin.create.hooks>] in het Subversion book voor meer informatie hierover. Lees ook **Paragraaf 3.3, “Server gesitueerde hook scripts”** voor meer informatie over het implementeren van hooks op een Windows machine.

Als je eenmaal je server het met de benodigde hooks geconfigureerd hebt, dan kun je met het context menu uit het bovenste deelvenster van het Log venster de auteur en logboekboodschap (of elke andere revprop) van een revisie wijzigen. De logboekboodschap kun je ook wijzigen via het context menu van het middelste deelvenster.



Waarschuwing

Omdat Subversion revisie eigenschappen niet van versiebeheer zijn voorzien, worden de oude waarden van een eigenschap (bijvoorbeeld de `svn:log` eigenschap) bij een wijzigingen voor *altijd* overschreven.

4.9.8. Filter op Logmeldingen

Als je de getoonde logboek berichten wilt beperken tot degene waarin jij geïnteresseerd bent, dan kun je hiervoor de instellingen boven aan het Logboekboodschappen venster gebruiken. Met de start- en einddatum instellingen kun je het datumbereik instellen. Met het zoekveld kun je de berichten laten zien die alleen een bepaalde term bevatten.

Klik op het zoeken pictogram om te selecteren welke informatie je wilt zoeken en om de *regex* mode te kunnen kiezen. Normaal gesproken heb je alleen een eenvoudige tekst zoekopdracht nodig, maar als je een flexibelere zoekmethode wilt gebruiken, dan kun je ook vaak gebruikte expressies gebruiken. Als je de muis boven een vak houdt, dan worden er in een tooltip tips gegeven voor het gebruik van regex functies. Je kunt online documentatie en een tutorial vinden op <http://www.regular-expressions.info/>. Het filter vergelijkt de zoekopdracht met logboek berichten en toont dan de berichten die *overeenkomen* met de zoekopdracht.

Om het filter in te stellen dat het alle logboek boodschappen toont die *niet* met het filter overeenkomen, begin dan met een uitroepteken (!). Bijvoorbeeld het filter `!username` toont alleen de objecten die niet vastgelegd zijn door `username`.

Merk op dat de filters alleen toegepast worden op de berichten die al zijn opgehaald. Hiermee wordt niet het ophalen van berichten uit het archief gefilterd.

Je kunt ook filteren op pad-namen in het onderste deelvenster door gebruik te maken van het **Verberg ongerelateerde wijzigingspaden** selectievak. Gerelateerde paden zijn de paden die gebruikt zijn om het logboek weer te geven. Als je het logboek voor een map ophaalt, dan houdt dat alles in die map en onderliggende mappen in. Bij een bestand gaat het dan alleen om het bestand. Dit selectievak heeft drie standen: je kunt alle paden tonen, ongerelateerde grijs maken, of ongerelateerde paden compleet verbergen.

In sommige gevallen vereist je werkmethode een bepaalde indeling van de logboekboodschappen. Dit kan tot gevolg hebben dat de beschrijvende tekst van de wijziging niet zichtbaar is in de samenvatting, die weergegeven wordt in het bovenste deelvenster. De eigenschap `tsvn:logsummary` kan gebruikt worden om een deel van de logboekboodschap te extraheren en in het bovenste deelvenster te laten tonen. Lees **Paragraaf 4.17.2, “TortoiseSVN Project Eigenschappen”** om te weten te komen hoe je deze eigenschap moet gebruiken.



Geen Logopmaak van de archief-verkenner

Aangezien de indeling afhankelijk is van het kunnen bereiken van subversion eigenschappen, zul je alleen de resultaten zien als je een opgehaalde werkkopie gebruikt. Eigenschappen op afstand ophalen is een trage actie, daarom zul je deze functie niet in actie zien vanuit de archief-verkenner.

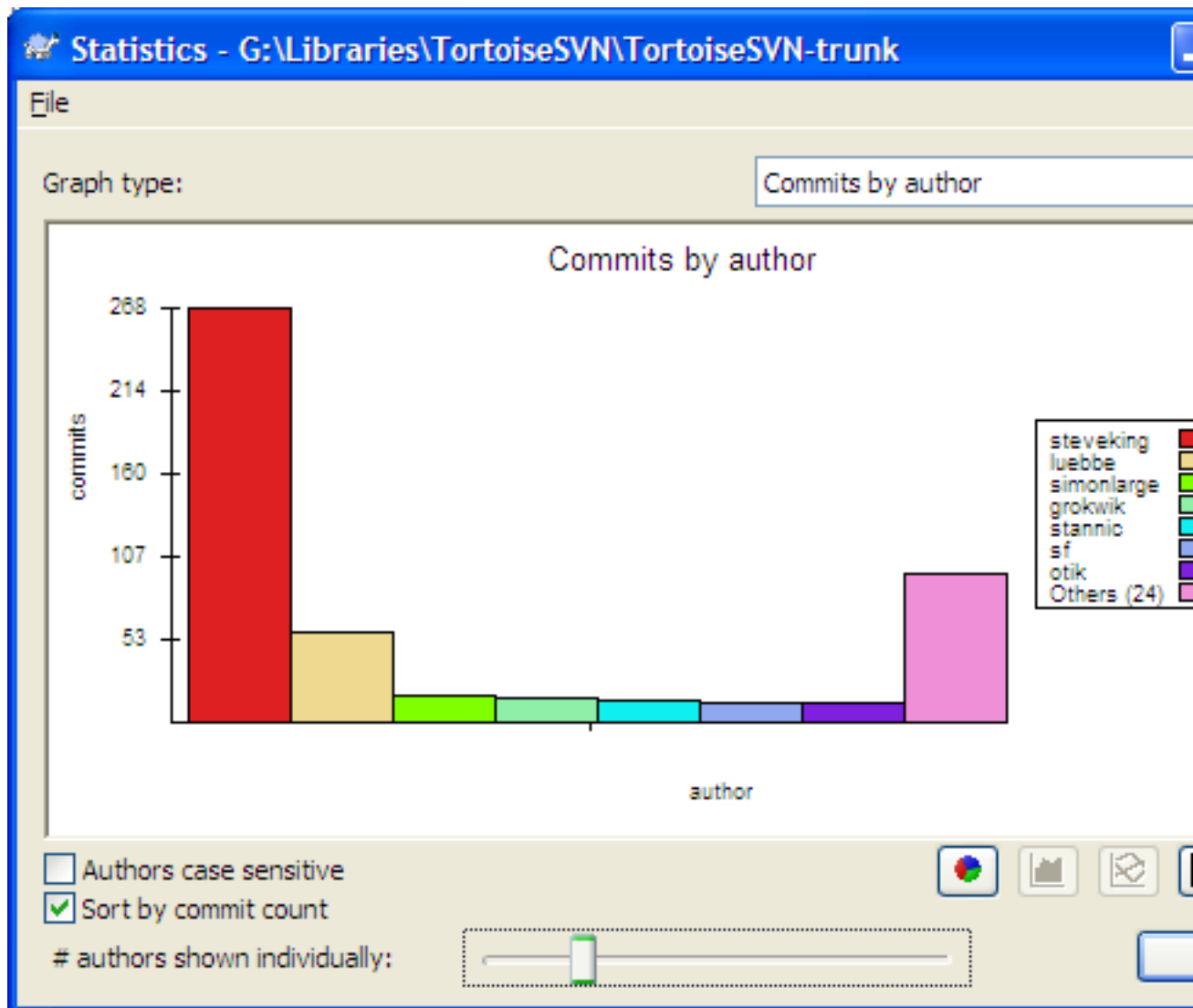
4.9.9. Statistische informatie

Met de **Statistics** knop kun je een scherm oproepen met interessante informatie over revisies die weergegeven worden in het logboekboodschappenvenster. In dit scherm wordt weergegeven hoeveel auteurs er aan het werk geweest zijn, hoeveel vastleggingen zijn hebben gedaan, de voortgang per week en nog veel meer. Nu kun je in één oogopslag zien wie het hardst heeft gewerkt en wie er loopt te treuzelen. ;-)

4.9.9.1. Statistiekenpagina

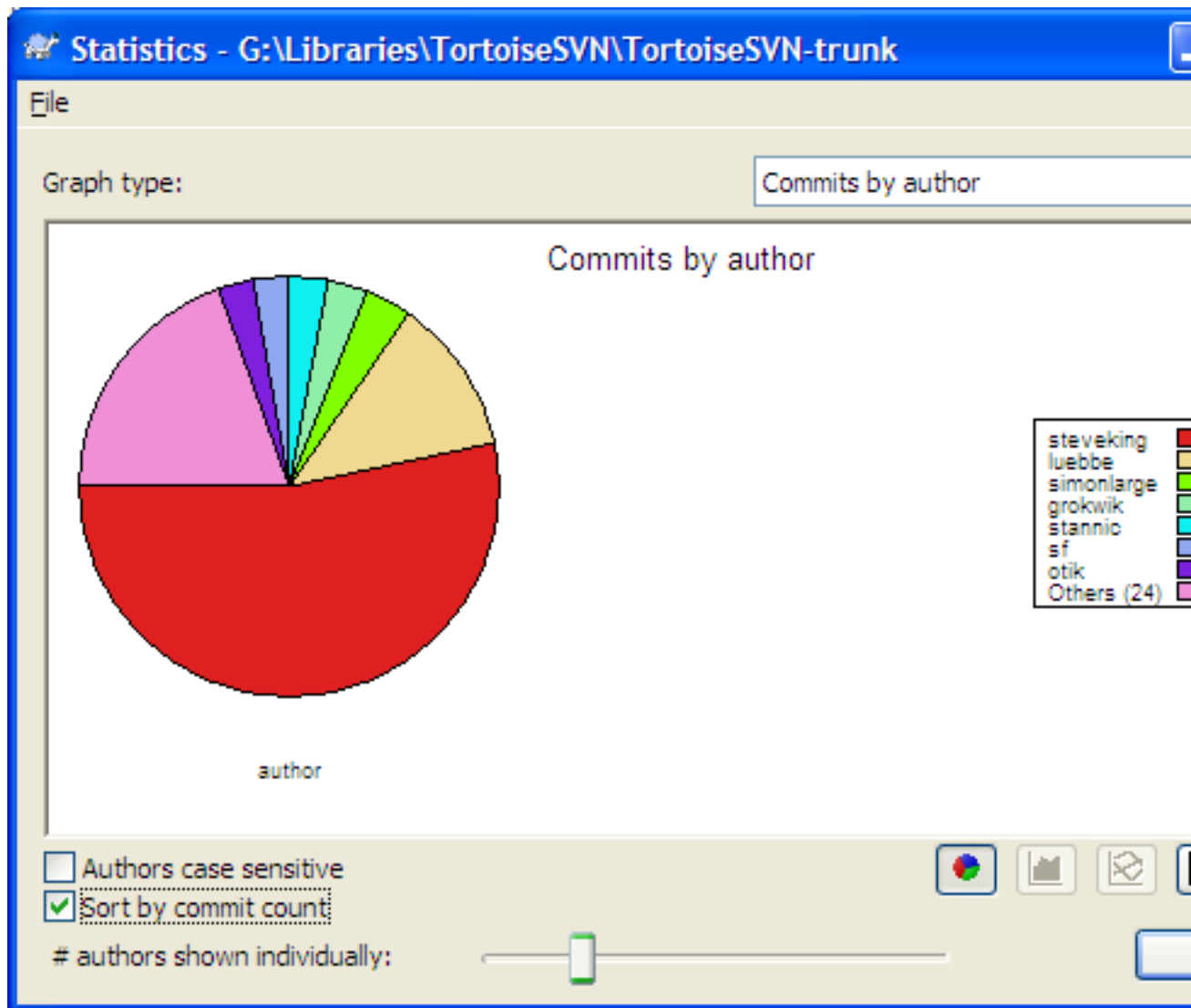
Op deze pagina vind je alle getallen die je maar kunt bedenken, maar in het bijzonder die van een bepaalde periode en aantal revisies die geselecteerd zijn en bijbehorende min/max/gemiddelde waardes.

4.9.9.2. Vastleggingen per Auteur



Figuur 4.19. Vastleggingen per Auteur Histogram

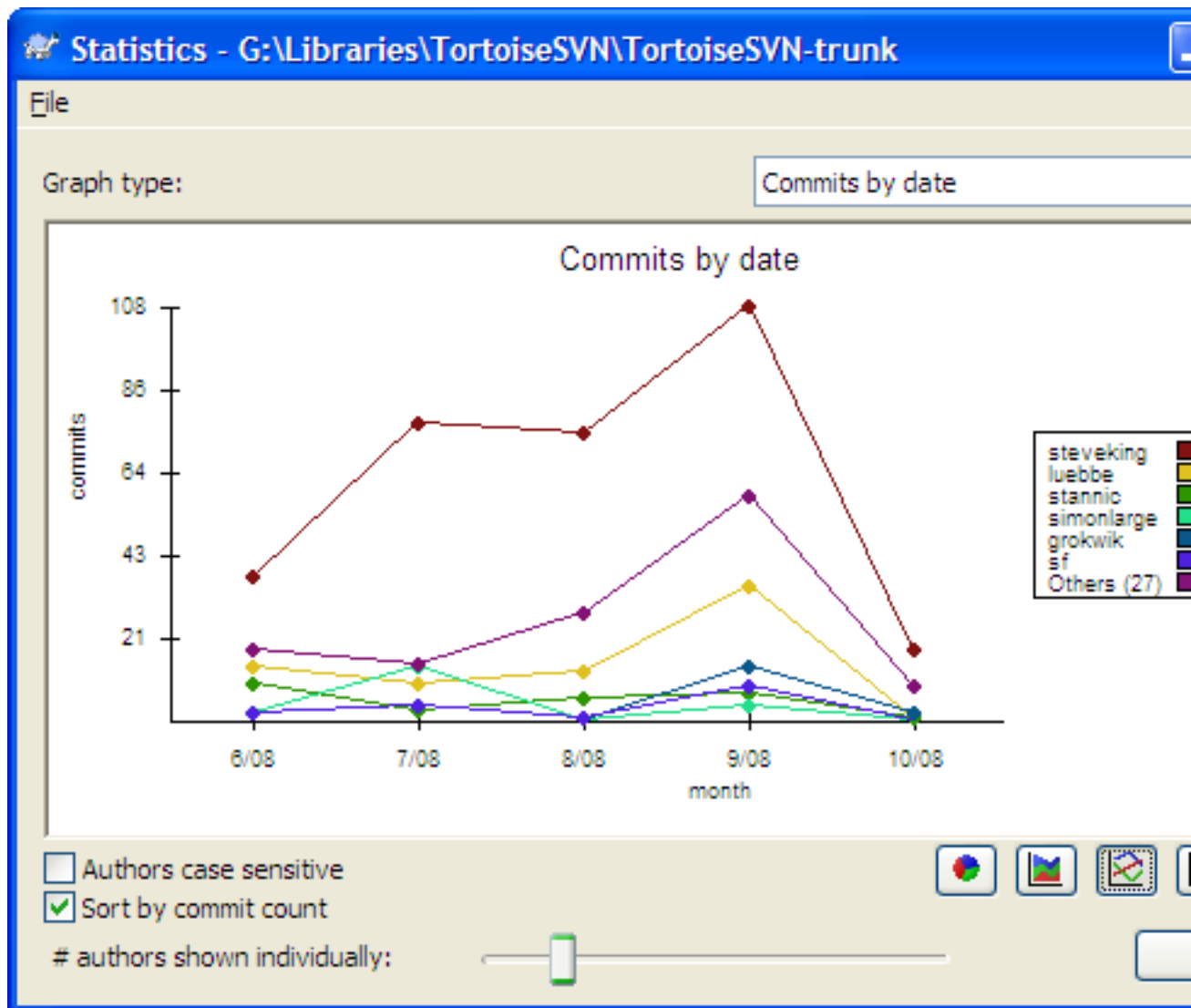
Dit figuur laat zien welke auteurs actief zijn geweest aan het project in de vorm van een eenvoudige histogram, gestapelde histogram, of taartdiagram.



Figuur 4.20. Vastleggingen-per-Auteur taartdiagram

Als er enkele zeer actieve auteurs zijn en vele minder actieve, dan wordt door het aantal kleine segmenten de grafiek snel onleesbaar. Met de schuifbalk onderaan het venster kun je de drempel bepalen (als percentage van het totaal aantal vastleggingen) waaronder activiteiten gegroepeerd moeten worden in een categorie met *Anderen*.

4.9.9.3. Vastleggingen per datum



Figuur 4.21. Vastleggingen-per-datum Grafiek

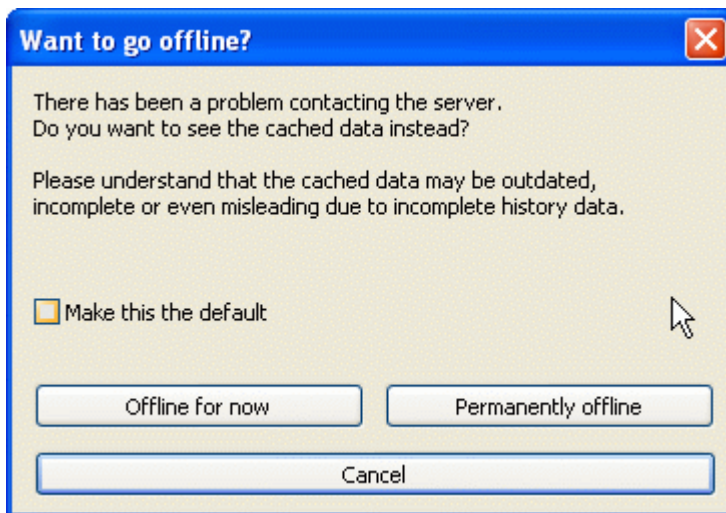
Deze grafiek is een grafische weergave van de project activiteiten in de zin van aantal vastleggingen *en* auteur. Hiermee kun je een idee krijgen wanneer er aan een project gewerkt wordt en wie er op welk moment aan werkt.

Er zullen veel lijnen in de grafiek weergegeven worden als er meerdere auteurs zijn. Er zijn hier twee weergavemodi beschikbaar: *normaal*, waarbij de activiteit van elke auteur ten opzichte van de nullijn wordt weergegeven, en *gestapeld*, waarbij de activiteit ten opzichte van de lijn eronder wordt weergegeven. Met de laatste modus voorkom je dat lijnen elkaar kruisen. Hiermee wordt de grafiek overzichtelijker, maar ook meteen lastiger om de individuele bijdrage te bekijken.

De analyse is standaard ingesteld als hoofdlettergevoelig, dus de gebruikers PeterEgan en PeteRegan worden als aparte auteurs behandeld. Echter zijn in veel gevallen de gebruikersnamen niet hoofdlettergevoelig en dus worden ze vaak inconsequent ingevoerd. In die gevallen wil je dat DavidMorgan en davidmorgan als dezelfde persoon behandeld worden. Gebruik het Exacte auteursnamen selectievak om dit in te stellen.

Merk op dat statistieken over dezelfde periode weergegeven worden zoals deze is ingesteld in het Logboek boodschappen venster. Als hierin maar één revisie wordt weergegeven, dan zullen de statistieken niet veel vertellen.

4.9.10. Offline Modus



Figuur 4.22. Schakel naar Offline werken dialoog

Als de server niet bereikbaar is en je hebt log caching aangezet, dan kun je het logboek boodschappen venster en de revisie grafieken in offline modus gebruiken. Hiervoor worden de gegevens uit de cache gebruikt, waardoor het mogelijk is om door te werken ook al is de informatie niet bijgewerkt of misschien zelfs incompleet.

Je hebt hier drie mogelijkheden:

Tijdelijk offline

Maak de huidige actie af in offline modus, maar probeer het archief opnieuw te bereiken als er weer informatie opgevraagd wordt.

Permanent offline

Blijf in offline modus werken, totdat er een specifieke actie op het archief gevraagd wordt. Lees ook [Paragraaf 4.9.11, "De weergave verversen"](#).

Annuleren

Als je niet verder wilt gaan met de actie met mogelijk achterhaalde informatie, selecteer dan annuleren.

Met het **Maak dit de standaard** selectievak voorkom je dat dit venster telkens weer terug komt en selecteert altijd de optie die je eerder hebt gekozen. Je kunt je keuze later altijd nog wijzigen via TortoiseSVN → Instellingen.

4.9.11. De weergave verversen

Als je wilt kijken of er nieuwe logboek boodschappen zijn op de server, dan kun je het beeld eenvoudig verversen door op **F5** te drukken. Als je gebruikt maakt van de log cache (standaard ingeschakeld), dan zal het archief op nieuwe berichten worden gecontroleerd en worden alleen de nieuwe berichten opgehaald. Als de log cache offline was, dan wordt er ook meteen geprobeerd om weer online te gaan.

Als je de log cache gebruikt en je vermoed dat de inhoud van een bericht of de auteur gewijzigd is, dan kun je **Shift-F5** of **Ctrl-F5** gebruiken om de weergegeven berichten opnieuw van de server op te halen en de log cache bij te werken. Merk op dat hiermee alleen de getoonde berichten worden verversen, de rest van de cache van het archief wordt hierdoor niet gewijzigd of gecontroleerd.

4.10. Tonen van Verschillen

Een van de meest voorkomende eisen in de ontwikkeling van projecten is om te kunnen zien wat er is veranderd. Je zou bijvoorbeeld de verschillen tussen twee herzieningen van hetzelfde bestand, of de verschillen tussen twee aparte bestanden willen bekijken. TortoiseSVN biedt een ingebouwd gereedschap genaamd TortoiseMerge om de verschillen tussen tekstbestanden te kunnen bekijken. Voor het vergelijken van afbeeldingen heeft TortoiseSVN ook een gereedschap genaamd TortoiseIDiff. Natuurlijk kun je hier ook je eigen favoriete programma voor gebruiken als je wilt.

4.10.1. Bestandsverschillen

Lokale wijzigingen

Als je wilt bekijken welke veranderingen *jij* gedaan hebt in je werkkopie, gebruik je simpelweg het context menu van de verkenner en selecteert TortoiseSVN → **Verschil**.

Verschil met een andere tak/label

Als je wilt zien wat er veranderd is in de basislijn (als je aan een tak werkt) of in een specifieke tak (als je werkt op de basislijn), kun je gebruik maken van het context menu van de verkenner. Je houdt gewoon de **Shift**-toets ingedrukt terwijl je met de rechtermuisknop op het bestand klikt. Vervolgens selecteer je TortoiseSVN → **Verschil met URL**. In het dialoogvenster dat volgt, geef je de URL in het archief aan waarmee je het lokale bestand wilt vergelijken.

Je kunt ook gebruik maken van de archief-verkenner. Selecteer hier twee bomen, twee labels, of een tak/label en de basislijn om te vergelijken. Via het daar aanwezige contextmenu kun je ze vergelijken met **Vergelijk revisies**. Lees meer in [Paragraaf 4.10.3, “Vergelijken van mappen”](#).

Verschil met een vorige revisie

Als je het verschil wilt zien tussen een bepaalde revisie en je werkkopie, gebruik je het Logboek dialoogvenster, selecteer je de gewenste revisie, en kies je vervolgens **Vergelijk met werkkopie** uit het contextmenu.

Als je het verschil wilt zien tussen de laatst vastgelegde revisie en je werkkopie, ervan uitgaande dat de werkkopie niet is gewijzigd, klik je met de rechtermuisknop op het bestand. Vervolgens selecteer je TortoiseSVN → **Vergelijk met vorige revisie**. Dit zal een vergelijking uitvoeren tussen de revisie voor de laatste-vastleg-datum (zoals vastgelegd in je werkkopie) en de werkbasis. Deze toont je de laatste wijziging uitgevoerd op het bestand welke nodig is om deze in de staat te brengen welke je in je werkkopie ziet. Het zal geen wijzigingen tonen die nieuwer zijn dan je werkkopie.

Verschil tussen twee eerdere revisies

Als je het verschil wil zien tussen twee revisies die al vastgelegd zijn, gebruik dan het Revisie Log dialoog en selecteer de twee revisies die je wil vergelijken (met behulp van de gebruikelijke **Ctrl**-wijziger). Selecteer daarna **Vergelijk revisies** uit het contextmenu.

Als je dit gedaan hebt vanuit het revisie log van een map, verschijnt er een **Vergelijk Revisies** dialoog welke een lijst toont met gewijzigde bestanden uit deze map. Lees meer in [Paragraaf 4.10.3, “Vergelijken van mappen”](#).

Alle wijzigingen die zijn gemaakt in een vastlegging

Wanneer je de wijzigingen van alle bestanden in een bepaalde revisie in één weergave wil zien, kun je hier de Unified-Diff uitvoer (GNU patch formaat) voor gebruiken. Deze toont alleen de wijzigingen met een aantal regels voor de context. Het is moeilijker te lezen als een visuele vergelijking, maar het toont alle wijzigingen bij elkaar. Selecteer vanuit het revisie log dialoog de gewenste revisie en selecteer dan **Toon verschillen als unified diff** uit the contextmenu.

Verschillen tussen bestanden

Als je de verschillen tussen twee verschillende bestanden wil bekijken, dan kun je dat rechtstreeks via de verkenner doen door beide bestanden te selecteren (gebruik makend van de standaard **Ctrl**-wijziger). Selecteer daarna uit het verkenner contextmenu TortoiseSVN → **Verschil**.

Verschil tussen een werkkopie bestand/map en een URL

Als je de verschillen wil zien tussen een bestand in je werkkopie en een bestand in een willekeurig Subversion archief, kun je dat rechtstreeks vanuit de verkenner doen door het bestand te selecteren

en terwijl je de **Shift** ingedrukt houdt met de rechter muisknop erop klikt zodat het contextmenu verschijnt. Selecteer dan TortoiseSVN → Verschil met URL. Je kan hetzelfde doen met een folder uit je werkkopie. TortoiseMerge toont deze verschillen op dezelfde manier als patch bestanden worden getoond - een lijst met gewijzigde bestanden die je in één keer kan zien.

Verschil met verklaringen

Als je niet alleen de verschillen, maar ook de auteur, de revisie en de wijzigingen die gemaakt zijn wil zien, kun je de verschil en verklaar rapporten vanuit het revisie log dialoog combineren. Ga voor meer informatie naar [Paragraaf 4.23.2, “Verklaar Verschillen”](#).

Verschillen tussen mappen

Met de ingebouwde gereedschappen die met TortoiseSVN worden meegeleverd is het niet mogelijk om verschillen tussen map hiërarchieën te bekijken, maar als je een extern gereedschap hebt welke dat ondersteund, kun je deze hier ook voor gebruiken. In [Paragraaf 4.10.5, “Externe Verschil/Samenvoeg gereedschappen”](#) worden een aantal gereedschappen beschreven die wij hebben gebruikt.

Als je een verschil gereedschap van een externe partij hebt geconfigureerd, kun je **Shift** gebruiken bij het selecteren van het Verschil commando om het alternatieve gereedschap te gebruiken. Lees [Paragraaf 4.30.5, “Externe Programma Instellingen”](#) voor meer informatie over het configureren van alternatieve verschil gereedschappen.

4.10.2. Opties voor regeleinden, spaties en witregels

Gedurende de loop van een project, zou je kunnen beslissen om de regeleinden te veranderen van CRLF naar LF, of je zou het inspringen van een sectie kunnen aanpassen. Dit zal jammer genoeg een groot aantal bestanden markeren als gewijzigd, ook al is er geen verandering in de betekenis van de code. De opties hieronder helpen je om deze wijzigingen te hanteren tijdens het vergelijken en toepassen van verschillen. Je komt deze instellingen tegen in de **Samenvoeg** en **Verklaar** dialogen, maar ook bij de instellingen van TortoiseMerge.

Negeer regeleinden negeert wijzigingen die uitsluitend veroorzaakt worden door een verschil in stijl van regeleinden.

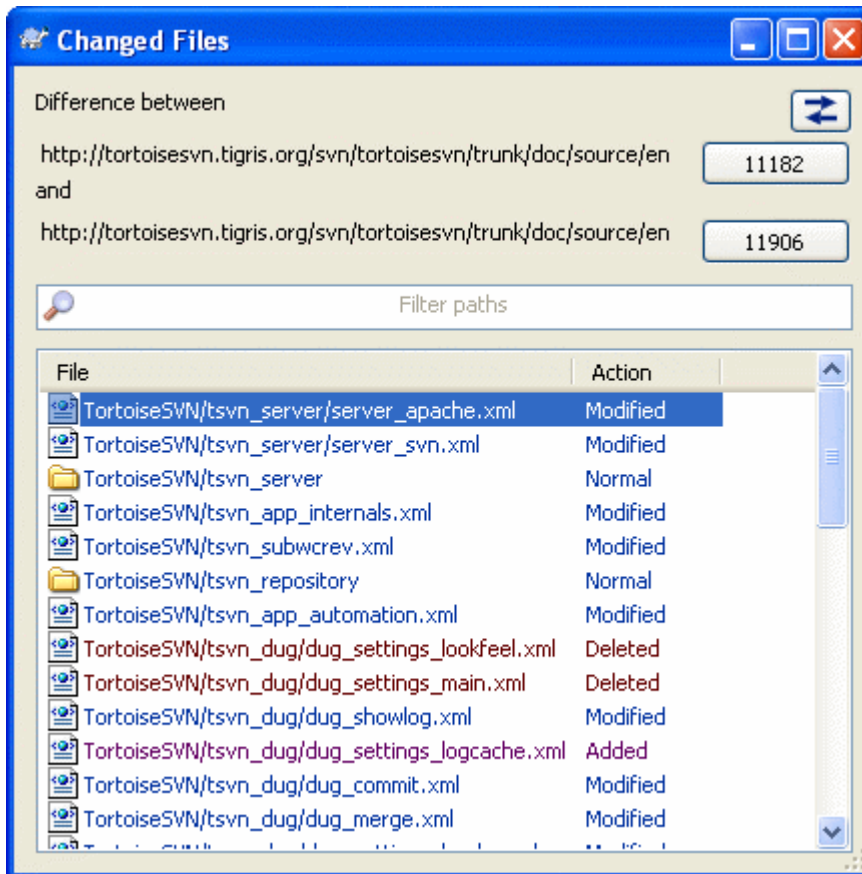
Vergelijk witregels toont alle wijzigingen in inspringen en witregels als toegevoegde/verwijderde regels.

Negeer wijzigingen in spaties en lege regels sluit de wijzigingen uit die louter door een verandering van het aantal of het type spatie/lege regel worden veroorzaakt, zoals bijvoorbeeld een verandering van de inspringing, of het veranderen van tabs naar spaties. Toevoegen van een spatie of lege regel waar er geen was, of volledig verwijderen van een spatie of lege regel wordt nog steeds getoond als een wijziging.

Alle spaties en witregels negeren sluit alles uit waar alleen wijzigingen aan spaties en/of witregels zijn gemaakt.

Natuurlijk wordt elke regel met gewijzigde inhoud altijd aan het verschil toegevoegd.

4.10.3. Vergelijken van mappen



Figuur 4.23. De Vergelijk Revisies Dialoog

Als je twee bomen selecteert in de archief verkenners, of wanneer je twee revisies van een map selecteert in het log dialoog, kun je Contextmenu → Vergelijk revisies.

Deze dialoog toont een lijst met alle bestanden die zijn veranderd en staat je toe om deze individueel te vergelijken of te verklaren via het context menu.

Je kan een *wijzigings boom* exporteren, welke handig kan zijn als je iemand anders de boomstructuur van je project wilt sturen, maar daar alleen de gewijzigde bestanden in wilt hebben. Deze operatie werkt alleen op de geselecteerde bestanden, dus dien je de gewenste bestanden - meestal betekent dit alle bestanden - te selecteren en daarna Contextmenu → Exporteer selectie naar.... Je krijgt de vraag op welke locatie je deze wijzigingsboom wilt opslaan.

Je kan de *lijst* met gewijzigde bestanden ook exporteren naar een tekstbestand via Contextmenu → Sla lijst van geselecteerde bestanden op in....

Als je de lijst met bestanden *en* de bijbehorende acties (gewijzigd, toegevoegd, verwijderd) wil exporteren, kun je dat doen via Contextmenu → Naar klembord kopiëren.

Met de knop bovenaan kun je de richting waarin vergeleken wordt aanpassen. Je kan de wijzigingen zien die nodig zijn om van A naar B te komen, maar als je dat liever hebt, van B naar A.

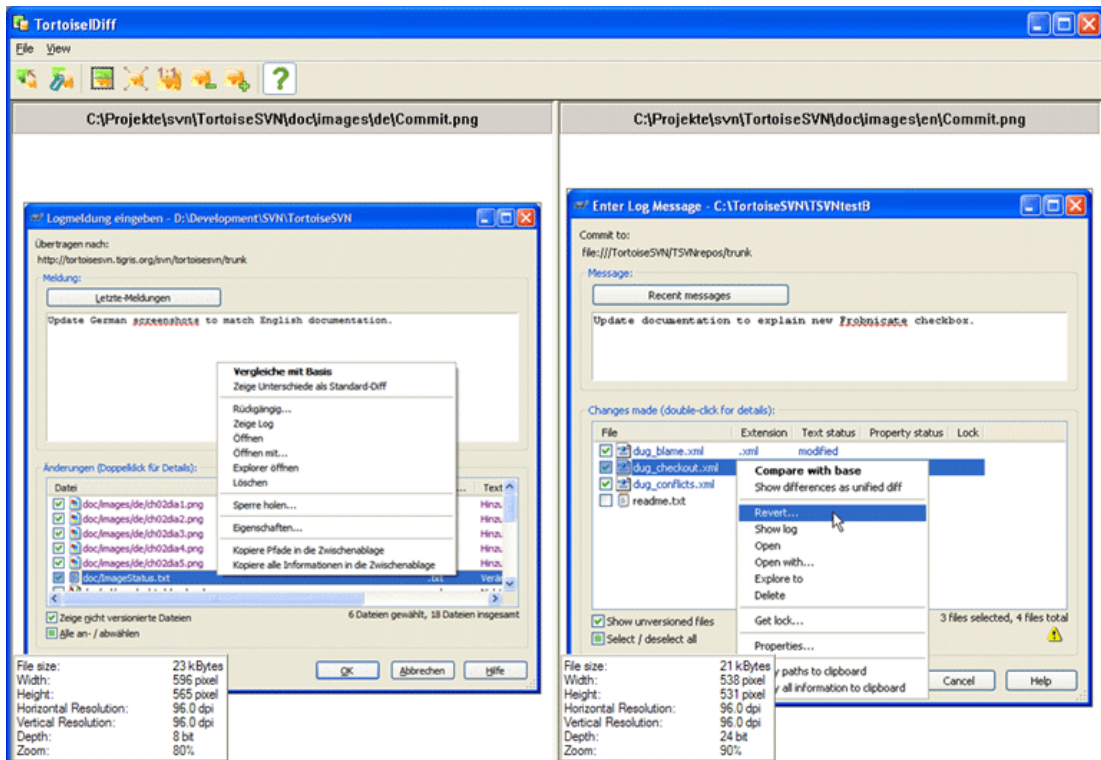
De knoppen met de revisie nummers erop kunnen gebruikt worden om het revisie bereik te veranderen. Als je dit bereik verandert, zal de lijst met items die verschillen tussen deze twee revisies automatisch ververs worden.

Als de lijst met bestandsnamen erg lang is, kun je het zoekvak gebruiken om de lijst te reduceren tot bestandsnamen die een specifieke tekst bevatten. Bedenk hierbij dat een simpele manier van zoeken op

tekst wordt gebruikt, dus als je de lijst wil beperken tot C bron bestanden, zul je moeten zoeken op `.c` in plaats van `*.c.`"

4.10.4. Vergelijk afbeeldingen met TortoiseIDiff

Er zijn veel gereedschappen beschikbaar waarmee verschillen tussen tekst bestanden bekeken kunnen worden, inclusief onze eigen TortoiseMerge, maar vaak willen we ook kunnen zien hoe een afbeelding veranderd is. Daarom hebben we TortoiseIDiff gemaakt.



Figuur 4.24. De beeld-verschil viewer

TortoiseSVN → Verschil uitvoeren op één van de meest voorkomende formaten voor afbeeldingen zal TortoiseIDiff starten om de verschillen tussen afbeeldingen te tonen. Standaard zullen de afbeeldingen naast elkaar worden getoond, maar je kan via het View menu of de werkbalk instellen dat ze boven elkaar getoond worden. Als je dat wilt, kun je ze zelfs over elkaar leggen en doen of je een lightbox gebruikt.

Natuurlijk kan je ook in en uit zoomen en de afbeelding verplaatsen. De afbeelding kan ook eenvoudig verplaatst worden door te slepen met de linker muisknop. Als je de optie Koppel afbeeldingsposities selecteert, dan worden de verplaats regelaars (schuifbalken, muiswiel) voor beide afbeeldingen gekoppeld.

Een afbeeldingsinformatie scherm toont details over de afbeelding, zoals de grootte in pixels, resolutie en kleurdiepte. Als dit scherm in de weg zit, gebruik dan Beeld → Afbeelding Info om het te verbergen. Je krijgt dezelfde informatie in een tooltip wanneer je de muis over de titel van de afbeelding laat zweven.

Als de afbeeldingen over elkaar zijn gelegd, wordt de intensiteit van de afbeeldingen (alpha mix) bestuurd met de schuifbalk aan de linkerkant. Je kan op een willekeurige plek op de schuifbalk klikken om de mix direct te veranderen. Ook kan je de balk verschuiven om de mix interactief te veranderen.

De knop boven de schuifbalk schakelt tussen 0% mix en 100% mix. Als je dubbel klikt op de knop, zal de mix elke seconde automatisch schakelen totdat je nog een keer op de knop klikt. Dit kan handig zijn als je verschillende kleine wijzigingen zoekt.

Soms wil je een verschil zien in plaats van een mix. Je zou bijvoorbeeld van twee herzieningen van een afbeelding voor een printplaat willen zien welke paden zijn gewijzigd. Als je de alpha-mix modus uitzet, zal het verschil getoond worden als een *XOR* van de kleur van de pixel waarden. Ongewijzigde gebieden zullen getoond worden als wit en veranderingen zullen worden gekleurd.

4.10.5. Externe Verschil/Samenvoeg gereedschappen

Als de gereedschappen die wij aanbieden niet doen wat je nodig hebt, kun je één van de vele beschikbare open-source of commerciële programma's gebruiken. Iedereen heeft zo zijn eigen favoriet, en deze lijst is verre van compleet, maar hier zijn er een aantal die je zou kunnen overwegen:

WinMerge

WinMerge [<http://winmerge.sourceforge.net/>] is een handig open-source verschil gereedschap welke ook mappen kan vergelijken.

Perforce Merge

Perforce is een commercieel RCS, maar je kan het verschil/samenvoeg gereedschap gratis gebruiken. Ga voor meer informatie naar *Perforce* [<http://www.perforce.com/perforce/products/merge.html>].

KDiff3

KDiff3 is een gratis verschil gereedschap welke ook mappen kan vergelijken. Je kan het *hier* [<http://kdiff3.sf.net/>] downloaden.

ExamDiff

ExamDiff Standard is freeware. Het kan omgaan met bestanden, maar niet met mappen. ExamDiff Pro is shareware en voegt een aantal 'goodies' toe waaronder het vergelijken van mappen en een bewerken functionaliteit. Beide smaken, versie 3.2 en hoger, kunnen omgaan met unicode. Je kan ze downloaden van *PrestoSoft* [<http://www.prestosoft.com/>].

Beyond Compare

Dit is een uitstekend shareware diff gereedschap, vergelijkbaar met ExamDiff Pro, dat kan omgaan met verschillen in mappen en unicode. Download het van *Scooter Software* [<http://www.scootersoftware.com/>].

Araxis Merge

Araxis Merge is een handig commercieel gereedschap voor het vergelijken en samenvoegen van zowel bestanden als mappen. Het voert een drie-weg vergelijking uit bij het samenvoegen en heeft synchronisatie links die gebruikt kunnen worden als je de volgorde van functies hebt veranderd. Download het van *Araxis* [<http://www.araxis.com/merge/index.html>].

SciTE

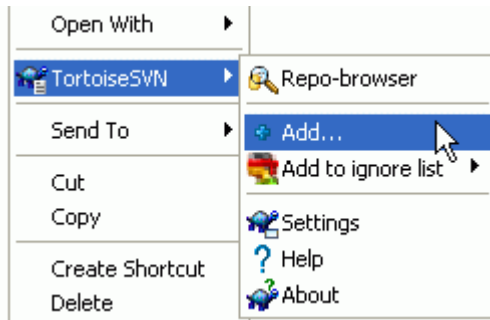
Deze tekstverwerker bevat syntax inkleuring voor unified diffs, waardoor ze veel makkelijker te lezen zijn. Download het programma via *Scintilla* [<http://www.scintilla.org/SciTEDownload.html>].

Notepad2

Notepad2 is ontworpen als alternatief voor het standaard Windows programma Kladblok en is gebaseerd op het Scintilla open-source edit component. Het programma is niet alleen goed te gebruiken voor het uitvoeren van unified diffs, maar is in veel gevallen ook veel beter dan het Windows Kladblok. Download *hier* [<http://www.flos-freeware.ch/notepad2.html>] het gratis programma.

Lees **Paragraaf 4.30.5, "Externe Programma Instellingen"** voor informatie over het gebruiken van deze gereedschappen binnen TortoiseSVN."

4.11. Nieuwe Bestanden En Mappen Toevoegen



Figuur 4.25. Verkenner context menu voor bestanden zonder versiebeheer

Als je tijdens het ontwikkelproces nieuwe bestanden en/of mappen hebt gemaakt, moeten die ook onder broncodebeheer worden gebracht. Selecteer de bestanden en/of de mappen, en ga naar TortoiseSVN → Toevoegen.

Nadat je de bestanden of mappen onder versiebeheer hebt gebracht, verschijnt er een icoontje in de vorm van een plusje over het normale icoontje in Windows Verkenner. Dit betekent dat je de werkkopie van het bestand moet vastleggen voordat de bestanden/mappen beschikbaar zijn voor andere ontwikkelaars. Een bestand of map toevoegen heeft *geen* effect op het archief!



Vele toevoegingen

Je kunt het commando 'Toevoegen' ook gebruiken op reeds geversioneerde mappen. In dat geval zal het toevoegen-dialogscherm alle niet-geversioneerde bestanden in die geversioneerde mappen weergeven. Dit is handig als je veel nieuwe bestanden hebt en die allemaal in één keer wilt toevoegen.

Bestanden van buiten je werkkopie kun je daaraan toevoegen door middel van de volgende drag-and-drop-handeling:

1. selecteer de bestanden die je wilt toevoegen
2. sleep ze met de rechter muisknop naar de nieuwe locatie in je huidige werkkopie
3. laat de rechter muisknop los
4. ga naar Contextmenu → SVN Kopiëren en toevoegen aan deze werkkopie. De bestanden worden dan naar de werkkopie gekopieerd en onder versiebeheer gebracht.

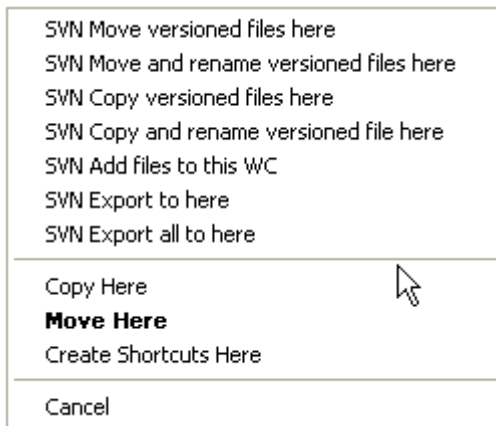
Ook binnen een werkkopie kun je bestanden toevoegen. Dit doe je door de bestanden (of mappen) met de linkermuisknop op te pakken, naar het vastleggen-dialogscherm te slepen en ze vervolgens los te laten.

Het toevoegen van bestanden en mappen kun je, als je bijvoorbeeld per ongeluk een verkeerd bestand hebt toegevoegd, ongedaan maken voordat je je wijzigingen vastlegt. Dit doe je door het volgende uit te voeren: TortoiseSVN → Toevoeging opheffen....

4.12. Kopiëren/Verplaatsen/Hernoemen Bestanden en Mappen

Het komt vaak voor dat u de benodigde bestanden al heeft in een ander project in het archief, en u ze eenvoudigweg daarvandaan wilt kopiëren. U zou de bestanden eenvoudig kunnen kopiëren en kunnen toevoegen zoals hierboven beschreven, maar dat biedt u geen geschiedenis. En als u vervolgens een fout oplost in de originele bestanden, kunt u de oplossing alleen nog automatisch samenvoegen indien de nieuwe kopie is gerelateerd aan het origineel in Subversion.

De eenvoudigste wijze om bestanden en folders te kopiëren vanuit een werkkopie is door gebruik te maken van het rechts-sleep menu. Indien u een bestand of folder met rechts-sleept van de ene werkkopie naar de andere, of zelfs binnen dezelfde folder, verschijnt er een contextmenu wanneer u de muis loslaat.



Figuur 4.26. Rechts-slepen-menu voor een map met versiebeheer

Nu kunt u de bestaande geversioneerde inhoud naar een nieuwe locatie kopiëren, en deze waarschijnlijk tegelijk hernoemen.

U kunt ook bestanden met versiebeheer binnen een werkkopie kopiëren of verplaatsen, of tussen twee werkkopieën, met behulp van de gebruikelijke knip-en-plak methode. Gebruik de standaard Windows Kopiëren of Knippen om een of meerdere geversioneerde items naar het klipbord te kopiëren. Als het klipbord zulke geversioneerde items bevat, kunt u vervolgens TortoiseSVN → Plakken gebruiken (noot: niet de standaard Windows Plakken) om de items naar de nieuwe werkkopie locatie te kopiëren of verplaatsen.

U kunt bestanden en mappen van uw werkkopie naar een andere locatie in het archief kopiëren door middel van TortoiseSVN → Tak/Label. Raadpleeg [Paragraaf 4.19.1, “Aanmaken van Tak of Label”](#) voor meer informatie.

U kunt een oudere versie van een bestand of map vinden in het log dialoog en deze direct vanuit het log dialoog kopiëren naar een nieuwe locatie in het archief met behulp van Context menu → Maak tak/label van revisie. Raadpleeg [Paragraaf 4.9.3, “Extra Informatie Opvragen”](#) voor meer informatie.

U kunt ook de archief-verkenner gebruiken om de gewenste inhoud te vinden, en deze direct vanuit het archief naar uw werkkopie kopiëren, of tussen twee locaties binnen het archief te kopiëren. Raadpleeg [Paragraaf 4.24, “De Archief-verkenner”](#) voor meer informatie.

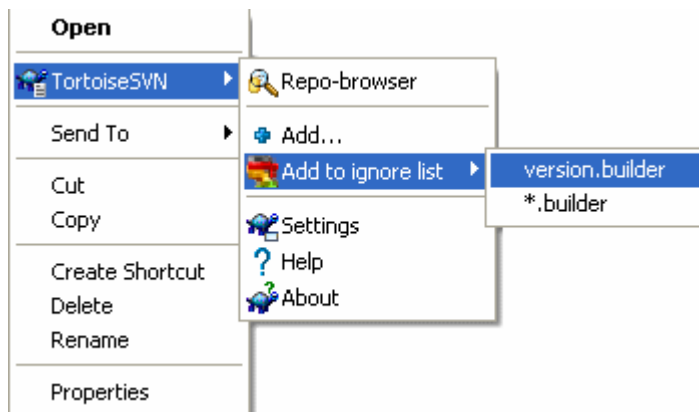


Kan niet tussen archieven kopiëren

Hoewel je bestanden en mappen kunt kopiëren of verplaatsen *binnen* in een archief, kun je bestanden en mappen *niet* kopiëren tussen twee archieven zonder dat de geschiedenis verloren gaat. Zelfs niet als de archieven op dezelfde server staan. Het enige wat je kunt doen is de inhoud in de huidige staat kopiëren en als nieuwe inhoud aan het tweede archief toevoegen.

Als u er niet zeker van bent of twee URL's op dezelfde server naar dezelfde of verschillende archieven verwijzen, gebruik dan de archiefverkenner om één URL te openen en bepaal hiermee waar de bron van het archief is. Als u beide locaties kunt herkennen in de één archiefverkennervenster, dan bevinden ze zich in hetzelfde archief.

4.13. Negeren bestanden en mappen



Figuur 4.27. Verkenner context menu voor bestanden zonder versiebeheer

In de meeste projecten zijn er ook bestanden of mappen die je niet onder controle van subversion wilt brengen. Denk bijvoorbeeld aan compiler bestanden, `*.obj`, `*.lst`, of output bestanden en backup versies. Als je wijzigingen commit, laat TortoiseSVN jou de bestanden zien die (nog) niet meegenomen zullen worden. Deze kunnen je dialoog box vullen. Natuurlijk je de weergave ervan uitzetten, maar dan zou je belangrijke bestanden kunnen gaan vergeten.

De beste manier om deze problemen te vermijden is om een bestanden en/of extensies aan de negerlijst toe te voegen. Zo kunnen ze nooit per ongeluk gecommited worden of de weergave in de commit dialog box overvol maken zodat je andere belangrijke bestanden voor versie beheer zou vergeten.

Als je met rechts klikt op een nog niet beheerd bestand, en het commando TortoiseSVN → Voeg toe aan negerlijst, krijg je een submenu waarmee alleen dat bestand of alle bestanden met diezelfde extensie kunt selecteren. Indien je meerdere bestanden hebt geselecteerd, krijg je het submenu niet te zien en neger je alleen de geselecteerde bestanden en mappen.

Wil je items van je negerlijst afhalen, selecteer dan die bestanden of mappen, rechts klik op de selectie en kies TortoiseSVN → Verwijder van negerlijst. Voor mappen kun je ook de `svn:ignore` eigenschap direct zetten. Daarmee kun je algemene patronen specificeren van te negeren bestanden, zoals hieronder beschreven zal worden. Lees [Paragraaf 4.17, “Project Instellingen”](#) voor meer informatie over het zetten van eigenschappen op mappen. Let op: iedere negerpatroon moet op zijn eigen regel komen te staan. Scheiden met spaties werkt niet.



De Globale Negeerlijst

Een andere methode om bestanden te negeren is door ze toe te voegen aan de *globale negerlijst*. Het grote verschil is dat de globale negerlijst een eigenschap van de gebruiker is. Deze lijst wordt toegepast op *alle* Subversion projecten, maar dan alleen die op de PC van de gebruiker. Over het algemeen is het beter om de `svn:ignore` eigenschap te gebruiken waar mogelijk, omdat het toegepast kan worden op specifieke project gebieden en het wordt gebruikt bij iedereen die het project ophaalt. Lees [Paragraaf 4.30.1, “Algemene instellingen”](#) voor meer informatie.



Negeren van geversioneerde items

Bestanden en mappen die al onder versiebeheer vallen kunnen nooit genegeerd worden. Dat zou subversion tegenspreken. Mocht je een bestand per ongeluk hebben toegevoegd, lees

dan **Paragraaf B.8, "Negeer bestanden die al opgenomen zijn in het versiebeheer"** om dit ongedaan te maken.

4.13.1. Patroonherkenning in negerlijsten

Subversion's neger patronen maken gebruik van globbing bij bestandsnamen, een techniek die van origine gebruikt is in Unix om bestanden aan te wijzen met meta-karakters als wildcards. De volgende karakters hebben een speciale betekenis:

*

Matcht elke string van karakters, inclusief de lege string.

?

Matcht elk enkel letterteken.

[...]

Zoekt overeenkomsten met elk van de karakters welke zijn omsloten door blokhaken. Binnen de blokhaken, een karakterpaar gescheiden door "-" komen overeen met ieder karakter dat lexicaal tussen beide karakters in ligt. Bijvoorbeeld [AGm-p] komt overeen met elk van de karakters A, G, m, n, o of p.

Patroonherkenning is hoofdletter gevoelig, dus let op onder Windows. Je kunt ongevoeligheid voor hoofd- of kleine letters afdwingen door expliciet en uitputtend letterparen te specificeren, bv. om *.tmp te negeren gebruik je het patroon *. [Tt] [Mm] [Pp].

Voor de officiële definitie van globbing, raadpleeg de IEEE specificaties voor de 'shell command language' *Pattern Matching Notation* [http://www.opengroup.org/onlinepubs/009695399/utilities/xcu_chap02.html#tag_02_13].

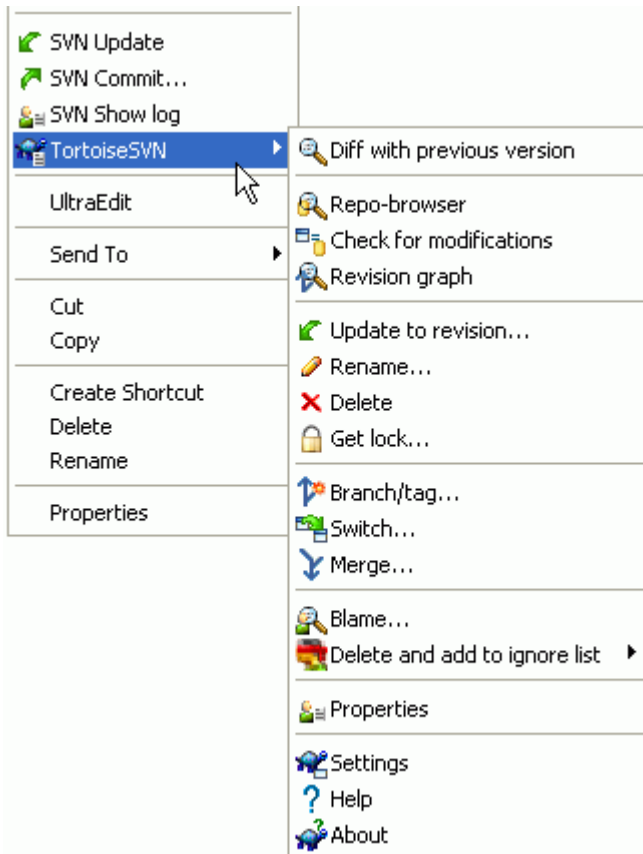


Geen Paden in de Globale Negeerlijst

Je moet eigenlijk geen paden in je patronen opnemen. Het patroonherkenningsproces is bedoeld om te filteren op platte bestands- en mapnamen. Als je alle CVS mappen wilt negeren, voeg dan allen CVS toe aan de neger lijst. Het is niet nodig om CVS */CVS te specificeren, zoals je gewend was van eerder versies. Als je alle tmp mappen wilt negeren als ze in een prog map aanwezig zijn, maar niet in een doc map, dan zou je de svn:ignore eigenschap moeten gebruiken. Er is geen betrouwbare manier om dit gedrag te bereiken met globale neger patronen.

4.14. Verwijderen, Verplaatsen en Hernoemen

In tegenstelling tot CVS, staat Subversion het hernoemen en verplaatsen van bestanden en mappen toe. Er zijn derhalve menu opties voor verwijderen en hernoemen in het TortoiseSVN submenu.



Figuur 4.28. Verkenner context menu voor bestanden met versiebeheer

4.14.1. Verwijderen van bestanden en mappen

Gebruik TortoiseSVN → Verwijderen om bestanden of mappen uit subversion te verwijderen.

Wanneer je TortoiseSVN → Verwijderen kiest op een bestand, wordt het direct uit je werkkopie verwijderd en tegelijkertijd ook gemarkeerd om bij een volgende vastlegging te worden verwijderd uit het archief. De map waar het bestand in stond toont nu een “verwijderd” overlay icoon. Tot op het moment dat je de wijziging vastlegt, kun je het bestand terugzetten via het uitvoeren van TortoiseSVN → Ongedaan maken op deze map

Wanneer je TortoiseSVN → Delete uitvoert op een map, blijft deze aanwezig in je werkkopie, maar het overlay icoon geeft aan dat het gemarkeerd is voor verwijdering. Tot het moment dat je de wijziging vastlegt, kun je de map weer terugkrijgen via het uitvoeren van TortoiseSVN → Ongedaan maken op deze folder. Het verschil in gedrag tussen bestanden en mappen is onderdeel van Subversion, niet van TortoiseSVN.

Als je een item wilt verwijderen uit het archief, maar wel lokaal wil behouden als een bestand/map zonder versiebeheer, gebruik dan Uitgebreide Context Menu → Verwijderen (lokaal behouden). Hiervoor dien je de **Shift** toets ingedrukt te houden terwijl je rechts klikt op het item in het lijst venster van de verkenner (rechter venster) om het uitgebreide context menu te kunnen zien.

Wanneer een *bestand* via de verkenner verwijderd is in plaats van via het TortoiseSVN context menu, wordt deze in het vastleggen dialoog getoond, zodat deze voor de vastlegging ook uit het versiebeheer gehaald kan worden. Als je echter je werkkopie ververs, zal Subversion een ontbrekend bestand detecteren en de meest recente versie vanuit het archief weer terugzetten. Wanneer je een bestand onder

versiebeheer wilt verwijderen, moet je daarom altijd TortoiseSVN → Verwijderen gebruiken, zodat Subversion niet hoeft te raden wat je echt wilt doen.

Als een *map* via de verkenner verwijderd is in plaats van via het TortoiseSVN context menu, zal je werkkopie kapot zijn geraakt en is het niet meer mogelijk om wijzigingen vast te leggen. Wanneer je je werkkopie ververst, zal Subversion de meest recente versie van deze map weer terugzetten, zodat je deze op de juiste manier kan verwijderen via TortoiseSVN → Verwijderen.



Een verwijderd bestand of map terughalen

Wanneer je een bestand of map hebt verwijderd en deze verwijdering al hebt vastgelegd, kan een standaard TortoiseSVN → Ongedaan maken deze niet meer terughalen. Maar het bestand of de map is nog niet verloren. Als je de revisie weet waarin het bestand of de map is verwijderd (als je dit niet weet, gebruik dan het log dialoog om erachter te komen), open dan de archief verkenner en schakel naar de betreffende revisie. Selecteer dan het bestand of de map welke verwijderd is, rechtsklik hierop en selecteer Context Menu → Kopiëren naar.... Selecteer het pad naar je werkkopie als doel van deze kopieër actie.

4.14.2. Verplaatsen van bestanden en mappen

Wanneer je een bestand of map gewoonweg wil hernoemen, gebruik dan Context Menu → Hernoemen.... Voer de nieuwe naam voor het item in en je bent klaar.

Wanneer je bestanden binnen je werkkopie wilt verplaatsen, wellicht naar een andere submap, gebruik dan de rechter-muisknop slepen en plaatsen methode:

1. selecteer de bestanden of mappen die je wil verplaatsen
2. sleep ze met de rechter muisknop naar de nieuwe locatie in je huidige werkkopie
3. laat de rechter muisknop los
4. selecteer vanuit het popup menu Contextmenu → SVN Verplaats objecten onder versiebeheer hierheen



Vastleggen van de bovenliggende map

Omdat het hernoemen en verplaatsen uitgevoerd wordt als een verwijdering gevolgd door een toevoeging, moet je de bovenliggende map vastleggen zodat het verwijderde gedeelte van de hernoeming/verplaatsing in het vastleggen dialoog verschijnt. Als je het verwijderde gedeelte niet vastlegt, zal het in het archief achterblijven en wanneer je collega's verversen, zal het oude bestand niet verwijderd zijn/worden. Zij zullen dus *beide* bestanden krijgen.

Je *moet* een naamswijziging in een map eerst vastleggen voordat je één van de bestanden in deze map gaat wijzigen, anders kan je werkkopie flink verknoeid worden.

Je kan ook de archief-verkenner gebruiken om objecten te verplaatsen. Lees [Paragraaf 4.24, "De Archief-verkenner"](#) voor meer informatie.



SVN Verplaats Geen Externen

Je moet *niet* de TortoiseSVN Verplaatsen of Hernoemen commando's gebruiken op een map welke gemaakt is door gebruik te maken van `svn:externals`. Deze actie zal ervoor zorgen dat het externe item verwijderd wordt uit de bovenliggende map wat waarschijnlijk

vele andere mensen van streek maakt. Als je een externe map moet verplaatsen, gebruik dan een gewone shell verplaatsing. Wijzig daarna de `svn:externals` eigenschappen van de bovenliggende mappen van de bron en het doel.

4.14.3. Wijzig hoofdlettergebruik in een bestandsnaam

Het wijzigen van een bestandsnaam door uitsluitend het veranderen van hoofd- en kleine letters is listig met Subversion onder Windows, omdat tijdens een hernoeming beide bestandsnamen voor een korte tijd moeten bestaan. Omdat Windows een bestandssysteem heeft wat niet hoofdlettergevoelig is, is dit niet mogelijk via het standaard Rename commando.

Gelukkig zijn er (minstens) twee mogelijke methodes om een bestand te hernoemen zonder de bijbehorende log geschiedenis kwijt te raken. Het is hierbij belangrijk om te hernoemen binnen Subversion. Het simpelweg hernoemen via de verkenner, zal je werkkopie verknoeien.

Oplossing A) (aanbevolen)

1. Vastleggen van de wijzigingen in uw werkkopie.
2. Hernoem het bestand van UPPERcase naar upperCASE rechstreeks in het archief via de archiefverkenner.
3. Ververs je werkkopie

Oplossing B)

1. Hernoem van UPPERcase naar UPPERcase_ via het hernoem commando in het TortoiseSVN submenu.
2. Vastleggen van veranderingen.
3. Hernoemen van HOOFDletter_ naar hoofdLETTER.
4. Vastleggen van veranderingen.

4.14.4. Omgaan met conflicten in hoofdlettergebruik in bestandsnamen

Als het archief al twee bestanden bevat met dezelfde naam, maar verschillend in hoofd-/kleine letters (bijvoorbeeld TEST.TXT en test.txt), zal het niet mogelijk zijn om de bovenliggende map te verversen of op te halen op een Windows client. Terwijl Subversion hoofdlettergevoelige bestandsnamen ondersteunt, doet Windows dit niet.

Dit kan soms gebeuren wanneer twee mensen bestanden vastleggen, vanuit verschillende werkkopiën, welke toevallig dezelfde naam hebben, maar verschillen in hoofd-/kleine letter. Het kan ook gebeuren wanneer bestanden worden vastgelegd vanuit een hoofdlettergevoelig bestandssysteem, zoals Linux.

In dat geval zul je moeten beslissen welk van deze bestanden je wilt behouden en de andere zul je moeten verwijderen uit het archief (of hernoemen).



Het voorkomen van twee bestanden met dezelfde naam

Er is een server hook-script beschikbaar op: <http://svn.collab.net/repos/svn/trunk/contrib/hook-scripts/> dat vastleggingen voorkomt welke zullen resulteren in conflicten in hoofd-/kleine letters.

4.14.5. Herstellen van Bestandhernoemingen

Soms zal je vriendelijke IDE bestanden voor je hernoemen als onderdeel van een refactoring en dit natuurlijk niet doorgeven aan Subversion. Als je dan probeert je wijzigingen vast te leggen, zal de oude bestandsnaam te zien zijn als ontbrekend, terwijl de nieuwe getoond wordt als een bestand zonder

versiebeheer. Je kan natuurlijk eenvoudig de nieuwe bestandsnaam aanvinken om het toe te voegen, maar omdat Subversion niet weet dat de twee bestanden gerelateerd zijn, verlies je wel alle achterliggende geschiedenis.

Een beter manier is om Subversion te vertellen dat deze wijziging feitelijk een hernoeming is, wat gedaan kan worden in de Vastleggen en Kijk of er updates zijn dialogen. Selecteer eenvoudig zowel de oude (ontbrekende) als de nieuwe (zonder versiebeheer) bestandsnaam en gebruik Context Menu → Repareer Verplaatsing om de twee bestanden te koppelen als een hernoeming.

4.14.6. Verwijderen van Bestanden Zonder Versiebeheer

Normaal gesproken zul je je negerlijst zo opzetten dat alle gegenereerde bestanden door Subversion genegeerd worden. Maar wat als je al deze bestanden wilt verwijderen om een schone build te kunnen produceren. Meestal zal je dit in je makefile zetten, maar als je de makefile aan het debuggen bent, of wanneer je het bouwsysteem veranderd, is het handig om een manier te hebben om het dek schoon te vegen.

TortoiseSVN voorziet in zo'n optie via Uitgebreide Context Menu → Verwijder objecten zonder versiebeheer.... Je moet de **Shift** toets ingedrukt houden als je rechts klikt op een map in het lijstvenster (rechter venster) van de verkenner om deze optie te kunnen zien in het uitgebreide context menu. Er wordt een scherm getoond waarin alle bestanden zonder versiebeheer in je gehele werkkopie in een lijst weergegeven worden. Je kan hieruit de bestanden die verwijderd moeten worden selecteren of deselecteren.

Als zulke items worden verwijderd wordt de prullenbak gebruikt, zodat als u hier een vergissing maakt en een bestand verwijdert dat zou moeten zijn geversioneerd, kunt u het nog terugzetten.

4.15. Wijzigingen ongedaan maken

Als je de wijzigingen die je sinds de laatste update in een bestand hebt gemaakt, ga je als volgt te werk. Selecteer het bestand, klik er met de rechter muisknop op zodat het contextmenu openspringt, en klik vervolgens op het commando TortoiseSVN → Ongedaan maken

Omdat een verwijderd bestand of een verwijderde map niet meer in je werkkopie bestaat, dien je, als je het verwijderen of hernoemen van een bestand of map ongedaan wilt maken, de actie 'Terugzetten' uit te voeren op de folder waarin deze items zich bevonden.

Als je het toevoegen van een bestand of map ongedaan wilt maken, dien je de volgende actie in het contextmenu uit te voeren: TortoiseSVN → Toevoegen opheffen.... Deze actie verschilt niet van de andere acties waarmee je handelingen ongedaan kunt maken, maar heeft een andere naam gekregen om het doel ervan te verhelderen.

Je kunt de kolommen in dit dialoogscherf configureren op dezelfde manier als die van het dialoogscherf Kijk of er updates zijn. Ga naar [Paragraaf 4.7.3, "Lokale en Server Status"](#) voor meer informatie.



Reeds vastgelegde wijzigingen ongedaan maken

Ongedaan maken maakt alleen lokale wijzigingen ongedaan. Wijzigingen die al zijn vastgelegd, worden *niet* ongedaan gemaakt. Als je wijzigingen die in een bepaalde revisie zijn vastgelegd, ongedaan wilt maken, lees dan [Paragraaf 4.9, "Logboekboodschappen Venster"](#).



Herstellen is langzaam

Als je wijzigingen ongedaan maakt, kan het gebeuren dat dit veel langer duurt dan verwacht. De reden hiervoor is dat de gewijzigde versie van het bestand wordt weggegooid en in de

prullenbak terechtkomt, zodat je je wijzigingen weer terug kunt halen als je TortoiseSVN per ongeluk instrueerde om die ongedaan te maken. Het kan echter voorkomen dat de prullenbak vol zit; in dat geval heeft Windows veel tijd nodig om een plaats voor het weggegooide bestand te vinden. Je kunt de lange wachttijd dus terugbrengen door de prullenbak te legen, of door de eigenschap **Gebruik prullenbak bij herstellen** in de instellingen van TortoiseSVN te deactiveren.

4.16. Schonen

Als een commando in Subversion niet succesvol kan worden afgerond, wat mogelijk met serverproblemen te maken heeft, kan je werkkopie in een foutieve staat geraken. In dat geval dien je de contextmenu-actie TortoiseSVN → **Schonen** op de map waarin zich je werkkopie bevindt, uit te voeren. Het is aan te raden om dit op zo hoog mogelijk niveau te doen.

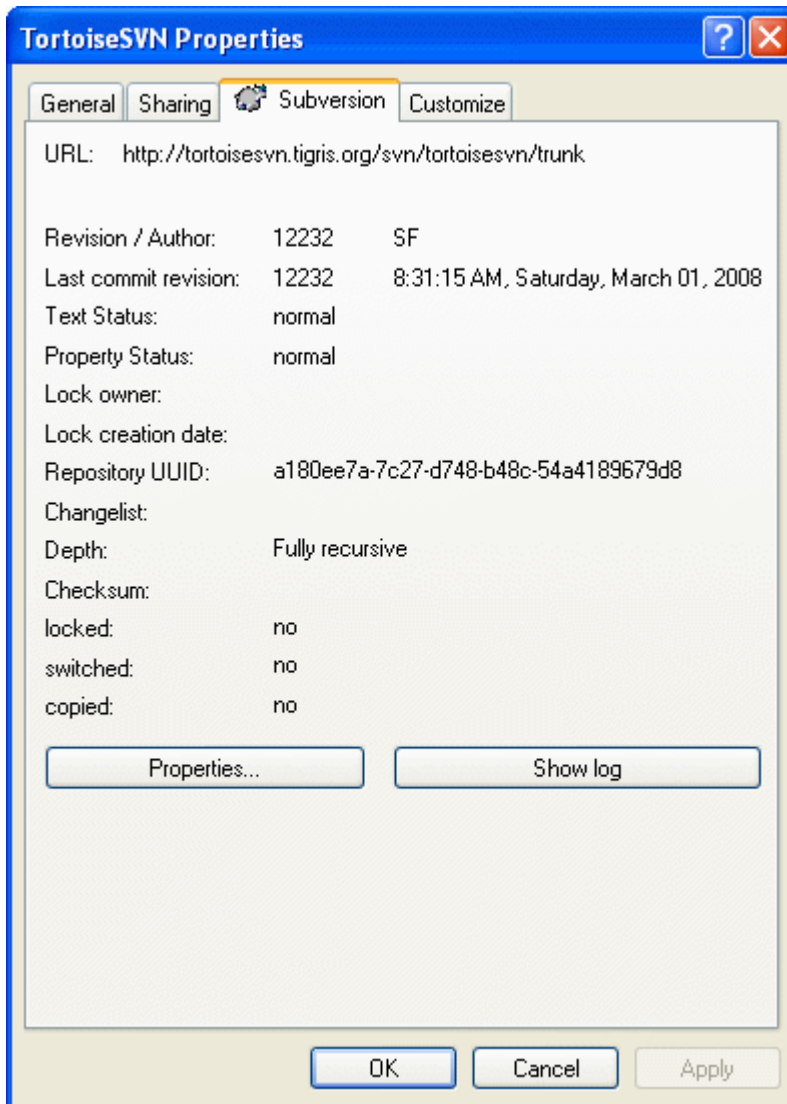
Schonen heeft een bruikbaar neveneffect. Als de datum van een bestand verandert zonder dat de inhoud dat doet, kan Subversion niet meer bepalen of het bestand werkelijk gewijzigd is, tenzij het dit bestand byte voor byte vergelijkt met een ongewijzigde versie ervan. Als je veel bestanden hebt die onder dit euvel lijden, kan dat het opvragen van statussen erg vertragen, waardoor veel dialoogschermen veel langzamer worden. Het commando 'Schonen' op je werkkopie uitvoeren repareert deze “kapotte” datum-/tijdsaanduidingen, waardoor een statuscontrole weer op volle snelheid kan plaatshebben.



Gebruik Vastleg-tijdstippen

In enkele oudere versies van Subversion zat een bug die ervoor zorgde dat de datum-/tijdsaanduiding van een bestand niet meer klopte wanneer je een werkkopie uit een archief ophaalde en daarbij de optie **Gebruik vastleg-tijdstippen** had aangevinkt. Gebruik het commando 'Schonen' om de werkkopiën die hieronder lijden, weer op snelheid te brengen.

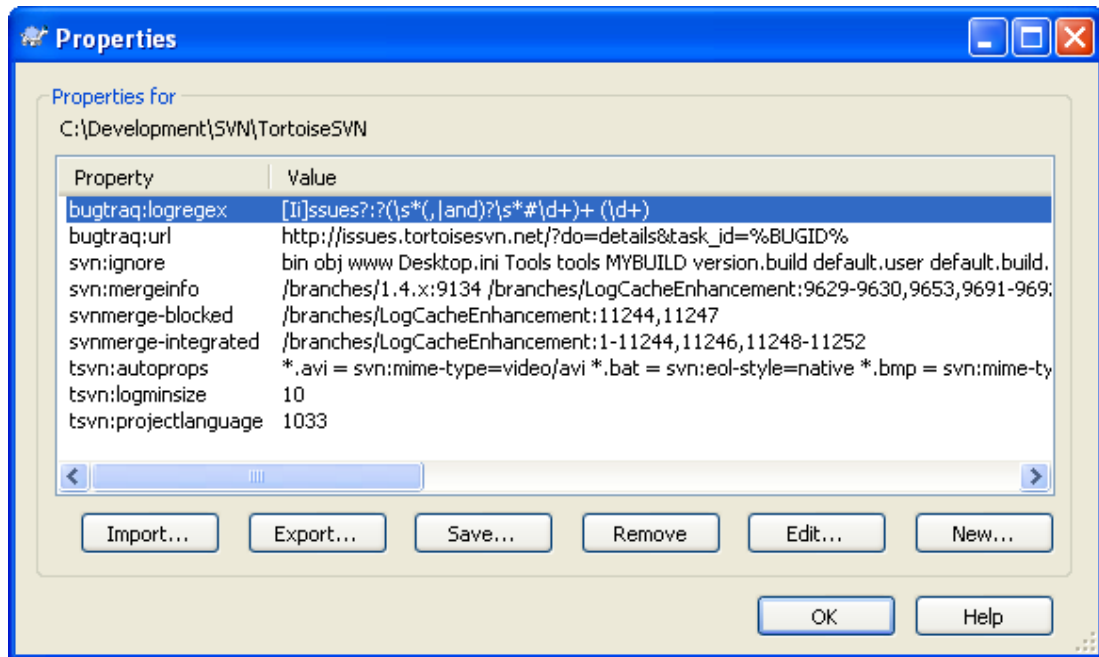
4.17. Project Instellingen



Figuur 4.30. Verkenner eigenschappen pagina, Subversion tab

In sommige gevallen wilt u meer gedetailleerde informatie over een bestand/map dan het overlappende pictogram. U kunt alle informatie welke Subversion levert verkrijgen in het eigenschappen dialoog van de verkenner. Selecteer het bestand of directory en selecteer **Windows Menu** → **eigenschappen** in het contextmenu (noot: dit is het normale eigenschappen menu-optie welke de verkenner levert, niet degene in het TortoiseSVN submenu!). In het eigenschappen dialoogvenster heeft TortoiseSVN een nieuwe eigenschappen pagina toegevoegd, voor bestanden/folders welke onder Subversion controle vallen, waar u alle relevante informatie over het geselecteerde bestand/directory kunt vinden.

4.17.1. Subversion Eigenschappen



Figuur 4.31. Subversion kenmerken pagina

U kunt de Subversion eigenschappen lezen en zetten vanuit het Windows eigenschappen dialoog, maar ook vanuit TortoiseSVN → eigenschappen en binnen TortoiseSVN's status lijsten, vanuit Context menu → eigenschappen.

U kunt uw eigen eigenschappen toevoegen, of enige eigenschappen met een speciale betekenis in Subversion. Deze beginnen met `svn:`. `svn:externals` is zo'n eigenschap; raadpleeg hoe externen af te handelen in [Paragraaf 4.18, "Externe Objecten"](#).

4.17.1.1. svn:keywords

Subversion ondersteunt CVS-achtige trefwoord uitbreidingen welke kunnen worden gebruikt om bestandsnaam en revisie informatie in het bestand zelf in te bedden. Trefwoorden welke momenteel worden ondersteund zijn:

`$Date$`

Datum van laatste vastlegging. Dit is gebaseerd op informatie verkregen tijdens het bijwerken van uw werkkopie. Het controleert het archief *niet* op meer recente wijzigingen.

`$Revision$`

Revisie van laatst bekende vastlegging.

`$Author$`

Auteur welke de laatst bekende vastlegging heeft gemaakt.

`$HeadURL$`

De volledige URL van dit bestand in het archief.

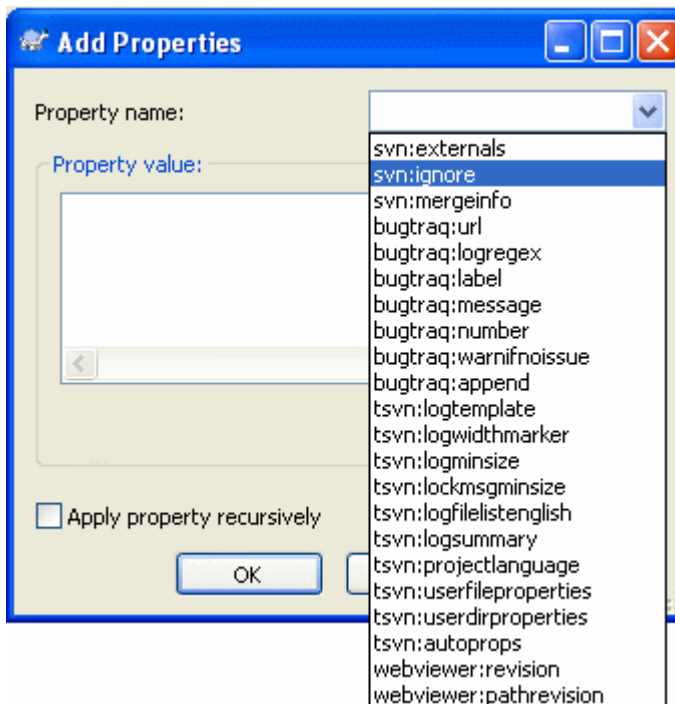
`Id`

Een gekomprimeerde combinatie van de vorige vier trefwoorden.

Om meer te weten te komen over hoe deze trefwoorden te gebruiken, bekijk [svn:keywords section](http://svnbook.red-bean.com/en/1.5/svn.advanced.props.special.keywords.html) [http://svnbook.red-bean.com/en/1.5/svn.advanced.props.special.keywords.html] in het Subversion boek, wat een volledige beschrijving van deze trefwoorden geeft en hoe ze te kunnen gebruiken.

Voor meer informatie over eigenschappen in Subversion bekijk [Speciale Eigenschappen](http://svnbook.red-bean.com/en/1.5/svn.advanced.props.html) [http://svnbook.red-bean.com/en/1.5/svn.advanced.props.html].

4.17.1.2. Toevoegen en Aanpassen van Eigenschappen



Figuur 4.32. Toevoegen van eigenschappen

Klik eerst op **Toevoegen...** om een nieuwe eigenschap toe te voegen. Selecteer de gewenste naam van de eigenschap uit de lijst of voer een naam van jouw keuze in en geef vervolgens de waarde op in het veld daaronder. Eigenschappen die meerdere waarden kunnen hebben, bijvoorbeeld de negeren lijst, kunnen meerdere regels hebben. Klik op **OK** om de eigenschap aan de lijst toe te voegen.

Als je een eigenschap op meerdere objecten tegelijk wilt toepassen, selecteer dan de bestanden/mappen in de verkenners en kies **Context menu** → **eigenschappen**.

Als je de eigenschappen wilt gebruiken voor *alle* bestanden en mappen in de onderliggende mappen van de huidige map, selecteer dan de optie **Recursief**.

Enkele eigenschappen, bijvoorbeeld `svn:needs-lock`, kunnen alleen aan bestanden toegekend worden, daarom worden deze namen van eigenschappen niet weergegeven in de lijst voor mappen. Het is wel mogelijk om zo'n eigenschap recursief op alle bestanden in de hiërarchie toe te passen, maar je moet dan wel de naam van eigenschap handmatig invoeren.

Als je een bestaande eigenschap wilt bewerken, selecteer dan die eigenschap uit de lijst van bestaande eigenschappen en klik dan op **Wijzigen...**

Als je een bestaande eigenschap wilt verwijderen, selecteer dan de eigenschap uit de lijst van bestaande eigenschappen en klik dan op **Verwijderen**.

De `svn:externals` eigenschap kan worden gebruikt om andere projecten uit hetzelfde archief of uit een compleet ander archief binnen te halen. Voor meer informatie, lees [Paragraaf 4.18, "Externe Objecten"](#).

4.17.1.3. Exporteren en Importeren van Eigenschappen

Je zult er achter komen dat je vaak dezelfde set van eigenschappen toepast, bijvoorbeeld `bugtraq:logregex`. Om het kopiëren van eigenschappen van het ene project naar het andere te vereenvoudigen, kun je gebruik maken van de **Export/Import** functionaliteit.

Kies voor het bestand of de map waar de eigenschap al ingesteld is TortoiseSVN → Eigenschappen en selecteer de eigenschappen die je wilt exporteren en klik op Exporteer. Je moet vervolgens de bestandsnaam opgeven voor het bestand waarin je namen en waardes wilt opslaan.

Kies vanuit de map(pen) waarop je deze eigenschappen wilt toepassen TortoiseSVN → Eigenschappen en klik op Importeren.... Er zal dan gevraagd worden vanuit welk bestand er geïmporteerd moet worden. Blader naar de locatie waar je eerder het export bestand hebt opgeslagen en selecteer deze. De eigenschappen zullen niet-recursief worden toegepast op de mappen.

Als je eigenschappen recursief aan een mapstructuur wilt toevoegen, volg dan de hierboven beschreven stappen, selecteer in het eigenschappen scherm elke aparte eigenschap en klik op Bewerken.... Vink de optie Eigenschap recursief toepassen aan en klik op OK.

De indeling van het import bestand is binair en een eigen indeling voor TortoiseSVN. Het enige nut van deze bestanden in het overzetten van eigenschappen met de Importeren en Exporteren functie. Er is dus geen reden om deze bestanden te kunnen bewerken.

4.17.1.4. Binaire eigenschappen

TortoiseSVN kan omgaan met binaire waardes van eigenschappen als bestanden worden gebruikt. Om een binaire waarde van een eigenschap te lezen, dan moet je deze Opslaan... in een bestand. Gebruik een Hex-editor of ander geschikt programma om een bestand aan te maken met de benodigde inhoud en kies dan Laden... om de inhoud van dit bestand op te halen.

Ondanks het feit dat binaire eigenschappen niet vaak gebruikt worden, kunnen ze in sommige gevallen erg handig zijn. Bijvoorbeeld als je hele grote grafische bestanden opslaat of de applicatie die gebruikt wordt om een bestand te laden erg groot is, dan wil je misschien een thumbnail opslaan als eigenschap, zodat je deze snel vooraf kunt bekijken.

4.17.1.5. Automatisch eigenschappen zetten

Je kunt Subversion en TortoiseSVN zo instellen dat ze automatisch eigenschappen instellen voor bestanden en mappen als deze worden toegevoegd aan het archief. Er zijn twee manieren om dit te doen.

Je kunt het subversion configuratie bestand bewerken om deze functie voor jouw eigen client aan te zetten. De Algemeen pagina van het TortoiseSVN instellingen scherm heeft een knop waarmee je dit bestand makkelijk kunt vinden. Het configuratie bestand is een plat tekstbestand, waarmee enkele werkinstellingen voor subversion worden gedaan. Je moet twee dingen aanpassen: haal eerst in de sectie met de naam `miscellany` de commentaar tekens weg voor de regel `enable-auto-props = yes`. Als tweede actie moet je de sectie daaronder bewerken om te definiëren welke eigenschappen je op welke bestandstypes wilt toe laten passen. Deze methode is een standaard functie van subversion en werkt met elke subversion client. Dit moet echter wel op elke client apart ingesteld worden - er is geen methode om dit door te laten propageren vanuit het archief.

Een alternatief is om de `tsvn:autoprops` eigenschap in te stellen op mappen, zoals omschreven in de volgende sectie. Deze methode werkt alleen met TortoiseSVN, maar zal wel propageren naar alle werkkopieën bij het verversen.

Welke methode je ook kiest, moet je er rekening mee houden dat auto-props alleen worden toegepast op het moment dat deze aan het archief worden toegevoegd. Auto-props zullen nooit de eigenschappen veranderen van bestanden die al in het archief zitten.

Als je er absoluut zeker van wilt zijn dat nieuwe bestanden de juist eigenschappen toegepast krijgen, moet je een pre-commit hook instellen, welke de vastlegging weigert als de benodigde eigenschappen niet ingesteld zijn.



Vastleggen eigenschappen

Subversion eigenschappen zijn voorzien van versiebeheer. Als je een eigenschap wijzigt of toevoegt, moet je je wijzigingen vastleggen in het archief.



Conflicten op eigenschappen

Als er een conflict optreedt bij het vastleggen van de wijzigingen, omdat een andere gebruiker dezelfde eigenschap heeft gewijzigd, maakt Subversion een `.prej>/filename>` bestand aan. Verwijder dit bestand nadat je het conflict opgelost hebt.

4.17.2. TortoiseSVN Project Eigenschappen

TortoiseSVN heeft een paar speciale eigenschappen van zichzelf, en deze beginnen met `tsvn:`.

- `tsvn:logminsize` stelt de minimum lengte in voor een logboek bericht bij een vastlegging. Als je minder karakters gebruikt dan gespecificeerd, dan kan de vastlegging niet afgerond worden. Deze functie is erg nuttig om je te helpen herinneren dat je een bericht achter laat met een goede beschrijving voor elke blokkade die je plaatst. Als deze eigenschap niet ingesteld is of op nul is ingesteld, dan worden lege logboek berichten toegestaan.

`tsvn:lockmsgminsize` stelt de minimum lengte in voor een blokkade bericht. Als je minder karakters gebruikt dan gespecificeerd, dan wordt de blokkade niet geplaatst. Deze functie is erg nuttig om je te helpen herinneren dat je een bericht achter laat met een goede beschrijving voor elke blokkade die je plaatst. Als deze eigenschap niet ingesteld is of op nul is ingesteld, dan worden lege blokkade berichten toegestaan.

- `tsvn:logwidthmarker` wordt gebruikt voor projecten waarbij het logboek bericht opgemaakt moet worden met een maximum breedte (meestal 80 karakters) voor een regeleinde. Het instellen van deze eigenschap op een waarde die niet nul is, resulteert in twee dingen in het logboek bericht venster: het plaatst een markering om de maximale breedte aan te geven en tekstterugloop wordt uigeschakeld, zodat je kunt zien of de tekst die je hebt ingevoerd te lang is. Noot: deze functie werkt alleen goed als je een lettertype hebt geselecteerd met een vaste tekenbreedte.
- `tsvn:logtemplate` wordt gebruikt voor project waarbij er regels zijn voor de indeling van het logboek bericht. De eigenschap bevat een meerregelige tekst die opgenomen wordt in het vastleg invoerveld als je een vastlegging start. Je kunt het vervolgens bewerken om de benodigde informatie in te vullen. Noot: als je ook `tsvn:logminsize` gebruikt, zorg er dan voor dat de minimale lengte hoger staat ingesteld dan het aantal karakters van de template, anders heeft die eigenschap helemaal geen nut meer.
- Subversion heeft de mogelijkheid om “autoprops” (automatisch eigenschappen) te configureren, welke automatisch toegevoegd worden aan nieuw toegevoegde of geïmporteerde bestanden op basis van de extensie van het bestand. Dit is afhankelijk van de instellingen van de autoprops in het subversion configuratiebestand van elke individuele client. `tsvn:autoprops` kunnen ingesteld worden op mappen. Deze worden dan samengevoegd met de lokale autoprops van de gebruiker bij het importeren of toevoegen van nieuwe bestanden. De indeling is gelijk aan die van de subversion autoprops, bijv. `*.sh = svn:eol-style=native;svn:executable` kent twee eigenschappen toe aan bestanden met de `.sh` extensie.

Als er een conflict is tussen de lokale autoprops en de `tsvn:autoprops`, dan krijgen de instellingen van het project voorrang, omdat die specifiek voor het project zijn.

- In het Vastleg scherm kun je de lijst met gewijzigde bestanden en de status (toegevoegd, gewijzigd, etc) invoegen. `tsvn:logfilelistenglish` definieert of de status van de bestanden in het Engels

of in de lokale taal opgenomen moet worden. Deze eigenschap krijgt, als hiervoor geen andere waarde is ingesteld, de standaard waard `true`.

- TortoiseSVN kan spellingscontrole modules gebruiken, zoals deze ook gebruikt worden in OpenOffice en Mozilla. Als je deze geïnstalleerd hebt, dan bepaalt deze eigenschap welke spellingscontrole gebruikt moet worden, oftewel in welke taal de logboek boodschappen voor je project geschreven moeten worden. `tsvn:projectlanguage` stelt de te gebruiken taal in voor de spellingscontrole van je logboek boodschappen. Je kunt de waarde voor jouw taal vinden op deze pagina: [MSDN: Language Identifiers](http://msdn2.microsoft.com/en-us/library/ms776260.aspx) [http://msdn2.microsoft.com/en-us/library/ms776260.aspx].

Je kunt deze waarde als decimale of hexadecimale, indien vooraf gegaan door `0x`, waarde ingevoerd worden. Bijvoorbeeld Engels (VS) kan ingevoerd worden als `0x0409` of als `1033`.

- De `tsvn:logsummary` eigenschap wordt gebruikt om een deel uit het logboek bericht te extraheren, welke dan getoond kan worden in het logboek venster als samenvatting.

De waarde van de `tsvn:logsummary` eigenschap moet een regelige reguliere expressie bevatten met één Regex-groep. De inhoud van die groep wordt als samenvatting gebruikt.

Een voorbeeld: `\[SUMMARY\]:\s+(.*)` Zal alles pakken na “[SUMMARY]” in het logbericht en zal dat gebruiken als de samenvatting.

- Als je een nieuwe eigenschap wilt toevoegen, dan kun je er eentje uit de lijst van de combo box kiezen of je kunt de gewenste naam voor de eigenschap invoeren. Als je project gebruik maakt van enkele eigen eigenschappen en je wilt dat deze in ook de lijst getoond worden (bijv. om typefouten te voorkomen), dan kun je een lijst met je eigen eigenschappen aanmaken door gebruik te maken van `tsvn:userfileproperties` en `tsvn:userdirproperties`. Pas deze eigenschappen toe op een map. Pas deze eigenschappen toe op een map. Als je dan de eigenschappen van onderliggende objecten bewerkt, dan worden je voorgedefinieerde eigenschappen weergegeven in de lijst met de namen van de beschikbare eigenschappen.

Sommige `tsvn:` eigenschappen vereisen een `true/false` waarde. TortoiseSVN begrijpt ook `yes` als een synoniem voor `true` en `no` als een synoniem voor `false`.

TortoiseSVN kan worden geïntegreerd met enkele fout zoek tools. Hiervoor worden de project-eigenschappen gebruikt welke beginnen met `bugtraq:`. Lees [Paragraaf 4.28, “Integratie met Bug Tracking Systemen / Issue Trackers”](#) voor meer informatie.

Het kan ook worden geïntegreerd met enkele web-gebaseerde archief verkenners door gebruik te maken van de project-eigenschappen welke beginnen met `webviewer:`. Lees [Paragraaf 4.29, “Integratie met Web-gebaseerde Archief Viewers”](#) voor meer informatie.



Stel de project eigenschappen in voor folders

Deze speciale project eigenschappen moeten ingesteld worden op *mappen* om te kunnen werken. Als je een bestand of map vastlegt, worden de eigenschappen van die map gelezen. Als daar de eigenschappen niet worden gevonden, dan zal TortoiseSVN in de bovenliggende mappen gaan zoeken om ze te vinden, totdat het een map zonder versiebeheer tegenkomt of bij de hoofdmap van de mapstructuur is aangekomen (bijv. `C:\`). Als je er zeker van kunt zijn dat elke gebruiker bijv. van de map `trunk/` ophaalt en nooit een sub-map, dan volstaat het om de eigenschap alleen op de map `trunk/` in te stellen. Als je daar niet zeker van kunt zijn, dan moet je de eigenschappen ook op elke sub-map instellen. Een eigenschap die dieper in de project hiërarchie is ingesteld, is belangrijker dan instellingen op een hoger niveau (dichter bij de map `trunk/`).

Bij eigenschappen welke *alleen* op projecten van toepassing zijn, bijv. `tsvn:`, `bugtraq:` en `webviewer:`, kunt u gebruik maken van de **Recursief vinkbox** om de eigenschap op alle sub-mappen binnen de hiërarchie in te stellen, zonder het ook op de bestanden in te stellen.

Als u met TortoiseSVN nieuwe sub-mappen aan een werkkopie toevoegd, zullen alle project eigenschappen welke in de bovenliggende map aanwezig zijn, automatisch aan de nieuwe sub-map worden toegevoegd.



Let op

Ondanks dat TortoiseSVN's project eigenschappen erg nuttig zijn, werken ze alleen met TortoiseSVN en sommige nieuwe eigenschappen zullen allen samenwerken met nieuwe versies van TortoiseSVN. Als er mensen met verschillende Subversion clients werken aan een project, of misschien een oude versie van TortoiseSVN gebruiken, dan wil je misschien archief hooks gebruiken om een project beleid te forceren. Project eigenschappen kunnen alleen helpen om een beleid te implementeren, eigenschappen kunnen een beleid niet afdwingen.

4.18. Externe Objecten

Soms is het handig om een werkkopie te hebben, die bestaat uit bestanden die in verschillende aparte archieven zijn opgeslagen. Bijvoorbeeld, je wilt dat bestanden en submappen van verschillende locaties uit het archief worden gehaald, of misschien zelfs vanuit verschillende archieven. Als je wilt dat elke gebruiker dezelfde indeling gebruikt, dan kun je gebruik maken van `svn:externals` eigenschappen om de benodigde bronnen naar de locaties te halen waar ze nodig zijn.

4.18.1. Externe Mappen

Stel je haalt je werkkopie van `/project1` op naar `D:\dev\project1`. Selecteer de map `D:\dev\project1`, klik rechts en kies **Windows Menu** → **Eigenschappen** in het context menu. Het venster met de eigenschappen komt naar voren. Ga dan naar de Subversion tab, waarin je eigenschappen kunt wijzigen. Klik **Add...** Selecteer de `svn:externals` eigenschap in de combobox en voer in het tekstveld de URL van het archief in volgend de indeling `url map` of als je een specifieke revisie wilt gebruiken `-rREV url folder`. Je kunt meerdere externe projecten toevoegen, op elke regel één. Stel je hebt deze eigenschappen ingesteld op `D:\dev\project1`:

```
http://sounds.red-bean.com/repos sounds
http://graphics.red-bean.com/repos/fast%20graphics "quick graphs"
-r21 http://svn.red-bean.com/repos/skin-maker skins/toolkit
```

Klik nu op **Set** en leg je wijzigingen vast in het archief. Als jij nu (of een andere gebruiker) de werkkopie ververst, zal Subversion de sub-map `D:\dev\project1\sounds` aanmaken en het `sounds` project, een extra sub-map `D:\dev\project1\quick_graphs` met het grafische deel van het project en als laatste een geneste sub-map `D:\dev\project1\skins\toolkit` met revisie 21 van het `skin-maker` project aanmaken.

URL's moeten netjes afgesloten worden, anders werken ze niet. Je moet bijvoorbeeld ook elke spatie vervangen door `%20`, zoals in het tweede voorbeeld weergegeven.

Als je in lokale paden spaties en andere speciale tekens wilt gebruiken, dan moet je deze tussen aanhalingstekens plaatsen, of je kunt de `\` (schuine streep naar links, ook wel backslash genoemd) gebruiken, zoals deze gebruikt worden Unix voor speciale karakters. Dit houdt ook in dat je de `/` (schuine streep naar rechts, ook wel forward slash genoemd) moet gebruiken als scheidingsteken voor paden. Merk op dat dit gedrag nieuw is voor Subversion 1.6 en dat dit niet werkt bij oudere versies.



Gebruik Explicite Revisie Nummers

Het is verstandig om serieus te overwegen expliciete revisienummers te gebruiken in alle externe definities, zoals hierboven beschreven. Dit houdt in dat je moet beslissen wanneer

je een andere momentopname van externe informatie op wilt halen en dan dus ook welke momentopname precies. Afgezien van het feit dat je dan niet verrast wordt door wijzigingen in archieven van derde partijen, waar je geen controle over hebt, biedt het gebruik van expliciete revisienummers ook het voordeel dat als je je werkkopie terugzet naar een oudere versie ook de betrokken externen naar de respectievelijke oudere versie worden teruggezet. Bij software projecten kan dit het verschil betekenen tussen een succesvolle of fout gelopen bouwproces van een oude versie van een complex project.



Oudere svn:externals definities

De indeling die hier weergegeven is, is in Subversion 1.5 geïntroduceerd. Je kunt ook de oudere indeling tegenkomen waarbij de informatie in een andere volgorde is weergegeven. De nieuwe indeling heeft de voorkeur, omdat daarbij enkele nuttige functies, zoals hieronder beschreven, worden ondersteund die niet in oudere versies ondersteund worden. De verschillen zijn weergegeven in het *Subversion Boek* [<http://svnbook.red-bean.com/en/1.5/svn.advanced.externals.html>].

Als het externe project zich in hetzelfde archief bevindt, dan worden de wijzigingen die je daarin aanbrengt meegenomen in de vastleglijst als je het hoofdproject vastlegt.

Als het externe project zich in een ander archief bevindt, dan zul je alleen een notificatie krijgen van de wijzigingen die je aan het externe project hebt doorgevoerd op het moment dat je het hoofd project vastlegt. Je moet de wijzigingen aan het externe project apart vastleggen.

Als je absolute URL's gebruikt in de `svn:externals` definities en je moet een keer je werkkopie verplaatsen (oftewel als de URL van je archief verandert), dan zullen de externe verwijzingen niet mee veranderen en dus zullen deze niet meer werken.

Om zulke problemen te voorkomen, ondersteunen de Subversion clients vanaf versie 1.5 relatieve externe URL's. Er zijn vier verschillende methodes voor het definiëren van relatieve URL's. Bij de volgende voorbeelden gaan we er van uit dat we twee archieven hebben: één op `http://example.com/svn/repos-1` en de ander op `http://example.com/svn/repos-2`. We hebben een werkkopie opgehaald van `http://example.com/svn/repos-1/project/trunk` naar de locatie `C:\Working` en de `svn:externals` eigenschap is ingesteld op de basislijn.

Relatief ten opzichte van bovenliggende map

Deze URL's beginnen altijd met `../`, bijvoorbeeld:

```
../../widgets/foo common/foo-widget
```

Hiermee wordt `http://example.com/svn/repos-1/widgets/foo` geëxtraheerd naar `C:\Working\common\foo-widget`.

Merk op dat de URL relatief is ten opzichte van de URL van de map met de `svn:externals` eigenschap, niet ten opzichte van de map waar de externe opgeslagen is op de schijf.

Relatief ten opzichte van de archief hoofdmap

Deze URL's beginnen altijd met `^/`, bijvoorbeeld:

```
^/widgets/foo common/foo-widget
```

Hiermee wordt `http://example.com/svn/repos-1/widgets/foo` naar `C:\Working\common\foo-widget` geëxtraheerd.

Je kunt eenvoudig naar andere archieven met dezelfde `SVNParentPath` (een gemeenschappelijke map met meerdere archieven) refereren. Bijvoorbeeld:

```
^/./repos-2/hammers/claw common/claw-hammer
```

Hiermee wordt `http://example.com/svn/repos-2/hammers/claw` naar `C:\Working\common\claw-hammer` geëxtraheerd.

Relatief ten opzichte van de indeling

URL's die beginnen met `//` kopiëren van de URL alleen het gedeelte met de indeling. Dit is nuttig als dezelfde hostnaam gebruikt wordt voor het benaderen van verschillende indelingen afhankelijk van de netwerk locatie. Bijvoorbeeld programma's op het intranet gebruiken `http://` terwijl externe programma's `svn+ssh://` gebruiken. Een voorbeeld:

```
//example.com/svn/repos-1/widgets/foo common/foo-widget
```

Hiermee wordt `http://example.com/svn/repos-1/widgets/foo` of `svn+ssh://example.com/svn/repos-1/widgets/foo` geëxtraheerd afhankelijk van de methode die gebruikt is voor het ophalen van `C:\Working`.

Relatief ten opzichte van de server hostnaam

URL's die beginnen met `/` kopiëren de gedeeltes met de indeling en de hostnaam van de URL, bijvoorbeeld:

```
/svn/repos-1/widgets/foo common/foo-widget
```

Hiermee wordt `http://example.com/svn/repos-1/widgets/foo` naar `C:\Working\common\foo-widget` geëxtraheerd. Als je echter je werkkopie ophaalt van een andere server vanaf `svn+ssh://another.mirror.net/svn/repos-1/project1/trunk`, dan zal de externe referentie `svn+ssh://another.mirror.net/svn/repos-1/widgets/foo` extraheren.

Je kunt ook een vastgepinde revisie opgeven achteraan de URL indien nodig, bijvoorbeeld `http://sounds.red-bean.com/repos@19`.

Als je meer informatie wilt hebben over hoe TortoiseSVN omgaat met Eigenschappen, lees dan [Paragraaf 4.17, "Project Instellingen"](#).

Lees voor meer informatie over de verschillende methodes voor het benaderen van gemeenschappelijke sub-projecten [Paragraaf B.6, "Een veelgebruikt subproject gebruiken"](#).

4.18.2. Externe Bestanden

Vanaf Subversion 1.6 kun je enkele externe bestanden aan je werkkopie toevoegen, door gebruik te maken van dezelfde syntax als voor mappen. Er zijn wel enkele beperkingen.

- Het pad naar het externe bestand moet het bestand in een bestaande map met versiebeheer plaatsen. Over het algemeen is het het handigst om het bestand te plaatsen in de map waarin de `svn:externals` is ingesteld, maar dit kan ook in een sub-map met versiebeheer indien nodig. Dit in tegenstelling tot externe mappen, waarbij automatisch tussentijdse mappen zonder versiebeheer worden aangemaakt als deze nodig zijn.
- De URL van een extern bestand moet in hetzelfde archief zijn als de URL van het archief waarin het externe bestand wordt opgehaald, externe bestanden tussen verschillende archieven worden niet ondersteund.

Een extern bestand gedraagt zich net zoals elk ander bestand met versiebeheer, maar deze kunnen niet verplaatst of verwijderd worden met normale commando's; de `svn:externals` eigenschap moet in de plaats daarvan gewijzigd worden.



Incomplete ondersteuning van externe bestanden in Subversion 1.6

Het is in subversion 1.6 niet mogelijk om een extern bestand van je werkkopie te verwijderen als je deze toegevoegd hebt, zelfs als je ook de `svn:externals` eigenschap verwijderd. Je moet een verse werkkopie ophalen als je het bestand wilt verwijderen.

4.19. Vertakken / labelen

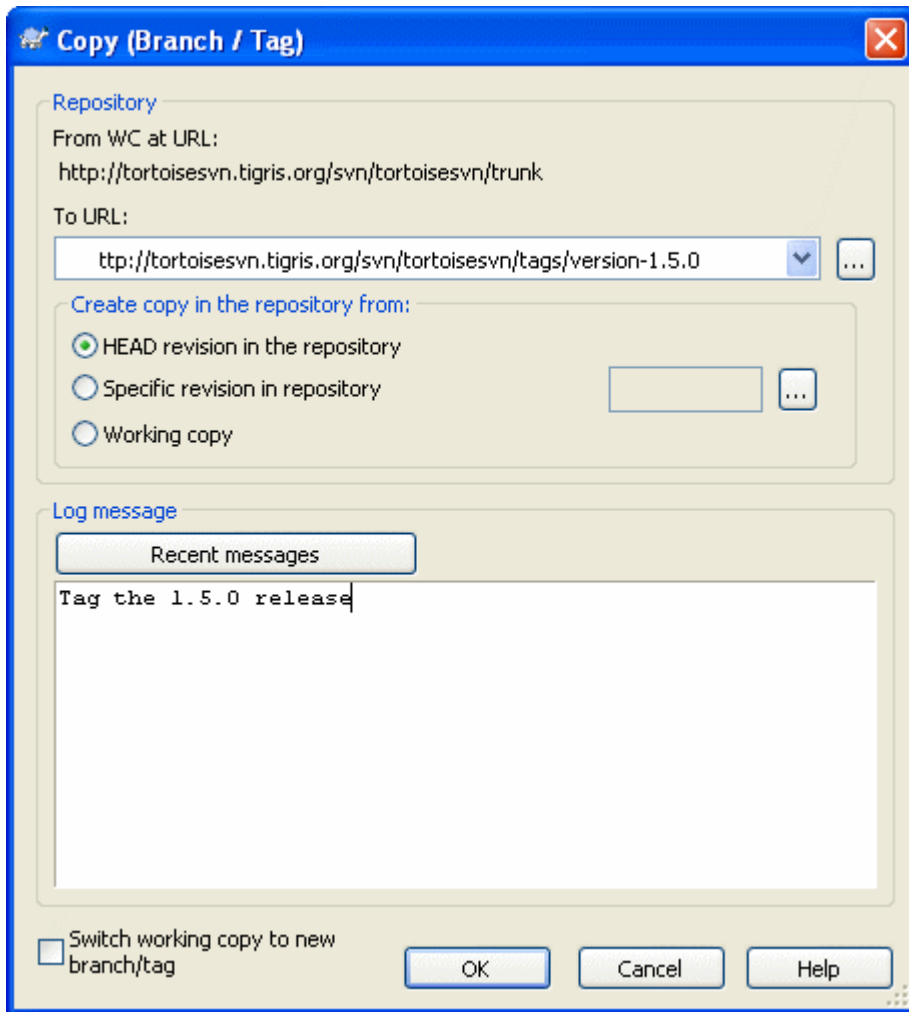
Eén van de mogelijkheden van versiebeheersystemen is het isoleren van wijzigingen in een aparte ontwikkellijijn. Zo'n lijn wordt *tak* (in het Engels *branch*) genoemd. Takken worden vaak gebruikt om nieuwe functies uit te proberen, zonder dat daarbij de hoofdontwikkeling verstoord wordt met compiler fouten en bugs. Zodra de nieuwe functie stabiel genoeg is, wordt deze ontwikkel tak *samengevoegd* met de basislijn (in het Engels de *trunk* genoemd).

Een andere mogelijkheid van versiebeheersystemen is het markeren van bepaalde revisies (bijv. een vrijgegeven versie), zodat op elk gewenst moment een bepaald pakket of een omgeving opnieuw (op) kunt bouwen. Dit proces wordt *labelen* genoemd (in het Engels *tagging*).

Subversion heeft geen speciale commando's voor vertakken of labelen, maar gebruikt hiervoor zogenaamde “goedkope kopieën”. Goedkope kopieën zijn vergelijkbaar met de harde links in Unix, wat inhoudt dat er in plaats van een gehele kopie te maken van het archief, er een interne link wordt aangemaakt. Deze link verwijst naar een specifieke boom of revisie. Hierdoor kunnen takken en labels heel snel gemaakt worden en hebben ze bijna geen extra ruimte nodig in het archief.

4.19.1. Aanmaken van Tak of Label

Als je je project met de aanbevolen mapstructuur hebt geïmporteerd, dan is het aanmaken van een tak of label erg eenvoudig:



Figuur 4.33. De Tak/Label Dialoog

Selecteer de map in je werkkopie, waarvan je een tak of label wilt maken. Selecteer vervolgens het commando TortoiseSVN → Maak Tak/label....

De standaard doel URL voor de nieuwe tak is de URL van de bron van je werkkopie. Je moet deze URL aanpassen naar het nieuwe pad voor je tak/label. Dus in plaats van

```
http://svn.collab.net/repos/ProjectName/trunk
```

zul je mogelijk nu

```
http://svn.collab.net/repos/ProjectName/tags/Release_1.10
```

gebruiken. Als je je niet meer kunt herinneren welke structuur je de vorige keer hebt gebruikt, klik dan op de knop aan de rechterkant om de archief-verkenner te openen. Je kunt dan de structuur van het archief bekijken.

Nu moet je de bron van de kopie selecteren. Er zijn daarvoor drie mogelijkheden:

HEAD revisie in het archief

De nieuwe tak wordt direct binnen het archief gekopieerd van de HEAD revisie in het archief. Er hoeven geen gegevens verstuurd te worden vanuit je werkkopie. De tak is dan erg snel gemaakt.

Specifieke revisie in het archief

De nieuwe tak wordt direct binnen het archief gekopieerd, maar je kunt een oudere revisie selecteren. Dit is handig als je vergeten bent een label te maken toen je een nieuwe versie vrijgaf vorige week. Als je het revisienummer niet meer weet, klik dan op de knop aan de rechterkant om het revisie logboek te bekijken en te selecteren. Er worden ook hierbij geen gegevens vanuit je werkkopie gehaald en dus is de tak of label erg snel gemaakt.

Werkkopie

De nieuwe tak moet een identieke kopie zijn van je lokale werkkopie. Als je bijvoorbeeld van enkele bestanden een oudere versie naar je werkkopie hebt gehaald, of als je lokale wijzigingen hebt aangebracht, dan worden exact die bestanden die je in je werkkopie hebt in de tak of label meegenomen. Uiteraard heeft een label van deze complexiteit mogelijk het gevolg dat er informatie vanuit je werkkopie naar het archief gestuurd moet worden, mocht de informatie daar nog niet bekend zijn.

Als je wilt dat je werkkopie automatisch overschakelt naar de nieuw aangemaakte tak, gebruik schakel dan de optie **Schakel werkkopie naar de nieuwe tak/label** in. Voordat je dat doet, moet je er zeker van zijn dat er geen wijzigingen meer in je werkkopie zitten. Als er wijzigingen in zitten, dan worden deze ook meteen samengevoegd in de werkkopie van de tak als je omschakelt.

Klik op OK om de nieuwe kopie vast te leggen in het archief. Vergeet niet een logboek boodschap achter te laten. Merk op dat de kopie gemaakt wordt *in het archief*.

Merk op dat, tenzij je ervoor gekozen hebt je werkkopie om te laten schakelen naar de nieuw aangemaakte tak, het aanmaken van een Tak of Label je werkkopie *niet* verandert. Ook als je een tak aanmaakt van je werkkopie, dan worden de wijzigingen in de tak vastgelegd, niet in de basislijn. Het is dus mogelijk dat je werkkopie nog steeds als gewijzigd wordt aangemerkt ten opzicht van de basislijn.

4.19.2. Ophalen of Wisselen...

.. dat is (niet echt) de vraag. Terwijl bij het ophalen alles van de gewenste tak uit het archief naar je werkkopie wordt gehaald, haalt TortoiseSVN → Wissel... alleen de gewijzigde gegevens op naar je werkkopie. Dat is gunstig voor de belasting van het netwerk en goed voor je geduld. :-)

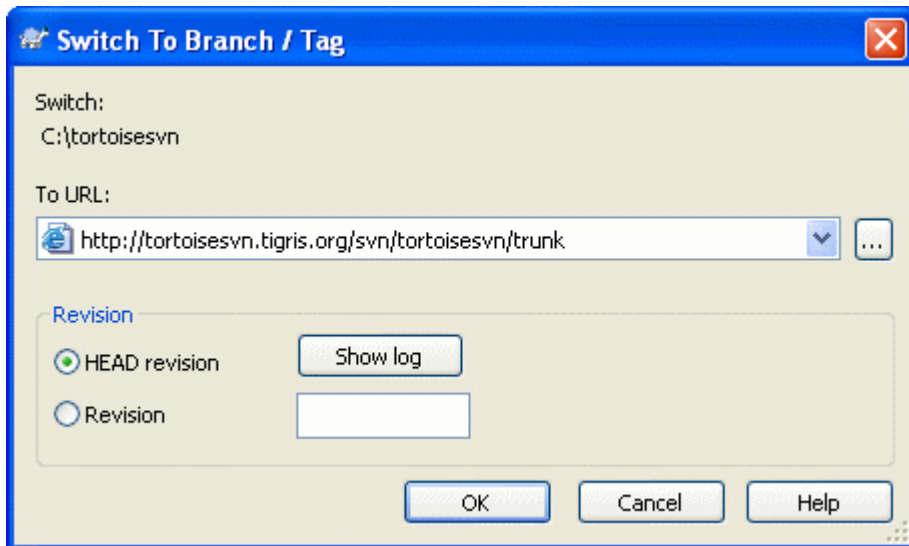
Er zijn verschillende mogelijkheden om met je nieuw aangemaakte tak of label te gaan werken. Je kunt:

- TortoiseSVN → Ophalen gebruiken om een versie kopie te maken in een lege map. Je kunt naar elke gewenste locatie op je harde schijf je gegevens ophalen en je kunt zoveel werkkopieën maken als je wilt.
- Je huidige werkkopie wisselen/schakelen naar de nieuw in het archief aangemaakte kopie. Selecteer weer de hoogste map van je project en kies TortoiseSVN → Wissel... vanuit het context menu.

Voer in het volgende venster de URL in van de tak die je net aangemaakt hebt. Selecteer het **HEAD Revisie** keuzerondje en klik op OK. Je werkkopie wordt dan gewisseld naar de nieuwe tak/label.

Wisselen werkt net zoals Verversen in zoverre dat het nooit de lokale wijzigingen negeert. Elke wijziging die je in je werkkopie hebt aangebracht en nog niet het vastgelegd in het archief, zal worden samengevoegd zodra je wisselt. Als je wilt voorkomen dat dit gebeurt, moet je of je wijzigingen vastleggen voordat je wisselt, of je werkkopie terugzetten naar een eerder vastgelegde revisie (meestal de HEAD revisie).

- Werken aan een basislijn en een tak, zonder dat je een verse werkkopie op moet halen. Je kunt dan met de Windows Verkenner een kopie maken van je basislijn naar een andere map en vervolgens met TortoiseSVN → Wissel... de werkkopie omschakelen naar de nieuwe tak.



Figuur 4.34. De Wissel Dialoog

Ondanks dat Subversion geen verschil maakt tussen labels en takken, verschilt de manier waarop ze gebruikt worden iets.

- Labels worden in principe gemaakt om een momentopname te maken van een project op een bepaald moment. Als gevolg hiervan, worden deze normaal gesproken niet gebruikt voor ontwikkeling. Hiervoor worden takken gebruikt. Daarom adviseren we gebruik te maken van de mapstructuur met de mappen `/trunk` `/branches` `/tags`. Het werken aan een label *is geen goed idee*, maar omdat er op je lokale bestanden geen schrijf beveiliging zit, zal niets je tegenhouden. Als je echter probeert wijzigingen vast te leggen naar een pad waarin `/tags/` voor komt, dan zal TortoiseSVN je wel waarschuwen.
- Het is soms nodig om extra wijzigingen aan te brengen in vrijgaves die je al gelabeld hebt. De juiste manier is om eerst een nieuwe tak te maken van de label en vervolgens de tak vast te leggen. Voor je wijzigingen door op deze tak en maak dan opnieuw een label aan van deze tak, bijv. `Versie_1.0.1`.
- Als je een werkkopie van een tak wijzigt en vastlegt, dan zullen de wijzigingen in de tak worden vastgelegd en *niet* in de basislijn. Alleen de wijzigingen worden opgeslagen. De rest blijft een goedkope kopie.

4.20. Samenvoegen

Nadat takken zijn gebruikt om meerdere ontwikkelrichtingen uit te gaan kan het op gegeven moment nodig zijn om de parallel ontwikkelde wijzigingen uit een tak weer terug samen te voegen in de basislijn of omgekeerd.

Aangezien het erg complex kan worden is het erg belangrijk om goed te begrijpen hoe het vertakken en samenvoegen in Subversion werkt voordat van start wordt gegaan. Het is sterk aan te bevelen om het hoofdstuk *Branching and Merging* [<http://svnbook.red-bean.com/en/1.5/svn.branchmerge.html>] uit het Subversion boek te lezen dat een volledige beschrijving inclusief voorbeelden geeft.

Een volgend belangrijk punt is dat samenvoegen *altijd* vanuit een werkkopie begint. Indien wijzigingen uit een tak *naar* een andere tak samengevoegd moeten worden, moet deze laatste op uw computer zijn afgehaald en moet van hier uit het samenvoegen worden geïnitieerd met het menu TortoiseSVN → Samenvoegen....

Er wordt over het algemeen aangeraden om samen te voegen naar een werkkopie waar alle wijzigingen zijn vastgelegd. Indien het samenvoegen niet naar verwachting zou gebeuren zullen *alle* niet vastgelegde

wijzigingen anders mee ongedaan worden gemaakt bij gebruik van het **Ongedaan Maken** commando, en dus ook de wijzigingen van voor het samenvoegen.

Er zijn drie situaties waar samenvoegen telkens op een verschillende manier werkt, zoals hieronder beschreven. De eerste vraag die wordt gesteld bij het Samenvoegen is welke methode gewenst is.

Een revisiebereik samenvoegen

Deze methode wordt gebruikt als u één of meer wijzigingen in een tak (of de basislijn) heeft aangebracht en deze ook wilt toepassen op een andere tak.

Wat hier aan Subversion wordt gevraagd is het volgende: “Bereken de benodigde wijzigingen om vanuit [FROM] revisie 1 van tak A [TO] revisie 7 van tak A te gaan, en pas deze wijzigingen toe op mijn werkkopie (uit basislijn of tak B).”

Een tak reïntegreren

Deze methode wordt gebruikt als u een 'feature branch' heeft zoals besproken in het Subversion boek. Wekelijks zijn alle wijzigingen uit de basislijn ook in de 'feature branch' samengevoegd. Nu de nieuwe ontwikkeling in de 'feature branch' is afgewerkt mag deze naar de basislijn worden samengevoegd. Aangezien de 'feature branch' gedurende zijn ontwikkeling steeds is gesynchroniseerd met de basislijn, is de basislijn ook zijn werkbasis gebleven.

Dit is een bijzonder geval van het boom samenvoegen dat in het volgende punt wordt besproken. Vanuit de basislijn is enkel de URL naar de desbetreffende 'feature' tak nodig. Subversion kan dankzij zijn wijziging-controle-routines zowel zelf terugvinden welk revisiebereik nodig is evenals indien alle wijzigingen van de basislijn correct zijn samengevoegd naar de tak. Dit voorkomt dat eventueel werk van anderen in de basislijn na de laatste synchronisatie niet per ongeluk ongedaan wordt gemaakt.

Na het samenvoegen zijn alle wijzigingen die in de tak zijn gebeurd meegenomen in de basislijn. De tak is nu overtollig geworden en kan worden verwijderd.

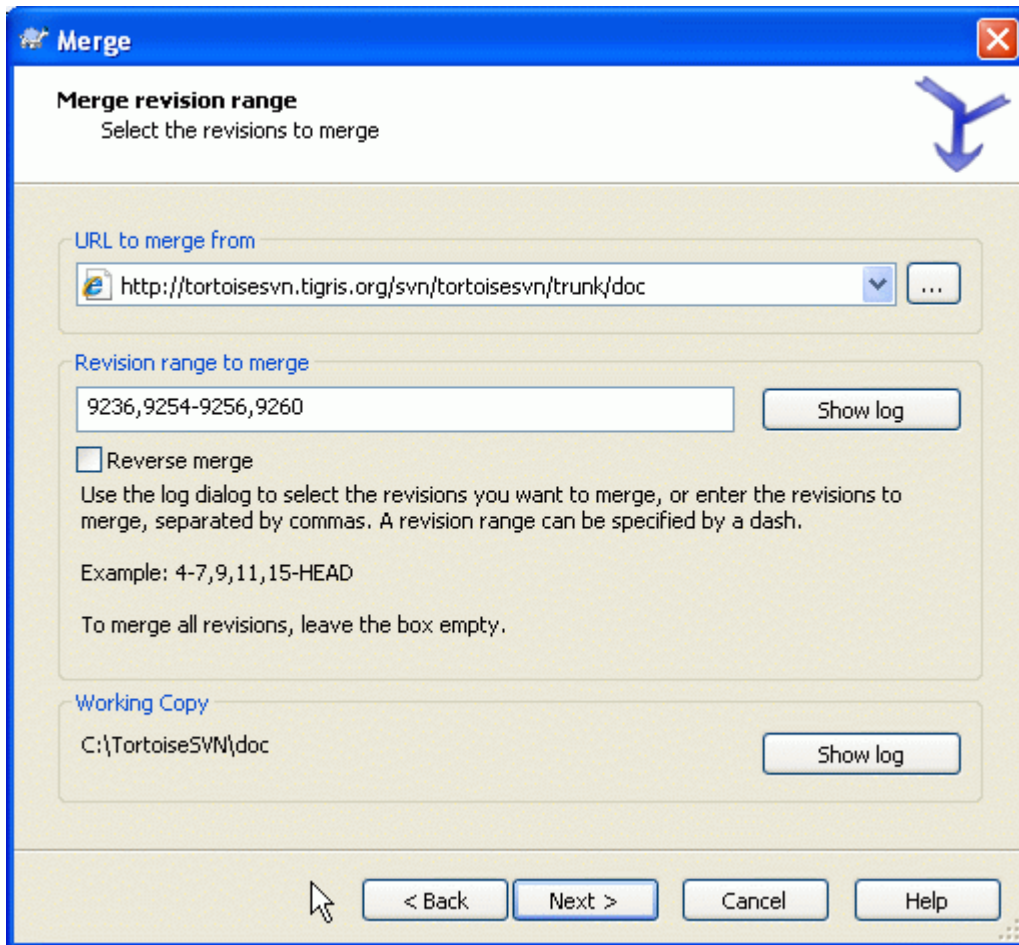
Eenmaal de tak terug in de basislijn is samengevoegd mag deze niet meer opnieuw worden gebruikt voor een volgende ontwikkeling. De reden is dat wanneer later opnieuw zou worden samengevoegd naar de basislijn, de wijziging-controle-routines van Subversion zullen merken dat de wijzigingen aan de basislijn van de vorige reïntegratie zelf nog niet zijn toegepast op de tak, en zal het de tak-naar-basislijn wijzigingen opnieuw in de tak proberen toepassen! De enige correcte oplossing is eenvoudigweg een nieuwe tak te maken om vervolgonwikkelingen in te doen.

Twee verschillende takken samenvoegen

Dit is een meer algemene variant van vorige tak reïntegratie methode. Hierbij wordt aan Subversion het volgende gevraagd: “Bereken wijzigingen die nodig zijn om [VAN] de HEAD revisie van de basislijn [NAAR] de HEAD revisie van de tak te gaan, en pas deze wijzigingen toe op mijn werkkopie (uit de basislijn).” Het resultaat is dat de basislijn precies hetzelfde bevat als de tak.

Indien uw Subversion server geen wijziging-controle-routines ('merge-tracking') ondersteunt is dit de enige manier om een tak opnieuw in de basislijn samen te voegen. Een andere situatie waar deze vorm van samenvoegen wordt gebruikt is wanneer een project zogenaamde verkopers-takken ('vendor branches') heeft en een nieuwe versie van de verkopers-tak in de basislijn moet worden gevoegd. Meer informatie is te vinden in het Subversion boek hoofdstuk *vendor branches* [<http://svnbook.red-bean.com/en/1.5/svn.advanced.vendorbr.html>].

4.20.1. Een Revisie-Bereik Samenvoegen



Figuur 4.35. Het Samenvoeg Hulpprogramma - Selecteer Revisie Bereik

Geef in het Van: veld de complete folder URL van de tak of label, welke de wijzigingen bevat die u naar uw werkkopie wilt doorvoeren. U kunt ook klikken op ... om door het archief te bladeren naar de gewenste tak. Als u eerder van deze tak heeft samengevoegd, kunt u de uitklaplijst gebruiken welke een geschiedenis van eerder gebruikte URL's toont.

Geef in het Revisiebereik om samen te voegen veld de lijst aan van revisies welke u wilt samenvoegen. Dit kan een enkele revisie zijn, een lijst van specifieke revisies gescheiden door komma's, een bereik van revisies gescheiden door een min-teken, of een combinatie van deze manieren.



Belangrijk

Er is een belangrijk verschil in de manier waarop een revisiebereik wordt gespecificeerd in TortoiseSVN of in de commandoregel client. De gemakkelijkste manier om dit te visualiseren is om een omheining voor ogen te nemen compleet met toegangspoorten en hekwerk.

Met de commandoregel client geeft u de vast te leggen wijzigingen aan door gebruik te maken van twee “toegangspoort” revisies welke de *voor* en *na* punten aangeven.

Met TortoiseSVN geeft u de vast te leggen wijzigingsset aan door gebruik te maken van “hekwerken”. De reden hiervoor wordt duidelijk wanneer u het log-dialoog gebruikt om revisies aan te geven om samen te voegen, waarbij elke revisie zichtbaar is als een wijzigingsset.

Als u samenvoegt in blokken, laat de methode welke in het subversion boek wordt getoond u 100-200 nu samenvoegen en 200-300 de volgende keer. Met TortoiseSVN zou u 100-200 nu samenvoegen en 201-300 de volgende keer.

Dit verschil heeft een hoop tam-tam in de mailing lijsten opgeleverd. We geven toe dat er een verschil is met de commandoregel client, maar we geloven dat het voor de meerderheid van de GUI-gebruikers de methode welke wij hebben toegepast gemakkelijker is om te begrijpen.

De makkelijkste methode om een reeks van revisies te selecteren die je nodig hebt, is door op de **Toon logboek** te klikken. Hiermee wordt een lijst met recente wijzigingen en de bijbehorende logboek boodschappen getoond. Als je de wijzigingen van een enkele revisie wilt samenvoegen, selecteer dan die ene revisie. Als je de wijzigingen van meerdere revisies wilt samenvoegen, selecteer dan de reeks terwijl je de **Shift** knop gebruikt. Klik op de **OK** knop en lijst met revisienummers die samengevoegd worden, wordt voor je gevuld.

Als je wijzigingen van een samenvoeging weer *uit* je werkkopie wilt halen, om een wijziging die al vastgelegd is weer ongedaan te maken, select dan de revisies die ongedaan gemaakt moeten worden en zorg er dan voor dat de optie **Omgekeerd samenvoegen** aangevinkt is.

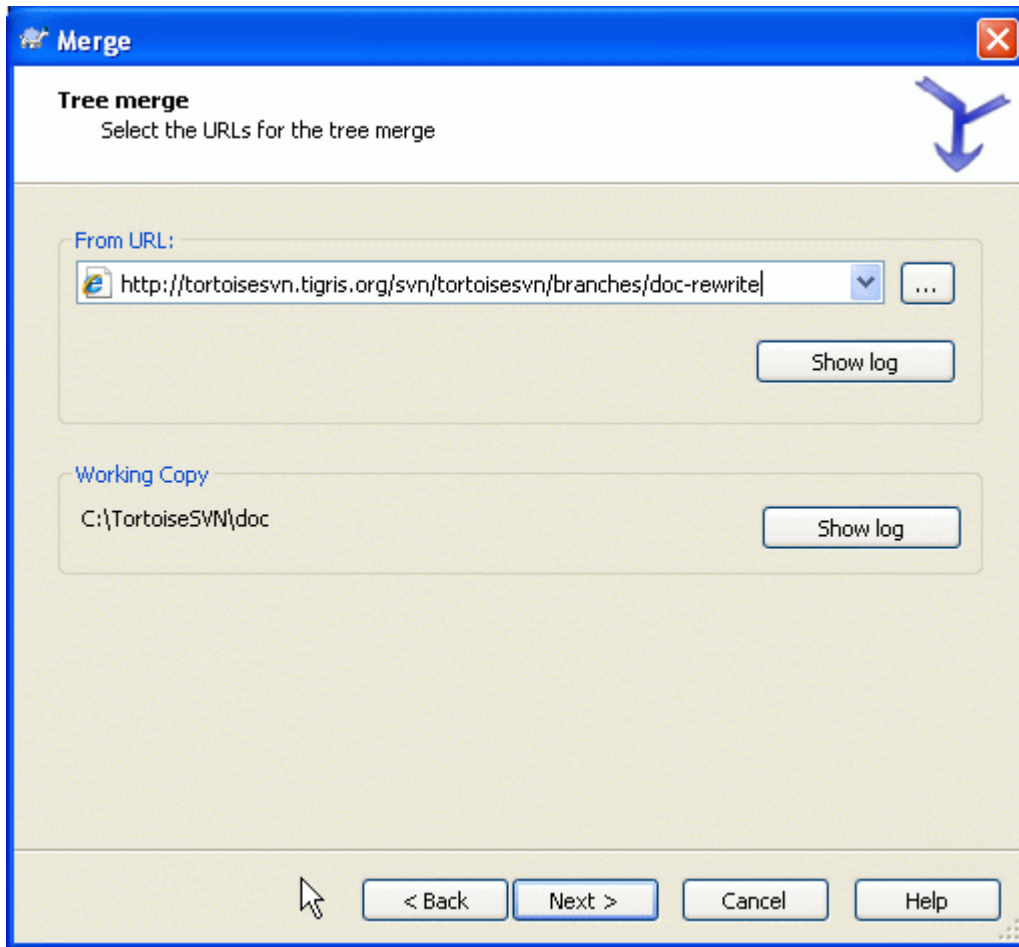
Als je al enkele wijzigingen van deze tak hebt samengevoegd, hopelijk heb toen je in het logboek bericht opgenomen wat de laatste samengevoegde revisie was toen je de wijzigingen vastlegde. Je kunt dan **Toon logboek** gebruiken voor de werkkopie, om zo de logboek berichten te kunnen bekijken. Hou in gedachten dat we revisies als een set van wijzigingen zien, waardoor je als startpunt voor de nieuwe samenvoeging de eerste revisie na de vorige samenvoeging moet selecteren. Als je bijvoorbeeld de vorige keer revisie 37 naar revisie 39 hebt samengevoegd, dan moet je de nieuwe samenvoeging beginnen op revisie 40.

Als de samenvoeg volg functies van Subversion gebruikt, dan hoeft je niet te onthouden welke revisies je al samengevoegd hebt - Subversion onthoudt dat voor je. Als je de revisie reeks leeg laat, worden alle revisies die nog niet samengevoegd zijn meegenomen. Lees [Paragraaf 4.20.6, "Samenvoegen traceren"](#) voor meer informatie.

Als er andere personen mogelijk wijzigingen aan het vastleggen zijn, wees dan voorzichtig met het gebruik van de HEAD revisie. Het kan zijn dat deze niet wijst naar de revisie die je verwacht, bijvoorbeeld als er iemand wijzigingen heeft vastgelegd nadat jij de laatste keer ververst hebt.

Klik op **Volgende** en lees verder bij [Paragraaf 4.20.4, "Samenvoegopties"](#).

4.20.2. Een tak reïntegreren



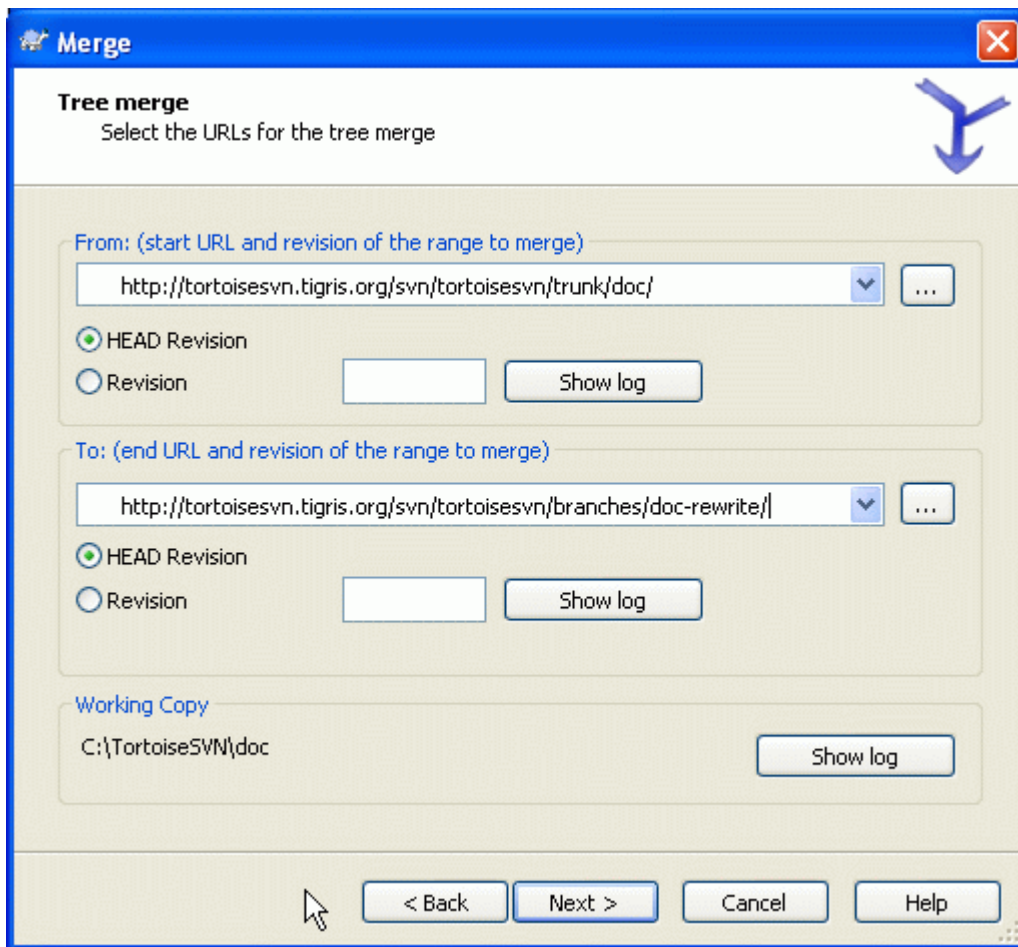
Figuur 4.36. Het Samenvoeg Hulpprogramma - Herintegreer Samenvoegen

Om een tak met nieuwe functies naar de basislijn samen te voegen, moet je het samenvoeg hulpprogramma opstarten vanuit een werkkopie van de basislijn.

Voer in het Van URL: veld de volledige URL in van de map van de tak die je wilt samenvoegen. Je kunt ook op de ... klikken om door het archief te bladeren.

Er zijn enkele voorwaarden voor een reïntegratie samenvoeging. Ten eerste moet de server samenvoeg tracking ondersteunen. De werkkopie moet een onbegrensde diepte hebben (dus geen gedeeltelijk opgehaalde versie), er mogen geen lokale wijzigingen zijn gemaakt, objecten gewisseld zijn of objecten van een andere revisie dan de HEAD revisie. Alle wijzigingen die aan de basislijn zijn doorgevoerd gedurende de ontwikkeling van de tak, moeten in de tak samengevoegd zijn (of gemarkeerd als samengevoegd zijn). De reeks van de samen te voegen revisies wordt automatisch berekend.

4.20.3. Twee Verschillende Takken Samenvoegen



Figuur 4.37. Het Samenvoeg Hulpprogramma - Samenvoegen van Boomstructuren

Als je deze methode gebruikt om een functie tak samen te voegen naar de basislijn, dan moet je het samenvoeg hulpprogramma starten vanuit de werkkopie van de basislijn.

Voer in het Van: veld de volledige URL van de map van de *basislijn*. Dit lijkt verkeerd te zijn, maar onthoudt dat de basislijn het startpunt is waar je de wijzigingen van de tak aan toe wilt voegen. Je kunt ook op de ... klikken om het archief te verkennen.

Voer in het Naar: veld de volledige URL van de map van de nieuwe functie tak.

Vul in zowel het Van Revisie veld als in het Naar Revisie veld het laatste revisie nummer in, waarop de beide boomstructuren gesynchroniseerd zijn. Als je er zeker van bent dat niemand anders wijzigingen vastlegt, dan kun je voor beide de HEAD revisie gebruiken. Als er een kans is dat iemand anders wijzigingen na de laatste synchronisatie heeft vastgelegd, gebruik dan de exacte revisie nummers om verlies van recente wijzigingen te voorkomen.

Je kunt ook met Toon logboek de revisie selecteren.

4.20.4. Samenvoegopties

Dit scherm van het hulpprogramma geeft je de mogelijkheid om geavanceerde opties in te stellen, voordat het samenvoegproces start. In de meeste gevallen kun je de standaard instellingen gebruiken.

Je kunt de diepte instellen die gebruikt moet worden voor het samenvoegen, dat houdt in hoe diep in je werkkopie de samenvoegactie doorgevoerd moet worden. De definitie van de diepte wordt beschreven

in **Paragraaf 4.3.1, “Ophaaldiepte”**. De standaard diepte staat ingesteld op **Werkkopie**, waardoor de bestaande diepte instelling gebruikt wordt. Dit is eigenlijk altijd de diepte die je wilt gebruiken.

Meestal wil je dat er bij het samenvoegen rekening gehouden wordt met de geschiedenis van een bestand, zodat wijzigingen aan een gemeenschappelijke oorsprong worden meegenomen en samengevoegd. Soms zul je bestanden samenvoegen die mogelijk gerelateerd zijn, maar niet in jouw archief. Bijvoorbeeld als je versie 1 en 2 van een bibliotheek van een derde partij hebt geïmporteerd in verschillende mappen. Ondanks dat ze een logische relatie hebben, heeft Subversion hier geen weet van omdat deze alleen de tarballs ziet die je hebt geïmporteerd. Als je probeert de verschillen tussen deze twee boomstructuren samen te voegen, dan zul je een complete verwijdering en een complete toevoeging zien plaatsvinden. Om er voor te zorgen dat Subversion alleen pad-gebaseerde in plaats van geschiedenis-gebaseerde verschillen gebruikt, selecteer dan de **Negeer oorsprong** optie. Lees hier meer over in het Subversion boek, *Noticing or Ignoring Ancestry* [<http://svnbook.red-bean.com/en/1.5/svn.branchmerge.advanced.html#svn.branchmerge.advanced.ancestry>].

Je kunt ook opgeven hoe regeleinden, spaties en witregels behandeld moeten worden. Deze opties worden beschreven in **Paragraaf 4.10.2, “Opties voor regeleinden, spaties en witregels”**. Standaard worden alle regeleinden, spaties en witregels als echte wijzigingen behandeld en samengevoegd.

Als je gebruik maakt van het traceren van samenvoegingen en je wilt een revisie markeren als zijnde samengevoegd, zonder dat de feitelijke samenvoeging hier heeft plaatsgevonden, selecteer dan de **Samenvoegen alleen vastleggen** optie. Er zijn twee mogelijke redenen waarom je deze optie zou willen gebruiken. Het kan zijn dat het samenvoegen te gecompliceerd is voor de gebruikte algoritmes. Je codeert de wijzigingen dan handmatig, vervolgens markeer je de wijzigingen als samengevoegd, zodat het samenvoeg traceer algoritme dit ook ziet. De andere mogelijke reden is dat je wilt voorkomen dat een bepaalde revisie samengevoegd wordt. Door deze te markeren als reeds samengevoegd voorkom je dat de samenvoeging plaatsvindt, zolang de samenvoeg programma's rekening kunnen houden met de samenvoeg traceer informatie.

Alles is nu ingesteld en je hoeft alleen nog maar op de **Samenvoegen** knop te drukken. Als je vooraf het resultaat wilt beoordelen, kun je met **Test Samenvoegen** de samenvoeging simuleren. Hiermee wijzig je niets in je werkkopie. Het toont een lijst met bestanden die gewijzigd zullen worden bij de echte samenvoeg actie en geeft de bestanden aan waarmee conflicten op zouden *kunnen* treden. Omdat het traceren van de samenvoegingen het proces veel gecompliceerder maakt, is er geen manier om van te voren zeker te weten of er een conflict zal optreden of niet. Oftewel bestanden die als conflicterend worden aangemerkt bij een samenvoeg test hoeven in de praktijk helemaal geen probleem op te leveren.

In het voortgangsscherm wordt elke stadium van de samenvoeging getoond met de betrokken revisie reeksen. Het kan zijn dat er hier meer revisies getoond worden, dan je had verwacht. Bijvoorbeeld als je revisie 123 ging samenvoegen, kan het voortgangsscherm “Samenvoegen van revisie 122 tot 123” tonen. Om dit te begrijpen moet je je realiseren dat Samenvoegen gerelateerd is met Verschillen/Diff. Het samenvoegproces maakt eerst een lijst met verschillen tussen twee punten in het archief en past deze toe op je werkkopie. Het voortgangsscherm geeft het start- en eindpunt van de verschil/diff weer.

4.20.5. De Resultaten van het Samenvoegen Bekijken

Het samenvoegen is nu klaar. Het is verstandig om het resultaat te bekijken om na te gaan of het is zoals verwacht. Samenvoegen is vaak erg gecompliceerd. Conflicten komen vaak voor als een tak ver van de basislijn is afgeweken.

Bij Subversion clients en servers van voor versie 1.5, is er geen samenvoeg informatie opgeslagen en samengevoegde revisies moeten handmatig achterhaald worden. Als je de wijzigingen getest hebt en klaar bent om deze revisie vast te leggen, zou je logboek boodschap *altijd* de revisienummers van de geporteerde revisies moeten bevatten. Als je op een later moment nog een samenvoeging wilt uitvoeren, dan moet je weten wat er al samengevoegd is, omdat je wijzigingen niet vaker dan één maal wilt doorvoeren. Lees voor meer informatie *Best Practices for Merging* [<http://svnbook.red-bean.com/en/1.4/svn.branchmerge.copychanges.html#svn.branchmerge.copychanges.bestprac>] in het Subversion boek.

Als je server en alle clients Subversion versie 1.5 of hoger gebruiken, zullen de samenvoeg acties getraceerd worden, zodat voorkomen wordt dat wijzigingen meer dan eens worden toegepast. Dit maakt

het leven een stuk makkelijker, omdat je dan elke keer de hele revisie reeks kunt samenvoegen en dat dan alleen de nieuwe revisies samengevoegd zullen worden.

Het managen van takken is belangrijk. Als je deze tak dicht bij de basislijn wilt houden, zul je vaak moeten samenvoegen, zodat de tak en de basislijn niet te ver uit elkaar drijven. Natuurlijk moet je het herhaaldelijk samenvoegen van wijzigingen voorkomen, zoals eerder beschreven.



Tip

Als je net een functie tak hebt samengevoegd naar de basislijn, dan heeft de basislijn nu alle nieuwe functionaliteit en is de tak verouderd. Je kunt de tak nu verwijderen uit het archief indien nodig.



Belangrijk

Subversion kan niet bestanden met mappen samenvoegen en vice versa - alleen mappen naar mappen en bestanden naar bestanden. Als je op een bestand klikt en het samenvoeg scherm opent, dan moet je het pad naar een bestand opgeven. Als je een map selecteert en samenvoeg scherm opent, dan moet je een URL van een map opgeven om samen te voegen.

4.20.6. Samenvoegen traceren

In Subversion 1.5 is functionaliteit geïntroduceerd voor het traceren van samenvoegingen. Als je wijzigingen vanuit de ene boomstructuur naar de andere samenvoegt, worden de samengevoegde revisie nummers opgeslagen. Deze informatie kan dan voor verschillende doeleinden gebruikt worden.

- Het is mogelijk om het gevaar van het twee keer samenvoegen van een revisie te voorkomen. Als eenmaal een revisie gemarkeerd is als samengevoegd, zal er bij toekomstige samenvoegen waarbij deze revisie in de reeks is opgenomen, deze revisie overgeslagen worden.
- Als je een tak terug naar de basislijn samenvoegt, dan kan het logboek scherm de vastleggingen aan de tak deel laten uitmaken van het logboek van de basislijn. Hierdoor is het makkelijker om wijzigingen te traceren.
- Als je het logboek vanuit het samenvoeg scherm opvraagt, worden de revisies die al samengevoegd zijn grijs weergegeven.
- Als je de verklaren informatie bekijkt voor een bestand, dan kun je er voor kiezen de originele auteur te tonen van samengevoegde revisie, in plaats van de persoon die de samenvoeging uitvoerde.
- Je kunt revisies markeren met *niet samenvoegen* door deze al in de lijst van samengevoegde revisies op te nemen, zonder de echte samenvoeging uit te voeren.

De samenvoeg informatie is opgeslagen in de `svn:mergeinfo` eigenschap door de client als deze een samenvoeging uitvoert. Als de samenvoeging vastgelegd wordt, dan slaat de server de gegevens op in de database. Als je samenvoeg, logboek of verklaren informatie opvraagt, dan kan de server deze informatie doorgeven. Om dit goed te laten werken moet je er voor zorgen dat de server, het archief en alle clients naar de nieuwe versie bijgewerkt worden. Oudere clients zullen de `svn:mergeinfo` eigenschappen niet opslaan en oudere servers zullen de door de clients opgevraagde informatie niet opgeven.

Voor meer informatie over het traceren van samenvoegingen, lees Subversion's [Merge tracking documentatie](http://subversion.tigris.org/merge-tracking/index.html) [http://subversion.tigris.org/merge-tracking/index.html].

4.20.7. Conflicten Oplossen tijdens Samenvoegen

Het samenvoegen gaat niet altijd soepel. Soms treedt er een conflict op en als je meerdere reeksen samenvoegt, wil je meestal een conflict oplossen voordat het samenvoegen van de volgende reeks start. TortoiseSVN helpt je door dit proces heen met het *Samenvoeg Conflict Onderzoeksscherm*.



Figuur 4.38. Het Samenvoeg Conflict Onderzoeksscherm

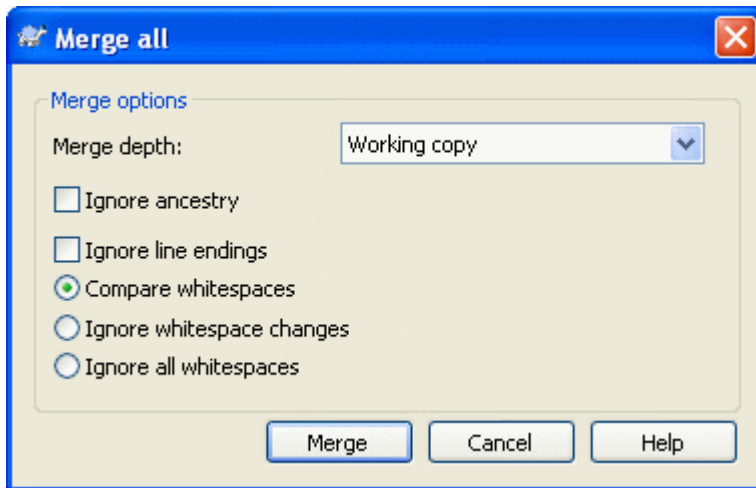
Als er een conflict optreedt tijdens het samenvoegen, dan zijn er drie methodes om daarmee op te gaan.

1. Je kunt er voor kiezen dat je lokale wijzigingen belangrijker zijn, zodat je de versie in het archief negeert en je lokale versie behoudt, of andersom. Dit heeft alleen betrekking op conflicterende wijzigingen. Wijzigingen die zonder problemen samengevoegd kunnen worden, worden gewoon samengevoegd. Alleen voor de onderdelen die conflicteren wordt je keuze gebruikt.
2. Normaliter zou je de conflicten willen bekijken en deze oplossen. In dat geval klik je op **Bewerk Conflict**, waardoor je samenvoeg programma gestart wordt. Als je tevreden bent met het resultaat klik je op **Opgelost**.
3. De laatste mogelijkheid is om het oplossen uit te stellen en door te gaan met het samenvoegen. Je kunt hiervoor kiezen voor het huidige conflicterende bestand of voor alle bestanden voor de rest van de samenvoeging. Echter, als er verder nog wijzigingen zijn in dat bestand, dan zal het niet mogelijk zijn om de samenvoeging af te ronden.

Als je niet gebruik wilt maken van deze interactieve onderzoeksmethode, dan kun je gebruik maken van het selectievak **Samenvoegen zonder tussenkomst** in het samenvoeg voortgangsscherm. Als deze optie aangevinkt is en er treedt een conflict op met een bestand, dan zal dat bestand als conflicterend gemarkeerd worden en gaat de samenvoeging verder. Je moet het conflict dan oplossen nadat de hele samenvoeging afgerond is. Als deze optie niet aangevinkt is, dan krijg je de kans om een conflict met een bestand op te lossen *tijdens* het samenvoegen, voordat het als conflicterend gemarkeerd wordt. Dit heeft als voordeel, dat als een bestand meerdere keren wordt samengevoegd (meerdere revisies voeren een wijziging door aan dat bestand), de opeenvolgende samenvoegingen succesvol kunnen verlopen afhankelijk van de regels die veranderd zijn. Maar je kunt natuurlijk niet even koffie gaan halen als een samenvoeging bezig is. ;)

4.20.8. Samenvoegen van een Afgehandelde Tak

Als je alle wijzigingen van een tak met nieuwe functies wilt samenvoegen naar de basislijn, dan kun je gebruik maken van TortoiseSVN → **Reïntegratie samenvoeging...** vanuit het uitgebreide context menu (houdt de **Shift** knop ingedrukt als je met rechts op een bestand klikt).



Figuur 4.39. De Samenvoeg Re-integratie Dialoog

Dit scherm is heel eenvoudig. Het enige wat je hoeft te doen is de opties voor de samenvoeging in te stellen, zoals beschreven in [Paragraaf 4.20.4, “Samenvoegopties”](#). De rest wordt automatisch door TortoiseSVN gedaan met gebruik van samenvoeg traceer informatie.

4.20.9. Onderhoud aan een Tak met Nieuwe Functies

Als je een nieuwe functie ontwikkelt in een aparte tak, dan is het verstandig om een beleid af te spreken voor het reïntegreren als de functie klaar is. Als er op het zelfde moment gewoon doorgewerkt wordt aan de basislijn, dan kun je er tegenaan lopen dat wijzigingen steeds meer gevolgen hebben, waardoor het samenvoegen echt een nachtmerrie kan worden.

Als de nieuwe functie relatief eenvoudig is en de ontwikkeling niet lang duurt, dan kun je een vrij eenvoudige aanpak gebruiken. Hou de tak helemaal gescheiden totdat de functie klaar is en voeg de wijzigingen in de tak dan samen naar de basislijn. In het samenvoeg hulpprogramma zou die een eenvoudige **Voeg een reeks revisies samen** actie zijn met de reeks van revisies van de tak.

Als de ontwikkeling langer duurt en je rekening moet gaan houden met wijzigingen in de basislijn, dan moet je je tak synchroniseren met de basislijn. Dit houdt gewoon in dat je regelmatig de wijzigingen in de basislijn moet samenvoegen naar je tak, zodat de tak de wijzigingen van de basislijn *plus* de nieuwe functie heeft. Het synchronisatie proces gebruikt de **Voeg een reeks revisies samen** functie. Als de functie klaar is, kun je deze samenvoegen naar de basislijn met **Reïntegreer een tak** of met **Twee verschillende bomen samenvoegen**.

4.21. Blokkeren

Subversion werkt het beste als er geen gebruik wordt gemaakt van blokkeren, als je gebruik maakt van de “Kopieer-Wijzig-Samenvoeg” methode die eerder beschreven is in [Paragraaf 2.2.3, “De Kopieer-Wijzig-Samenvoeg Oplossing”](#). Er zijn echter enkele gevallen waarbij je een zekere vorm van blokkeerbeleid moet gebruiken.

- Je maakt gebruik van “niet samen te voegen” bestanden, bijvoorbeeld grafische bestanden. Als twee mensen hetzelfde bestand wijzigen, dan is samenvoegen niet mogelijk. Oftewel, één van de twee zal de wijzigingen kwijtraken.
- Jouw bedrijf heeft in het verleden altijd gebruik gemaakt van een versiebeheersysteem met blokkeringsmethode en er is een management besluit genomen dat “blokkeren het beste is”.

Allereerst moet je er zeker van zijn dat je Subversion server is bijgewerkt tot minimaal versie 1.2. Eerdere versies ondersteunen blokkeren helemaal niet. Als je gebruik maakt van `file://`, hoeft natuurlijk alleen maar je eigen programma bij te werken.

4.21.1. Hoe Het Blokkeren Werkt In Subversion

Standaard wordt er niets geblokkeerd en kan iedereen die de rechten heeft wijzigingen op elk moment vastleggen in het archief. Anderen zullen hun werkkopieën regelmatig moeten verversen en de wijzigingen in het archief worden dan samengevoegd met de lokale wijzigingen.

Als je een bestand *blokkeert*, dan heb jij alleenrecht op het vastleggen van dat bestand. Vastleggingen van andere gebruikers zullen geblokkeerd worden totdat jij de blokkade opheft. Een geblokkeerd bestand kan op geen enkele manier in het archief gewijzigd worden, dat houdt ook in dat niemand anders dan de eigenaar van de blokkade het bestand kan verwijderen of hernoemen.

Andere gebruikers zullen echter niet direct weten dat jij een bestand geblokkeerd hebt, tenzij ze regelmatig de blokkade status opvragen. Het eerste moment waarop ze er meestal pas achter komen is als ze wijzigingen proberen vast te leggen, wat niet altijd handig is. Om het werken met blokkades te vereenvoudigen is er een nieuwe Subversion eigenschap geïntroduceerd, `svn:needs-lock`. Als deze eigenschap (met een willekeurige waarde) aan een bestand is gekoppeld, dan wordt het lokale bestand altijd als alleen-lezen gemarkeerd als deze opgehaald wordt uit het archief, *tenzij/totdat* die werkkopie het bestand blokkeert. Op deze manier wordt je gewaarschuwd dat je dit bestand niet moet wijzigen voordat je het geblokkeerd hebt. Bestanden, die voorzien zijn van versiebeheer en als alleen-lezen zijn gemarkeerd, hebben in TortoiseSVN een apart overlappend pictogram waarmee aangegeven wordt dat je eerst het bestand moet blokkeren.

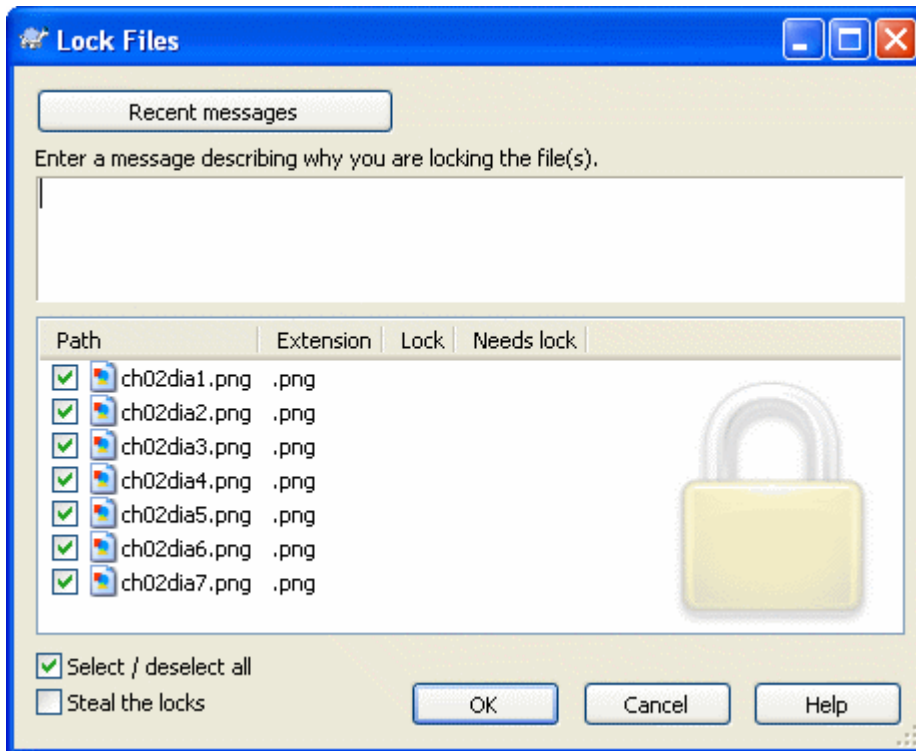
Blokkades worden geadmineistreerd met de werkkopie locatie en de eigenaar. Als je meerdere werkkopieën hebt (thuis, op het werk), dan kun je in maar *één* werkkopie zo'n blokkade initiëren.

Stel dat één van je medewerkers zo'n blokkade instelt en vervolgens op vakantie gaat, wat doe je dan? Subversion heeft de mogelijkheid om blokkades te forceren. Het vrijgeven van een blokkade die door iemand anders is ingesteld wordt aangeduid als het *Verbreken* van een blokkade. Het geforceerd toe-eigenen van een blokkade die door iemand anders is ingesteld, wordt het *Stelen* van een blokkade genoemd. Hier moet je natuurlijk voorzichtig mee omgaan als je vrienden wilt blijven met je collega's.

Blokkades worden geadmineistreerd in het archief en er wordt ook een merkteken (een zogenaamde token) in je lokale werkkopie geplaatst. Als er een discrepantie is opgetreden, bijvoorbeeld als iemand anders de blokkade geforceerd heeft, dan zal het lokale blokkade merkteken ongeldig worden. De status in het archief is altijd leidend.

4.21.2. Blokkade verkrijgen

Selecteer de bestanden in je werkkopie die je wilt blokkeren. Vervolgens selecteer je het commando TortoiseSVN → Plaats blokkade....



Figuur 4.40. Het blokkerings scherm

Een apart venster verschijnt, waarin je aan anderen kunt aangeven waarom je de bestanden geblokkeerd hebt. Het invullen is optioneel en wordt op dit moment alleen gebruikt in archieven die op basis van Svnserve opgezet zijn. Alleen als je *echt* een blokkade moet stelen van iemand anders, moet je de **Steal de blokkades** selectie aanvinken om de blokkade te stelen. Klik vervolgens op OK om de actie in gang te zetten.

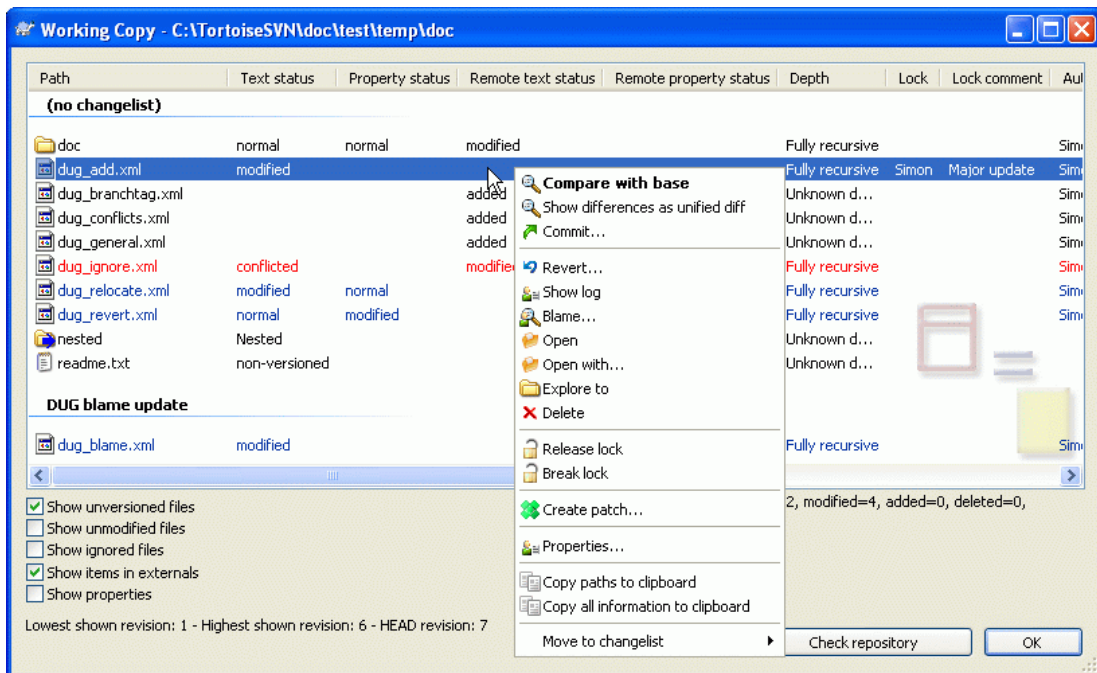
Als je een map selecteert en vervolgens het commando TortoiseSVN → Plaats blokkade... kiest, zal het blokkerings scherm geopend worden voor *elk* bestand in de map *en alle* bestanden in de sub-mappen. Als je echt de hele hiërarchie wilt blokkeren, dan is dat de methode. Je collega's zullen er mogelijk niet blij van worden als je hun helemaal buiten het project sluit. Gebruik dit dus met verstand...

4.21.3. Een Blokkade Vrijgeven

Om te voorkomen dat je vergeet een blokkade vrij te geven, worden geblokkeerde bestanden weergegeven in het vastleggen venster en standaard geselecteerd. Als je dan doorgaat met het vastleggen, zullen blokkades van de geselecteerde bestanden verwijderd worden, zelfs als er niets gewijzigd is. Als je de blokkade op bepaalde bestanden niet wilt vrijgeven, dan kun je deze de-selecteren (als ze niet gewijzigd zijn). Als je de blokkade op gewijzigde bestanden wilt behouden, dan moet je het **Blokkades handhaven** selectievak aanvinken voordat je je wijzigingen vastlegt.

Als je een blokkade handmatig wilt opheffen, moet je de bestanden in je werkkopie selecteren waarvoor je de blokkade wilt opheffen. Vervolgens kies je het commando TortoiseSVN → Blokkade opheffen. Er hoeft verder geen informatie ingevoerd te worden, dus TortoiseSVN zal het archief benaderen en de blokkades opheffen. Je kunt dit commando ook op mappen toepassen om alle blokkades recursief op te heffen.

4.21.4. Blokkadestatus Controleren



Figuur 4.41. Het Kijk of er updates zijn scherm

Om na te gaan welke blokkades er door jou of anderen geplaatst zijn, kun je gebruik maken van TortoiseSVN → Kijk of er updates zijn.... Lokale blokkade merktekens worden meteen getoond. Om de blokkades van anderen te zien (en om na te gaan of jouw blokkades verbroken of gestolen zijn) moet je op **Controleer archief** klikken.

Via het context menu, kun je hier ook blokkades plaatsen en opheffen, maar ook blokkades van anderen verbreken en stelen.



Voorkom het verbreken en stelen van blokkades

Het verbreken of stelen van iemand anders blokkade zonder dat die persoon daarvan op de hoogte is, kan verlies van gegevens tot gevolg hebben. Als je iemand zijn blokkade steelt en werkt met niet samenvoegbare bestandsformaten, dan kan iemand anders jouw wijzigingen overschrijven op het moment dat jij jouw blokkade opheft en zij hun wijzigingen vastleggen in het archief. Subversion raakt geen informatie kwijt, maar je verliest wel de beveiliging de blokkade functionaliteit je biedt bij het werken in teamverband.

4.21.5. Niet-geblokkeerde bestanden als Alleen-lezen instellen

Zoals eerder beschreven, kun je blokkades het effectiefst gebruiken door gebruik te maken van de `svn:needs-lock` eigenschap. Lees [Paragraaf 4.17, “Project Instellingen”](#) voor meer informatie over het instellen van eigenschappen. Bestanden die deze eigenschap hebben, krijgen meteen de alleen-lezen eigenschap mee als ze worden opgehaald uit het archief, tenzij jouw werkkopie het bestand geblokkeerd heeft.



Ter herinnering, TortoiseSVN gebruikt een apart overlappend pictogram om deze situatie aan te geven.

Als je een beleid hanteert waardoor elk bestand geblokkeerd moet worden, dan is het waarschijnlijk makkelijker om automatische-eigenschappen functie van Subversion te gebruiken. Hiermee wordt

automatisch dezelfde eigenschap ingesteld als je een nieuwe bestand toevoegt. Lees ook [Paragraaf 4.17.1.5, “Automatisch eigenschappen zetten”](#) voor meer informatie.

4.21.6. Hook Scripts Voor Blokkeringen

Als je een nieuw archief aanmaakt met Subversion 1.2 of hoger, worden er vier hook sjablonen aangemaakt in de map `hooks` van het archief. Deze worden aangeroepen voor en na het plaatsen van een blokkade, en voor en na het opheffen van een blokkade.

Het is verstandig om op de server een `na-het-blokkeren` en `na-het-deblokkeren` hook script te installeren, waardoor een email verstuurd wordt waarin aangegeven wordt welk bestand ge(de)blokkeerd is. Met zo'n script kunnen alle gebruikers geïnformeerd worden over het feit dat er een blokkade geplaatst of opgeheven wordt. Je kunt een voorbeeld `hooks/post-lock.tmpl` vinden in je archief.

Je kunt ook hook scripts gebruiken om het verbreken of stelen van blokkades te voorkomen, of dat alleen een beheerder dit mag doen. Je kunt ook een email laten sturen naar de eigenaar van een blokkade als zijn blokkade verbroken of gestolen wordt.

Lees [Paragraaf 3.3, “Server gesitueerde hook scripts”](#) voor meer informatie hierover.

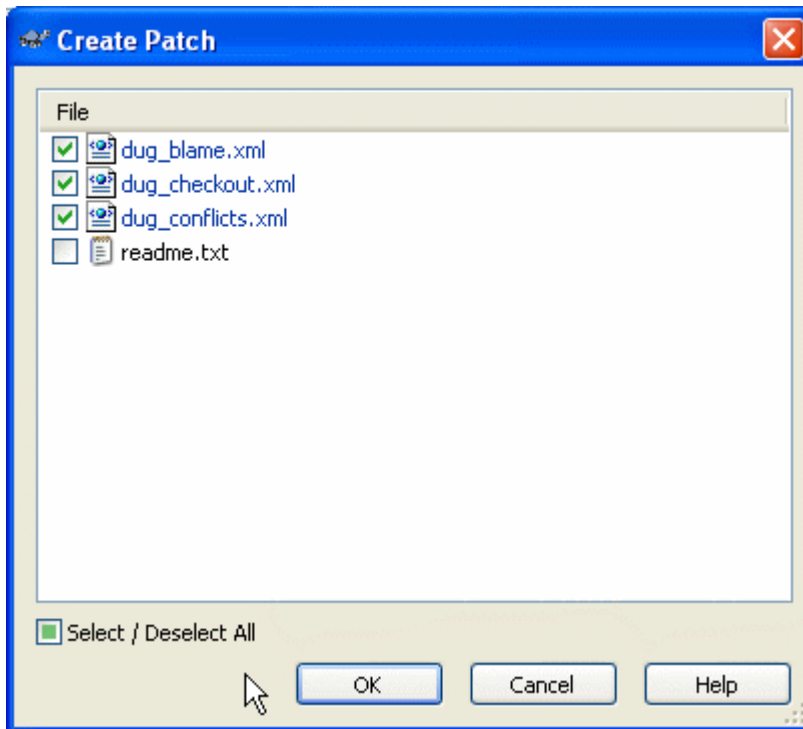
4.22. Een Patch Maken en Toepassen

Bij Open bron projecten (zoals deze) heeft iedereen leesrechten voor het archief en kan iedereen een bijdrage leveren aan het project. Hoe worden die bijdrages behandeld? Als iedereen wijzigingen kan vastleggen, dan zal het project bijna altijd onstabiel zijn en zelfs in de meeste gevallen niet werken. Om dit te voorkomen wordt er gebruik gemaakt van *patch* bestanden die naar het ontwikkelteam gestuurd kunnen worden, die natuurlijk wel schrijfrechten heeft. Zij kunnen de patch dan controleren en in het archief inbrengen of de auteur laten weten dat deze afgewezen is.

Patch bestanden zijn Unified-Diff bestanden waarin de verschillen tussen jouw werkkopie en de basis revisie zijn opgenomen.

4.22.1. Maken van een Patchbestand

Eerst moet je natuurlijk je wijzigingen maken *en testen*. Vervolgens moet je op de bovenliggende map TortoiseSVN → Maak Patch... selecteren in plaats van TortoiseSVN → Vastleggen...



Figuur 4.42. Het Maak Patch Venster

Nu kun je de bestanden selecteren die je in je patch wilt opnemen, net zoals je zou doen bij een normale vastlegging. Hiermee wordt dan een enkel bestand gemaakt met daarin een samenvatting van alle wijzigingen die je aan de geselecteerde bestanden hebt aangebracht sinds de laatste verversing van het archief.

Je kunt de kolommen in dit dialoogscherm configureren op dezelfde manier als die van het dialoogscherm [Kijk of er updates zijn](#). Ga naar [Paragraaf 4.7.3, “Lokale en Server Status”](#) voor meer informatie.

Je kunt verschillende patches aanmaken met wijzigingen aan verschillende bestandsgroepen. Als je een patch bestand aanmaakt, dan nog enkele wijzigingen aan *dezelfde* bestanden doorvoert en vervolgens weer een patch bestand aanmaakt, dan zal in het tweede patch bestand *beide* sets met wijzigingen te zien zijn.

Sla het bestand op met een naam naar keuze. Patch bestanden kunnen elke extensie hebben die je leuk vindt, maar standaard wordt de extensie `.patch` of `.diff` gebruikt. Nu ben je klaar om je patch bestand op te sturen.

Je kunt je patch ook op het klembord opslaan in plaats van in een bestand. Dit is handig als je de patch bijvoorbeeld in een mail wilt plakken zodat iemand anders hem kan controleren. Je kunt deze methode ook gebruiken als je twee PC's gebruikt en je de wijzigingen van de ene naar de andere PC wilt verplaatsen.

4.22.2. Een Patchbestand Toepassen

Patch bestanden worden toegepast op je werkkopie. Dit moet gedaan worden vanaf hetzelfde map-niveau als waarop de patch gemaakt is. Als je niet zeker weet op welk niveau dat was, kijk dan naar de eerste regel van het patch bestand. Als bijvoorbeeld het eerste bestand waaraan gewerkt is `doc/source/english/chapter1.xml` is en de eerste regel in het patch bestand is `Index: english/chapter1.xml`, dan moet je de patch toepassen op de map `doc/source/`. Als je per ongeluk het verkeerde map-niveau kiest, dan zal TortoiseSVN je hiervan op de hoogte stellen en het juiste niveau voorstellen. Dit werkt alleen als je in de juiste werkkopie aan het werken bent.

Als je een patch bestand wilt toepassen op je werkkopie, dan heb je op zijn minst lees-rechten nodig bij het archief. De reden hiervoor is dat het samenvoegprogramma de revisie moet kunnen traceren waarop de wijzigingen door de ontwikkelaar zijn gemaakt.

Klik in het context menu van die folder op TortoiseSVN → Pas Patch toe.... Er wordt dan een nieuw scherm geopend waarin je het patch bestand, die je wilt toepassen, kunt selecteren. Standaard zullen alleen `.patch` of `.diff` bestanden weergegeven worden, maar je kunt ook “Alle bestanden” laten zien. Als je een patch op het klembord hebt staan, dan kun je de knop **Openen vanaf klembord...** klikken. Merk op dat deze optie alleen weergegeven wordt als je de patch op het klembord hebt gezet met TortoiseSVN → **Maak Patch...** De knop zal niet getoond worden als je een patch vanuit een ander programma naar het klembord hebt gekopieerd.

Als jouw patch bestand de extensie `.patch` of `.diff` heeft, kun je deze ook direct toepassen als je met rechts op het bestand klikt en TortoiseSVN → Pas Patch toe... selecteert.

Dit zijn twee verschillende manieren waarmee je hetzelfde bereikt. Met de eerste methode selecteer je de werkkopie en blader je naar het patch bestand. Met de tweede selecteer je het patch bestand en blader je naar de werkkopie.

Als je eenmaal het patch bestand en de locatie van de werkkopie hebt geselecteerd, zal TortoiseMerge de wijzigingen van het patch bestand met je werkkopie samenvoegen. In een klein scherm zullen de gewijzigde bestanden weergegeven worden. Dubbelklik op elk bestand om de wijzigingen van de samenvoeging te bekijken en deze op te slaan.

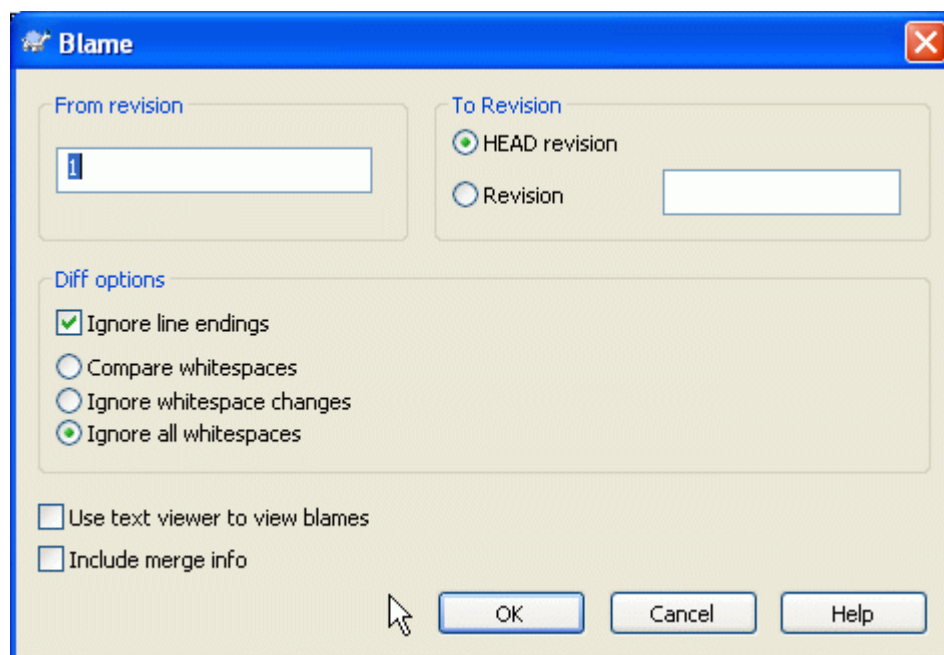
De patch van de andere ontwikkelaar is nu toegepast op jouw werkkopie. Je moet nu de wijzigingen vastleggen in het archief, zodat iedereen de wijzigingen kan ophalen.

4.23. Wie Heeft Welke Regel Gewijzigd?

Soms wil je niet alleen welke regels er gewijzigd zijn, maar ook wie er precies welke regel gewijzigd heeft. Hierbij komt het commando TortoiseSVN → Verklaren..., soms ook aangeduid als *annoteren*, goed van pas.

Dit commando laat voor elke regel in een bestand zien wie, en in welke revisie, een regel heeft gewijzigd.

4.23.1. Verklar Bestanden



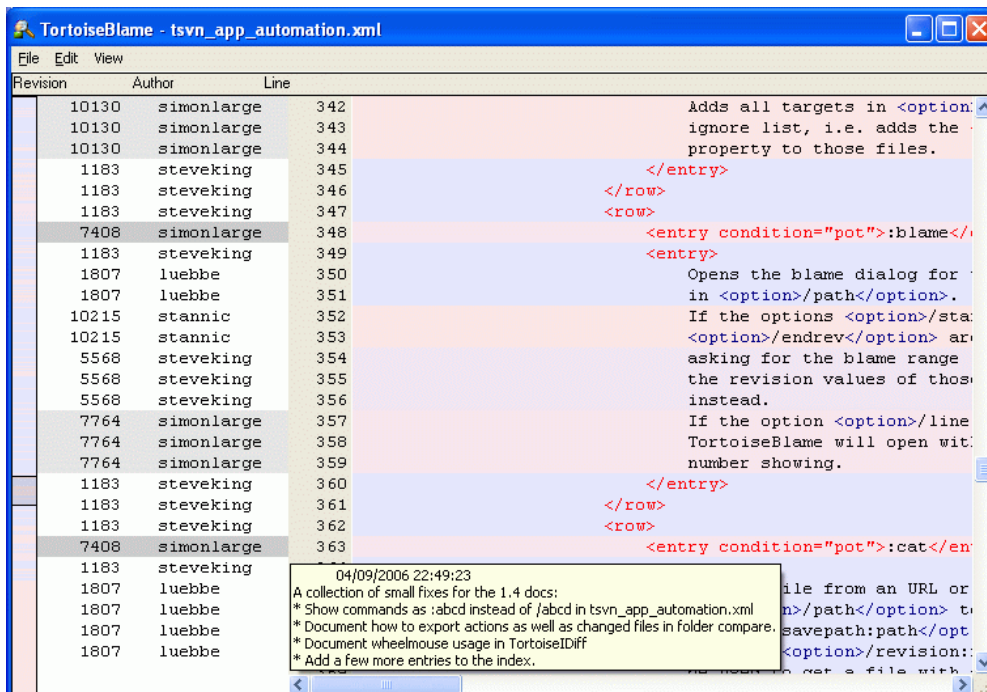
Figuur 4.43. Het Verklar/Annoteren Scherm

Als je niet geïnteresseerd bent in de wijzigingen van eerdere revisies, kun je aangegeven vanaf welke revisie het verklaren moet beginnen. Stel deze waarde in op 1 als je de wijzigingen voor *elke* revisie wilt zien.

Standaard wordt het verklaren-bestand getoond in *TortoiseBlame*, waarin de verschillende revisies gemarkeerd worden voor een beter overzicht. Als je het verklaren-bestand wilt afdrukken of bewerken, selecteer dan Gebruik tekst editor om verklaringen te bekijken.

Je kunt instellen hoe regeleinden en spaties behandeld moeten worden. Deze opties staan beschreven in [Paragraaf 4.10.2, “Opties voor regeleinden, spaties en witregels”](#). Standaard worden alle verschillen in spaties en regeleinden als echte wijzigingen behandeld. Als je een inspring wijziging wilt negeren en de originele auteur wilt vinden, dan kun je de benodigde instellingen hier kiezen.

Als je eenmaal op OK hebt geklikt, start TortoiseSVN met het ophalen van de gegevens om het verklaren-bestand aan te maken. Merk op: Het kan enkele minuten duren voordat dit proces klaar is. Dit hangt natuurlijk af van het aantal wijzigingen in een bestand en de netwerkverbinding naar je archief. Als het proces klaar is, worden de resultaten naar een tijdelijk bestand geschreven, waarna je de resultaten kunt bekijken.



Figuur 4.44. TortoiseBlame

TortoiseBlame, die met TortoiseSVN wordt meegeleverd, maakt het lezen van verklaren-bestanden makkelijker. Als je je muis boven een regel in de verklaren kolom houdt, dan worden alle regels van dezelfde revisie weergegeven met een donkere achtergrond. Regels van andere revisies die door dezelfde auteur gewijzigd zijn, worden weergegeven met een lichte achtergrond. De kleuren worden niet optimaal weergegeven als je je scherm hebt ingesteld op 256 kleuren mode.

Als je met links op een regel klikt, worden alle regels van dezelfde revisie gemarkeerd en regels van andere revisies van dezelfde auteur worden met een lichtere kleur gemarkeerd. De markeringen blijven actief, waardoor je je muis kunt bewegen zonder dat de markeringen verloren gaan. Als je weer op de revisie klikt, worden de markeringen weer uitgezet.

De revisie opmerkingen (logboek boodschappen) worden in een klein hint scherm weergegeven als je je muis boven de verklaring informatie kolom houdt. Als je de logboek boodschap van die revisie wilt kopiëren, gebruik je het context menu door met rechts te klikken op de verklaring informatie kolom.

Je kunt in het verklaren rapport zoeken met **Wijzigen** → **Zoeken...** Hiermee kun je zoeken op revisienummers, auteurs en de inhoud van het bestand zelf. Logboek boodschappen worden niet meegenomen in de zoekopdracht - hiervoor moet je het Logboek venster gebruiken.

Je kunt ook naar een specifiek regelnummer springen met **Wijzigen** → **Ga naar regel...**

In de verklaring informatie kolom is een context menu beschikbaar, waarmee je makkelijk revisies kunt vergelijken en de geschiedenis kunt onderzoeken met het revisienummer van de regel waar de muis boven gehouden wordt als referentie. **Context menu** → **Verklaar vorige revisie** genereert een verklaar rapport voor hetzelfde bestand, maar dan met de vorige revisie als bovenste limiet. Hiermee krijg je een verklaar rapport voor de staat van het bestand vlak voordat de regel die je bekijkt gewijzigd werd. **Context menu** → **Toon wijzigingen** start jou vergelijk programma, waarmee dan de wijzigingen van de gerefereerde revisie worden weergegeven. **Context menu** → **Toon logboek** toont het revisie log venster vanaf de gerefereerde revisie.

Als je een betere visuele indicatie wilt hebben van waar de oudste en nieuwe wijzigingen staan, selecteer dan **Bekijk** → **Kleur regels naar leeftijd**. Hiermee wordt een kleur verloop gebruikt om nieuwe regels met een rode kleur weer te geven en oudere met een blauwe kleur. De standaard kleurinstelling is vrij licht, maar dat kun je wijzigen bij de TortoiseBlame instellingen.

Als je **Samenvoeg** historie gebruikt, zal TortoiseBlame, als regels gewijzigd zijn als gevolg van het samenvoegen vanaf een ander pad, de revisie en auteur van de laatste wijziging weergeven in het originele bestand in plaats van de revisie waarin de samenvoeging plaatsvond. Bij deze regels worden het revisienummer en de auteur schuin gedrukt weergegeven. Als je niet wilt dat samengevoegde regels zo worden weergegeven, haal dan het vinkje weg bij **Samenvoeg-informatie meenemen**.

Als je de betrokken paden van het samenvoegen wilt zien, kies dan **Bekijk** → **Voeg paden samen**.

De instellingen van TortoiseBlame kunnen bekeken worden bij TortoiseSVN → **Instellingen...** onder de TortoiseBlame tab. Lees ook [Paragraaf 4.30.9, "TortoiseBlame Instellingen"](#).

4.23.2. Verklaar Verschillen

Een van de beperkingen van het verklaren-rapport is dat het alleen het bestand weergeeft zoals het op het moment van een bepaalde revisie was en de laatste persoon die een regel gewijzigd heeft. Soms wil je weten welke wijziging is doorgevoerd en wie dat gedaan heeft. Hiervoor heb je een combinatie van verschil- en verklaar-rapporten nodig.

Het revisie log venster heeft enkele opties waarmee je dit kunt doen.

Verklaar Revisies

Selecteer in het bovenste deelvenster 2 revisies en kies dan **Context menu** → **Verklaar revisies**. Hiermee worden de verklaar-gegevens voor die 2 revisies opgehaald. Gebruik daarna dan de diff viewer om de twee verklaarde revisies te vergelijken.

Verklaar Wijzigingen

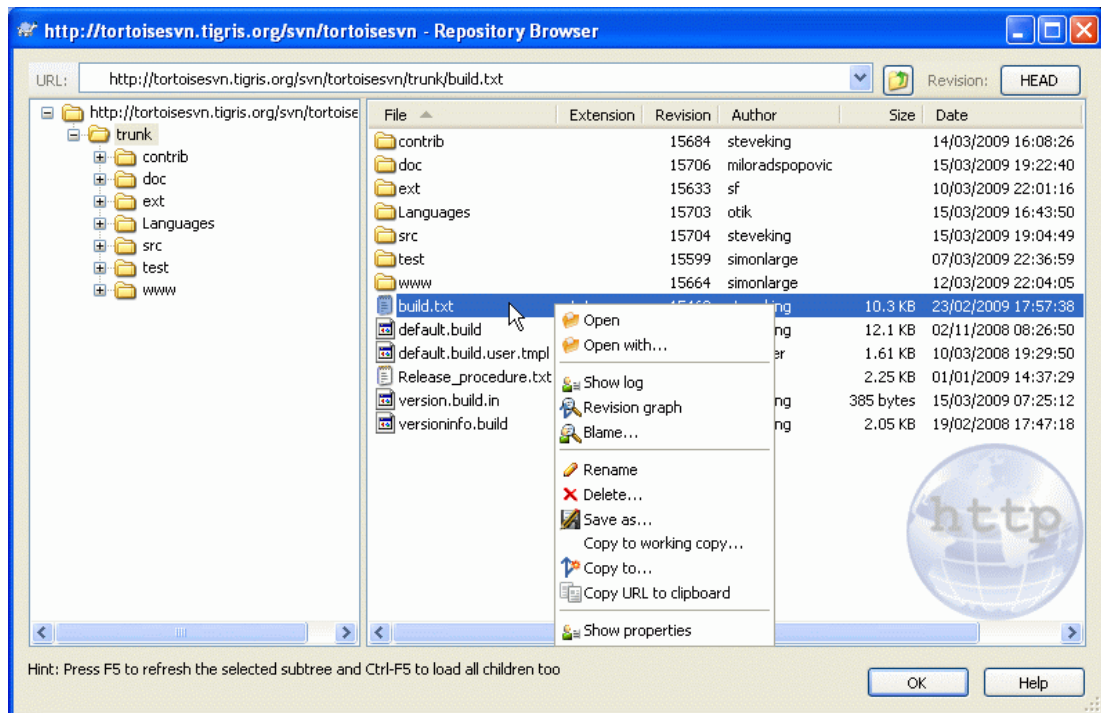
Selecteer één revisie in het bovenste deelvenster, kies vervolgens één bestand uit het onderste deelvenster en selecteer **Context menu** → **Verklaar wijzigingen**. Hiermee worden de verklaar-gegevens voor de geselecteerde revisie en de revisie daarvoor opgehaald. Gebruik daarna je diff programma om de twee verklaar-bestanden te vergelijken.

Vergelijk en Verklaar met Werkbasis

Toon het logboek voor een enkel bestand en selecteer in het bovenste deelvenster een enkele revisie. Kies vervolgens **Context menu** → **Vergelijk en verklaar met werkkopie (BASE)**. De verklaar-gegevens worden dan voor de geselecteerde revisie en het bestand in de werkbasis opgehaald. Gebruik dan het diff programma om de twee verklaar-bestanden te vergelijken.

4.24. De Archief-verkenner

Soms wil je direct dingen in het archief wijzigen, zonder dat je een werkkopie hebt. Daarvoor kun je de *Archief-verkenner* gebruiken. Met de archief-verkenner kun je de structuur en status van het archief bekijken, net zoals je met je verkenner en overlappende picotogrammen de status van je werkkopie kunt bekijken.



Figuur 4.45. De Archief-verkenner

Met de archiefverkenner kun je acties zoals kopiëren, verplaatsen, hernoemen, ... direct op het archief uitvoeren.

De archief-verkenner ziet er ongeveer hetzelfde uit als de windows verkenner, behalve dan dat het de inhoud van het archief van een bepaalde revisie laat zien, in plaats van bestanden op je computer. In het linker deelvenster wordt de mapstructuur weergegeven en in het rechter de inhoud van de geselecteerde map. Aan de bovenzijde van de archief-verkenner kun je de URL en de revisie van een archief invoeren, welke je wilt bekijken.

Net zoals bij de windows verkenner, kun je op de koptekst van een kolom klikken om de sorteervolgorde te wijzigen. Net zoals in de verkenner zijn er context menu's beschikbaar in beide deelvensters.

Vanuit het context menu kun je verschillende dingen doen met bestanden:

- Open het geselecteerde bestand met het standaard programma voor dit bestandstype of met een zelf gekozen programma.
- Een kopie zonder versiebeheer van een bestand opslaan op je harde schijf.
- Het revisie logboek laten tonen van een bestand of een figuur laten tonen met alle revisies, zodat je kunt zien waar het bestand vandaan is gekomen.
- Een bestand laten verklaren om te zien wie welke regel wanneer in een bestand gewijzigd heeft.
- Verwijderen of hernoemen van het bestand.

- Een kopie laten maken van een bestand naar een ander deel van het archief of naar een werkkopie gesitueerd in hetzelfde archief.
- Bekijken/Wijzigen van de kenmerken van het bestand.

Met het context menu van een map kun je dingen doen zoals:

- Het revisie logboek laten tonen van een map of een figuur laten tonen met alle revisies, zodat je kunt zien waar de map vandaan is gekomen.
- De map laten exporteren naar een locatie op je harde schijf zonder versiebeheer.
- Een map laten ophalen, zodat er een lokale werkkopie wordt gemaakt op je harde schijf.
- Een nieuwe map aanmaken in het archief.
- Bestanden of mappen direct in het archief zetten.
- Een map verwijderen of hernoemen.
- Een kopie laten maken van een map naar een ander deel van het archief of naar een werkkopie gesitueerd in hetzelfde archief. Hiermee kan ook een vertakking/label gemaakt worden, zonder dat daarvoor een werkkopie nodig is.
- Eigenschappen van een map bekijken of wijzigen.
- Een map markeren voor vergelijken. Een gemarkeerd map wordt vetgedrukt weergegeven.
- De map vergelijken met een eerder gemarkeerde map, als unified diff of als een lijst met gewijzigde bestanden, die dan met het standaard vergelijkingsprogramma vergeleken kan worden. Dit kan in het bijzonder handig zijn als je twee labels wilt vergelijken, of een basislijn (trunk) met een vertakking.

Als je twee mappen in het rechter deelvenster selecteert, dan kun je de verschillen als unified-diff bekijken of als lijst van bestanden, welke dan weer met een vergelijkingsprogramma bekeken kan worden.

Als je meerdere mappen in het rechter deelvenster selecteert, dan kun je deze allemaal tegelijk ophalen naar een zelfde hoofdmap.

Als je 2 labels selecteert die van dezelfde hoofdmap gekopieerd zijn (normaal gesproken de basislijn /trunk/), dan kun je met **Context Menu** → **Toon Logboek...** de lijst met revisies zien tussen die twee labels.

Je kunt zoals gebruikelijk met **F5** het beeld verversen. Hiermee wordt alles verversen wat weergegeven worden. Als je de informatie van andere delen, die je nog niet geopend hebt, wilt verversen of alvast wilt ophalen, gebruik dan **Ctrl-F5**. Daarna kun je elk ander deel direct openen, zonder dat daarvoor vertraging ontstaat voor het ophalen van informatie via het netwerk.

Je kunt de archief-verkenner ook gebruiken voor drag-and-drop acties. Als je een map vanuit de verkenner naar de archief-verkenner sleept, dan zal deze map in het archief geïmporteerd worden. Merk op dat als je meerdere objecten sleept, deze mappen als aparte vastleggingen worden behandeld.

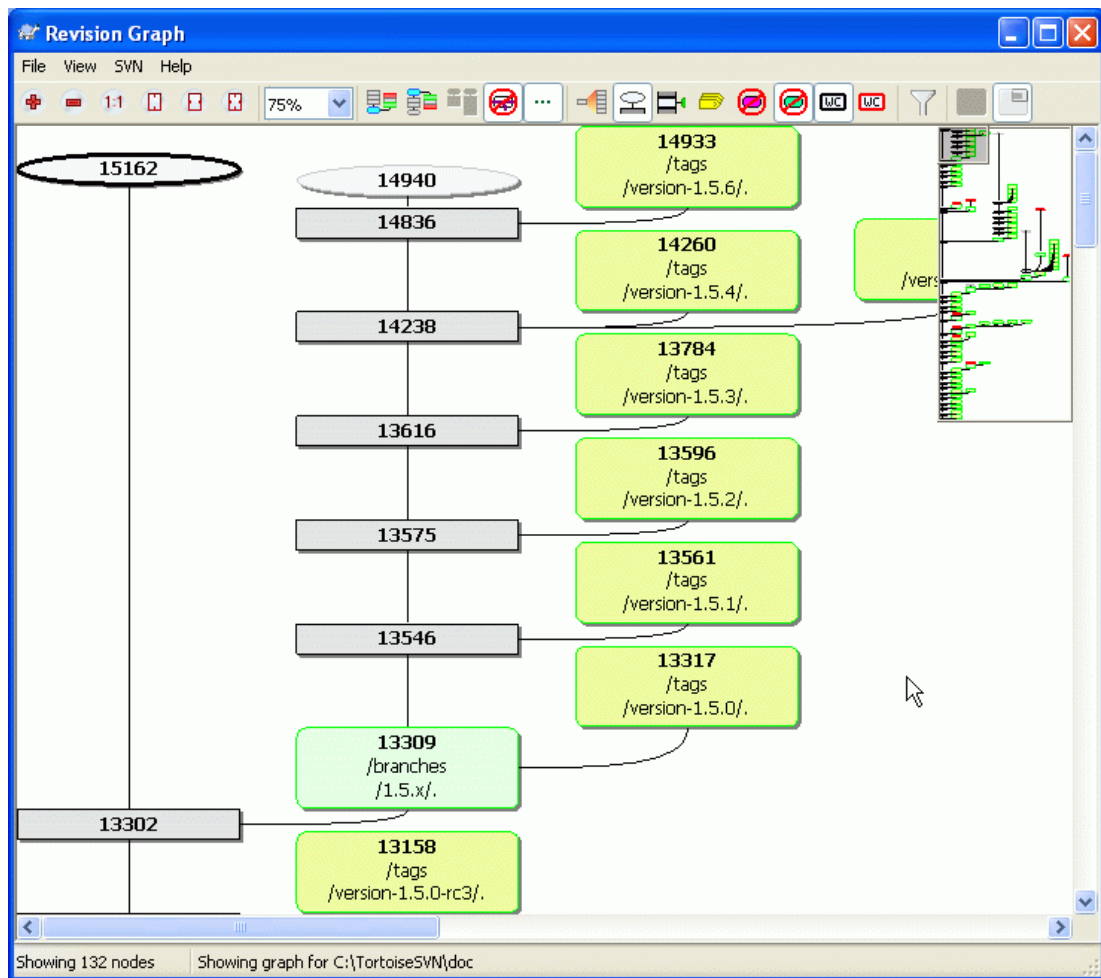
Als je een object binnen het archief wilt verplaatsen, dan kun je deze gewoon met links verslepen naar de nieuwe locatie. Als je een kopie van het object wilt hebben in plaats van het alleen te verplaatsen, houdt dan ook **Ctrl** ingedrukt tijdens het verslepen. Als je een object gaat kopiëren, dan heeft de cursor een “plus” teken net zoals in de windows verkenner.

Als je een bestand of map wilt verplaatsen/kopieren naar een andere locatie en ook tegelijkertijd de naam wilt wijzigen, dan moet je het object met rechts slepen of **Ctrl**-rechts slepen in plaats van links slepen. Er wordt dan een scherm getoond waarin je de nieuwe naam van de map of het bestand moet invoeren.

Telkens als je op deze manier wijzigingen in het archief doorvoert, zal er een logboek boodschappen venster getoond worden. Als je per ongeluk objecten hebt versleept, dan is dit je kans om die actie te stoppen.

Soms als je een pad wilt openen, krijg je een foutmelding in plaats van de objectdetails. Dit kan komen doordat je een verkeerde URL hebt ingevoerd of doordat je geen toegangsrechten hebt of omdat er een ander probleem met de server is. Als je de melding moet kopiëren naar bijvoorbeeld een email, klik dan met rechts en selecteer Context Menu → Kopieer foutmelding naar klembord of gebruikt **Ctrl+C**.

4.25. Revisie schema's



Figuur 4.46. Een Revisie Schema

Soms moet je weten op welk punt vertakkingen en labels gemaakt zijn van de basislijn (trunk). De ideale manier om dit te bekijken is door de informatie te sorteren in een grafiek of boomstructuur. Hiervoor kun je gebruik maken van TortoiseSVN → Revisie schema....

Dit commando analyseert de revisiegeschiedenis en probeert daarvan een boomstructuur te maken, waarin de punten aangegeven worden waarop kopieën gemaakt en vertakkingen/labels verwijderd zijn.



Belangrijk

Om de figuur te maken, moet TortoiseSVN alle logboek boodschappen ophalen vanuit de hoofdmap van het archief. Overbodig is het om te zeggen dat dit enkele minuten kan duren, zelfs bij een archief met maar enkele duizenden revisie. Dit hangt natuurlijk sterk af van de snelheid van de server en de bandbreedte van het netwerk. Als je dit probeert te doen op bijvoorbeeld het *Apache* project met op dit moment meer dan 500.000 revisies, dan moet je wel een tijd wachten.

Het goede nieuws is dat als je log caching gebruikt, je maar een keer op deze vertraging hoeft te wachten. Daarna worden de logboek boodschappen lokaal opgeslagen. Log caching kun je inschakelen bij de instellingen van TortoiseSVN.

4.25.1. Revisie Schema Knooppunten

Elke knoop in de revisie grafiek geeft een revisie in het archief weer, waarbij er iets gewijzigd is in de mapstructuur waar je naar kijkt. De verschillende soorten knopen zijn te onderscheiden door hun vorm en kleur. De vormen staan vast, maar de kleuren kunnen aangepast worden met TortoiseSVN → Instellingen

Toegevoegde of gekopieerde objecten

Objecten die toegevoegd of door het kopiëren van een ander bestand/map aangemaakt zijn, worden weergegeven met een rechthoek met ronde hoeken. De standaard kleur is groen. Labels en basislijnen worden behandeld als speciale gevallen en hebben een andere schaduw, afhankelijk van de TortoiseSVN → Instellingen.

Verwijderde objecten

Verwijderde objecten, bijvoorbeeld een tak die niet langer nodig is, worden weergegeven als een achthoek. De standaard kleur is rood.

Hernoemde objecten

Hernoemde objecten worden ook als een achthoek weergegeven, maar de standaard kleur daarvoor is blauw.

Revisie van het uiteinde van een tak

De grafiek laat normaal gesproken alleen de vertak-punten zien, maar soms is het ook handig om de HEAD revisie van een tak ook te zien. Als je optie **Toon HEAD revisies** selecteert, dan wordt elke HEAD revisie knoop getoond als ellips. Merk op dat hier met de HEAD revisie de laatste vastgelegde revisie van dat pad bedoeld wordt, niet de actuele HEAD revisie van het hele archief.

Werkkopie revisie

Als je de revisie grafiek van een werkkopie hebt opgevraagd, dan kun je de BASE (basis) revisie met **Toon revisie van werkkopie** laten weergeven. Hierdoor wordt de BASE knoop dikkere omtrek getoond.

Gewijzigde werkkopie

Als je de revisie grafiek van een werkkopie hebt opgevraagd, dan kun je additionele knopen, die jouw gewijzigde werkkopie representeren, laten zien met **Toon wijzigingen in werkkopie**. Dit wordt weergegeven als een ellipsvormige knoop met een dikkere omtrek en standaard een rode kleur.

Normaal object

Alle andere objecten worden als een normale rechthoek weergegeven.

Merk op dat standaard de grafiek alleen de punten laat zien waarop er objecten zijn toegevoegd, gekopieerd of verwijderd. Een grafiek laten maken van elke revisie van het project, levert een enorm grote grafiek op met niet triviale gegevens. Als je echt *alle* wilt zien waarin wijzigingen zijn doorgevoerd, dan kun je dat doen instellen in het **Beeld** menu of via de knoppenbalk.

De standaard weergave (groeperen uitgeschakeld) positioneert de knopen zo dat deze verticaal op volgorde gerangschikt zijn, waardoor je direct een overzicht hebt van de volgorde waarop de dingen gedaan zijn. Als er twee knopen in dezelfde kolom staan, dan spreekt de volgorde voor zich. Als er twee knopen zijn in aangrenzende kolommen, dan is de afstand veel kleiner. Er is dan geen noodzaak om te voorkomen dat de knopen elkaar overlappen. Het gevolg hiervan is dat de volgorde iets minder overzichtelijk kan zijn. Deze optimalisaties zijn helaas nodig om complexe grafieken een acceptabel formaat te laten houden. Merk op dat bij het rangschikken de *rand* van de knoop aan de *oude* kant van de knoop gebruikt wordt als referentiepunt. Dat is de onderkant van de knoop als de oudste knoop

onderaan de grafiek wordt weergegeven. Dit referentiepunt is belangrijk, omdat niet alle figuren van knopen dezelfde hoogte hebben.

4.25.2. Wijzigen van de Weergave

Een revisie grafiek is vaak erg complex. Er zijn daarom verschillende functies beschikbaar, waarmee je de weergave kunt aanpassen naar jouw wensen. Deze zijn beschikbaar in het **Beeld** menu en via de werkbalk.

4.25.3. Gebruik van de grafiek

Om het navigeren door een grote grafiek makkelijker te maken, kun je het overzicht scherm gebruiken. Hiermee wordt een klein scherm getoond, waarin de huidige weergave gemarkeerd is. Je kunt het gemarkeerde gebied verslepen om de actuele weergave te veranderen.

De revisie datum, auteur en het commentaar worden weergegeven in een hint box als je de muis boven een revisie houdt.

Als je twee revisies selecteert (gebruikt **Ctrl**-klik met links), dan kun je via het context menu de verschillen tussen deze revisies opvragen. Je kunt verschillen laten zien van het moment dat er vertakkingen zijn aangemaakt, maar meestal zul je de verschillen willen zien bij het eindpunt van een vertakking, bij de HEAD revisie.

Je kunt de verschillen in een Unified-Diff bestand bekijken, welke de exacte verschillen laat zien met minimale context. Als je voor **Context Menu** → **Vergelijk Revisies** kiest, zul je een lijst met wijzigingen krijgen. Dubbelklik op een bestandsnaam om beide revisies van dat bestand op te halen en te vergelijken met een visueel vergelijkingsprogramma.

If you right click on a revision you can use **Context Menu** → **Show Log** to view the history.

Je kunt ook de wijzigingen van de geselecteerde revisie(s) samenvoegen in een aparte werkkopie. Een scherm waarin je een map kunt selecteren, geeft de mogelijkheid om de werkkopie te selecteren waarin de selectie samengevoegd moet worden. Daarna is er geen bevestigingsscherm meer en geen kans meer om de samenvoeging te testen. Het is verstandig om de wijzigingen naar een niet gemodificeerde werkkopie samen te laten voegen, zodat je de wijzigingen ongedaan kunt maken als de samenvoeging niet oplevert wat je ervan had verwacht. Dit is handig als je functies van de geselecteerde revisie van de ene naar de andere tak wilt samenvoegen.



Hoe moet je de Revisie Grafiek lezen

Mensen die de grafiek voor het eerst gebruiken, kunnen verrast worden doordat grafisch weergave heel anders kan zijn dan het model wat de mensen zelf voor ogen hebben. Als een revisie bijvoorbeeld meerdere kopieën, takken, een bestand of map veranderd, dan zullen er verschillen knopen zijn voor die revisie. Het is verstandig om dan te beginnen met de meest linkse opties in de werkbalk en de grafiek zo stap voor stap aan te passen, totdat het het dichtst bij het model komt dat je voor ogen had.

Alle filteropties proberen zo min mogelijk informatie verloren te laten gaan. Hierdoor kunnen de kleuren van sommige knopen bijvoorbeeld veranderen. Als een resultaat onverwacht is, maak dan de laatste filteractie ongedaan en probeer te achterhalen wat er speciaal is aan die revisie of tak. In de meeste gevallen is het initiële resultaat van de filteractie onnauwkeurig of misleidend.

4.25.4. De weergave verversen

Als je wilt nagaan of er nieuwe informatie beschikbaar is op de server, dan kun je het beeld eenvoudig verversen met **F5**. Als je de log cache gebruikt (standaard ingeschakeld), dan zal het archief op nieuwe

vastleggingen gecontroleerd worden en worden ook alleen de nieuwe vastleggingen opgehaald. Als de log cache in offline mode werkte, dan zal deze ook proberen weer naar online mode om te schakelen.

Als je de log cache gebruikt en je vermoed dat de inhoud van een bericht of auteur gewijzigd is, gebruik dan het logboek venster om de berichten die je nodig hebt te verversen. Omdat de revisie grafiek werkt vanaf de basis van het archief, moeten we de hele log cache ongeldig verklaren. Het opnieuw vullen hiervan kan *erg* lang duren.

4.25.5. Boomstructuren inkorten

Het navigeren door een grote boomstructuur kan lastig zijn. Soms wil je onderdelen verbergen of de structuur opdelen in een bos met kleinere bomen. Als je je muis boven het punt houdt waar een verbinding een knoop binnenkomt of verlaat, zul je een popup te zien krijgen met knoppen waarmee je dit kunt doen.



Klik op de min-knop om de bijbehorende onderverdeling in te klappen.



Klik op de plus-knop om de ingeklapte boom uit te klappen. Als een boom is ingeklapt blijft deze knop zichtbaar om de verborgen onderverdeling aan te geven.



Klik op de kruis-knop om de gekoppelde onderverdeling te splitsen en deze als een afzonderlijke boom op de grafiek te tonen.

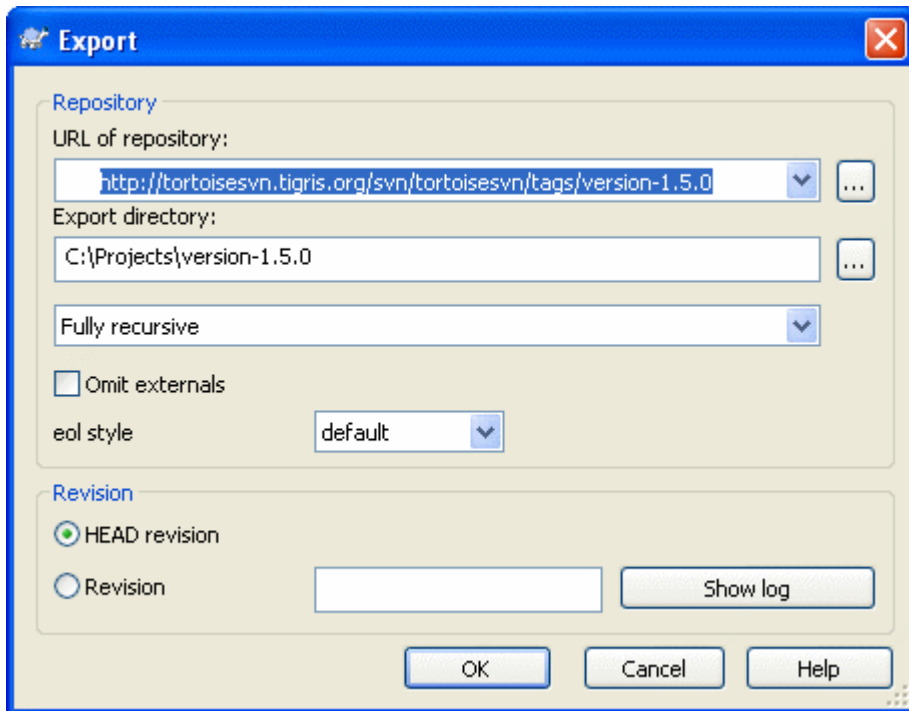


Klik op de cirkel-knop om een afgesplitste boom weer aan te koppelen. Wanneer een boom is afgesplitst blijft deze knop zichtbaar om aan te geven dat er een aparte onderverdeling aanwezig is.

Klik op de achtergrond van de grafiek voor het hoofd contextmenu, wat opties biedt naar **Alles uitklappen** en **Alle samenvoegen**. Als er geen tak is ingeklapt of afgesplitst zal het contextmenu niet worden getoond.

4.26. Exporteren van een Subversion Werkkopie

Soms wil je een kopie van je werk omgeving zonder al die `.svn` mappen, bijvoorbeeld om een gezippt tarball van je bron te maken of om deze te exporteren naar een web server. In plaats van het maken van een kopie en alle `.svn` mappen handmatig te verwijderen, heeft TortoiseSVN het commando **TortoiseSVN** → **Exporteren....** Het exporteren vanaf een URL en vanaf een werkkopie wordt iets anders behandeld.



Figuur 4.47. Het Exporteren-vanaf-een-URL scherm

Als je dit commando op een map zonder versiebeheer toepast, dan zal TortoiseSVN deze map als de doelmap zien en een scherm openen waarin je de URL en revisie in kunt voeren van waar er geëxporteerd moet worden. Er zijn opties beschikbaar waarmee kunt selecteren dat je alleen de hoofdmap wilt hebben, externen wilt uitsluiten en de stijl voor geregeleinden voor bestanden, die de eigenschap `svn:eol-style` hebben, kunt negeren.

Natuurlijk kun je ook direct vanuit het archief exporteren. Gebruik de archief-verkenner om naar de relevante submappen te gaan in je archief en gebruik dan Context Menu → Exporteren. Je krijgt dan het Exporteren van URL venster zoals eerder beschreven.

Als je dit commando uit laat voeren op je werkkopie, dan zal je gevraagd worden waar de *schone* werkkopie zonder `.svn.` mappen opgeslagen moet worden. Standaard worden alleen de bestanden met versiebeheer geëxporteerd, maar je kunt met het Exporteer ook bestanden zonder versiebeheer selectievak elk bestand zonder versiebeheer in je werkkopie (die dus niet in het archief zitten) mee laten exporteren.

Nog een andere manier om vanuit je werkkopie te exporteren is door de werkkopie met rechts te slepen naar een andere locatie en Context Menu → SVN Exporteer objecten onder versiebeheer hierheen of Context Menu → SVN Exporteer alle objecten hierheen te kiezen. Met de tweede optie worden ook bestanden zonder versiebeheer meegenomen.

Als je een werkkopie exporteert, krijg je, als de doelmap al een map heeft met dezelfde naam, de vraag of je de bestaande inhoud wilt overschrijven of dat er automatisch een nieuwe naam aangemaakt moet worden, bijv. Doe1 (1).



Exporteren van enkele bestanden

Het export scherm staat het exporteren van een enkel bestand niet toe, ondanks dat Subversion dat wel ondersteunt.

Voor het exporteren van enkele bestanden met TortoiseSVN, moet je de archief-verkenner gebruiken (Paragraaf 4.24, “De Archief-verkenner”). Sleep het bestand (of de bestanden)

die je wilt exporteren, van de archief-verkenner naar de plaats in de verkenner waar je deze wilt hebben. Je kunt ook bestanden exporteren vanuit het context menu in de archief-verkenner.



Exporteren van een gewijzigde boom

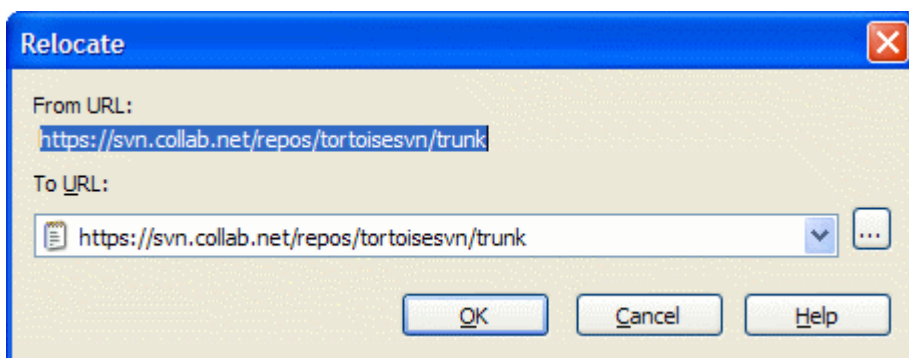
Als je een kopie van je project boomstructuur wilt exporteren met alleen de in een specifieke revisie (of tussen 2 revisies) gewijzigde bestanden, gebruik dan de revisies vergelijken functionaliteit best [Paragraaf 4.10.3, "Vergelijken van mappen"](#).

4.26.1. Versiebeheer van een werkkopie verwijderen

Soms heb je een werkkopie die je terug wilt zetten naar een normale map zonder de `.svn` mappen. Wat je eigenlijk nodig hebt is een `exporteer-op-deze-plaats` commando, waarmee gewoon de versiebeheer mappen worden verwijderd (in plaats van een nieuwe mapstructuur aan te maken).

Het antwoord is misschien onverwacht eenvoudig - exporteer de map naar zichzelf. TortoiseSVN detecteert deze speciale situatie en vraagt of je het versiebeheer van je werkkopie wilt verwijderen. Als je *Ja* antwoordt, worden de versiebeheer mappen verwijderd en heb je weer een normale mapstructuur zonder versiebeheer.

4.27. Verhuizen van een werkkopie



Figuur 4.48. De Verhuis Dialoog

Het kan zijn dat de locatie van je archief om één of andere reden veranderd is (IP/URL). Hierdoor kun je in de problemen komen, omdat je je wijzigingen niet meer vast kunt leggen. Je wilt eigenlijk ook niet een compleet nieuwe werkkopie ophalen vanaf de nieuwe locatie, om vervolgens jouw gewijzigde data naar die nieuwe werkkopie te verplaatsen. TortoiseSVN → **Verhuizen** is dan het commando die je nodig hebt. Dit commando doet eigenlijk heel weinig. Het doorzoekt alle `entries` bestanden in de `.svn` mappen en wijzigt de URL's naar de nieuwe waarde.

Misschien wordt je verrast als TortoiseSVN het archief benaderd gedurende deze actie. Het enige wat het doet, is controleren of de nieuwe URL daadwerkelijk naar het archief wijst van de bestaande werkkopie.



Waarschuwing

Dit is een zelden gebruikte methode. Het verhuizen commando wordt *alleen* gebruikt als de URL van de basis van het archief is gewijzigd. Mogelijke redenen hiervan zijn:

- Het IP-adres van de server is gewijzigd.
- Het protocol is gewijzigd (vb: `http://` naar `https://`).

- Het pad van de hoofdmap van het archief is veranderd in de server instellingen.

Anders gezegd, je moet verhuizen als je werkkopie refereert naar dezelfde locatie in hetzelfde archief, maar het archief zelf is verhuisd.

Het is niet van toepassing als:

- Je wilt verhuizen naar een ander Subversion archief. In zo'n geval moet je een verse werkkopie ophalen vanaf de nieuwe archief locatie.
- Je wilt schakelen naar een andere tak of map binnen hetzelfde archief. Om dat te doen moet je gebruik maken van TortoiseSVN → Wisselen.... Lees [Paragraaf 4.19.2, "Ophalen of Wisselen..."](#) voor meer informatie.

Als je verhuizen gebruikt bij één van de twee hierboven genoemde situaties, dan *zal je werkkopie beschadigd worden* en krijg je mogelijk onverklaarbare foutmeldingen bij het verversen, vastleggen, etc. Als dat gebeurt is, kun je dat alleen oplossen met een vers opgehaalde werkkopie.

4.28. Integratie met Bug Tracking Systemen / Issue Trackers

Het is gemeengoed in Software Ontwikkeling om wijzigingen te relateren aan een specifieke bug of issue ID. Gebruikers van bug tracking systemen (issue trackers) zullen de wijzigingen welke zij maken in Subversion willen associëren aan een specifiek ID in hun issue tracker. Het merendeel van de issue trackers levert hiervoor een pre-commit hook script welke het logbericht analyseert om het bug ID te vinden waarmee de vastlegging wordt geassocieerd. Dit is enigzins foutgevoelig omdat dit afhankelijk is van de correctheid van het door de gebruiker geschreven logbericht zodat het pre-commit hook script het correct kan analyseren.

TortoiseSVN kan de gebruiker op twee manieren helpen:

1. Wanneer de gebruiker een logbericht ingeeft kan automatisch een vooraf gedefiniëerde regel, met daarin het met de vastlegging geassocieerde issue-nummer, worden toegevoegd. Dit verkleint het risico dat de gebruiker het issue-nummer op een dusdanige wijze ingeeft dat dit niet correct door de bug tracking tools kan worden geanalyseerd.

Of TortoiseSVN kan het gedeelte van het ingegeven logbericht, wat wordt herkend door de issue tracker, accentueren. Op die manier is het voor de gebruiker duidelijk of het logbericht correct kan worden geanalyseerd.

2. Wanneer de gebruiker door de logberichten bladert, maakt TortoiseSVN van elke bug ID in het logbericht een verwijzing waarmee de genoemde issue in een browser kan worden geopend.

4.28.1. Issue-nummers aan Logberichten toevoegen

U kunt een bug tracking tool naar keuze in TortoiseSVN integreren. Hiervoor dient een aantal eigenschappen te worden gedefiniëerd, welke starten met `bugtraq:`. Ze dienen te worden ingesteld op Mappen: ([Paragraaf 4.17, "Project Instellingen"](#))

Er zijn twee manieren om TortoiseSVN te integreren met issue trackers. De een is gebaseerd op eenvoudige tekenreeksen, de ander is gebaseerd op *reguliere expressies*. De eigenschappen welke door beide benaderingen worden gebruikt zijn:

`bugtraq:url`

Stel deze eigenschap in op de URL naar uw bug tracking tool. Deze dient correct te zijn gecodeerd als URI en moet `%BUGID%`. `%BUGID%` wordt vervangen door het door u ingegeven issue-nummer. Dit biedt TortoiseSVN de mogelijkheid om een verwijzing in het log venster te tonen, waarmee direct naar uw bug tracking tool kan worden gesprongen als u de revisielog bekijkt. U hoeft deze eigenschap

niet aan te leveren, maar dan toont TortoiseSVN alleen het issue-nummer en niet de verwijzing er naar toe. Bijv. het TortoiseSVN project gebruikt `http://issues.tortoisesvn.net/?do=details&id=%BUGID%`

U kunt ook gebruik maken van relatieve URL's in plaats van absolute. Dit kan handig zijn in het geval dat uw issue tracker zich op hetzelfde domein/server bevindt als uw bron archief. In het geval dat de domeinnaam ooit wijzigt hoeft u de `bugtraq:url` niet aan te passen. Er zijn twee manieren om een relatieve URL aan te geven:

Wanneer deze begint met de tekenreeks `^/` wordt aangenomen dat deze relatief is tot de archief root. `^/../?do=details&id=%BUGID%` zal bijvoorbeeld resulteren in `http://tortoisesvn.net/?do=details&id=%BUGID%` indien uw archief zich op `http://tortoisesvn.net/svn/trunk/` bevindt.

Van een URL welke met de tekenreeks `/` begint wordt aangenomen dat deze relatief is tot de hostnaam van de server. `/?do=details&id=%BUGID%` zal bijvoorbeeld resulteren in `http://tortoisesvn.net/?do=details&id=%BUGID%` indien uw archief zich ergens op `http://tortoisesvn.net` bevindt.

`bugtraq:warnifnoissue`

Stel dit in op `true` indien u wilt dat TortoiseSVN u waarschuwt in het geval van een leeg issue-nummer tekst veld. Geldige waarden zijn `true/false`. *Indien niet gedefiniëerd, wordt aangenomen dat de waarde `false` is.*

4.28.1.1. Issue-nummer in Tekst Veld

In de eenvoudige benadering toont TortoiseSVN een apart invoerveld aan de gebruiker, waar een bug ID kan worden ingegeven. Daarna wordt een aparte regel toegevoegd/vooringevoegd aan het door de gebruiker ingegeven logbericht.

`bugtraq:message`

Deze eigenschap activeert het bug tracking systeem in *Invoerveld* modus. Als deze eigenschap is ingesteld zal TortoiseSVN u vragen om een issue-nummer in te geven wanneer u uw wijzigingen vastlegt. Dit wordt gebruikt om een regel aan het einde van het logbericht toe te voegen. Het moet `%BUGID%` bevatten, wat wordt vervangen door het issue-nummer bij vastleggen. Dit biedt de zekerheid dat uw vastleggingslog een referentie naar het issue-nummer bevat, welke altijd in een consistent formaat is en kan worden geanalyseerd door uw bug tracking tool, om het issue-nummer met een bepaalde vastlegging te associëren. Als voorbeeld zou u `Issue : %BUGID%` kunnen gebruiken, maar dat hangt van uw tool af.

`bugtraq:append`

Deze eigenschap definieert of het bug-ID wordt toegevoegd (`true`) aan het einde van het logbericht of ingevoegd (`false`) aan het begin van het logbericht. Geldige waarden zijn `true/false`. *Indien niet gedefiniëerd wordt aangenomen dat de waarde `true` is, zodat bestaande projecten niet mislukken.*

`bugtraq:label`

Deze tekst wordt door TortoiseSVN getoond op het vastleggings venster om het invoerveld aan te geven waar u het issue-nummer ingeeft. Wanneer het niet is ingesteld zal `Bug-ID / Issue-Nr :` worden getoond. Houdt in gedachten dat het venster niet in grootte wordt aangepast om het label te laten passen, dus houdt de grootte van het label onder de 20-25 tekens.

`bugtraq:number`

Indien ingesteld op `true` zijn alleen nummers in het issue-nummer tekstveld toegestaan. Een uitzondering is de komma, zodat meerdere nummers door een komma kunnen worden gescheiden. Geldige waarden zijn `true/false`. *Indien niet gedefiniëerd wordt aangenomen dat de waarde `true` is.*

4.28.1.2. Issue-nummers bij gebruik van Reguliere Expressies

In de benadering met *reguliere expressies* toont TortoiseSVN geen apart invoerveld, maar wordt het gedeelte van de door de gebruiker ingegeven logbericht aangemerkt wat door de issue tracker wordt

herkend. Dit wordt gedaan terwijl de gebruiker het logbericht ingeeft. Dit betekent ook dat het bug ID zich overal in het logbericht mag bevinden! Deze methode is veel flexibeler en is de methode welke door het TortoiseSVN project zelf wordt gebruikt.

bugtraq:logregex

Deze eigenschap activeert het bug tracking systeem in *Regex* modus. Het bevat of een enkele reguliere expressie, of twee door een nieuwe regel gescheiden reguliere expressies.

Als er twee expressies zijn ingesteld, dan wordt de eerste gebruikt als voor-filter om expressies te vinden welke bug ID's bevatten. De tweede expressie haalt de kale bug ID's uit het resultaat van de eerste reguliere expressie. Dit biedt de mogelijkheid om gebruik te maken van een lijst van bug ID's of desgewenst natuurlijke taal expressies. bijv. U zou enkele bugs kunnen oplossen en een tekenreeks kunnen toevoegen zoals: "Deze wijziging is de oplossing voor issues #23, #24 en #25"

Als u bug ID's wilt afvangen, zoals gebruikt in bovenstaande expressie binnen een logbericht, kunt u gebruik maken van de volgende regex tekenreeksen, welke door het TortoiseSVN project worden gebruikt: `[Ii]ssues?:?(\s*(, |and)?\s*#\d+)+ en (\d+)`

De eerste expressie haalt "issues #23, #24 and #25" uit het omringende logbericht. De tweede regex extraheert zuivere decimale nummers uit de uitvoer van de eerste regex, zodat het "23", "24" en "25" teruggeeft om te gebruiken als bug ID's.

Wanneer we de eerste regex in stukken opbreken, dan moet deze beginnen met het woord "issue", mogelijk in hoofdletters. Dit wordt eventueel gevolgd door een "s" (meerdere issues) en eventueel een dubbele punt. Dit wordt gevolgd door een of meerdere groepen welke elk worden voorafgegaan door 0 of meer witruimte tekens, een komma of "en" en meer optionele spaties. Als laatste is er een verplichte "#" en een verplicht decimaal nummer.

Indien slechts één expressie is ingesteld, moeten de kale bug ID's worden vergeleken met de groepen van de regex tekenreeks. Voorbeeld: `[Ii]ssue(?:s)? #?(\d+)` Deze methode is vereist door een aantal issue trackers, bijv. trac, maar het is moeilijker om de regex samen te stellen. We adviseren deze methode alleen te gebruiken wanneer de documentatie van de issue tracker dit voorschrijft.

Als u niet bekend bent met reguliere expressies kunt u de introductie bekijken op http://nl.wikipedia.org/wiki/Reguliere_expressie, en de online documentatie en handleiding op <http://www.regular-expressions.info/>.

Als beide eigenschappen `bugtraq:message` en `bugtraq:logregex` zijn ingesteld, krijgt `logregex` voorrang.



Tip

Zelfs als u geen issue tracker heeft welke uw logberichten analyseert met een pre-commit hook, kunt u dit gebruiken om de in uw logbericht genoemde issues om te zetten naar verwijzingen!

En zelfs als u de verwijzingen niet nodig heeft, worden de issue-nummers getoond als een aparte kolom in het log venster wat het gemakkelijker maakt om de wijzigingen te vinden welke aan een bepaalde issue zijn gerelateerd.

Sommige `tsvn`: eigenschappen vereisen een `true/false` waarde. TortoiseSVN begrijpt ook `yes` als een synoniem voor `true` en `no` als een synoniem voor `false`.



Stel de Eigenschappen in op Mappen

Om het systeem te laten werken dienen deze eigenschappen op mappen te worden ingesteld. Wanneer u een bestand of map vastlegt worden de eigenschappen vanuit die map gelezen.

Als de eigenschappen daar niet kunnen worden gevonden, zal TortoiseSVN opwaards door de mapstructuur lopen om ze te vinden totdat het een map zonder versiebeheer tegenkomt, of de mapstructuur basis (bijv. C:\) bereikt. Als u er zeker van bent dat alle gebruikers alleen van bijv. `trunk/` ophalen en niet van een losse sub-map, dan volstaat het om de eigenschappen op `trunk/` in te stellen. Als u er niet zeker van bent, dan moet u de eigenschappen recursief op elke sub-map instellen. Het instellen van een eigenschap dieper in de hiërarchie van het project overschrijft de instellingen van de hogere niveau's (dichter bij `trunk/`).

Voor `tsvn`: eigenschappen kun je het **Recursief** selectievak *alleen* gebruiken om de eigenschap op alle sub-mappen in de mapstructuur toe te passen, zonder het ook op alle bestanden in te stellen.



Geen Issue Tracker Informatie vanuit de Archief Verkenner

Omdat de issue tracker integratie afhankelijk is van de toegang tot subversion eigenschappen kunt u de resultaten alleen zien wanneer u een opgehaalde werkkopie gebruikt. Het vanaf afstand ophalen van de eigenschappen is een langdurige operatie, daarom zal u deze mogelijkheid niet in actie zien vanuit de archief-verkenner, tenzij u de archief-verkenner vanuit uw werkkopie hebt opgestart. Als u de archief verkenner heeft opgestart met ingave van de URL naar het archief zal u deze mogelijkheid niet zien.

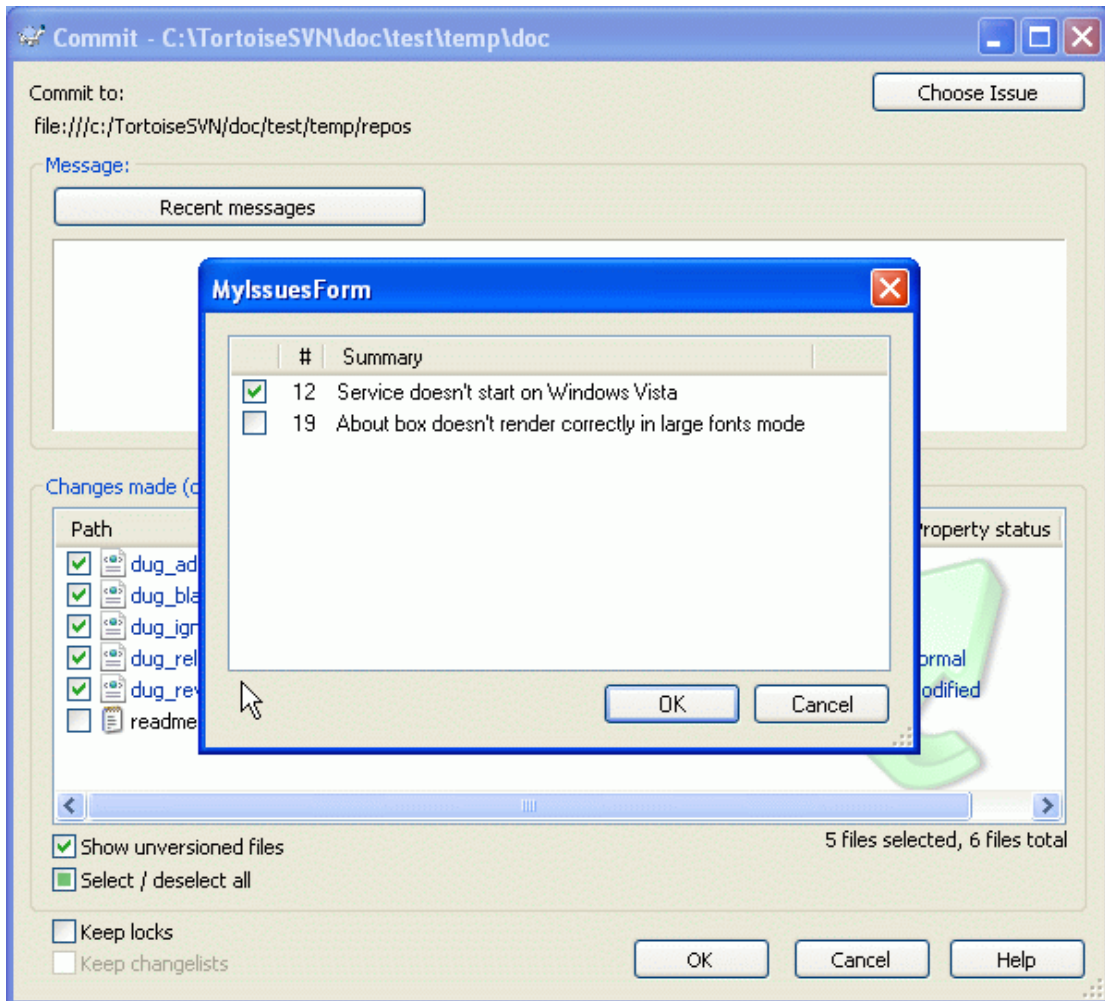
Deze issue tracker integratie is niet exclusief voor TortoiseSVN; het kan worden gebruikt met elke Subversion client. Voor meer informatie, lees de volledige *Issue Tracker Integratie Specificatie* [<http://tortoisesvn.googlecode.com/svn/trunk/doc/issuetrackers.txt>] in het TortoiseSVN bron archief. (Paragraaf 3, “**TortoiseSVN is gratis!**” geeft een uitleg over hoe het archief te benaderen).

4.28.2. Informatie Verkrijgen vanuit de Issue Tracker

De vorige sectie betrof het toevoegen van issue informatie aan de logberichten. Maar wat als u de informatie wilt verkrijgen uit de issue tracker? Het vastleg scherm bezit een COM interface welke de mogelijkheid biedt een extern programma, wat met uw tracker kan praten, te integreren. Gewoonlijk wilt u de tracker een lijst vragen met openstaande issues op uw naam, zodat u die issues eruit kunt pikken welke betrekking hebben op deze vastlegging.

Zo'n interface is natuurlijk specifiek voor uw issue tracker systeem, zodat we dit niet kunnen leveren. De beschrijving van hoe zo'n programma te schrijven valt buiten de scope van deze handleiding. De interface definitie en voorbeeld plugins in C# en C++/ATL zijn beschikbaar in de `contrib` map in het *TortoiseSVN archief* [<http://tortoisesvn.googlecode.com/svn/trunk/contrib/issue-tracker-plugins>]. (Paragraaf 3, “**TortoiseSVN is gratis!**” geeft uitleg over hoe het archief te benaderen). Ook is een samenvatting van de API beschikbaar op **Hoofdstuk 6, *IBugtraqProvider interface***. Een andere (werkende) voorbeeld plugin in C# is *Gurtle* [<http://code.google.com/p/gurtle/>] welke gebruik maakt van de vereiste COM interface om met de *Google Code* [<http://code.google.com/hosting/>] issue tracker samen te werken.

Als voorbeeld: Aangenomen dat uw systeemadministrator u een issue tracker plugin heeft aangeleverd welke u heeft geïnstalleerd, en dat u enkele van uw werkkopieën zo heeft ingesteld, met behulp van het TortoiseSVN instellingen scherm, dat ze gebruik maken van de plugin. Wanneer u het vastleggen scherm opent vanuit de werkkopie waaraan de plugin is toegekend, zult u een nieuwe knop zien aan de bovenkant van het venster.



Figuur 4.49. Voorbeeld issue tracker zoek scherm

In dit voorbeeld kunt u één of meerdere issues selecteren. De plugin kan vervolgens een speciaal opgemaakte tekst genereren welke deze toevoegt aan uw logbericht.

4.29. Integratie met Web-gebaseerde Archief Viewers

Er zijn meerdere web-gebaseerde archief viewers beschikbaar welke kunnen worden gebruikt met Subversion zoals [ViewVC](http://www.viewvc.org/) [http://www.viewvc.org/] en [WebSVN](http://websvn.tigris.org/) [http://websvn.tigris.org/]. TortoiseSVN biedt de mogelijkheid om deze viewers te koppelen.

U kunt een archief viewer naar keuze integreren in TortoiseSVN. Hiervoor dient u een aantal eigenschappen te definiëren welke de koppeling definiëren. Deze dienen op Mappen te worden ingesteld: (Paragraaf 4.17, "Project Instellingen")

webviewer:revision

Stel deze eigenschap in op de URL van uw archief viewer om alle wijzigingen in een specifieke revisie te zien. Deze dient correct te zijn gecodeerd als URI en dient %REVISION% te bevatten. %REVISION% wordt vervangen door het betreffende revisienummer. Hierdoor kan TortoiseSVN een contextmenu optie tonen in het log venster Contextmenu → Toon revisie in webviewer

webviewer:pathrevision

Stel deze eigenschap in op de URL van uw archief viewer om de wijzigingen aan een specifiek bestand in een specifieke revisie te zien. Deze dient correct te zijn gecodeerd als URI en dient %REVISION%

% en %PATH% te bevatten. %PATH% wordt vervangen door het pad relatief aan de archief hoofdmap. Hierdoor kan TortoiseSVN een contextmenu optie tonen in het log venster Contextmenu → Bekijk revisie voor dit pad in webviewer Voorbeeld: als u in het onderste gedeelte van het log scherm met de rechter muisknop klikt op een bestand /trunk/src/file wordt de %PATH% in de URL vervangen door /trunk/src/file.

U kunt ook gebruik maken van relatieve URL's in plaats van absolute. Dit komt van pas in het geval dat uw webviewer zich op het zelfde domein/server bevindt als uw bronarchief. In het geval dat de domeinnaam ooit wijzigt hoeft u de webviewer:revision en webviewer:pathrevision eigenschappen niet aan te passen. De schrijfwijze is dezelfde als van de bugtraq:url eigenschap. Zie ook [Paragraaf 4.28, "Integratie met Bug Tracking Systemen / Issue Trackers"](#).



Stel de Eigenschappen in op Mappen

Om het systeem te laten werken dienen deze eigenschappen op mappen te worden ingesteld. Wanneer u een bestand of map vastlegt worden de eigenschappen vanuit die map gelezen. Als de eigenschappen daar niet kunnen worden gevonden, zal TortoiseSVN opwaards door de mapstructuur lopen om ze te vinden totdat het een map zonder versiebeheer tegenkomt, of de mapstructuur basis (bijv. C:\) bereikt. Als u er zeker van bent dat alle gebruikers alleen van bijv. trunk/ ophalen en niet van een losse sub-map, dan volstaat het om de eigenschappen op trunk/ in te stellen. Als u er niet zeker van bent, dan moet u de eigenschappen recursief op elke sub-map instellen. Het instellen van een eigenschap dieper in de hiërarchie van het project overschrijft de instellingen van de hogere niveau's (dichter bij trunk/).

Voor tsvn: eigenschappen kun je het Recursief selectievak *alleen* gebruiken om de eigenschap op alle sub-mappen in de mapstructuur toe te passen, zonder het ook op alle bestanden in te stellen.



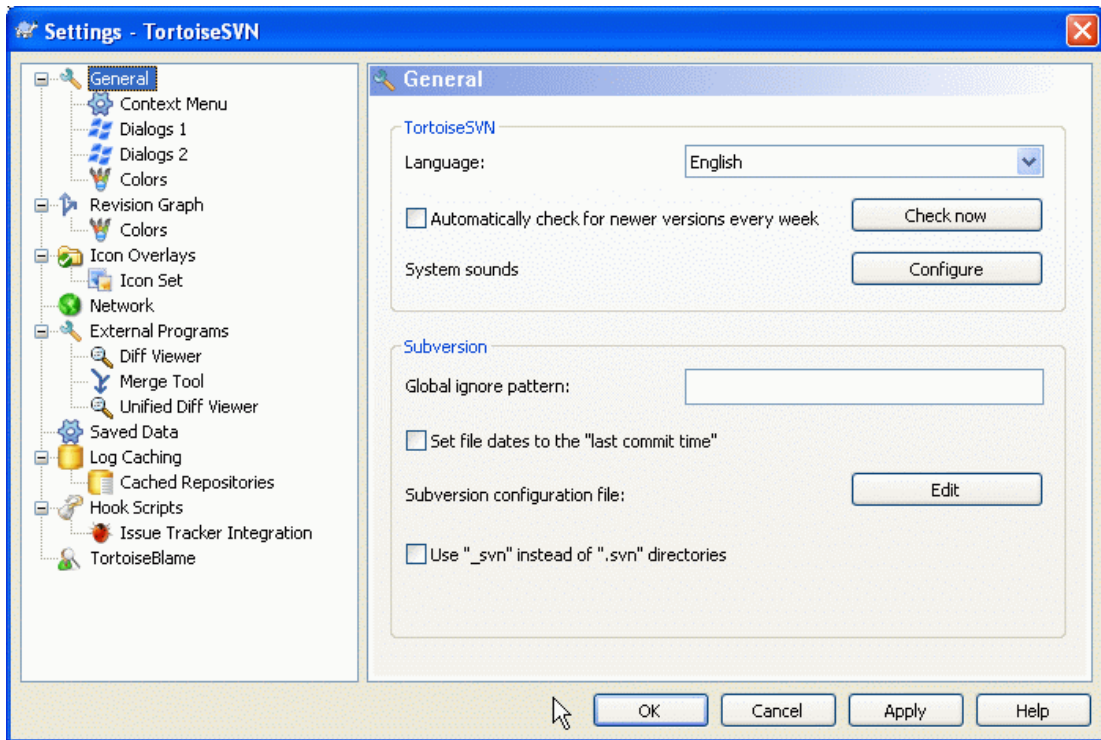
Geen Repo Viewer Koppelingen van de Archief-verkenner

Omdat de archief viewer integratie afhankelijk is van de toegang tot subversion eigenschappen kunt u de resultaten alleen zien wanneer u een opgehaalde werkkopie gebruikt. Het vanaf afstand ophalen van de eigenschappen is een langdurige operatie, daarom zal u deze mogelijkheid niet in actie zien vanuit de archief-verkenner, tenzij u de archief-verkenner vanuit uw werkkopie hebt opgestart. Als u de archief-verkenner heeft opgestart met ingave van de URL naar het archief zal u deze mogelijkheid niet zien.

4.30. TortoiseSVN Instellingen

Om uit te vinden waar de diverse instellingen voor dienen kun je met de muisaanwijzer naar het invoerveld wijzen.....en dan verschijnt er een handige "tooltip".

4.30.1. Algemene instellingen



Figuur 4.50. Het Instellingen Scherm, Algemene Pagina

In dit scherm kan de voorkeurstaal worden ingesteld en specifieke Subversion instellingen.

Taal

De taal voor de user-interface. Maar je had natuurlijk niet anders verwacht?

Automatisch (wekelijks) controleren op nieuwe versies

Als dit is aangevinkt zal TortoiseSVN elke week contact maken met de download site om te kijken of er een nieuwere versie van het programma beschikbaar is. Gebruik **Nu controleren** wanneer je direct wilt controleren op nieuwe versies. De nieuwe versie zal niet worden gedownload, maar er wordt alleen aangegeven dat er een nieuwe versie beschikbaar is.

Systeemgeluiden

TortoiseSVN heeft standaard drie eigen geluidseffecten geïnstalleerd.

- Fout
- Opmerking
- Waarschuwing

Om andere geluidseffecten te selecteren (of het geluid helemaal uit te zetten) kan gebruik gemaakt worden van het standaard Windows Configuratiescherm (Control Panel). Instellen is een koppeling naar dit Configuratiescherm.

Globaal neger patroon

Patroonherkenning kan worden gebruikt om ervoor te zorgen, dat bestanden die niet onder versiebeheer vallen ook niet zullen worden getoond bij (bijvoorbeeld) het uitvoeren van een vastlegging. Deze bestanden worden dan ook genegeerd bij het importeren. Bestanden of mappen kunnen worden genegeerd door de namen of extensies in te voeren. Deze patronen moeten worden gescheiden door spaties, bijvoorbeeld `bin obj *.bak .~?? *.jar *. [Tt]mp`. In deze

patronen mogen géén scheidingstekens voorkomen die ook in bestandspaden worden gebruikt. Houd er rekening mee, dat er geen onderscheid gemaakt kan worden tussen bestanden en mappen. Zie [Paragraaf 4.13.1, “Patroonherkenning in negerlijsten”](#) voor meer informatie over de syntax van deze patronen.

Houd er rekening mee, dat de patronen die je opgeeft ook effect zullen hebben op andere Subversion clients op dezelfde PC, inclusief de command-line client.



Let op

Wanneer je in de Subversion configuratiefile een globaal `negeer-patroon` instelt, dan heeft die instelling voorrang op waarden die je hier invoert. Het Subversion configuratiebestand kun je openen met behulp van de [Wijzigen](#) hieronder.

Dit `negeer-patroon` raakt alle projecten. Het valt niet onder versiebeheer, dus het heeft geen effect op andere gebruikers. Daarentegen kunnen met de `svn:ignore` optie ook bestanden uitgesloten worden van versiebeheer. Zie [Paragraaf 4.13, “Negeren bestanden en mappen”](#) voor meer informatie.

Bestandsdatum overschrijven met de “datum van laatste vastlegging”

Deze optie zorgt ervoor, dat TortoiseSVN de bestandsdatum van het lokale bestand aanpast aan de laatste datum waarop dit bestand is bijgewerkt (“last commit time”) in Subversion. Standaard zal TortoiseSVN hiervoor de huidige datum gebruiken. Wanneer je software ontwikkelt is het meestal de beste keuze om de huidige datum te laten gebruiken, want ontwikkel-tools maken vaak gebruik van de bestandsdatum om te beoordelen of het bestand weer opnieuw gecompileerd moet worden. Wanneer je dan de “last commit time” heb gekozen en dan een oudere revisie terugzet, dan kan dit tot een onverwacht resultaat leiden.

Subversion configuratiebestand

Gebruik [Wijzigen](#) om het Subversion configurationbestand direct te bewerken. Sommige settings kunnen niet rechtstreeks worden aangepast door TortoiseSVN en moeten daarom op deze manier gewijzigd worden. Kijk voor meer informatie over het Subversion `config` bestand in de [Runtime Configuration Area](http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html) [http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html]. Met name de sectie over [Automatic Property Setting](http://svnbook.red-bean.com/en/1.5/svn.advanced.props.html#svn.advanced.props.auto) [http://svnbook.red-bean.com/en/1.5/svn.advanced.props.html#svn.advanced.props.auto] is daarbij interessant, en die wordt hier geconfigureerd. Merk op, dat Subversion informatie over de configuratie vanaf diverse plaatsen ophaalt en het is daarom van belang om te weten welke daarvan de hoogste prioriteit heeft. Kijk op [Configuration and the Windows Registry](http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html#svn.advanced.confarea.windows-registry) [http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html#svn.advanced.confarea.windows-registry] voor meer informatie.

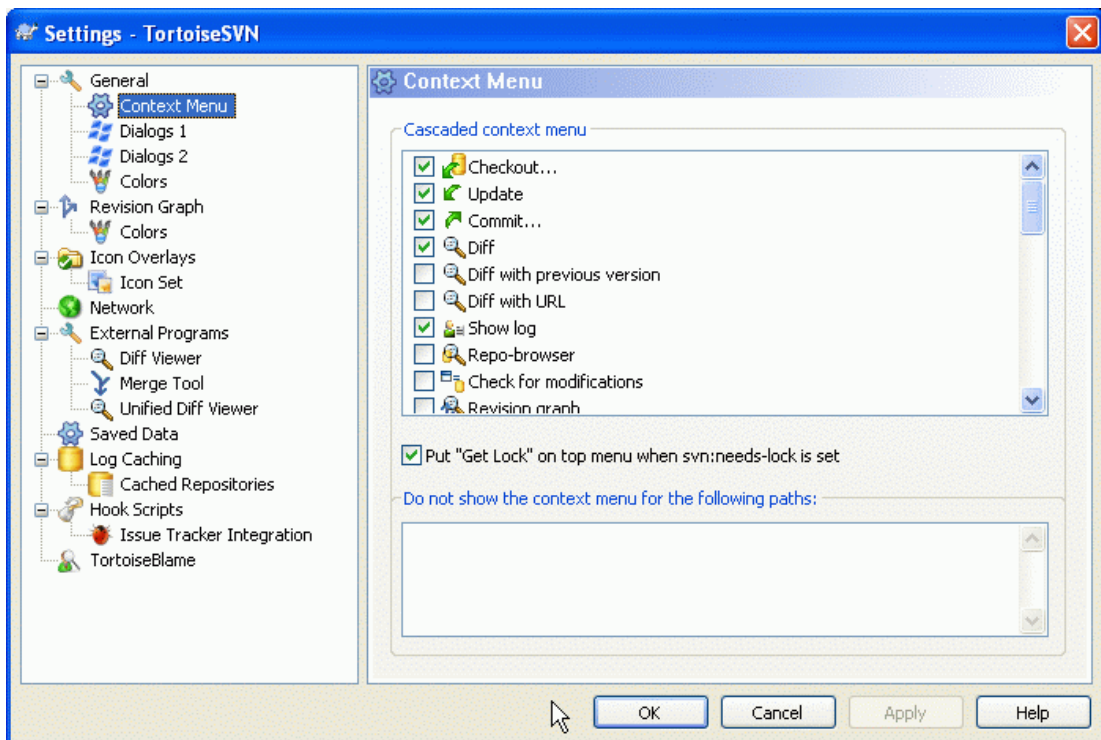
Gebruik `_svn` in plaats van `.svn` mappen

VS.NET kan - wanneer dat gebruikt wordt met web projecten - niet omgaan met `literal>.svn`

Wanneer je het gedrag van Subversion en TortoiseSVN wilt aanpassen, dan kun je dit selectievak gebruiken om de omgevingsvariabele in te stellen waarmee dit kan.

Merk op, dat het wijzigen van deze optie NIET automatisch de bestaande werkkopie zal aanpassen zodat deze gebruikmaakt van de nieuwe admin directory. Dat moet je zelf doen met behulp van een script (zie ook de veelgestelde vragen / FAQ) of maak gewoon een nieuwe werkkopie aan.

4.30.1.1. Context Menu Instellingen



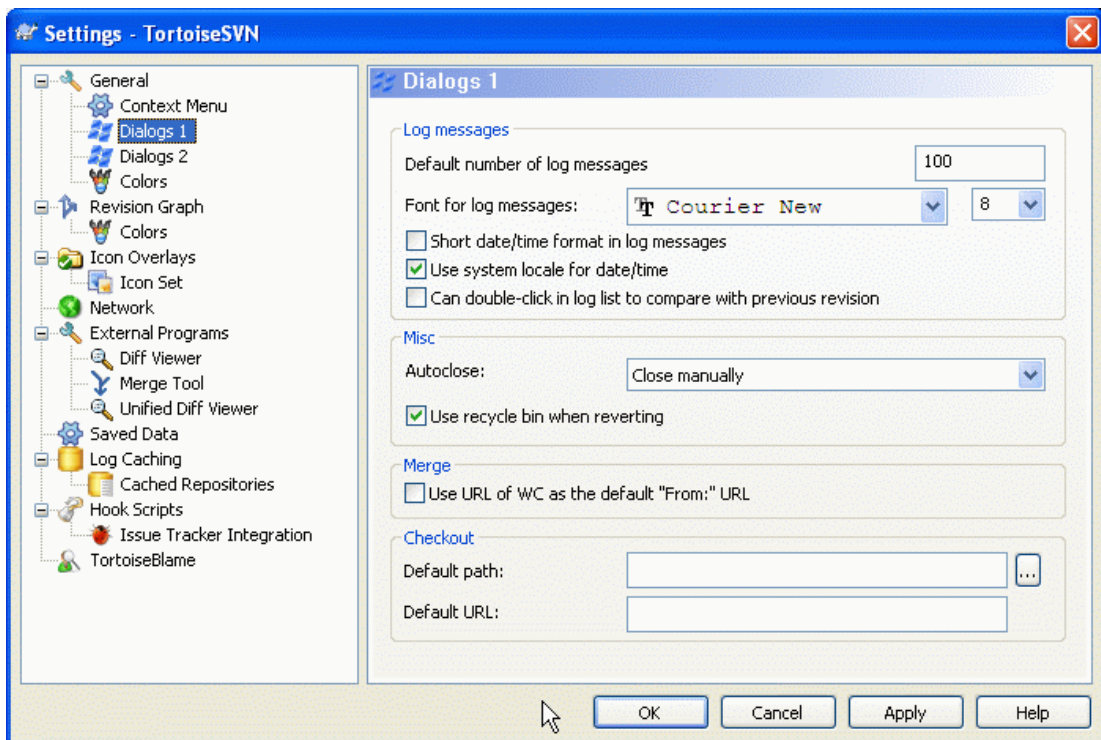
Figuur 4.51. Het Instellingen Scherm, Context Menu Pagina

Op deze pagina kun je instellen welke TortoiseSVN context menu onderdelen aan het standaard contextmenu toegevoegd moeten worden, en welke zullen verschijnen in het TortoiseSVN submenu. Standaard staan de meeste items uitgevinkt en verschijnen dus in het submenu.

Een uitzondering wordt gemaakt voor **Plaats Blokkade**. Je kunt deze menukeuze natuurlijk ook verplaatsen naar het hoogste niveau met behulp van de bovenstaande lijst, maar aangezien de meeste bestanden helemaal niet vergrendeld hoeven te worden wordt dit menu er niet overzichtelijker door. Aan de andere kant, wanneer een bestand met de `svn:needs-lock` eigenschap deze actie elke keer nodig heeft wanneer het wordt gewijzigd, dan is het wel weer handig dat deze menukeuze op het hoogste niveau getoond wordt. Het aanvinken van deze optie zorgt er voor, dat bij het selecteren van een bestand waarvoor de `svn:needs-lock` eigenschap is ingesteld, de **Plaats Blokkade** keuze getoond zal worden op het hoogste niveau.

Wanneer er mappen op je computer zijn waar je het TortoiseSVN context-menu helemaal niet wilt zien, dan kun je de paden daarnaar intoetsen in het vak onderaan.

4.30.1.2. TortoiseSVN Instellingen Venster 1



Figuur 4.52. Het Instellingen Scherm, Schermen 1 Pagina

Dit scherm geeft de mogelijkheid om de TortoiseSVN dialogen aan te passen aan je eigen voorkeuren.

Standaard aantal logbestanden

Beperkt het aantal log-regels dat TortoiseSVN toont wanneer je kiest voor TortoiseSVN → Toon logboek. Dit kan handig zijn bij trage verbindingen. Je kunt altijd kiezen voor Toon alles of Volgende 100 om meer regels te tonen.

Lettersoort voor logmeldingen

Instellen van het lettertype en de lettergrootte die worden gebruikt om de logregel te tonen in het middelste venster van de Revisie Log dialoog en bij het invoeren van de beschrijving in Vastleg dialoog.

Verkorte datum/tijd in logboek-boodschappen

Wanneer de standaard uitgebreide meldingen te veel ruimte innemen op het scherm gebruik dan de verkorte versie.

Dubbelklikken in de log-lijst vergelijkt met de vorige revisie

Wanneer je regelmatig revisies vergelijkt in het hoofdvenster van de log dialoog, dan kun je deze optie gebruiken om deze actie met een dubbelklik uit te voeren. Standaard is deze optie niet ingeschakeld, omdat het uitvoeren van de vergelijking vaak lang duurt en veel mensen dus liever voorkomen dat je deze optie per ongeluk kunt aanklikken.

Voortgangsdialoog

TortoiseSVN kan automatisch alle vensters met meldingen over de voortgang te sluiten wanneer de actie zonder fouten is afgerond. Deze instelling maakt het mogelijk om de condities hiervoor op te geven. De standaard (aanbevolen) instelling is Niet automatisch sluiten waarmee je alle meldingen te zien krijgt en dus kunt controleren wat er allemaal is gebeurd. Maar je kunt er ook voor kiezen om bepaalde typen meldingen te negeren en de bijbehorende vensters automatisch te laten sluiten wanneer er geen bijzonderheden zijn.

Automatisch sluiten, behalve na samenvoegingen, verwijderen of toevoegen wil zeggen dat de voortgangsvenster zal sluiten wanneer er eenvoudige updates zijn, maar wanneer er wijzigingen uit de repository werden samengevoegd met de jouwe, of wanneer er bestanden zijn toegevoegd of verwijderd, dan blijft het dialoogvenster geopend. Het blijft ook geopend wanneer er conflicten of fouten zijn opgetreden bij het uitvoeren van de operatie.

Automatisch sluiten, behalve na samenvoegingen, verwijderen of toevoegen voor lokale acties wil zeggen, dat het voortgangsvenster zal sluiten onder dezelfde voorwaarden als bij Automatisch sluiten, behalve na samenvoegingen, verwijderen of toevoegen maar dan uitsluitend voor lokale handelingen zoals het toevoegen van bestanden of het ongedaan maken van wijzigingen. Bij het uitvoeren van remote acties zal het dialoogvenster wel geopend blijven.

Automatisch sluiten, behalve na conflicten versoepelt de criteria nog iets en zal het dialoogvenster zelfs sluiten in het geval dat samenvoegingen, toevoegingen of verwijderingen zijn. Mocht zich echter een conflict of fout voordoen, dan blijft het dialoogvenster wel geopend.

Automatisch sluiten, behalve na fouten sluit het dialoogvenster zelfs wanneer er conflicten zijn opgetreden. Uitsluitend wanneer er een foutsituatie is opgetreden, zoals wanneer Subversion de actie niet kan voltooien, zal het dialoogvenster geopend blijven. Bijvoorbeeld wanneer een update mislukt omdat de server onbereikbaar is of een vastlegging mislukt doordat de werkkopie te oud is.

Gebruik prullenbak bij herstellen

Wanneer je lokale aanpassingen ongedaan maakt gaan de zelf aangebrachte wijzigingen weer verloren. TortoiseSVN geeft je een extra vangnet door de aangepaste file naar de prullenbak te zenden voordat deze door de voorgaande versie worden overschreven. Wanneer je deze stap wilt overslaan kun je deze optie uit zetten.

Gebruik URL van werkkopie als standaard “Van:” URL

In de samenvoeg-dialoog wordt standaard de Van:-URL onthouden voor de volgende keer. Maar sommige mensen willen samenvoegen vanaf veel verschillende plaatsen en willen hier liever de URL van hun huidige werkkopie ingevuld hebben. Deze kan dan vervolgens aangepast worden naar een vergelijkbaar pad in een andere aftakking.

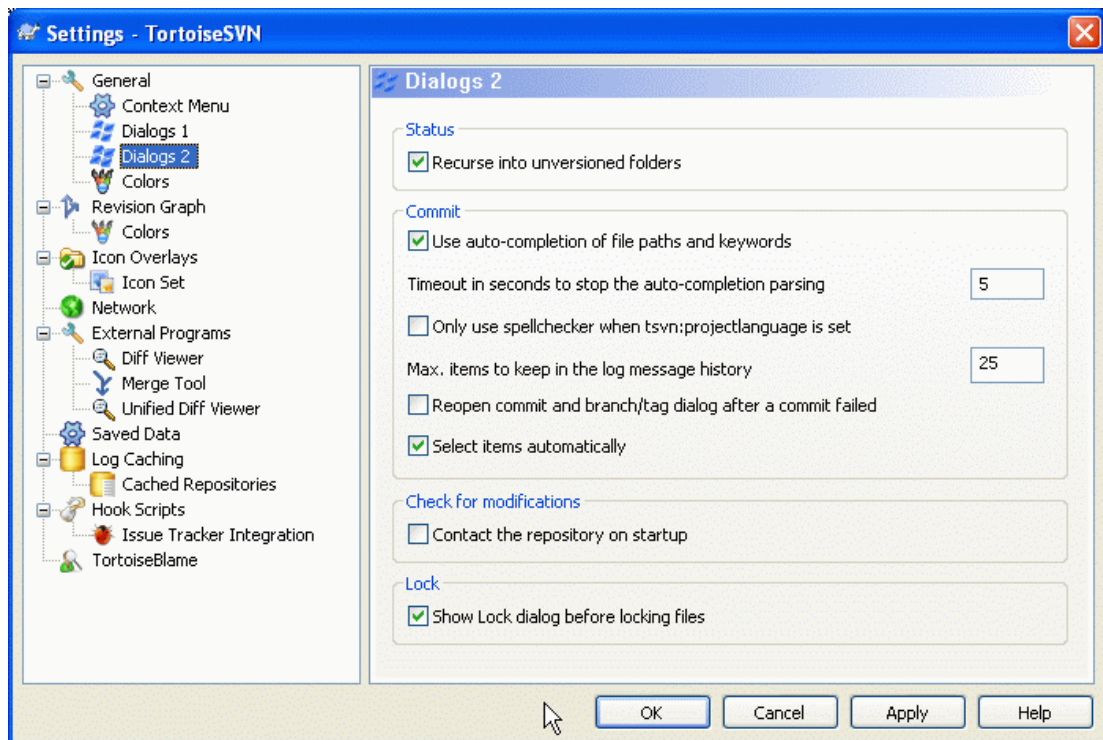
Standaard ophaalpad

Het standaard pad voor het ophalen kan hier opgegeven worden. Wanneer dit ophalen altijd op dezelfde plaats gedaan wordt, dan is het handig dat de schijfaanuiding en de map al ingevuld zijn en alleen nog de nieuwe mapnaam aan het einde ingevuld hoeft te worden.

Standaard ophaal URL

Je kunt ook de standaard URL voor het ophalen opgeven. Wanneer je vaak sub-projecten van een erg groot project ophaalt, kan het handig zijn dat de URL alvast ingevuld is, zodat je daaraan alleen nog het sub-project hoeft toe te voegen.

4.30.1.3. TortoiseSVN Instellingen Venster 2



Figuur 4.53. Het Instellingen Scherm, Schermen 2 Pagina

Recursie toepassen op mappen die niet onder versiebeheer vallen

Wanneer deze optie is aangevinkt (standaard) wordt telkens wanneer de status van een map zonder versiebeheer wordt getoond in het **Toevoegen**, **Vastleggen** of **Kijk of er updates zijn** venster, ook elk onderliggend bestand en map getoond. Wanneer deze optie uit staat wordt alleen de bovenste map zonder versiebeheer getoond. Uitzetten van deze optie vermindert vervuiling in de dialogen. In het geval dat je een map zonder versiebeheer selecteert voor toevoegen, dan wordt deze recursief toegevoegd.

Gebruik automatisch-aanvullen voor bestandslocaties en trefwoorden

De dialoog voor het vastleggen kent ook een mogelijkheid om een lijst met namen van de betreffende bestanden te kunnen verwerken. Wanneer je de eerste 3 letters van een item in de lijst intoetst verschijnt de "automatisch-aanvullen" pop-up en kun je Enter toetsen om de bestandsnaam aan te vullen. Vink dit vakje aan om deze functie te activeren.

Timeout (in seconden) waarna automatisch-aanvullen wordt gestopt

De automatische-aanvul functie kan erg langzaam zijn als er veel grote bestanden gecontroleerd moeten worden. Met deze timeout voorkom je dat het vastleg scherm te lang opgehouden wordt. Als je belangrijke automatische-aanvul gegevens mis, dan kun je de timeout verlengen.

Gebruik spellingscontrole alleen als `tsvn:projectlanguage` een waarde heeft

Als je de spellingscontrole niet voor elke vastlegging wilt gebruiken, vink dan deze optie aan. De spellingscontrole zal dan nog wel beschikbaar zijn als de project eigenschappen dat vereisen.

Maximum aantal boodschappen in de geschiedenis van logboek-boodschappen

Wanneer je een log opmerking intoetst in de vastleg-dialoog, slaat TortoiseSVN dit op voor eventueel hergebruik. Standaard zullen per archief de laatste 25 opmerkingen worden bewaard, maar dit aantal kan hier ingesteld worden. Wanneer je veel verschillende archieven hebt, wil je dit aantal wellicht verlagen om te voorkomen dat het register volloopt.

Merk op dat deze instelling uitsluitend betrekking op de opmerkingen die je intoetst op deze computer. Het heeft dus niets te maken met de log cache.

Heropen vastleg- en tak/label-dialoog na een mislukte poging

Wanneer een vastlegging om de een of andere reden mislukt (de werkkopie moet vernieuwd worden, de voorbewerking voorkomt vastlegging, netwerkfout, etc), dan kun je er met deze optie voor zorgen, dat de dialoog geopend blijft voor een eventuele herhaalde poging. Maar wees je ervan bewust, dat dit kan zorgen voor problemen. Wanneer de fout vereist dat je de werkkopie moet actualiseren en die actie zorgt voor conflicten, dan moeten deze eerst worden opgelost.

Selecteer items automatisch

Normaal gesproken worden in de vastleggingsvensters alle gewijzigde (en onder versiebeheer vallende) objecten al automatisch geselecteerd voor vastlegging. Wanneer je er echter liever hebt dat er niets geselecteerd is zodat je de items handmatig kunt selecteren, zet deze optie dan uit.

Leg contact met het archief bij het opstarten

De Kijk of er updates zijn dialoog vergelijkt standaard met de werkkopie, en vergelijkt alleen met de archief wanneer je op **Controleer archief** klikt. Wanneer je altijd de archief wilt controleren, dan kun je er met deze instelling voor zorgen, dat dit standaard zo is.

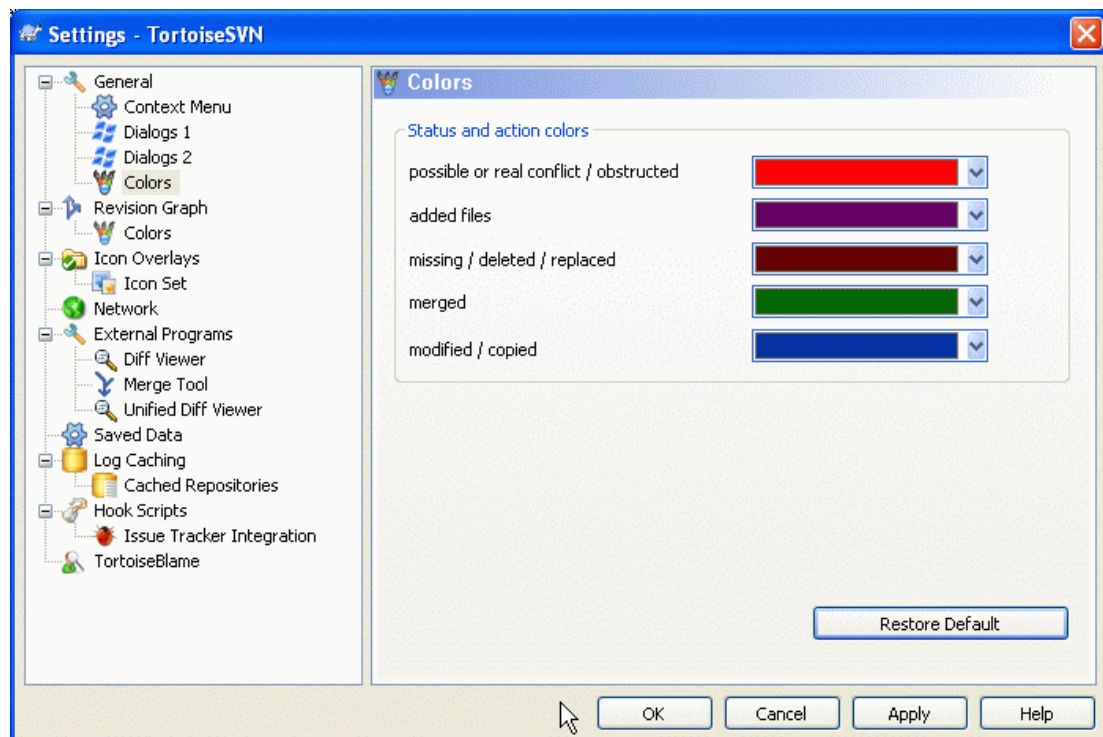
Toon de Blokkeer dialoog voordat bestanden geblokkeerd worden

Wanneer je een of meer bestanden selecteert en dan kiest voor TortoiseSVN → **Plaats blokkade** om deze bestanden te blokkeren, dan is het bij sommige projecten gebruikelijk om een blokkade-notitie achter te laten met de verklaring waarom je deze bestanden geblokkeerd hebt. Wanneer je deze blokkade-notities niet gebruikt, kun je deze optie uitschakelen om deze dialoog over te slaan en de bestanden meteen te blokkeren.

Wanneer je het blokkerr commando gebruik voor een map, dan wordt altijd de plaats blokkade dialog getoond aangezien die de mogelijkheid geeft om bestanden voor de blokkade te selecteren.

Wanneer jouw project de `tsvn:lockmsgminsize` eigenschap gebruikt, dan zal de plaats blokkade dialoog getoond worden ongeacht deze instelling aangezien het project blokkade notities *verplicht*.

4.30.1.4. TortoiseSVN kleurinstellingen



Figuur 4.54. Het Instellingen Scherm, Kleuren Pagina

Deze dialoog maakt het mogelijk om de tekstkleuren die in TortoiseSVN dialogen worden gebruikt naar eigen voorkeur in te stellen.

Mogelijk of echt conflict / geblokkeerd

Er is een conflict opgetreden tijdens een vastlegging of kan mogelijk optreden bij een samenvoeging. De vastlegging wordt geblokkeerd door een bestand/map zonder versiebeheer die dezelfde naam heeft als een met versiebeheer.

Deze kleur wordt tevens gebruikt voor foutmeldingen in de voortgangsvensters.

Toegevoegde bestanden

Aan het archief toegevoegde objecten.

Ontbreekt / verwijderd / vervangen

Items die verwijderd zijn uit het archief, ontbreken in de werkkopie of uit de werkkopie verwijderd en vervangen zijn door een gelijknamig bestand.

Samengevoegd

Wijzingen van het archief die met succes zijn samengevoegd in de werkkopie zonder dat er conflicten zijn opgetreden.

Gewijzigd / gekopieerd

Toevoegingen met historie, of gekopieerde paden in het archief. Wordt ook gebruikt in de log vensters voor onderdelen die gekopieerde objecten bevatten.

Verwijderde knoop

Een item dat is verwijderd uit het archief.

Toegevoegde knoop

Een item dat is toegevoegd aan het archief met behulp van een toevoeg, kopiër of verplaats actie.

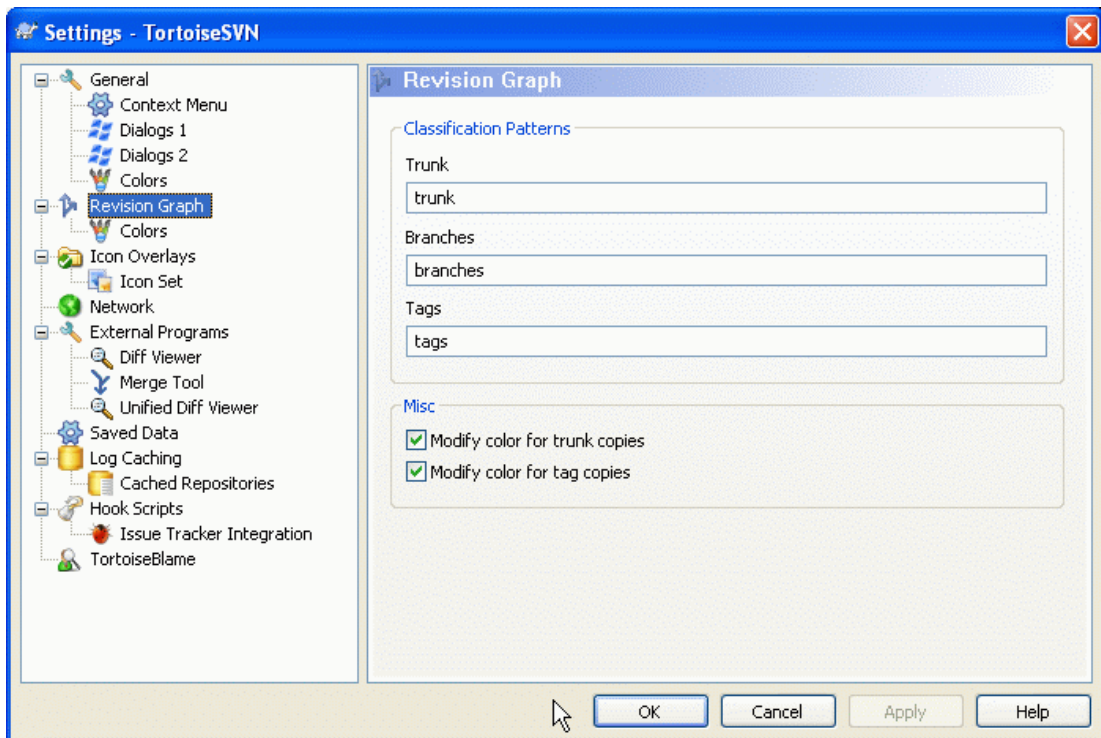
Hernoemde knoop

Een item dat is hernoemd in het archief.

Vervangen knoop

Het oorspronkelijke item is verwijderd en een nieuw item met dezelfde naam is ervoor in de plaats gekomen.

4.30.2. Revisie Grafiek Instellingen



Figuur 4.55. Het instellingen scherm, Revisie schema pagina

Classificatie patronen

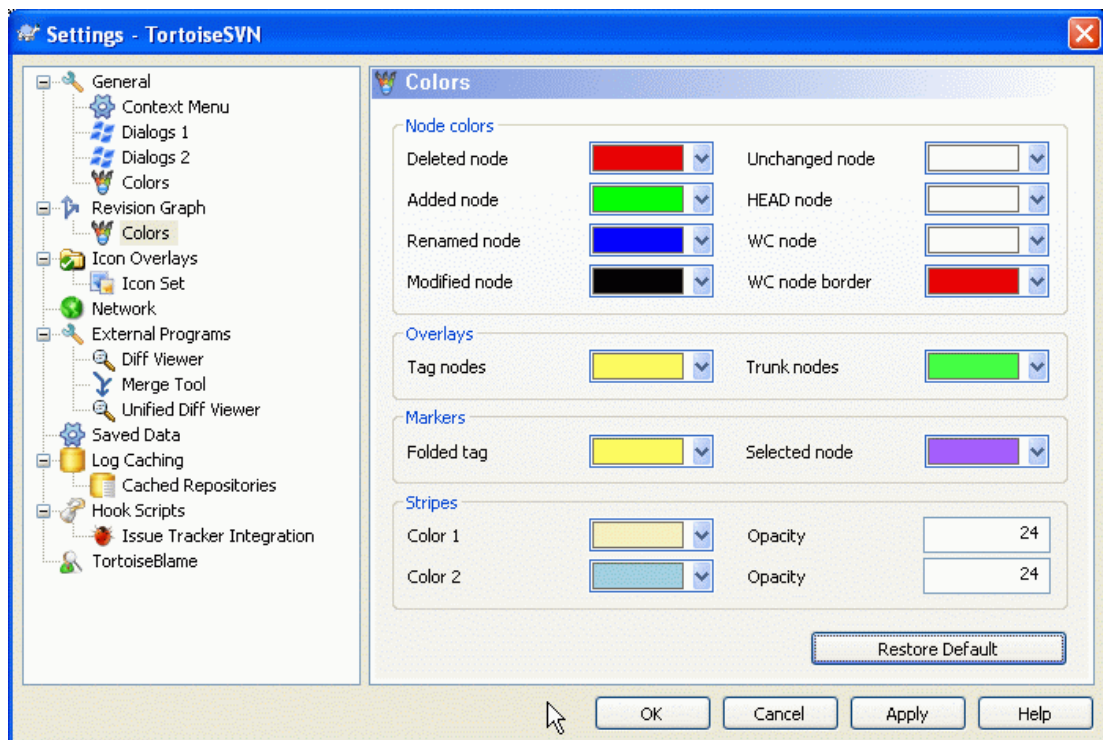
Het revisieschema tracht een duidelijker beeld te geven van uw archiefstructuur door onderscheid te maken tussen basis, vertakkingen en labels. Omdat dergelijke classificaties niet in Subversion zijn ingebouwd, wordt deze informatie uit de namen van de paden gehaald. De standaard instellingen verwachten dat u conventionele engelse benamingen gebruikt zoals voorgesteld in de documentatie van Subversion, maar kan het gebruik hiervan afwijken.

Specificeer de patronen om deze paden te herkennen in de drie gegeven velden. De patronen zullen worden niet hoofdlettergevoelig worden vergeleken, maar u dient ze in kleine letters in te geven. Jokertekens * en ? werken als gewoonlijk en u kunt ; gebruiken om meerdere patronen van elkaar te scheiden. Voeg geen extra witruimte in omdat dit wordt meegenomen in de vergelijkingsspecificatie.

Bewerk Kleuren

Kleuren worden in het revisieschema gebruikt om het type knoop aan te geven, bijv. of een knoop is toegevoegd, verwijderd of hernoemd. Om u te laten assisteren in het selecteren van de knoop classificatie kunt u het revisieschema toestaan om kleuren te mengen om een indicatie te geven van de combinatie knooptype en classificatie. Als het selectievak is aangevinkt wordt menging toegepast. Als het selectievak niet is aangevinkt wordt de kleur gebruikt om alleen het knooptype aan te geven. Gebruik het kleurselectiescherm om de gebruikte specifieke kleuren vast te leggen.

4.30.2.1. Revisie Grafiek Kleuren



Figuur 4.56. Het instellingen scherm, Revisieschema kleuren pagina

Deze pagina biedt u de mogelijkheid om de te gebruiken kleuren te configureren. Let op dat de aangegeven kleur de ondoorzichtige kleur is. De meeste knopen worden gekleurd door een mengsel van de knooptype kleur, achtergrondkleur en eventueel de classificatiekleur.

Verwijderde knoop

Items welke zijn verwijderd en niet naar elders zijn gekopieëerd binnen dezelfde revisie.

Toegevoegde knoop

Nieuw toegevoegde of gekopieëerde (toevoegen met geschiedenis) items.

Hernoemde knoop

Items welke zijn verwijderd uit de ene locatie en toegevoegd aan een andere locatie binnen dezelfde revisie.

Aangepaste knoop

Eenvoudige wijzigingen zonder enige toevoeging of verwijdering.

Ongewijzigde knoop

Kan worden gebruikt om de revisie aan te geven welke wordt gebruikt als bron van een kopie, zelfs als er geen wijziging (aan het item dat schematisch wordt weergegeven) heeft plaatsgevonden binnen die revisie.

HEAD knoop

Huidige HEAD revisie in het archief.

Werkkopie knoop

Als u er voor kiest een extra knoop weer te geven voor uw aangepast werkkopie, gekoppeld aan zijn laatste-vastlegging revisie in het schema, gebruik deze kleur.

Werkkopie knoop rand

Als u ervoor kiest om weer te geven of een werkkopie is aangepast, gebruik deze gekleurde rand voor de werkkopieknoop wanneer aanpassingen worden gevonden.

Label knopen

Knopen welke worden geclassificeerd als labels kunnen worden gemengt met deze kleur.

Basislijn knopen

Knopen welke worden geclassificeerd als basislijn kunnen worden gemengt met deze kleur.

Ingeklapt label merktekens

Als u labels inklapt om ruimte te winnen worden de labels gemarkeerd in de bronkopie door een blok in deze kleur.

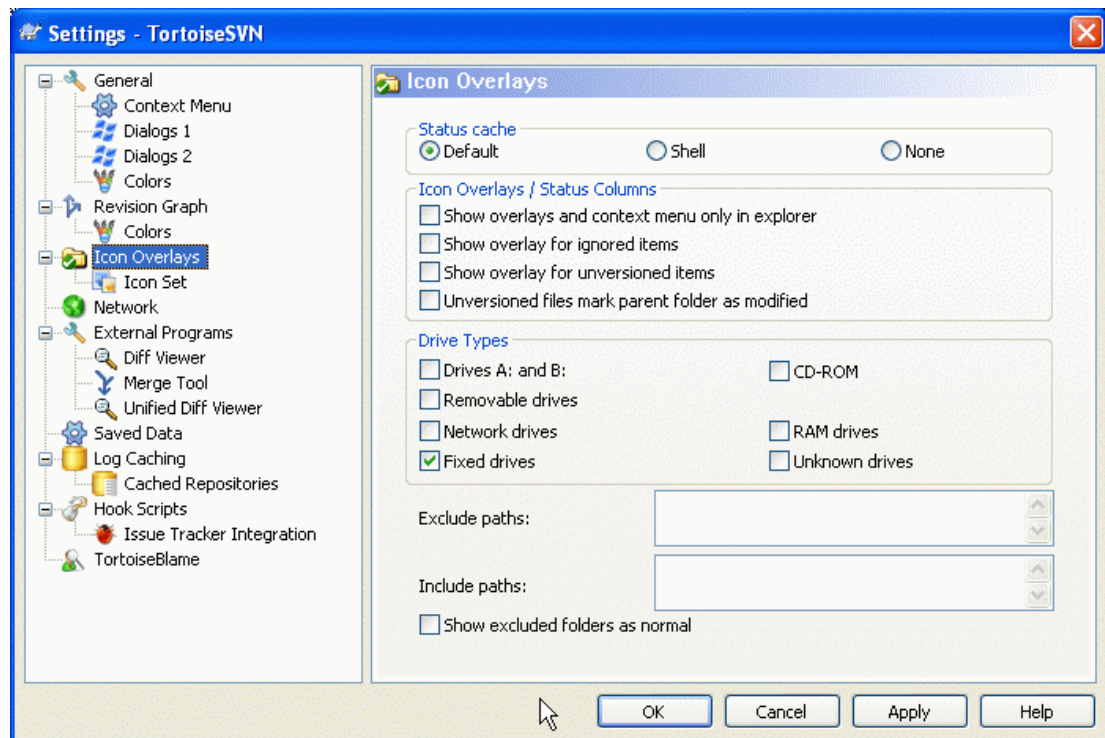
Geselecteerde knoop markering

Als u met links op een knoop klikt om deze te selecteren, is het merkteken dat wordt gebruikt om de selectie aan te geven een blok in deze kleur.

Strepen

Deze kleuren worden gebruikt wanneer het schema wordt opgesplitst in sub-boomstructuren en de achtergrond wordt gekleurd als afwisselende strepen om te helpen de verschillende boomstructuren te onderscheiden.

4.30.3. Overlappende Pictogram Instellingen



Figuur 4.57. Het instellingen scherm, Overlappend pictogram pagina

Deze pagina biedt u de mogelijkheid om de items te kiezen waarvoor TortoiseSVN de overlappende pictogrammen weergeeft.

Standaard zullen de overlappende pictogrammen en contextmenu's in alle openen/opslaan schermen alsook in Windows Verkenner verschijnen. Als u wilt dat ze *alleen* in Windows Verkenner verschijnen, vink dan het Toon overlays en contextmenu alleen in verkenner selectievak aan.

Genegeerde items en items zonder versiebeheer krijgen normaal gesproken geen overlappend pictogram. Als u wilt dat in deze gevallen een overlappend pictogram wordt getoond, vink dan gewoon de selectievakken aan.

U kunt er ook voor kiezen om mappen aan te merken als aangepast als deze items zonder versiebeheer bevatten. Dit kan handig zijn om u eraan te herinneren dat u nieuwe bestanden heeft aangemaakt welke nog niet onder versiebeheer vallen. Deze optie is enkel beschikbaar als u de *standaard* status cache optie (zie hieronder) gebruikt.

Omdat het een behoorlijke tijd kost om de status van een werkkopie op te halen, gebruikt TortoiseSVN een cache om de status in op te slaan, zodat de verkenner niet teveel stukt tijdens het tonen van de overlappende pictogrammen. U kunt het type cache hier kiezen dat TortoiseSVN moet gebruiken volgens uw systeem en grootte van werkkopie:

Standaard

Plaatst alle statusinformatie in een apart proces (*TSVNCache.exe*) in de cache. Dat proces observeert alle schijven op wijzigingen en haalt de status opnieuw op wanneer bestanden binnen een werkkopie worden gewijzigd. Het proces wordt met de laagst mogelijke prioriteit uitgevoerd zodat andere programma's er niet door worden gehinderd. Dat betekent ook dat de statusinformatie niet *realtime* is, maar het enige seconden kan duren voordat de overlappende pictogrammen wijzigen.

Voordeel: de overlappende pictogrammen tonen de status recursief, bijv. als een bestand diep in een werkkopie wordt aangepast, zullen alle mappen tot aan de basis van de werkkopie ook het aangepaste overlappend pictogram tonen. En doordat het proces berichten naar de shell kan sturen worden de overlappende pictogrammen in het linker boomstructuur venster gewoonlijk ook wijzigen.

Nadeel: het proces wordt continu uitgevoerd, zelfs als u niet aan uw projecten werkt. Het gebruikt ook zo'n 10-50 Mb RAM, afhankelijk van het aantal en grootte van uw werkkopieën.

Shell

Cachen wordt direct binnen de shell-extensie dll uitgevoerd, maar alleen voor de op dat moment zichtbare map. Iedere keer dat u naar een andere map navigeert wordt de statusinformatie opnieuw opgehaald.

Voordeel: heeft slechts weinig geheugen nodig (rond de 1Mb RAM) en kan de status in *realtime* tonen.

Nadeel: Doordat slechts één map in de cache wordt geplaatst, geven de overlappende pictogrammen de status niet recursief weer. Voor grote werkkopieën kan het meer tijd kosten om een map in de verkenner te tonen dan met de standaard cache. Ook is de mime-type kolom niet beschikbaar.

Geen

Met deze instelling haalt TortoiseSVN geen enkele status op in Verkenner. Hierdoor krijgen bestanden geen overlappend pictogram en mappen krijgen alleen een 'normaal' overlappend pictogram als zij onder versiebeheer vallen. Andere overlappende pictogrammen worden niet getoond en extra kolommen zijn ook niet beschikbaar.

Voordeel: gebruikt absoluut geen extra geheugen en vertraagt de Verkenner niet tijdens bladeren.

Nadeel: Statusinformatie van bestanden en mappen wordt in de Verkenner niet getoond. Om te zien of uw werkkopieën zijn aangepast dient u het "Kijk of er wijzigingen zijn" scherm te raadplegen.

De volgende groep biedt de mogelijkheid om de aan te geven welke opslagklassen overlappende pictogrammen moeten weergeven. Standaard zijn alleen harde schijven geselecteerd. U kunt zelfs alle overlappende pictogrammen uitzetten, maar wat kan is daar de lol van zijn?

Netwerkschijven kunnen erg traag zijn, dus worden de overlappende pictogrammen standaard niet voor werkkopieën, die zich op netwerkschijven bevinden, getoond.

USB Flash schijven blijken een speciaal geval te vormen doordat het schijftype door het apparaat zelf wordt geïdentificeerd. Sommige doen zich voor als harde schijf en andere als verwijderbare schijf.

De **Exclusief paden** worden gebruikt om aan TortoiseSVN door te geven voor welke paden *geen* overlappende pictogrammen en statuskolommen moeten worden getoond. Dit is handig als u enkele zeer

grote werkkopieën heeft, welke enkel bibliotheken bevatten die u nooit wilt wijzigen en waarvoor u daarom geen overlappende pictogrammen nodig heeft. Bijvoorbeeld:

`f:\development\SVN\Subversion` zal de overlappende pictogrammen *alleen* voor die specifieke map uitschakelen. U kunt echter de overlappende pictogrammen nog steeds zien voor alle bestanden en mappen binnen die map.

`f:\development\SVN\Subversion*` zal de overlappende pictogrammen voor *alle* bestanden en mappen uitschakelen, waarvan het pad begint met `f:\development\SVN\Subversion`. Dat betekent dat u geen overlappende pictogrammen zal zien voor elk bestand en map onder die map.

Hetzelfde geldt voor de Inclusief paden. Behalve dan dat voor die paden de overlappende pictogrammen worden getoond, zelfs als de overlappende pictogrammen zijn uitgeschakeld voor dat specifieke schijftype, of door een uitgesloten pad zoals hierboven aangegeven.

Gebruikers vragen soms hoe deze drie instellingen op elkaar reageren, en het definitieve antwoord hierop is:

```
if (pad is in de inclusief-lijst)
  toon overlappende pictogrammen
if (pad is toegestane schijftype) AND (pad is niet in exclusief-lijst)
  toon overlappende pictogrammen
```

De inclusief-lijst zorgt er *altijd* voor dat de overlappende pictogrammen worden getoond. Anders worden de overlappende pictogrammen getoond voor alle geselecteerde schijftypes *tenzij* het pad in de exclusief-lijst is aangegeven.

TSVNCache.exe gebruikt deze paden ook om zijn doorzoeken van de paden te beperken. Als u wilt dat slechts in bepaalde mappen wordt gekeken, schakel dan alle schijftypes uit en includeer alleen die mappen welke u ook daadwerkelijk wilt laten doorzoeken.



Uitsluiten van SUBST schijven

Het is vaak gemakkelijk om een SUBST schijf te gebruiken om uw werkkopieën te benaderen, bijv. door gebruik van het commando

```
subst T: C:\TortoiseSVN\trunk\doc
```

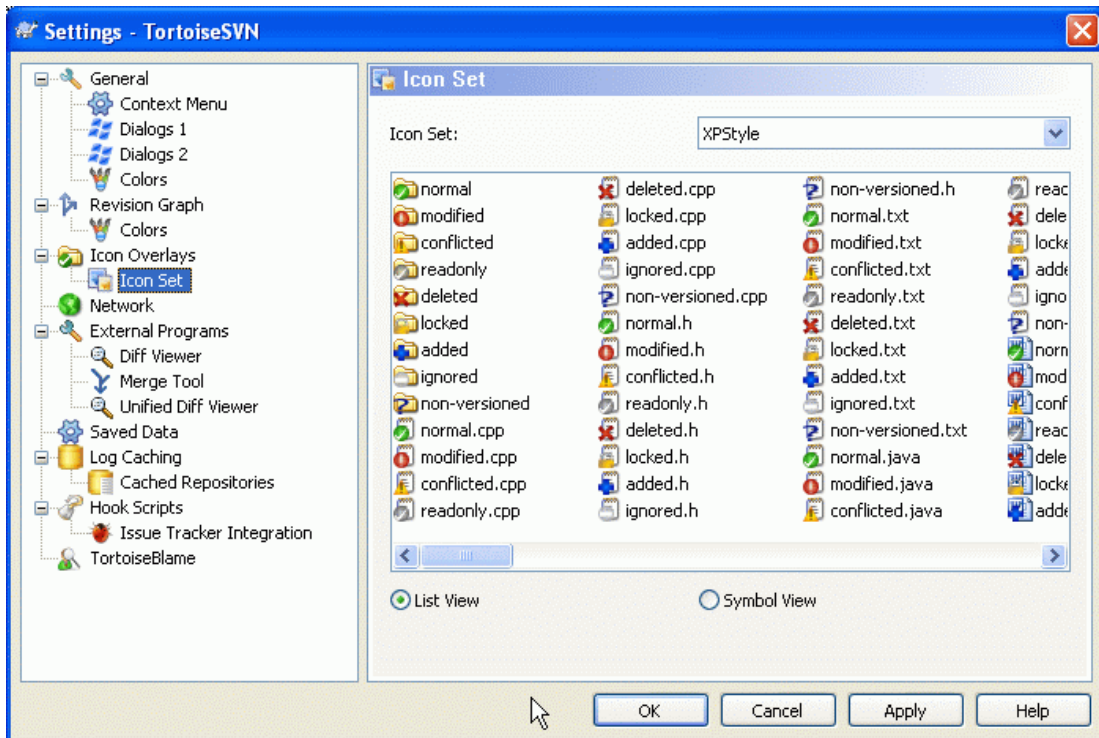
Dit kan echter leiden tot het niet bijwerken van de overlappende pictogrammen, doordat TSVNCache slechts één melding zal ontvangen wanneer een bestand wijzigt, en dat is normaal gesproken voor het originele pad. Dit zou kunnen betekenen dat de overlappende pictogrammen op het subst pad nooit worden bijgewerkt.

Een gemakkelijke manier om dit te omzeilen is door het originele pad uit te sluiten voor het tonen van overlappende pictogrammen, zodat, in de plaats hiervan, de overlappende pictogrammen worden getoond op het subst pad.

Soms zal u gebieden uitsluiten die werkkopieën bevatten, wat TSVNCache het doorzoeken en observeren van wijzigingen bespaart, maar u toch een visuele indicatie wilt hebben dat zulke mappen onder versiebeheer vallen. Dit wordt geboden door het selectievak **Toon uitgesloten mappen als 'normaal'**. Met deze optie worden mappen onder versiebeheer in elk uitgesloten gebied (schijftype niet geselecteerd of nadrukkelijk uitgesloten) tevoorschijn komen als normaal en bijgewerkt, met een groen vinkteken. Dit herinnert u eraan dat u naar een werkkopie kijkt, ook al zijn de overlappende pictogrammen voor de map mogelijk niet correct. Bestanden zullen in het geheel geen overlappende pictogrammen krijgen. Zie ook dat de contextmenu's nog steeds werken, ondanks dat de overlappende pictogrammen niet worden getoond.

Een speciale uitzondering op de regel is dat voor de schijven A : en B : nooit de optie Toon uitgesloten mappen als 'normaal' zal worden toegekend. Dit is omdat Windows wordt gedwongen om op de schijf te kijken, wat kan resulteren in een vertraging van meerdere seconden wanneer Verkenner wordt gestart, zelfs als uw PC een floppy-drive bezit.

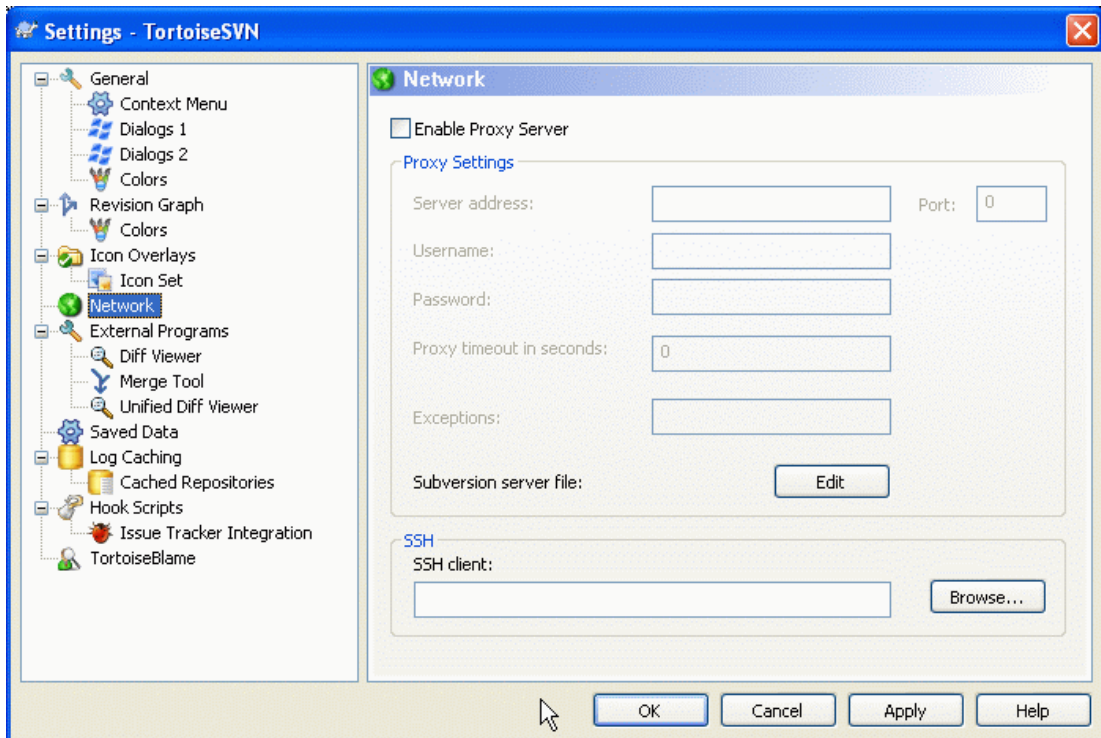
4.30.3.1. Pictogram Set Selectie



Figuur 4.58. Het instellingen scherm, Pictogramset pagina

U kunt de overlappende pictogrammen set wijzigen naar degene van uw wens. Onthoud dat wanneer u de overlappende pictogrammen set wijzigt, u mogelijk uw computer opnieuw moet opstarten om de wijzigingen te effectueren.

4.30.4. Netwerk Instellingen



Figuur 4.59. Het instellingen scherm, Netwerk pagina

Hier kunt u uw proxy-server configureren als u er een nodig heeft om door uw bedrijfs-firewall te geraken.

Als u per archief de proxy instellingen moet aangeven, dient u het Subversion servers bestand te gebruiken om dit te configureren. Gebruik Wijzigen om hier direct bij te komen. Raadpleeg de [Runtime Configuration Area](http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html) [http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html] voor details omtrent het gebruik van dit bestand.

U kunt ook aangeven welk programma TortoiseSVN moet gebruiken om een veilige verbinding te leggen met een svn+ssh archief. We raden aan dat u TortoisePlink.exe gebruikt. Dit is een versie van het populaire Plink programma en wordt meegeleverd met TortoiseSVN, maar is gecompileerd als een schermloze applicatie, dus u krijgt geen DOS-venster dat tevoorschijn komt bij elke keer dat u zich authenticceerd.

U dient het volledige pad naar het uitvoerbare bestand op te geven. In het geval van TortoisePlink.exe is dit de standaard TortoiseSVN bin map. Gebruikt de Bladeren knop om u te helpen het te localiseren. Onthoud dat als het pad spaties bevat, u het tussen aanhalingstekens moet plaatsten, bijv.

```
"C:\Program Files\TortoiseSVN\bin\TortoisePlink.exe"
```

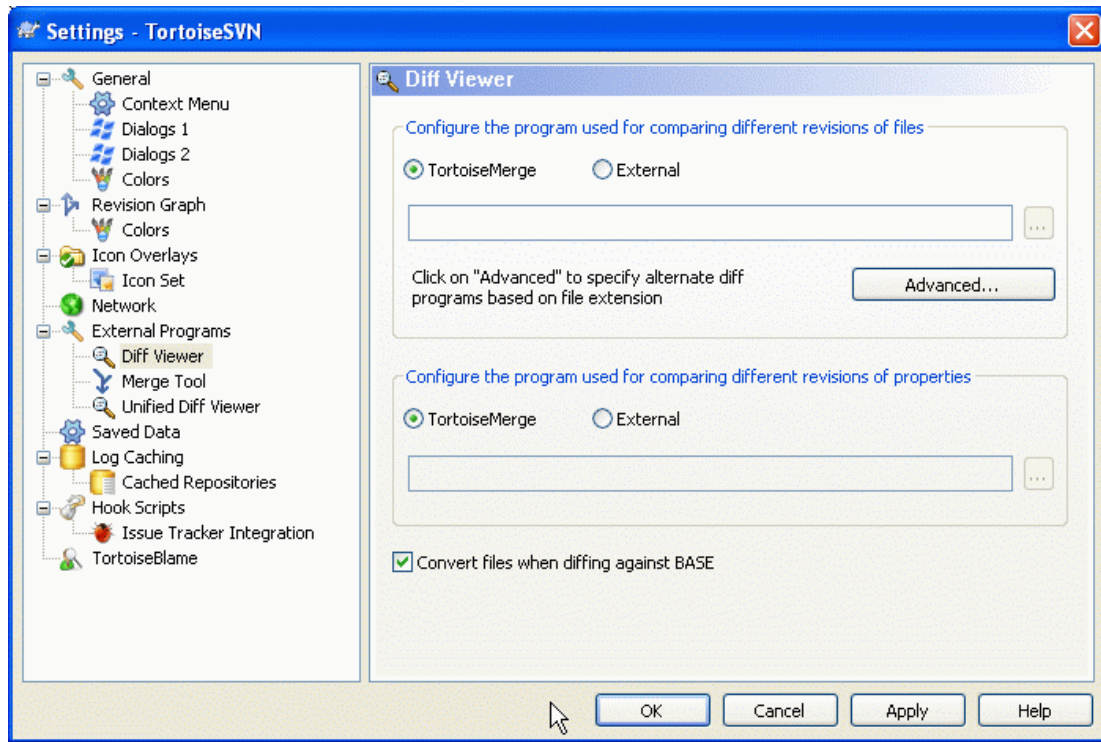
Een bijwerking van het niet hebben van een scherm is dat foutberichten nergens heen kunnen, dus als authenticatie mislukt zal u gewoon een melding krijgen welke iets aangeeft als “Kan niet schrijven naar standaard uitvoer”. Hiervoor adviseren we dat u eerst het standaard Plink instelt. Als alles werkt, kunt u TortoisePlink met exact dezelfde parameters gebruiken.

TortoisePlink heeft van zichzelf geen enkele documentatie omdat het slechts een eenvoudigere variant op Plink is. Zie voor informatie over commandoregel parameters op de [PuTTY website](http://www.chiark.greenend.org.uk/~sgtatham/putty/) [http://www.chiark.greenend.org.uk/~sgtatham/putty/]

Om te voorkomen dat u telkens om een wachtwoord wordt gevraagd, kunt u er ook voor kiezen om een wachtwoord caching hulpmiddel te gebruiken zoals Pageant. Dit is ook beschikbaar voor binnenhalen vanaf de PuTTY website.

Als laatste. Het instellen van SSH op server en clients is een niet-onbelangrijk proces dat buiten de scope van dit helpbestand valt. U kunt echter een handleiding vinden in de TortoiseSVN FAQ welke in de lijst staat onder *Subversion/TortoiseSVN SSH How-To* [http://tortoisesvn.net/ssh_howto].

4.30.5. Externe Programma Instellingen



Figuur 4.60. Het instellingen scherm, Diff viewer pagina

Hier kunt u uw eigen verschil/samenvoeg programma's aangeven welke TortoiseSVN moet gebruiken. De standaard instelling is om TortoiseMerge te gebruiken, welke tesamen met TortoiseSVN is geïnstalleerd.

Lees [Paragraaf 4.10.5, “Externe Verschil/Samenvoeg gereedschappen”](#) voor een lijst met enkele externe verschil/samenvoeg programma's welke door mensen worden gebruikt in combinatie met TortoiseSVN.

4.30.5.1. Diff Viewer

Er kan een extern verschil-programma worden gebruikt om de verschillende revisies van bestanden met elkaar te vergelijken. Het externe programma dient de bestandsnamen van de commandoregel uit te lezen, tesamen met eventuele andere commandoregel opties. TortoiseSVN gebruikt substitutie parameters voorafgegaan door een %. Als deze één van deze parameters tegenkomt, zal de parameter worden vervangen door de geschikte waarde. De volgorde van de parameters hangt van het door u gebruikte verschil-programma af.

%base

Het originele bestand zonder uw wijzigingen

%bname

De schermtitel voor het basis bestand

%mine

Uw eigen bestand, met uw wijzigingen

%yname

De schermtitel voor uw bestand

De schermtitels zijn niet de werkelijke bestandsnamen. TortoiseSVN behandelt dat als een weergavenaam en maakt de namen daarvoor aan. Dus als u bijvoorbeeld de verschillen analyseert tussen een bestand in revisie 123 en een bestand in uw eigen werkkopie, zullen de namen luiden `bestandsnaam: revisie 123` en `bestandsnaam : werkkopie`

Bijvoorbeeld, met ExamDiff Pro:

```
C:\Path-To\ExamDiff.exe %base %mine --left_display_name:%bname
--right_display_name:%yname
```

of met KDiff3:

```
C:\Path-To\kdiff3.exe %base %mine --L1 %bname --L2 %yname
```

of met WinMerge:

```
C:\Path-To\WinMerge.exe -e -ub -dl %bname -dr %yname %base %mine
```

of met Araxis:

```
C:\Path-To\compare.exe /max /wait /title1:%bname /title2:%yname
%base %mine
```

of met UltraCompare:

```
C:\Path-To\uc.exe %base %mine -title1 %bname -title2 %yname
```

Als u de `svn:keywords` eigenschap gebruikt om trefwoorden uit te breiden, en dan vooral de *revisie* van een bestand, kan er een verschil bestaan tussen bestanden welke slechts wordt veroorzaakt door de huidige waarde van het trefwoord. Ook wanneer u `svn:eol-style = native` gebruikt, heeft het BASE bestand pure LF regeleinden terwijl uw bestand CR-LF regeleinden zal hebben. TortoiseSVN zal normaal gesproken deze verschillen automatisch verbergen door eerst het BASE bestand te doorlopen om trefwoorden uit te breiden en de regeleinden aan te passen, voordat de verschil-operatie wordt uitgevoerd. Dit kan echter veel tijd kosten bij grote bestanden. Als **Bestanden converteren bij het vergelijken met BASE** niet is aangevinkt zal TortoiseSVN het vooraf bewerken van de bestanden overslaan.

U kunt ook een ander verschil-gereedschap aangeven om te gebruiken bij Subversion eigenschappen. Omdat deze gewoonlijk korte eenvoudige tekenreeksen zijn, zou u een eenvoudiger en kleinere viewer kunnen gebruiken.

Als u een alternatief verschil-gereedschap heeft geconfigureerd, kunt u TortoiseMerge *en* het gereedschap van derden via het contextmenu benaderen. Contextmenu → Verschil gebruikt het primaire verschil-gereedschap, en **Shift+** Contextmenu → Verschil gebruikt het secundaire verschil-gereedschap.

4.30.5.2. Merge Tool

Een extern samenvoegprogramma dat wordt gebruikt om conflicterende bestanden op te lossen. Er wordt op dezelfde manier van parameter substitutie gebruik gemaakt als bij het verschil-programma.

`%base`
het originele bestand zonder uw of andermans wijzigingen

`%bname`
De schermtitel voor het basis bestand

%mine
uw eigen bestand, met uw wijzigingen

%yname
De schermtitel voor uw bestand

%theirs
het bestand zoals het is in het archief

%tname
de schermtitel voor het bestand in het archief

%merged
het conflicterende bestand, het resultaat van de samenvoeg-operatie

%mname
de schermtitel voor het samengevoegde bestand

Bijvoorbeeld, met Perforce Merge:

```
C:\Path-To\P4Merge.exe %base %theirs %mine %merged
```

of met KDiff3:

```
C:\Path-To\kdiff3.exe %base %mine %theirs -o %merged  
--L1 %bname --L2 %yname --L3 %tname
```

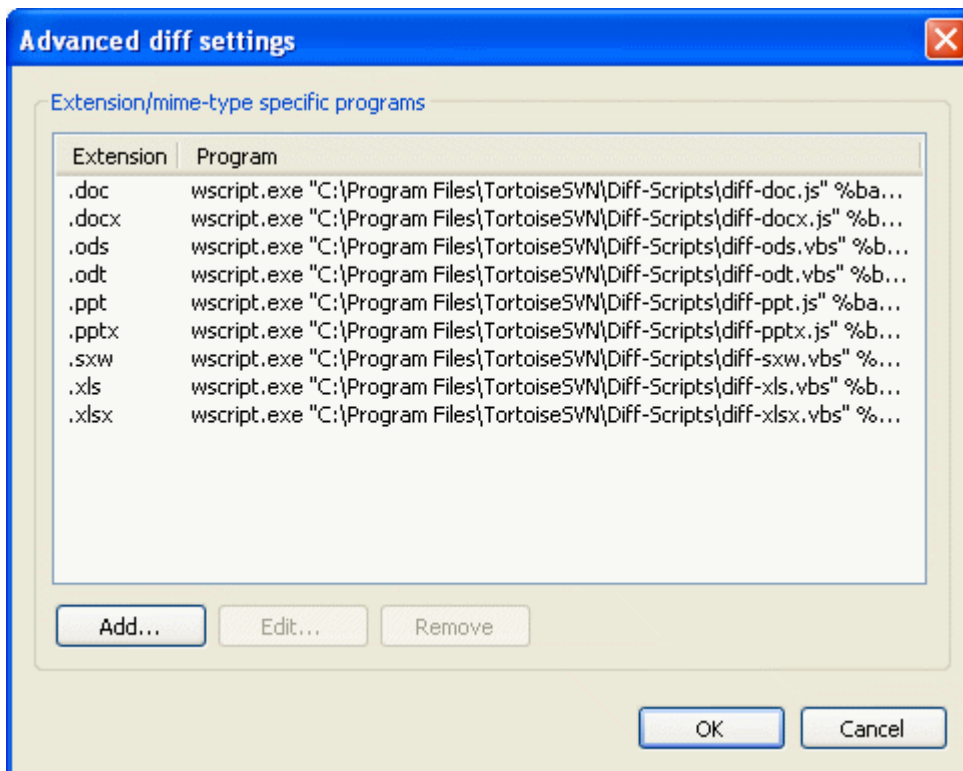
of met Araxis:

```
C:\Path-To\compare.exe /max /wait /3 /title1:%tname /title2:%bname  
/title3:%yname %theirs %base %mine %merged /a2
```

of met WinMerge (2.8 of later):

```
C:\Path-To\WinMerge.exe %merged
```

4.30.5.3. Verschil/Samenvoeg uitgebreide instellingen



Figuur 4.61. Het instellingen scherm, Verschil/Samenvoeg uitgebreid scherm

In de uitgebreide instellingen kun u voor elke bestandsextensie een ander verschil-programma en samenvoeg-programma opgeven. U kunt bijvoorbeeld Photoshop associëren als het “verschil”-programma voor `.jpg` bestanden :-). U kunt ook de `svn:mime-type` eigenschap associëren met een verschil- of samenvoegprogramma.

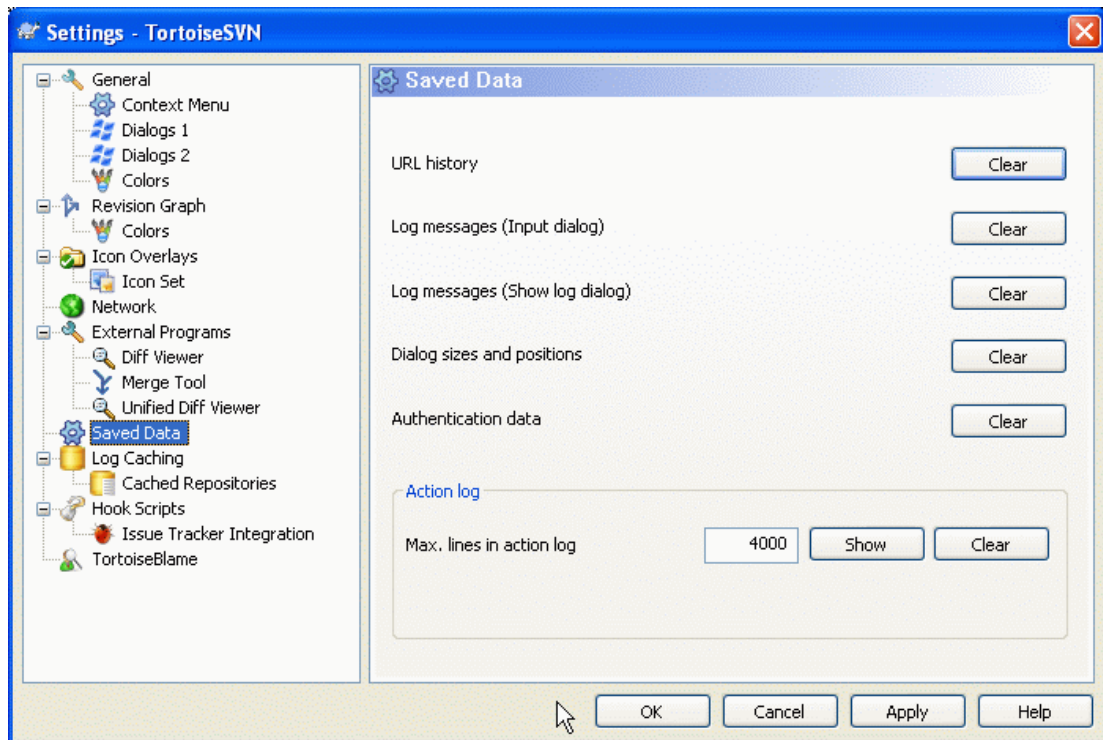
Om middels een bestandsextensie te associëren dient u de extensie op te geven. Gebruik `.bmp` om Windows bitmap bestanden aan te geven. Om te associëren middels de `svn:mime-type` eigenschap, geef dan het mime-type (inclusief een slash) op. Bijvoorbeeld `text/xml`.

4.30.5.4. Unified Diff viewer

Een viewer programma voor unified-diff bestanden (patch bestanden). Er zijn geen parameters vereist. De Standaard optie is om te zoeken naar een bestandsassociatie voor `.diff` bestanden, en daarna voor `.txt` bestanden. Als u geen viewer heeft voor `.diff` bestanden, krijgt u zeer waarschijnlijk Notepad.

Het originele Windows NotePad programma reageert niet goed op bestanden welke geen standaard CR-LF regeleinden bevat. Omdat de meeste unified diff bestanden pure LF regeleinden hebben, worden ze niet goed weergegeven in NotePad. U kunt echter een gratis NotePad vervanger binnenhalen [Notepad2](http://www.flos-freeware.ch/notepad2.html) [http://www.flos-freeware.ch/notepad2.html] welke niet alleen de regeleinden correct weergeeft, maar ook de toegevoegde en verwijderde regels kleurt.

4.30.6. Bewaarde Gegevens Instellingen



Figuur 4.62. Het instellingen scherm, Opgeslagen gegevens pagina

Voor uw gemak bewaart TortoiseSVN vele van de door u gebruikte eigenschappen en onthoudt waar u laatst bent geweest. Als u de cache van gegevens wilt schonen, kunt u dat hier doen.

URL geschiedenis

Telkens als u een werkkopie binnenhaalt, wijzigingen samenvoegt of de archief-verkenner gebruikt, houdt TortoiseSVN een lijst van de recent gebruikte URL's bij en stelt deze beschikbaar in een keuzelijst. Soms raakt die lijst vervuult met oude en ongeldige URL's, het is hierom raadzaam de lijst met regelmaat te legen.

Als u een enkel item uit een van de keuzelijsten wilt verwijderen kunt u dat ter plekke doen. Klik hiervoor op de pijl om de keuzelijst uit te klappen, beweeg de muis over het item dat u wilt verwijderen en gebruik de **Shift + Del** combinatie.

Recente boodschappen (Invoer dialoog)

TortoiseSVN bewaart de door u recent ingegeven logberichten bij vastleggen. Deze worden per archief opgeslagen, dus wanneer u veel archieven benadert kan de lijst tot een behoorlijke grootte aangroeien.

Logboek boodschappen (Toon log dialoog)

TortoiseSVN houdt de uit het Toon log scherm verkregen logberichten bij in cache om tijd uit te sparen wanneer u de volgende keer de log bekijkt. Als iemand anders een logbericht aanpast terwijl u dat bericht al in de cache heeft staan, zal u die aanpassing niet zien totdat u de cache heeft geleegd. Het cachen van logberichten wordt aangezet op de Log cache tab.

Venster afmetingen en posities

Veel vensters onthouden de afmetingen en vensterpositie die u het laatste heeft gebruikt.

Authenticatie gegevens

Als u zich authenticceerd met een Subversion server worden de gebruikersnaam en wachtwoord lokaal in de cache geplaatst zodat u deze niet elke keer hoeft in te geven. U kunt ze willen verwijderen om

veiligheids redenen of omdat u het archief onder een andere gebruikersnaam wilt benaderen ... weet Jan dat u zijn PC gebruikt ?

Als u de authenticatie gegevens wilt wissen voor slechts één specifieke server, lees [Paragraaf 4.1.5, "Authenticatie"](#) voor instructies omtrent hoe de cache gegevens te vinden.

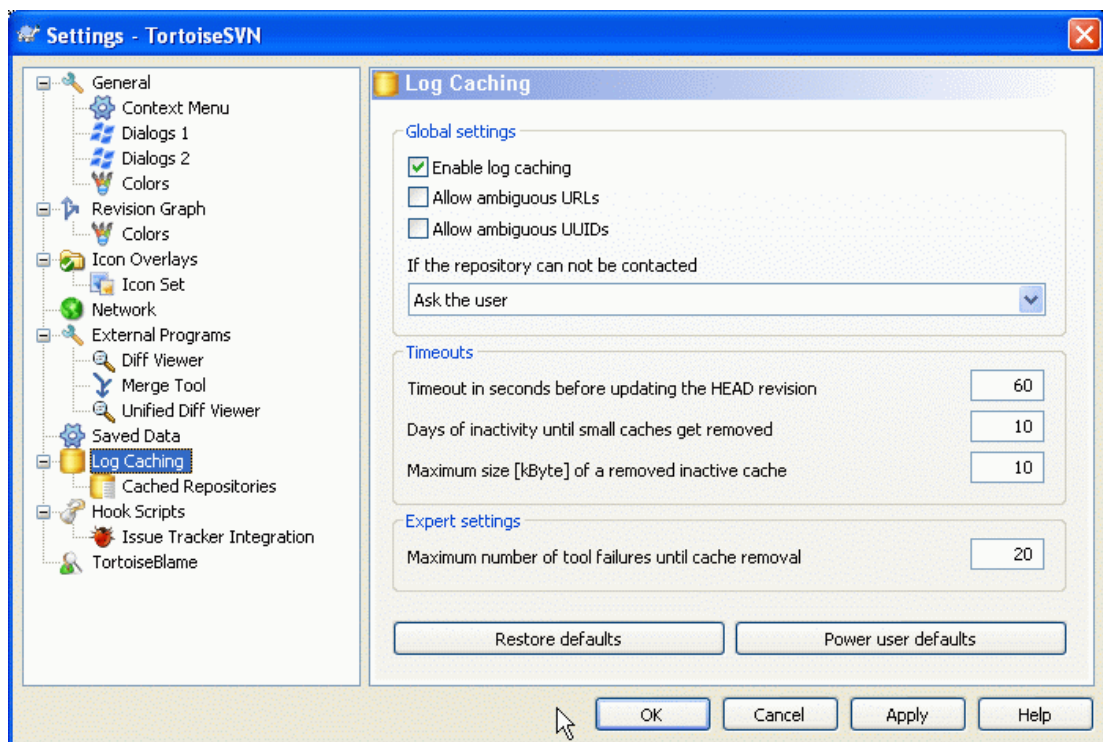
Actie-log

TortoiseSVN houdt een log bij van alles wat naar het voortgangsscherm wordt geschreven. Dit kan handig zijn voor het geval dat u bijvoorbeeld wilt controleren wat er in een recent bijwerk-commando is gebeurd.

Het logbestand is in lengte begrensd en als het te groot wordt, worden de oudste gegevens verwijderd. Standaard worden 4000 regels bewaard, maar u kunt dit aantal zelf aanpassen.

Van hieruit kunt u de inhoud van het logbestand bekijken en ook wissen.

4.30.7. Log Caching



Figuur 4.63. Het Instellingen Scherm, Log Cache Pagina

In dit scherm kun je de log cache functionaliteit van TortoiseSVN instellen, waarmee een lokale kopie van logboek boodschappen en gewijzigde paden gemaakt wordt om lange wachttijden voor het ophalen van de gegevens van een server te voorkomen. De log cache kan het gebruik van het logboek venster en het revisie schema sterk versnellen. Een andere nuttige toepassing is dat logboek boodschappen ook nog bekeken kunnen worden in offline mode.

Aanzetten van log caching

Maakt het gebruik van de log cache mogelijk als er logboek gegevens opgevraagd worden. Indien aangevinkt, worden berichten uit de cache gebruikt, als ze daarin beschikbaar zijn, en anders worden de berichten van de server gehaald en toegevoegd aan de cache.

Als caching uitgeschakeld is, dan worden de gegevens altijd direct vanaf de server opgehaald en wordt er niets lokaal opgeslagen.

Sta dubbelzinnige URLs toe

Soms moet je waarschijnlijk een server benaderen die dezelfde URL gebruikt voor alle archieven. Oudere versies van `svnbridge` doen dit bijvoorbeeld. Als je zulke archieven moet benaderen, dan moet je deze optie aanvinken. Zo niet, laat hem dan uit staan voor betere prestaties.

Sta dubbelzinnige UUIDs toe

Enkele hosting services gebruiken voor alle archieven dezelfde UUID. Het kan ook zijn dat je dit zelf gedaan hebt doordat je een archiefmap gekopieerd hebt om een nieuw archief te maken. Dit is onverstandig om verschillende redenen - een UUID moet eigenlijk *uniek* zijn. De log cache zal nog steeds werken als je deze optie selecteert. Als je dit niet nodig hebt, laat deze optie dan uitgeschakeld voor betere prestaties.

Indien het archief niet bereikt kan worden

Als je offline werkt of als de server onbereikbaar is, dan kan de log cache nog steeds gebruikt worden voor het gebruiken van logboek boodschappen die al in de cache staan. Natuurlijk kan het zijn dat de cache niet de laatste stand van zaken weergeeft. Er zijn instellingen waarmee je in kunt stellen of deze functionaliteit gebruikt kan worden.

Als er logboek gegevens uit de cache worden gebruikt zonder contact te maken met de server, wordt in de titelbalk van het venster de offline status weergegeven.

Timeout in seconden voordat de HEAD revisie ververs wordt

Als je het logboek opent wil je eigenlijk ook dat er op de server gekeken wordt of er nieuwe logboek boodschappen zijn. Als de timeout hier op een niet-nul waarde is ingesteld, dan zal de server alleen gecontacteerd worden als de timeout tijd verstreken is sinds het laatste contact. Hiermee verlaag je de belasting van de server als je server langzaam is en je het logboek venster vaak opent, maar er wordt dus misschien niet altijd de laatste stand van zaken weergegeven. Als je deze functie wilt gebruiken, adviseren wij om een waarde van 300 (5 minuten) als compromis te gebruiken.

Dagen van inactiviteit totdat kleine caches verwijderd worden

Als je door veel verschillende archieven bladert, zal je veel log caches hebben. Als je deze niet actief gebruikt, zal de cache niet echt groot groeien. TortoiseSVN zal deze standaard na een bepaalde tijd wissen. Gebruik deze instelling om het wissen van cache te configureren.

Maximale grootte van een verwijderde inactieve cache

Grotere caches zijn duurder om opnieuw te vullen, daarom wist TortoiseSVN alleen kleinere caches. Je kunt de drempel met deze waarde verfijnen.

Maximum aantal fouten tot de cache verwijderd wordt

Soms gaat er weleens wat mis met de cache met een crash tot gevolg. Als dit gebeurt, wordt de cache normaliter automatisch verwijderd om herhaling van het probleem te voorkomen. Als je de minder stabiele nachtelijk versies gebruikt, kun je er voor kiezen om de cache toch te bewaren.

4.30.7.1. Opgeslagen (cache) Archieven

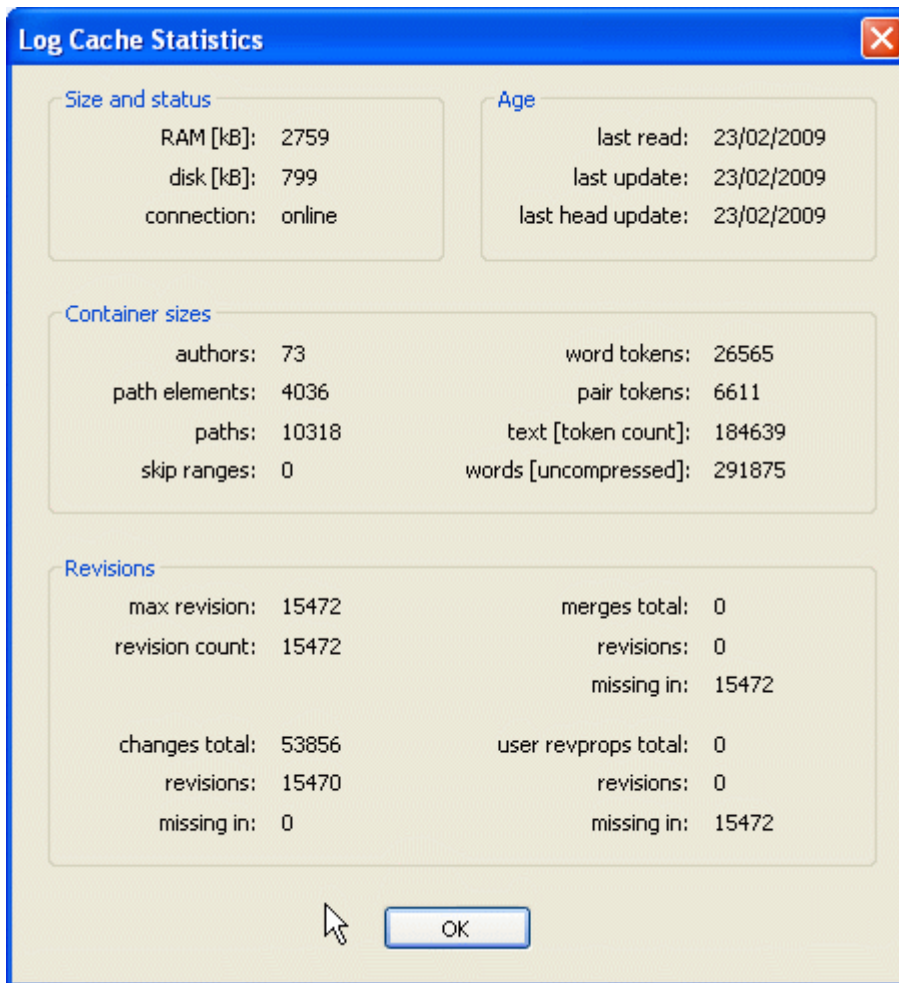
Op deze pagina zie je de lijst met archieven die in de lokale cache opgenomen zijn en de ruimte die gebruikt wordt voor de cache. Als je een van de archieven selecteert, kun je de volgende knoppen gebruiken.

Klik op de **Verversen** om de hele cache te verversen en de gaten op te vullen. Bij een groot archief kan dit veel tijd kosten, maar wel nuttig als je offline gaat en een goede cache beschikbaar wilt hebben.

Klik op de **Exporteren** knop om de gehele cache te exporteren naar een set van CSV bestanden. Dit kan handig zijn als je de logboek gegevens wilt bewerken met een extern programma, terwijl het hoofdzakelijk bedoeld is voor ontwikkelaars.

Klik op de **Verwijderen** om alle cache gegevens voor de geselecteerde archieven te verwijderen. Hiermee wordt de cache niet uitgeschakeld voor het archief. De volgende keer dat je logboek gegevens opvraagt, wordt er weer een nieuwe cache aangemaakt.

4.30.7.2. Log Cache Statistieken



Figuur 4.64. Het Instellingen Scherm, Log Cache Statistieken

Klik op de **Details** knop om gedetailleerde statistieken te zien van een bepaalde cache. Veel van de hier getoonde velden zijn alleen interessant voor ontwikkelaars van TortoiseSVN, daarom zijn ze niet allemaal in detail beschreven.

RAM

De hoeveelheid geheugen die nodig is voor het gebruik van deze cache.

Schijf

De diskruimte die in gebruik is voor de cache. Gegevens worden gecomprimeerd, gebruik van ruimte is dus beperkt.

Verbinding

Geeft weer of het archief beschikbaar was op het moment dat de cache voor het laatst gebruikt werd.

Laatste verversing

De laatste keer dat de inhoud van de cache veranderd is.

Laatste hoofd verversing

De laatste keer dat de HEAD revisie is opgevraagd bij de server.

Auteurs

Het aantal verschillende auteurs waarvan berichten zijn opgeslagen in de cache.

Paden

Het aantal paden opgenomen in de lijst, zoals je die zou zien met `svn log -v`.

Overgeslagen bereiken

Het aantal revisie bereiken die we niet opgehaald hebben, omdat ze eenvoudigweg niet opgevraagd zijn. Dit is een maatstaf voor het aantal gaten in de cache.

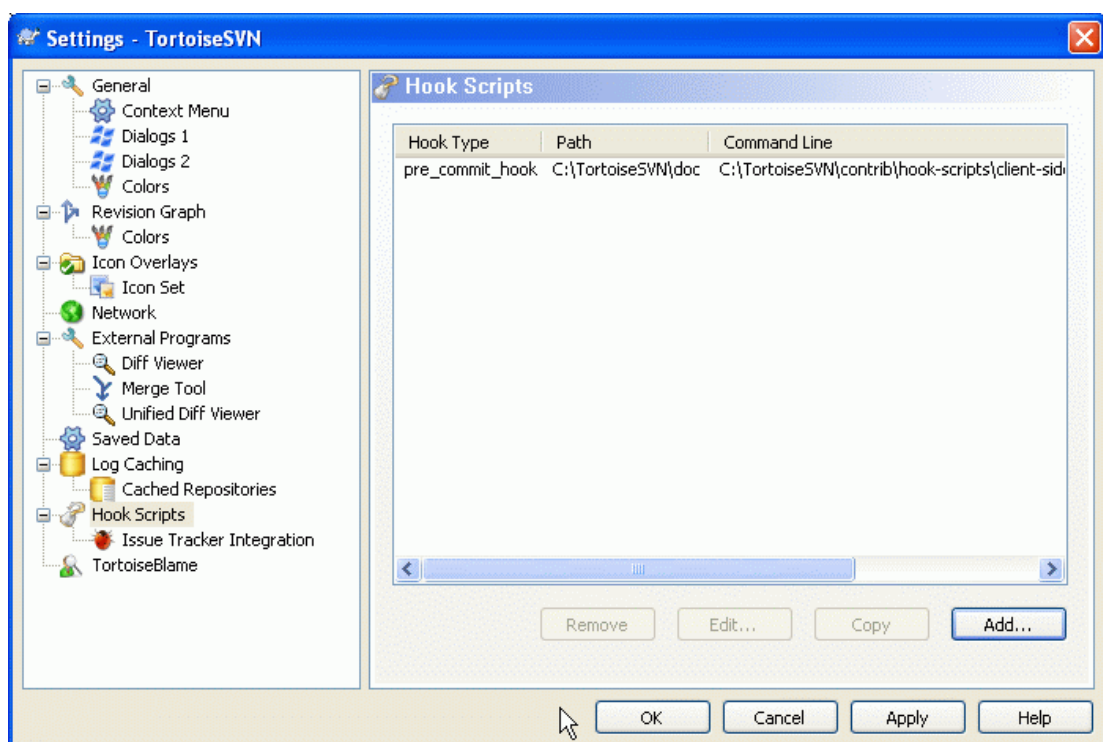
Maximale revisie

Het hoogste revisie nummer opgeslagen in de cache.

Revisie aantal

Het aantal revisies opgeslagen in de cache. Dit is een andere maatstaf voor de compleetheid van de cache.

4.30.8. Hook Scripts op de Client

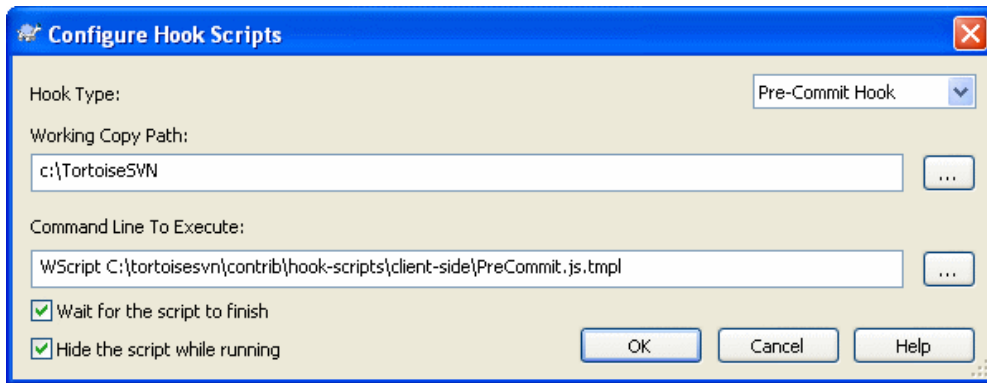


Figuur 4.65. Het Instellingen Scherm, Hook Scripts Pagina

In dit scherm kun je hook scripts instellen, welke automatisch uitgevoerd worden als een bepaalde Subversion actie plaatsvindt. In tegenstelling tot de hook scripts behandeld in [Paragraaf 3.3, “Server gesitueerde hook scripts”](#), worden deze scripts lokaal op de client uitgevoerd.

Een mogelijke toepassing voor zulke hooks kan zijn het aanroepen van een programma zoals `SubWCRev.exe` om de revisie nummers bij te kunnen werken na een vastlegging, en misschien ook om het bouwproces op te triggeren.

Vanwege verschillende veiligheid en implementatie redenen, worden hook scripts op de lokale machine gedefinieerd, in plaats van als project eigenschap. Je definieert wat er moet gebeuren, onafhankelijk van wat iemand anders in het archief vastlegt. Natuurlijk kun je er altijd voor kiezen om een script aan te roepen, die zelf ook van versiebeheer is voorzien.



Figuur 4.66. Het Instellingen Scherm, Hook Scripts Configureren

Klik op **Toevoegen** om een nieuw hook script toe te voegen en voer de details in.

Er zijn op dit moment 6 type hook scripts beschikbaar:

Start-commit

Wordt aangeroepen voordat het vastleg venster getoond wordt. Je wilt deze mogelijk gebruiken als de hook een bestand wijzigt, wat gevolg kan hebben voor de lijst met bestanden die vastgelegd moeten worden en/of voor het vastleg bericht. Hou wel in de gaten dat, omdat deze hook vroeg in het proces wordt aangeroepen, nog niet de hele lijst met geselecteerde objecten voor de vastlegging beschikbaar is.

Pre-commit

Wordt aangeroepen nadat de gebruiker op de **OK** klikt in het vastleg venster en nog voordat de echte vastlegging start. Deze hook heeft een lijst met wat er precies vastgelegd gaat worden.

Post-commit

Wordt aangeroepen nadat de vastlegging afgelopen is (succesvol of niet).

Start-update

Wordt aangeroepen voordat het verversen-naar-revisie scherm weergegeven wordt.

Pre-update

Wordt aangeroepen voordat de Subversion verversing begint.

Post-update

Wordt aangeroepen nadat een verversing klaar is (succesvol of niet).

Een hook wordt gedefinieerd voor een bepaalde werkkopie. Je hoeft alleen het pad van de basismap op te geven; als je een actie uitvoert in een sub-map, dan zoekt TortoiseSVN automatisch in de bovenliggende mappen naar het bijbehorende pad.

Vervolgens moet je de opdrachtregel specificeren die uitgevoerd moet worden, beginnend met het pad naar het hook script of de toepassing. Dit kan een batch bestand zijn, een toepassing of elk ander bestand die een geldig associatie in windows heeft, bijv. een perl script.

De opdrachtregel bevat verschillende parameters, die door TortoiseSVN ingevuld worden. Welke parameters doorgegeven worden hangt af van de aangeroepen hook. Elke hook heeft zijn eigen parameters, welke doorgegeven worden in de volgende volgorde:

Start-commit

PATHMESSAGEFILECWD

Pre-commit

PATHDEPTHMESSAGEFILECWD

Post-commit

PATHDEPTHMESSAGEFILEREVISIONERRORCWD

Start-update

PATHCWD

Pre-update

PATHDEPTHREVISIONCWD

Post-update

PATHDEPTHREVISIONERRORCWD

De betekenis van elke parameter is hier beschreven:

PATH

Een pad met een tijdelijk bestand met daarin alle paden waarvoor de operatie gestart werd. Elk pad staat op een aparte regel in het tijdelijke bestand.

DEPTH

De diepte waarmee de vastlegging/verversing uitgevoerd wordt.

Mogelijke waarden zijn:

- 2
 svn_depth_unknown
- 1
 svn_depth_exclude
- 0
 svn_depth_empty
- 1
 svn_depth_files
- 2
 svn_depth_immediates
- 3
 svn_depth_infinity

MESSAGEFILE

Pad naar een bestand met het logboek bericht van de vastlegging. Het bestand bevat de tekst met een UTF-8 codering. Na succesvolle uitvoering van de start-vastlegging hook, wordt het logboek bericht teruggelezen, waardoor de hook de kans krijgt het te wijzigen.

REVISION

De revisie van het archief naar welke de verversing uitgevoerd zal moeten worden of nadat een vastlegging gereed is.

ERROR

Pad naar een bestand met daarin de foutmelding. Als er geen fout was, dan zal het bestand leeg zijn.

CWD

De huidige werk map waarmee het script gedraaid wordt. Deze is ingesteld op de gemeenschappelijke hoofdmap voor alle betrokken paden.

Merk op dat ondanks dat we deze parameters namen hebben gegeven voor het gemak, je niet naar deze namen hoeft te refereren in de hook instellingen. Alle parameters die beschikbaar zijn voor een bepaalde hook worden altijd doorgegeven, of je ze nu wilt hebben of niet ;-)

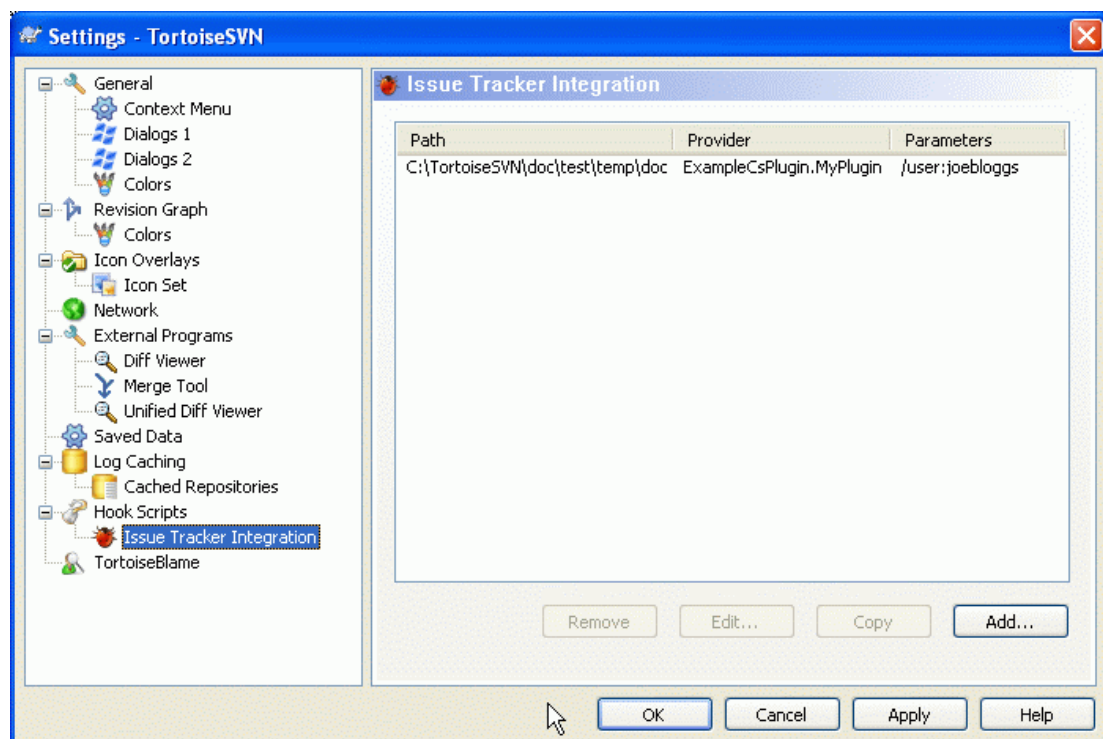
Als je wilt dat het Subversion proces wacht totdat de hook klaar is, selecteer dan Wacht op het beëindigen van het script.

Normaliter wil je de lelijke DOS schermen verbergen als je scripts gebruikt, daarom is Verberg script bij uitvoeren standaard aangevinkt.

Voorbeeld client hook scripts kunnen gevonden worden in de `contrib` map in het *TortoiseSVN* archief [http://tortoisesvn.googlecode.com/svn/trunk/contrib/hook-scripts]. Paragraaf 3, “TortoiseSVN is gratis!” beschrijft hoe je het archief benadert.

4.30.8.1. Issue Tracker Integratie

TortoiseSVN kan gebruik maken van een COM plugin om te communiceren met issue trackers vanuit het vastleg scherm. Het gebruik van plugins is beschreven in Paragraaf 4.28.2, “Informatie Verkrijgen vanuit de Issue Tracker”. Als jouw systeembeheerder een plugin beschikbaar heeft gesteld, die je al geïnstalleerd en geregistreerd hebt, dan kun je hier specificeren hoe deze moet integreren met je werkkopie.



Figuur 4.67. Het Instellingen Scherm, Issue Tracker Integratie Pagina

Klik op de Toevoegen... om de plugin met een bepaalde werkkopie te gebruiken. Je kunt hier het pad van de werkkopie specificeren, de kiezen welke plugin gebruikt moet worden uit een drop down lijst met alle geregistreerde issue tracker plugins en de parameters die meegegeven moeten worden. De parameters zijn specifiek voor een plugin, maar kunnen je gebruikersnaam voor je issue tracker bevatten, zodat de plugin problemen kan vinden dan aan jou zijn toegerekend.

Als je wilt dat alle gebruikers dezelfde COM plugin gebruiken voor je project, dan kun je de plugin ook specificeren met de eigenschap `bugtraq:provideruuid` en `bugtraq:providerparams`.

`bugtraq:provideruuid`

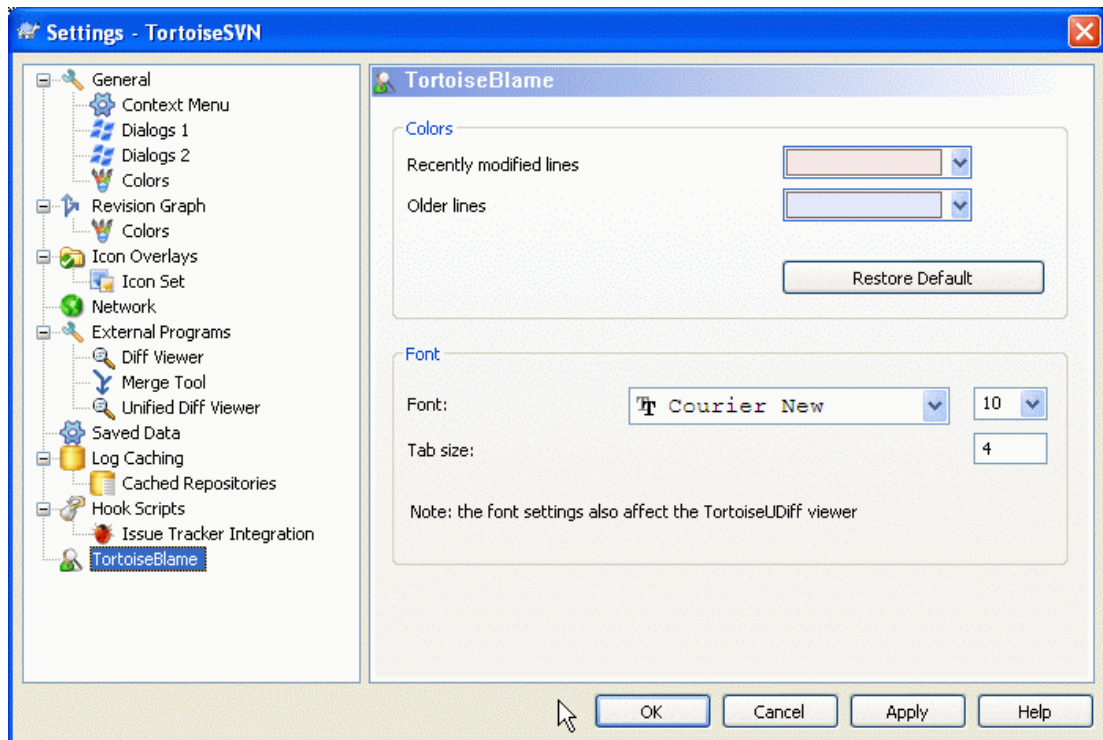
Deze eigenschap specificeert de COM UUID van de `IBugtraqProvider`, bijvoorbeeld `{91974081-2DC7-4FB1-B3BE-0DE1C8D6CE4E}`. Dit voorbeeld is de UUID van de *Gurtle bugtraq provider* [http://code.google.com/p/gurtle/], die een provider is voor de *Google Code* [http://code.google.com/hosting/] issue tracker.

bugtraq:providerparams

Deze eigenschap de eigenschappen die doorgegeven moeten worden aan de IBugtraqProvider.

Sla de documentatie van je IBugtraqProvider er op na om te achterhalen wat je moet invullen bij deze twee eigenschappen.

4.30.9. TortoiseBlame Instellingen



Figuur 4.68. Het Instellingen Scherm, TortoiseBlame pagina

De instellingen voor TortoiseBlame worden gedaan vanuit het hoofd context menu, niet direct in TortoiseBlame zelf.

Kleuren

TortoiseBlame kan gebruik maken van achtergrond kleur om de leeftijd van de regels in een bestand aan te geven. Je kunt de eindpunten aangeven door de kleuren voor de nieuwste en oudste revisies te kiezen. TortoiseBlame zal een lineaire interpolatie gebruiken tussen deze twee kleuren volgens de revisie van het archief voor elke regel.

Lettertype

Je kunt het lettertype en grootte selecteren voor tekst die weergegeven wordt. Deze instelling wordt gebruikt voor inhoud van een bestand en de auteur en revisie informatie in het linker deelvenster.

Tabs

Definieert hoeveel spaties gebruikt moeten worden als een tab karakter gevonden is in de inhoud van een bestand.

4.30.10. Register Instellingen

VS.NET 2003 kan bij web projecten niet omgaan met de .svn mappen, waarin Subversion de interne informatie in opslaat. Dit is geen bug van Subversion. De bug zit in VS.NET 2003 en de frontpage extenties die daarbij worden gebruikt.

Configuratie

Merk op dat deze bug opgelost is in VS2005 en latere versies.

Vanaf Subversion en TortoiseSVN versie 1.3.0 kun je de omgevingsvariabele `SVN_ASP_DOT_NET_HACK` gebruiken. Als deze variabele ingesteld is, dan gebruikt Subversion `_svn` mappen in plaats van `.svn` mappen. Je moet je shell opnieuw start voordat de nieuwe waarde van de variabele gebruikt wordt. Meestal moet je dan je PC opnieuw starten. Om dit te vereenvoudigen, kun je deze instelling nu wijzigen in de generieke instellingenpagina met een selectievak - lees hiervoor [Paragraaf 4.30.1, "Algemene instellingen"](#).

Voor meer informatie en andere methodes om dit probleem op voorhand te voorkomen, lees het artikel hierover in onze [FAQ](http://tortoisesvn.net/aspdotnethack) [http://tortoisesvn.net/aspdotnethack].

Enkele niet vaak gebruikte instellingen zijn alleen beschikbaar in de geavanceerde pagina van het instellingen scherm. Deze instellingen wijzigen het register en je moet weten waarvoor elke instelling gebruikt wordt en wat die doet. Wijzig deze instellingen niet, tenzij je zeker weet dat je ze wijzigen moet.

Hier kun je een andere map specificeren voor het Subversion configuratiebestand. Dit heeft effect op alle TortoiseSVN acties.

Context Menu Pictogrammen

Stel deze waarde in op `true` om een cache taakbalk pictogram toe te laten voegen voor het TSVNCache programma. Dit wordt echt alleen maar door ontwikkelaars gebruikt omdat het de mogelijkheid geeft het programma netjes af te sluiten.

Blokkeer Overlappende Pictogram Status

Als je niet wilt dat TortoiseSVN pictogrammen laat zien voor de context menu's in zijn eigen schermen, stel deze waarde dan in op `false`.

Ververs Controle URL

Het logboek venster toont de revisie van het pad van de werkkopie vetgedrukt, maar hiervoor moet het logboek venster de status van het pad van de werkkopie ophalen. Omdat dit voor grote werkkopieën lang kan duren, kun je deze waarde op `false` instellen, zodat deze functie uitgeschakeld wordt.

De besturing van de status lijst, welke in verschillende schermen wordt gebruikt (bijv. vastleggen, kijk-of-er-updates-zijn, toevoegen, ongedaan maken, ...), selecteert altijd hele rijen (dat wil zeggen, als je een entry selecteert de hele rij geselecteerd wordt, niet alleen de eerste kolom). Op zich is dat niet erg, maar de geselecteerde rij bedekt dan ook het achtergrond plaatje aan de rechter onderzijde. Dit kan er lelijk uitzien. Om de selectie van de gehele rij uit te schakelen, stel je deze waarde in op `false`

Het vastleg en logboek venster gebruiken stijlen (bijv. vet, schuingedrukt) bij vastleg berichten (lees [Paragraaf 4.4.4, "Vastleg Log Meldingen"](#) voor meer details). Als je dit niet wilt hebben, stel deze waarde dan in op `false`.

Verkenner kolommen overal

TortoiseSVN gebruikt sneltoetsen voor de opties in de context menu's. Dit kan leiden tot dubbele toewijzingen van sneltoetsen (bijv. de `SVN Vastleggen` het de **Alt-C** sneltoets, maar de `Kopieer` optie van de verkenner gebruikt die ook). Als je de sneltoetsen niet wilt gebruiken of nodig hebt voor de TortoiseSVN opties, stel deze waarde dan in op `false`.

Net zoals de verkenner, toont TortoiseSVN extra commando's als de **Shift** knop is ingedrukt als het context menu geopend is. Om TortoiseSVN altijd die uitgebreide commando's te laten zien, stel je deze waarde in op `true`.

Als je niet wilt dat de verkenner de overlappende status pictogrammen bijwerkt terwijl er een TortoiseSVN commando loopt (bijv. Verversen, Vastleggen, ...), stel deze waarde dan in op `true`.

Deze waarde bevat de URL waar vandaan TortoiseSVN een tekstbestand probeert op te halen om te achterhalen of er nieuwe versies beschikbaar zijn. Dit kan handig zijn voor beheerders van bedrijven, die willen voorkomen dat gebruikers hun TortoiseSVN bijwerken voordat zij dat geaccepteerd hebben.

De extra kolommen die TortoiseSVN toevoegt aan de detail weergave in de Windows verkenner, worden normaliter alleen weergegeven in een werkkopie. Als je deze overal wilt zien, niet alleen

in werkkopieën, stel je deze waarde in op `true`. Merk op dat de extra kolommen alleen in XP beschikbaar zijn. Visa en latere versies ondersteunen deze functie niet meer.

Als een verversing aan een werkkopie een nieuw bestand uit het archief toevoegt, van welke al een kopie aanwezig is in de werkkopie als bestand zonder versiebeheer, dan wordt standaard het lokale bestand behouden en aangegeven als (mogelijk) gewijzigd bestand. Als je prefereert dat TortoiseSVN hiervoor het bestand als conflict markeert in zulke situaties, stel deze waarde dan in op `false`.

TortoiseSVN controleert één keer per week of er een nieuwe versie beschikbaar is. Als een nieuwe versie gevonden is, dan wordt dit in het vastleg scherm aangegeven. Als je het oude gedrag terug wilt, waarbij er een apart scherm wordt weergegeven, stel deze waarde dan in op `true`.

TortoiseSVN controleert één keer per week of er een nieuwe versie beschikbaar is. Als je niet wilt dat TortoiseSVN deze controle uitvoert, stel deze waarde dan in op `false`.

De auto-aanvul lijst die weergegeven wordt in het vastleg bericht scherm, toont de namen van bestanden die meegenomen worden bij de vastlegging. Om deze name ook mee te nemen zonder de extensies, stel je deze waarde in op `true`.

4.30.11. Subversion Werk Mappen

Als je revisies van een andere tak samenvoegt en samenvoeg traceer informatie is beschikbaar, dan worden de logboek berichten van de samen te voegen revisies opgehaald om een nieuw vastleg bericht te maken. Een vooraf gedefinieerde tekst wordt gebruikt om de individuele berichten van de revisies te scheiden. Je kunt deze waarde aanpassen naar een tekst of indeling die je zelf wilt.

TortoiseSVN geeft de mogelijkheid om een extern diff programma in te stellen. De meeste programma's zijn echter niet geschikt voor het verklaren van verschillen (**Paragraaf 4.23.2, "Verklaar Verschillen"**), waardoor je misschien terug wilt vallen op TortoiseMerge in die gevallen. Stel hiervoor deze waarde in op `true`

Standaard neemt TortoiseSVN ook altijd de externen mee bij het verversen. Dit voorkomt consistentie problemen met werkkopieën. Als je echter veel externen hebt ingesteld, kan een verversing erg lang duren. Stel deze waarde in op `false` om standaard bij de verversing de externen niet mee te nemen. Om dan externen toch te verversen gebruik je `Verversen` naar `revisie...` of stel deze waarde weer in op `true`.

In de meeste schermen van TortoiseSVN kun je **Ctrl+Enter** gebruiken om het scherm te sluiten alsof je op de OK knop gedrukt hebt. Als je dit niet wilt, stel deze waarde dan in op `false`.

4.31. Laatste stap

Doneer!

Hoewel TortoiseSVN en TortoiseMerge niets kosten, kun je zelf ook helpen door het opsturen van patches en door een actieve rol te spelen bij de ontwikkeling. Je kunt ook helpen om ons op te vrolijken tijdens de eindeloze uren die we doorbrengen achter onze computers.

Tijdens het werk aan TortoiseSVN luisteren we graag naar muziek. En omdat we vele uren in dit project steken is dat *zeer veel* muziek. Daarom hebben we een aantal verlanglijstjes gemaakt met onze favoriete muziek CD's en DVD's: <http://tortoisesvn.tigris.org/donate.html> Kijk ook eens naar de lijst met mensen die hebben bijgedragen aan de ontwikkeling van dit project door het opsturen van patches of vertalingen.

Hoofdstuk 5. Het SubWCRev Programma

SubWCrev is een Windows opdrachtprompt programma, welke gebruikt kan worden voor het uitlezen van de status van een Subversion werkkopie en optioneel trefwoordvervanging met een template bestand kan uitvoeren. Dit wordt meestal gebruikt als onderdeel van het bouwproces om informatie van de werkkopie mee te nemen in het object wat je aan het bouwer bent. Meestal wordt het gebruikt om het revisienummer op te nemen in het “Over” scherm.

5.1. De SubWCRev Command Line

SubWCRev leest de Subversion status van alle bestanden in een werkkopie, standaard zonder hierbij de externen mee te nemen. Het onthoudt het hoogst gevonden vastgelegde revisienummer en de tijdstempel van die revisie. Het detecteert ook of er lokale wijzigingen in de werkkopie zijn doorgevoerd en of er bestanden verschillende revisies in de werkkopie aanwezig zijn. Het revisienummer, de reeks van verversingen en status van modificaties worden doorgegeven via stdout.

SubWCRev.exe wordt aangeroepen via de opdrachtprompt of een script en wordt ingesteld met behulp van opdrachtregel opties.

```
SubWCRev WorkingCopyPath [SrcVersionFile DstVersionFile] [-nmdfe]
```

WorkingCopyPath is het pad van de werkkopie die gecontroleerd moet worden. Je kunt SubWCRev alleen op werkkopieën gebruiken, niet direct op het archief. Het pad mag absoluut en relatief ten opzichte van de huidige werkmap zijn.

Als je SubWCRev trefwoordvervanging wilt laten uitvoeren, zodat bijvoorbeeld de velden archief revisie en URL in een bestand opgeslagen worden, dan moet je een template bestand SrcVersionFile en een resultaat bestand DstVersionFile voor de ingevulde resultaten specificeren.

Er zijn verschillen optionele opties die de manier waarop SubWCRev werkt beïnvloeden. Als je er meer dan één gebruikt, moeten ze telkens als een enkele groep gespecificeerd worden, bijv. -nm, en niet -n -m.

Optie	Beschrijving
-n	Als deze optie gebruikt wordt, zal SubWCRev stoppen met ERRORLEVEL 7 als de werkkopie lokale wijzigingen heeft. Dit kan gebruikt worden om te voorkomen dat er een pakket gebouwd wordt met niet vastgelegde wijzigingen.
-m	Als deze optie gebruikt wordt, zal SubWCRev stoppen met ERRORLEVEL 8 als de werkkopie meerdere revisies bevat. Dit kan gebruikt worden om te voorkomen dat er een pakket gebouwd wordt met een gedeeltelijk bijgewerkte werkkopie.
-d	Als deze optie gebruikt wordt, zal SubWCRev stoppen met ERRORLEVEL 9 als het doel bestand al bestaat.
-f	Als deze optie gebruikt wordt, zal SubWCRev ook de laatst gewijzigde revisie van de mappen ophalen. Het standaard gedrag is zo dat alleen de revisienummers van bestanden opgehaald worden.
-e	Als deze optie gebruikt wordt, zal SubWCRev mappen onderzoeken met <code>svn:externals</code> , maar alleen als ze van hetzelfde archief zijn. Standaard worden externen genegeerd.
-x	Als deze optie gebruikt wordt, zal SubWCRev het revisienummer in HEX formaat teruggeven.

Optie	Beschrijving
-X	Als deze optie gebruikt wordt, zal SubWCRev het revisienummer in HEX formaat teruggeven, vooraf gegaan door '0X'.

Tabel 5.1. Lijst met beschikbare opdrachtregel opties

5.2. Trefwoordvervangning

Als een bron en doel bestand ondersteund wordt, kopieert SubWCRev de bron naar de bestemming, terwijl het trefwoordvervangning als volgt gebruikt:

Trefwoord	Beschrijving
\$WCREV\$	Vervangen door de hoogste vastgelegde revisie in de werkkopie.
\$WCDATES\$	Vervangen door de vastleg datum/tijd van de hoogste vastgelegde revisie. Standaard wordt de internationale indeling gebruikt: jjjj-mm-dd uu:mm:ss. Je kunt ook een eigen indeling specificeren die gebruikt worden met <code>strftime()</code> , bijvoorbeeld: \$WCDATE=%a %b %d %I:%M:%S %p\$. Bekijk de online referentie [http://www.cppreference.com/stddate/strftime.html] voor een lijst met karakters die gebruikt kunnen worden voor het instellen van de indeling.
\$WCNOW\$	Vervangen door de huidige datum/tijd van het systeem. Dit kan gebruikt worden als de tijd van het bouwen van het pakket. Tijdsindeling kan gebruikt worden zoals beschreven in \$WCDATE\$.
\$WCRANGE\$	Vervangen door de ververs revisiereeks in de werkkopie. Als de werkkopie consistent is, dan zal dit één revisienummer zijn. Als de werkkopie verschillende revisies bevat, door of achterhaalde informatie of door een opzettelijke verversen-naar-revisie opdracht, dan zal de reeks weergegeven worden in de vorm 100:200.
\$WCMIXED\$	\$WCMIXED?TText:FText\$ wordt vervangen door TText als er verschillende ververs revisies zijn, FText als dat niet het geval is.
\$WCMODS\$	\$WCMODS?TText:FText\$ wordt vervangen door TText als er lokale wijzigingen zijn, of FText als geen wijzigingen zijn.
\$WCURL\$	Vervangen door de URL van het archief van het pad van de werkkopie doorgeven aan SubWCRev.
\$WCINSVN\$	\$WCINSVN?TText:FText\$ wordt vervangen door TText als er voor deze optie versiebeheer gebruikt wordt, of FText als er geen versiebeheer gebruikt wordt.
\$WCNEEDSLOCK\$	\$WCNEEDSLOCK?TText:FText\$ wordt vervangen door TText als voor de optie de <code>svn:needs-lock</code> eigenschap ingesteld is, of FText als deze niet ingesteld is.
\$WCISLOCKED\$	\$WCISLOCKED?TText:FText\$ wordt vervangen door TText als het object geblokkeerd is, of FText als het niet geblokkeerd is.
\$WCLOCKDATES\$	Vervangen door de datum van de blokkade. Tijdsindeling kan gebruikt worden als beschreven voor \$WCDATE\$.
\$WCLOCKOWNERS\$	Vervangen door de naam van de eigenaar van de blokkade.
\$WCLOCKCOMMENT\$	Vervangen door het commentaar van de blokkade.

Tabel 5.2. Lijst met beschikbare opdrachtregel opties



Tip

Enkele van deze trefwoorden zijn toepassing op bestanden in plaats van een hele werkkopie. Het is dus alleen logisch om deze te gebruiken als SubWCREv wordt aangeroepen op een enkel bestand. Dit heeft betrekking op \$WCINSVN\$, \$WCNEEDSLOCK\$, \$WCISLOCKED\$, \$WCLOCKDATE\$, \$WCLOCKOWNER\$ en \$WCLOCKCOMMENT\$.

5.3. Voorbeeld trefwoord

Dit voorbeeld laat zien hoe trefwoorden in een template bestand opgenomen worden in een resultaat bestand.

```
// Test bestand voor SubWCREv: testfile.tmpl

char *Revision = "$WCREV$";
char *Modified = "$WCMODS?Modified:Not modified$";
char *Date      = "$WCDATE$";
char *Range     = "$WCRANGE$";
char *Mixed     = "$WCMIXED?Mixed revision WC:Not mixed$";
char *URL       = "$WCURL$";

#if $WCMODS?1:0$
#error Source is modified
#endif

// End of file
```

Nadat je SubWCREv.exe path\to\workingcopy testfile.tmpl testfile.txt hebt uitgevoerd, zal het resulterende bestand testfile.txt er zo uit zien:

```
// Test bestand voor SubWCREv: testfile.txt

char *Revision = "3701";
char *Modified = "Modified";
char *Date      = "2005/06/15 11:15:12";
char *Range     = "3699:3701";
char *Mixed     = "Mixed revision WC";
char *URL       = "http://project.domain.org/svn/trunk/src";

#if 1
#error Source is modified
#endif

// End of file
```



Tip

Een dergelijk bestand zal meegenomen worden in het bouwproces, je kunt dus verwachten dat ook onder versiebeheer valt. Zorg er voor dat alleen het template bestand van versiebeheer wordt voorzien, niet het gegenereerde bestand. Je moet anders elke keer dat je het versie bestand opnieuw genereert de wijziging vastleggen. Dat heeft dan weer tot gevolg dat je het versie bestand weer bij moet werken (omdat bijv. het revisienummer erin vermeld is).

5.4. COM interface

Als je Subversion revisie informatie vanuit andere programma's wilt gebruiken, dan kun je de COM interface van SubWCRev gebruiken. Het aan te maken object is `SubWCRev.object` en de volgende methodes kunnen gebruikt worden:

Methode	Beschrijving
<code>.GetWCInfo</code>	Deze methode doorzoekt de werkkopie en verzamelt de revisie informatie. Je moet deze natuurlijk eerst aanroepen voordat je de informatie gebruikt voor de andere methodes. De eerste parameter is het pad. De tweede parameter moet waar zijn als je de revisies van de mappen mee wilt nemen. Dit komt overeen met de <code>-f</code> opdrachtregel optie. De derde optie moet waar zijn als je <code>svn:externals</code> mee wilt nemen. Dit komt overeen met de <code>-e</code> opdrachtregel optie.
<code>.Revision</code>	De laatste vastgelegde revisie in de werkkopie. Komt overeen met <code>\$WCREV\$</code>
<code>.Date</code>	De vastleg datum/tijd van de laatste vastgelegde revisie. Komt overeen met <code>\$WCDATE\$</code>
<code>.Author</code>	De auteur van de hoogste vastgelegde revisie, oftewel de laatste persoon die wijzigingen heeft vastgelegd.
<code>.MinRev</code>	De minimale bijgewerkte versie, zoals weergegeven in <code>\$WCRANGE\$</code>
<code>.MaxRev</code>	De maximale bijgewerkte revisie, zoals weergegeven in <code>\$WCRANGE\$</code>
<code>.HasModifications</code>	True, wanneer er lokale aanpassingen zijn
<code>.Url</code>	Wordt vervangen door de URL van het archief van het pad van de werkkopie gebruikt in <code>GetWCInfo</code> . Deze is gelijk aan <code>\$WCURL\$</code>
<code>.IsSvnItem</code>	True, wanneer het item opgenomen is in versiebeheer.
<code>.NeedsLocking</code>	True, als het item de eigenschap <code>svn:needs-lock</code> heeft ingesteld.
<code>.IsLocked</code>	True, als het item vergrendeld is.
<code>.LockCreationDate</code>	De regel geeft de datum waarop de blokkade geplaatst is of is leeg als er geen blokkade is.
<code>.LockOwner</code>	De regel bevat de eigenaar van de blokkade of is leeg als het object niet geblokkeerd is.
<code>.LockComment</code>	Het bericht dat meegegeven werd toen een blokkade geplaatst is.

Tabel 5.3. COM/automatiseringsmethoden worden ondersteund

Het volgende voorbeeld geeft aan hoe je de interface zou kunnen gebruiken.

```
// testCOM.js - javascript file
// test script voor de SubWCRev COM/Automation-object

filesystem = new ActiveXObject("Scripting.FileSystemObject");

revObject1 = new ActiveXObject("SubWCRev.object");
revObject2 = new ActiveXObject("SubWCRev.object");
revObject3 = new ActiveXObject("SubWCRev.object");
revObject4 = new ActiveXObject("SubWCRev.object");

revObject1.GetWCInfo(
    filesystem.GetAbsolutePathName("."), 1, 1);
revObject2.GetWCInfo(
```

```

    filesystem.GetAbsolutePathName(".."), 1, 1);
revObject3.GetWCInfo(
    filesystem.GetAbsolutePathName("SubWCRev.cpp"), 1, 1);
revObject4.GetWCInfo(
    filesystem.GetAbsolutePathName("../.."), 1, 1);

wcInfoString1 = "Revision = " + revObject1.Revision +
    "\nMin Revision = " + revObject1.MinRev +
    "\nMax Revision = " + revObject1.MaxRev +
    "\nDate = " + revObject1.Date +
    "\nURL = " + revObject1.Url + "\nAuthor = " +
    revObject1.Author + "\nHasMods = " +
    revObject1.HasModifications + "\nIsSvnItem = " +
    revObject1.IsSvnItem + "\nNeedsLocking = " +
    revObject1.NeedsLocking + "\nIsLocked = " +
    revObject1.IsLocked + "\nLockCreationDate = " +
    revObject1.LockCreationDate + "\nLockOwner = " +
    revObject1.LockOwner + "\nLockComment = " +
    revObject1.LockComment;
wcInfoString2 = "Revision = " + revObject2.Revision +
    "\nMin Revision = " + revObject2.MinRev +
    "\nMax Revision = " + revObject2.MaxRev +
    "\nDate = " + revObject2.Date +
    "\nURL = " + revObject2.Url + "\nAuthor = " +
    revObject2.Author + "\nHasMods = " +
    revObject2.HasModifications + "\nIsSvnItem = " +
    revObject2.IsSvnItem + "\nNeedsLocking = " +
    revObject2.NeedsLocking + "\nIsLocked = " +
    revObject2.IsLocked + "\nLockCreationDate = " +
    revObject2.LockCreationDate + "\nLockOwner = " +
    revObject2.LockOwner + "\nLockComment = " +
    revObject2.LockComment;
wcInfoString3 = "Revision = " + revObject3.Revision +
    "\nMin Revision = " + revObject3.MinRev +
    "\nMax Revision = " + revObject3.MaxRev +
    "\nDate = " + revObject3.Date +
    "\nURL = " + revObject3.Url + "\nAuthor = " +
    revObject3.Author + "\nHasMods = " +
    revObject3.HasModifications + "\nIsSvnItem = " +
    revObject3.IsSvnItem + "\nNeedsLocking = " +
    revObject3.NeedsLocking + "\nIsLocked = " +
    revObject3.IsLocked + "\nLockCreationDate = " +
    revObject3.LockCreationDate + "\nLockOwner = " +
    revObject3.LockOwner + "\nLockComment = " +
    revObject3.LockComment;
wcInfoString4 = "Revision = " + revObject4.Revision +
    "\nMin Revision = " + revObject4.MinRev +
    "\nMax Revision = " + revObject4.MaxRev +
    "\nDate = " + revObject4.Date +
    "\nURL = " + revObject4.Url + "\nAuthor = " +
    revObject4.Author + "\nHasMods = " +
    revObject4.HasModifications + "\nIsSvnItem = " +
    revObject4.IsSvnItem + "\nNeedsLocking = " +
    revObject4.NeedsLocking + "\nIsLocked = " +
    revObject4.IsLocked + "\nLockCreationDate = " +
    revObject4.LockCreationDate + "\nLockOwner = " +
    revObject4.LockOwner + "\nLockComment = " +
    revObject4.LockComment;

```

```
WScript.Echo(wcInfoString1);  
WScript.Echo(wcInfoString2);  
WScript.Echo(wcInfoString3);  
WScript.Echo(wcInfoString4);
```

Hoofdstuk 6. IBugtraqProvider interface

Om een betere integratie met issue trackers te krijgen dan alleen de `bugtraq:` te gebruiken, kan TortoiseSVN gebruik maken van COM plugins. Met deze plugins is het mogelijk om direct informatie uit het issue tracker systeem te halen, met de gebruiker te communiceren, om informatie aan TortoiseSVN terug te geven over open issues, door de gebruiker ingevoerde logboek berichten te controleren en zelfs het starten van acties na een succesvolle vastlegging, bijv. het sluiten van een issue.

We kunnen je geen informatie of handreiking geven over hoe je een COM object moet implementeren in de programmeertaal van jouw voorkeur, maar we hebben wel voorbeelden in C++/ATL en C# in ons archief in de `contrib/issue-tracker-plugins` map. In die map vind je ook de benodigde include bestanden, die je nodig hebt om je plugin te bouwen. (Paragraaf 3, “TortoiseSVN is gratis!” legt uit hoe je bij het archief komt).

6.1. De IBugtraqProvider interface

TortoiseSVN 1.5 en later kunnen werken met plugins die gebruik maken van de IBugtraqProvider interface. Deze interface biedt enkele methodes die plugins kunnen gebruiken om de communiceren met een issue tracker.

```
HRESULT ValidateParameters (
    // Hoofd scherm elke UI die weergegeven
    // moet worden tijdens validatie.
    [in] HWND hParentWnd,

    // De parameter tekst die gevalideerd moet worden.
    [in] BSTR parameters,

    // Is de tekst geldig?
    [out, retval] VARIANT_BOOL *valid
);
```

Deze methode wordt aangeroepen vanuit het instellingen scherm waar de gebruiker de plugin kan toevoegen en instellen. De `parameters` tekst kan door de plugin gebruikt worden om meer informatie te verkrijgen, bijv. de URL naar de issue tracker, inlog gegevens, etc. De plugin moet de `parameters` controleren en een foutmeldingsscherm tonen als er een fout in de `parameters` zit. De `hParentWnd` parameter moet gebruikt worden voor elk scherm dat door de plugin getoond wordt. De plugin moet `TRUE` teruggeven als de validatie van de `parameters` succesvol is. Als de plugin `FALSE` teruggeeft, zal het instellingen scherm de gebruiker verhinderen de plugin aan het pad van de werkkopie toe te voegen.

```
HRESULT GetLinkText (
    // Hoofdscherm voor elke (foutmeldings) UI die getoond moet worden.
    [in] HWND hParentWnd,

    // De parameter tekst, voor het geval je informatie moet geven aan je
    // web service (bijv.) om uit te vinden wat de juiste tekst is.
    [in] BSTR parameters,

    // Welke tekst wilt je tonen?
    // Gebruik de huidige thread locale.
    [out, retval] BSTR *linkText
```

```
);
```

De plugin kan een tekst aanbieden, welke gebruikt wordt in het TortoiseSVN vastleg scherm voor de knop waarmee de plugin gestart wordt, bijv. "Kies issue" of "Selecteer ticket". Zorg er voor dat de tekst niet te lang is, anders past het misschien niet in de knop. Als de methode een foutmelding teruggeeft (bijv. E_NOTIMPL), dan zal er een standaard tekst gebruikt worden.

```
HRESULT GetCommitMessage (
    // Hoofdscherm voor je provider's UI.
    [in] HWND hParentWnd,

    // Parameters voor je provider.
    [in] BSTR parameters,
    [in] BSTR commonRoot,
    [in] SAFEARRAY(BSTR) pathList,

    // De tekst die al in het vastleg venster staat.
    // Je provider moet deze tekst opnemen in het
    // nieuwe bericht, indien gewenst.
    [in] BSTR originalMessage,

    // De nieuwe vastleg tekst.
    // Deze vervangt het originele bericht.
    [out, retval] BSTR *newMessage
);
```

Dit is de hoofd methode van de plugin. Deze methode wordt aangeroepen vanuit het TortoiseSVN vastleg scherm als de gebruiker op de plugin knop drukt. De parameter tekst is de tekst die de gebruiker moet invoeren in het instellingen scherm, als hij de plugin configureert. Meestal gebruikt de plugin deze voor de URL van de issue tracker en/of inlog informatie of meer. De commonRoot tekst bevat het bovenliggende pad voor alle geselecteerde objecten waarvoor het vastleg scherm opgeroepen wordt. Merk op dat dit *niet* het basispad is voor alle objecten die de gebruiker heeft geselecteerd in het vastleg scherm. De pathList parameter bevat een array van paden (als tekst), die de gebruiker heeft geselecteerd voor de vastlegging. De originalMessage parameter bevat de tekst die in het tekst invoerveld voor het logboek bericht van het vastleg scherm is ingevoerd. Als de gebruiker nog geen tekst heeft ingevoerd, dan zal deze parameter leeg zijn. In de newMessage parameter bevat de nieuwe informatie die door de methode terug wordt gegeven, welke in het invoerveld voor het logboek bericht de al aanwezig tekst vervangt. Als de plugin de originalMessage tekst niet veranderd, dan moet de methode dezelfde tekst teruggeven, anders gaat de informatie die de gebruiker heeft ingevoerd verloren.

6.2. De IBugtraqProvider2 interface

In TortoiseSVN 1.6 is een nieuwe interface toegevoegd, welke het gebruik van plugins mogelijk maakt. Deze IBugtraqProvider2 interface erft functionaliteit van IBugtraqProvider.

```
HRESULT GetCommitMessage2 (
    // Hoofd scherm voor je provider UI.
    [in] HWND hParentWnd,

    // Parameters voor jouw provider.
    [in] BSTR parameters,
    // De gemeenschappelijke URL van de vastlegging
    [in] BSTR commonURL,
    [in] BSTR commonRoot,
    [in] SAFEARRAY(BSTR) pathList,
```

```

// De tekst die al aanwezig is in het vastleg bericht.
// Je provider moet deze tekst opnemen in het nieuwe bericht,
// indien gewenst.
[in] BSTR originalMessage,

// Je kunt eigen revisie eigenschappen toekennen aan een vastlegging
// door de volgende twee parameters in te stellen.
// noot: Beide Both safearrays moeten dezelfde lengte hebben.
//      Elke eigenschas naam moet er een waarde hebben!

// De inhoud van het bugID veld(indien weergegeven)
[in] BSTR bugID,

// Gewijzigde inhoud van het bugID veld
[out] BSTR * bugIDOut,

// De lijst van revision eigenschapsnamen.
[out] SAFEARRAY(BSTR) * revPropNames,

// De lijst van revisie eigenschap waardes.
[out] SAFEARRAY(BSTR) * revPropValues,

// De nieuwe tekst van het vastleg bericht.
// Deze vervangt het originele bericht
[out, retval] BSTR * newMessage
);

```

Deze functie wordt aangeroepen vanaf het TortoiseSVN vastleg scherm, zodra de gebruiker op de plugin knop drukt. Deze actie wordt aangeroepen in plaats van de `GetCommitMessage()`. Lees de documentatie over `GetCommitMessage()` voor de benodigde parameters die gebruikt moeten worden. De parameter `commonURL` is de hoofd URL van alle geselecteerde objecten waarvoor het vastlegvenster geopend is. Dit is dus eigenlijk de URL voor het `commonRoot` pad. De parameter `bugID` bevat de inhoud van het bug-ID veld (als het getoond is, dit wordt ingesteld met `bugtraq:message`). De return parameter `bugIDOut` wordt gebruikt om het bug-ID veld te vullen als de functie klaar is. De `revPropNames` en `revPropValues` return parameters kunnen een naam/waarde paar bevatten voor revisie eigenschappen die veranderd moeten worden bij die vastlegging. De plugin moet er voor zorgen dat beide arrays dezelfde grootte hebben bij de return! Elke eigenschapsnaam in `revPropNames` moet ook een corresponderende waarde hebben in `revPropValues`. Als er geen revisie eigenschappen ingesteld/veranderd moeten worden, dan moet de plugin lege arrays teruggeven.

```

HRESULT CheckCommit (
    [in] HWND hParentWnd,
    [in] BSTR parameters,
    [in] BSTR commonURL,
    [in] BSTR commonRoot,
    [in] SAFEARRAY(BSTR) pathList,
    [in] BSTR commitMessage,
    [out, retval] BSTR * errorMessage
);

```

Deze functie wordt aangeroepen voordat het vastleg scherm afgesloten worden en het vastleggen start. Een plugin kan deze functie gebruiken om de geselecteerde bestanden/mappen te valideren voor vastlegging en/of het vastlegbericht van de gebruiker. De parameters worden net zo ingesteld als bij `GetCommitMessage2()`, maar dan met het verschil dat `commonURL` nu de gemeenschappelijke URL is van alle *geselecteerde* objecten, en de `commonRoot` het basispad van alle geselecteerde objecten. De return paramter `errorMessage` moet of een foutmeldig bevatten, die door TortoiseSVN aan de gebruiker getoond wordt, of leeg zijn, zodat de vastlegging kan starten. Als er een foutmelding terug wordt gegeven, zal TortoiseSVN de foutmelding in een scherm weergeven en houdt blijft het vastleg

scherm geopend, zodat de gebruiker kan corrigeren wat er fout is. Een plugin moet dus een foutmelding geven die aan de gebruiker laat weten *wat* er fout is en hoe het gecorrigeerd moet worden.

```
HRESULT OnCommitFinished (
    // Hoofdscherm voor elk (foutmelding) UI welke getoond moet worden.
    [in] HWND hParentWnd,

    // De gemeenschappelijke bron voor alle paden die vastgelegd worden.
    [in] BSTR commonRoot,

    // Alle paden die vastgelegd zijn.
    [in] SAFEARRAY(BSTR) pathList,

    // De reeds aanwezige tekst in het vastleg bericht.
    [in] BSTR logMessage,

    // De revisie van de vastlegging.
    [in] ULONG revision,

    // Een fout getoond aan de gebruiker als deze functie
    // iets anders teruggeeft dan S_OK
    [out, retval] BSTR * error
);
```

Deze functie wordt aangeroepen na een succesvolle vastlegging. Een plugin kan deze functie gebruiken om bijv. de geselecteerde bevinding de sluiten of informatie aan de bevinding toe te voegen over de vastlegging. De parameters zijn gelijk aan die voor `GetCommitMessage2`.

```
HRESULT HasOptions(
    // Of de provider opties ondersteund
    [out, retval] VARIANT_BOOL *ret
);
```

Deze functie wordt aangeroepen vanuit het instellingen scherm waar de gebruiker de plugins kan instellen. Als een plugin een eigen configuratiescherm heeft met `ShowOptionsDialog`, dan moet deze `TRUE` teruggeven, anders moet deze `FALSE` teruggeven.

```
HRESULT ShowOptionsDialog(
    // Parent window for the options dialog
    [in] HWND hParentWnd,

    // Parameters voor je provider.
    [in] BSTR parameters,

    // De parameter tekst
    [out, retval] BSTR * newparameters
);
```

Deze functie wordt aangeroepen vanuit het instellingen scherm als de gebruiker op de "Options" knop klikt, welke getoond wordt als `HasOptions` `TRUE` antwoord. Een plugin kan een instellingenschermbereik weergeven om het configureren van de plugin te vereenvoudigen. De parameter tekst bevat de plugin parameters die al ingesteld zijn. De `newparameters` antwoordparameter moet de parameter tekst bevatten die de plugin samengesteld heeft uit de informatie van het instellingenschermbereik. De parameter tekst wordt doorgegeven aan alle `IBugtraqProvider` and `IBugtraqProvider2` functies.

Bijlage A. Veelgestelde Vragen (FAQ)

Omdat TortoiseSVN continue doorontwikkeld wordt, is het soms lastig om de documentatie helemaal bij te houden. We onderhouden daarom een [online veel gestelde vragen pagina \(ook bekend als FAQ\)](http://tortoisesvn.tigris.org/faq.html) [http://tortoisesvn.tigris.org/faq.html] met een selectie van vragen die we het meest binnen krijgen op de TortoiseSVN maillijst <dev@tortoisesvn.tigris.org> en <users@tortoisesvn.tigris.org>.

We hebben ook een project [Issue Tracker](http://issues.tortoisesvn.net) [http://issues.tortoisesvn.net] waarin enkele van onze plannen in een lijst zijn opgenomen en welke bugs er al opgelost zijn. Als je denkt dat je een bug gevonden hebt of een nieuwe functie wilt aanvragen, kijken dan eerst hier om na te gaan of er niet iemand je voor is geweest.

<users@tortoisesvn.tigris.org> is het adres dat je moet gebruiken als je vragen over het gebruik van TortoiseSVN hebt.

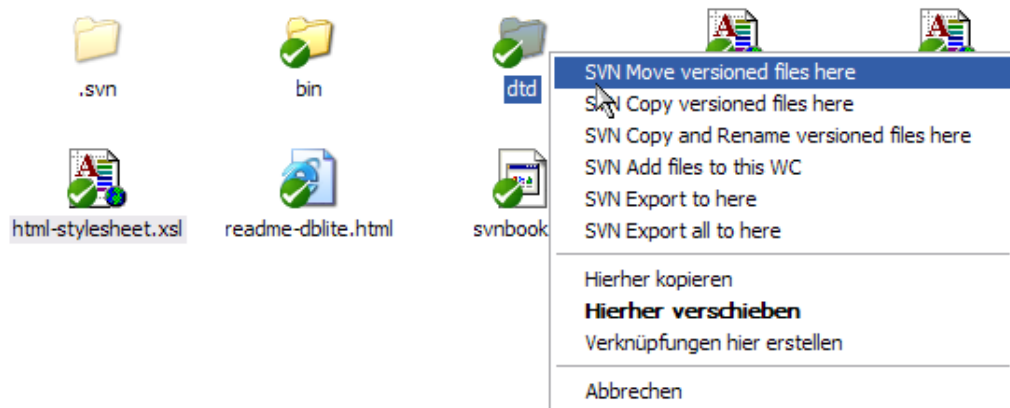
Bijlage B. Hoe kan ik...

Deze appendix bevat oplossingen voor problemen/vragen die je mogelijk hebt bij het gebruik van TortoiseSVN.

B.1. Verplaats een hoop bestanden in één keer

Verplaatsen/Kopiëren van enkele bestanden kan gedaan worden met TortoiseSVN → Hernoem.... Als je echter veel bestanden wilt verplaatsen/kopiëren, dan is deze methode veel te langzaam en kost het te veel werk.

De aanbevolen methode is door de bestanden met rechts te slepen naar de nieuwe locatie. Klik met rechts op de bestanden die je wilt verplaatsen/kopiëren zonder de muisknop los te laten. Versleep de bestanden naar de nieuwe locatie en laat de muisknop los. Een context menu zal getoond worden waarin je kunt kiezen voor Context Menu → SVN Kopiëer objecten onder versiebeheer hierheen. or Context Menu → SVN Verplaats object(en) onder versiebeheer hierheen.



B.2. Forceer gebruikers een log-bericht in te typen

Er zijn twee methodes om te voorkomen dat gebruikers wijzigingen vastleggen zonder logboek boodschap. Eentje is specifiek voor TortoiseSVN, de andere werkt op alle Subversion programma's, maar deze heeft dan weer directe toegang tot de server nodig.

B.2.1. Hook-script op de server

Als je direct toegang tot de server hebt, kun je een vastleg hook script installeren, welke een vastlegging weigert als er een leeg of te kort logboek boodschap wordt gebruikt.

In de archief map op de server is een sub-map `hooks` aanwezig. Hierin staan enkele hook script voorbeelden, die je kunt gebruiken. Het bestand `pre-commit.tmpl` bevat een voorbeeld script waarmee een vastlegging geweigerd wordt als er geen logboek boodschap ingevuld is of als het bericht te kort is. In dit bestand staat ook omschreven hoe je het script moet installeren of gebruiken. Volg de instructies die in dat bestand staan.

Deze methode is de aanbevolen methode als je gebruikers ook andere Subversion programma's gebruiken dan TortoiseSVN. Het nadeel van deze methode is dat vastlegging geweigerd wordt door de server en de gebruikers dus een foutmelding krijgen. Het programma weet niet dat de vastlegging geweigerd wordt, voordat er contact is gemaakt met de server. Als je wilt dat TortoiseSVN de OK inactief maakt, totdat er

een logboek boodschap is ingevuld die lang genoeg is, gebruik dan de methode die hieronder beschreven wordt.

B.2.2. Project eigenschappen

TortoiseSVN maakt voor enkele functies gebruik van eigenschappen. Eén van die eigenschappen is de `tsvn:logminsize` eigenschap.

Als je deze eigenschap aan een map toekent, dan zal TortoiseSVN de OK inactief maken bij alle vastleg vensters, totdat de gebruiker een logboek boodschap heeft ingevoerd met de minimale lengte zoals aangegeven in de eigenschap.

Lees voor meer informatie over zulke project eigenschappen [Paragraaf 4.17, "Project Instellingen"](#).

B.3. Update Geselecteerde Bestanden in het Archief

Normaliter ververs je je werkkopie met TortoiseSVN → Verversen. Je hebt een iets andere aanpak nodig, als je maar enkele nieuwe bestanden, die een collega heeft toegevoegd, wilt ophalen zonder meteen wijzigingen aan andere bestanden samen te laten voeren.

Gebruik TortoiseSVN → Kijk of er updates zijn en klik op Controleer archief om te zien wat er gewijzigd is in het archief. Selecteer de bestanden die je lokaal wilt verversen en gebruik dan het context menu om deze bestanden te verversen.

B.4. Draai Revisies Terug in het Archief (ongedaan maken)

B.4.1. Gebruik het logboek boodschappen scherm

De snelste methode om wijzigingen van één of een reeks van revisies ongedaan te maken, is door gebruik te maken van het logboek venster. Deze methode moet je ook gebruiken als je recentelijke wijzigingen wilt negeren en oudere revisie de nieuwe HEAD revisie wilt laten worden.

1. Selecteer het bestand of de map waarvoor je de wijzigingen ongedaan wilt maken. Als je alle wijzigingen ongedaan wilt maken, moet je de hoogste/basis map selecteren.
2. Selecteer TortoiseSVN → Toon logboek om de lijst met revisies weer te geven. Je moet misschien de Toon Alles of Volgende 100 knoppen gebruiken om de revisie(s) te zien, waarin je geïnteresseerd bent.
3. Selecteer de revisie die je ongedaan wilt maken. Als je een reeks van revisies ongedaan wilt maken, selecteer de eerste en houdt de **Shift** knop ingedrukt terwijl je de laatste selecteert. Merk op dat voor meerdere revisies de reeks niet onderbroken mag zijn met gaten. Klik met rechts op de geselecteerde revisie(s) en kies Context Menu → Revisie wijzigingen ongedaan maken.
4. Of als je een eerdere revisie de nieuwe HEAD revisie wilt maken, klik met rechts op de geselecteerde revisie en selecteer vervolgens Context Menu → Ongedaan maken tot deze revisie. Hiermee worden *alle* wijzigingen na de geselecteerde revisie ongedaan gemaakt.

Je hebt nu de wijzigingen in je werkkopie ongedaan gemaakt. Controleer het resultaat en leg de wijzigingen vast.

B.4.2. Gebruik het samenvoeg scherm

Voor ongedaan maken van wijzigingen in een groter bereik van revisies gebruik je het samenvoeg scherm. De vorige methode voegt achter de schermen dingen samen. Met deze methode wordt het expliciet gedaan.

1. Selecteer in je werkkopie TortoiseSVN → **Samenvoegen**.
2. Vul in het **Van**: veld de volledige URL van de map van de tak of label in, waarvoor je de wijzigingen ongedaan wilt maken in jouw werkkopie. Deze zal als standaard URL getoond moeten worden.
3. Vul in het **Van revisie** veld het huidige revisienummer van je werkkopie in. Als je er zeker van bent dat niemand anders wijzigingen aan het doorvoeren is, dan kun je de HEAD revisie gebruiken.
4. Zorg er voor dat het **Gebruik "Van:" URL** selectievak aangevinkt is.
5. Vul in het **Naar Revisie** veld het revisienummer in tot waar je de wijzigingen ongedaan wilt maken, degene *voor* de eerste revisie die ongedaan gemaakt moet worden.
6. Klik op **OK** om de samenvoeging af te ronden.

Je hebt nu de wijzigingen in je werkkopie ongedaan gemaakt. Controleer het resultaat en leg de wijzigingen vast.

B.4.3. Gebruik `svndumpfilter`

Omdat TortoiseSVN nooit gegevens zal verliezen, bestaan je “ongedaan gemaakt” revisies nog steeds als tussentijdse revisies in het archief. De HEAD revisie is alleen veranderd naar een eerdere status. Als je wilt dat de revisies volledig uit het archief verdwijnen, inclusief alle informatie over het bestaan ervan, dan moet je extreme maatregelen nemen. Tenzij er hiervoor echt een hele goede reden is, wordt dit *met klem afgeraden*. Een mogelijke reden is bijvoorbeeld dat iemand een vertrouwelijk document in een publiek archief heeft vastgelegd.

De enige manier waarop gegevens echt uit het archief verwijderd kunnen worden, is door het Subversion opdrachtprompt programma `svnadmin` te gebruiken. Je kunt een beschrijving van de werking vinden op *Repository Maintenance* [<http://svnbook.red-bean.com/en/1.5/svn.reposadmin.maint.html>].

B.5. Vergelijk Twee Revisies van een Bestand of Map

Als je de geschiedenis van twee revisies van een object wilt vergelijken, bijvoorbeeld voor revisie 100 en 200 van hetzelfde bestand, gebruik je TortoiseSVN → **Toon logboek** om de geschiedenis van dat bestand te bekijken. Selecteer de twee revisies die je wilt vergelijken en gebruik dan **Context Menu** → **Vergelijk Revisies**.

Als je hetzelfde object in twee verschillende mapstructuren wilt bekijken, bijvoorbeeld in de basislijn en een tak, gebruik dan de archief-verkenner om beide boomstructuren te openen. Selecteer de bestanden in beide locaties en gebruik **Context Menu** → **Vergelijk Revisies**.

Als je twee boomstructuren wilt vergelijken om na te gaan wat er gewijzigd is, bijvoorbeeld de basislijn en een gelabelde vrijgave, gebruik dan het TortoiseSVN → **Revisie schema**. Selecteer de twee te vergelijken knopen en kies **Context Menu** → **Vergelijk HEAD Revisies**. Er wordt dan een lijst met gewijzigde bestanden getoond waarin je individuele bestanden kunt selecteren, waarvoor je de wijzigingen in detail wilt zien. Je kunt ook de mapstructuur met alle gewijzigde bestanden exporteren, of simpelweg een lijst met gewijzigde bestanden. Lees **Paragraaf 4.10.3, “Vergelijken van mappen”** voor meer informatie. Als alternatief kun je ook **Context Menu** → **GNU diff van HEAD revisies** om een samenvatting met minimale context van de wijzigingen te bekijken.

B.6. Een veelgebruikt subproject gebruiken

Soms wil je een ander project in je werkkopie opnemen, bijvoorbeeld een bepaalde bibliotheek. Je wilt geen duplicaat maken van deze broncode in je archief, omdat je dan de verbinding met de originele (en

onderhouden) broncode verliest. Of je misschien heb je verschillende projecten die van dezelfde stukken broncode gebruik maken. Er zijn tenminste 3 methodes om hiermee om te gaan.

B.6.1. svn:externals Gebruiken

Stel de `svn:externals` eigenschap in op een map in je project. Deze eigenschap kan één of meer regels hebben; elke regel bevat de naam van een sub-map welke je wilt gebruiken als de doelmap voor gemeenschappelijke broncode en de URL van het archief waar vandaan je de gegevens op wilt halen. Lees voor meer details [Paragraaf 4.18, “Externe Objecten”](#).

Leg de nieuwe map vast in het archief. Zodra je nu ververst, zal Subversion een kopie van dat project maken vanuit het archief naar jouw werkkopie. Sub-mappen worden automatisch aangemaakt als dat nodig is. Elke keer dat je je werkkopie ververst, wordt ook de laatste wijzigingen van het externe project opgehaald.

Als het externe project zich in hetzelfde archief bevindt, dan worden de wijzigingen die je daarin aanbrengt meegenomen in de vastleglijst als je het hoofdproject vastlegt.

Als het externe project zich in een ander archief bevindt, dan zul je alleen een notificatie krijgen van de wijzigingen die je aan het externe project hebt doorgevoerd op het moment dat je het hoofd project vastlegt. Je moet de wijzigingen aan het externe project apart vastleggen.

Van de drie beschreven methodes, is dit de enige methode die geen instellingen vereist aan de kant van de gebruikers. Als de externen eenmaal toegekend zijn aan de map-eigenschappen, zullen alle gebruikers automatisch de betreffende externe gegevens krijgen als zij hun werkkopie verversen.

B.6.2. Een geneste werkkopie gebruiken

Maak een nieuwe map aan binnen je project, waarin de gemeenschappelijke broncode opgeslagen kan worden. Voeg deze map echter niet toe aan Subversion.

Selecteer TortoiseSVN → Ophalen voor de nieuwe map en haal een kopie van de gemeenschappelijke broncode op. Je hebt nu een aparte werkkopie genest in jouw hoofd werkkopie.

De twee werkkopieën zijn onafhankelijk. Als je wijzigingen vastlegt van de bovenliggende map, dan worden wijzigingen in de geneste werkkopie genegeerd. Ook als je de bovenliggende map ververst, wordt de geneste werkkopie niet ververst.

B.6.3. Gebruik een relatieve lokatie

Als je dezelfde gemeenschappelijke broncode voor verschillende projecten gebruikt en je niet meerdere werkkopieën wilt hebben voor elk project dat van die code gebruik maakt, dan kun je die broncode ook ophalen naar een aparte locatie die bekend is bij alle projecten die er gebruik van maken. Bijvoorbeeld:

```
C:\Projects\Proj1
C:\Projects\Proj2
C:\Projects\Proj3
C:\Projects\Common
```

en verwijst dan naar de gemeenschappelijke broncode door een relatief pad te gebruiken, bijv. `..\..\Common\DSPcore`.

Als je projecten over verschillende niet gerelateerde locaties verspreid zijn, dan kun je de volgende variant gebruiken. Je kunt je gemeenschappelijke broncode op een bepaalde locatie opslaan en door gebruik te maken van schijfletter vervanging de locatie een hard pad geven, welke je direct kunt opnemen in de code van je projecten. Haal bijvoorbeeld de gemeenschappelijke broncode op naar `D:\Documents\Framework of C:\Documents and Settings\{login}\My Documents\framework` en gebruik dan

```
SUBST X: "D:\Documents\framework"
```

om de toegewezen schijfletter te kunnen gebruiken in je broncode. Je kunt dan dus absolute locaties gebruiken in je code.

```
#include "X:\superio\superio.h"
```

Deze methode zal alleen een PC-omgeving en je moet nauwkeurig aangeven welke schijfletters er toegewezen moeten worden, zodat je team weer waar ze de mysterieuze bestanden kunnen vinden. Deze methode moet je alleen gebruiken in een gesloten ontwikkelomgeving en is niet aanbevolen voor generiek gebruik.

B.7. Maak een snelkoppeling naar een archief

Als je de archief-verkenner regelmatig op dezelfde locatie wilt openen, dan kun je een bureaublad snelkoppeling maken en de parameters direct doorgeven aan het TortoiseProc programma. Maak eenvoudigweg een nieuwe snelkoppeling aan en stel het doel in op:

```
TortoiseProc.exe /command:repobrowser /path:"url/to/repository"
```

Natuurlijk moet je wel de echte URL naar het archief gebruiken.

B.8. Negeer bestanden die al opgenomen zijn in het versiebeheer

Stel, je hebt per ongeluk enkele bestanden toegevoegd aan het archief, terwijl die eigenlijk genegeerd zouden moeten worden. Hoe krijg je die bestanden dan weer uit het versiebeheer systeem? Je hebt bijvoorbeeld je eigen IDE configuratie bestand die geen onderdeel is van het project, maar wat je wel veel tijd heeft gekost om alles naar jouw wens in te stellen.

Als je de toevoeging nog niet vastgelegd hebt, dan hoeft je alleen maar TortoiseSVN → Ongedaan maken te gebruiken om de toevoeging ongedaan te maken. Je moet dan het bestand (of de bestanden) aan de negerlijst toevoegen, zodat deze later niet alsnog per ongeluk worden toegevoegd aan het archief.

Als de bestanden toch al in het archief terecht zijn gekomen, moet je iets meer werk verrichten.

1. Houdt de **Shift** ingedrukt om in het uitgebreide context menu te komen. Gebruik dan TortoiseSVN → Verwijderen (lokaal behouden) om het bestand of de map te markeren als te verwijderen uit het archief zonder de lokale kopie te verliezen.
2. De bovenliggende map TortoiseSVN → Vastleggen.
3. Het bestand of de map aan de negerlijst toevoegen, zodat je niet nogmaals in de problemen belandt.

B.9. Versiebeheer van een werkkopie verwijderen

Als je een werkkopie hebt en die wilt herstellen naar een normale mapstructuur zonder de .svn mappen, dan kun je deze werkkopie eenvoudig naar dezelfde locatie exporteren. Lees [Paragraaf 4.26.1, "Versiebeheer van een werkkopie verwijderen"](#) voor meer informatie hierover.

B.10. Een werkkopie verwijderen

Hoe kom je op een nette manier van een werkkopie, waar je niets meer mee wilt doen, af? Simpel - gewoon verwijderen met je Windows Verkenner! Werkkopieën zijn eigen afzonderlijke lokale entiteiten.

Bijlage C. Handige tips voor beheerders

Deze bijlage bevat oplossingen voor problemen/vragen die je zou kunnen hebben wanneer je verantwoordelijk bent voor het verspreiden van TortoiseSVN over meerdere client computers.

C.1. Verspreid TotoiseSVN via groepsbeleid

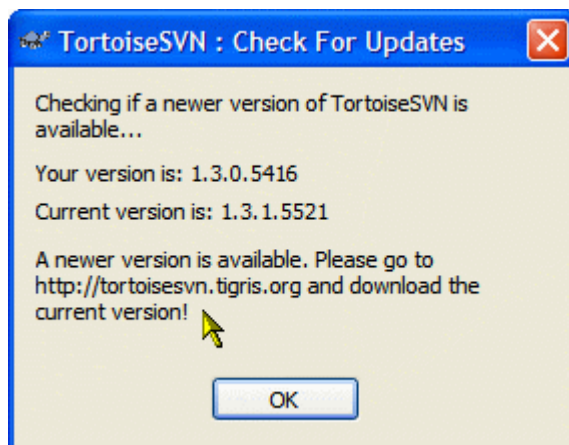
De TortoiseSVN installatie wordt geleverd als een MSI-bestand, wat inhoudt dat je geen problemen zou moeten ondervinden bij het toevoegen van dat MSI-bestand aan het groepsbeleid van je domeincontroller.

Een goede handleiding over hoe dat te doen kan gevonden worden in kennisbank artikel 314934 van Microsoft: <http://support.microsoft.com/?kbid=314934>.

Versies 1.3.0 en hoger van TortoiseSVN moeten geïnstalleerd worden onder *Computerconfiguratie* en niet onder *Gebruikersconfiguratie*. Dit komt omdat die versies de nieuwe CRT en MFC DLL's nodig hebben, welke alleen kunnen worden verspreid *per computer* en niet *per gebruiker*. Als je TortoiseSVN echt moet installeren per gebruiker, dan zul je eerst versie 8 van de MFC- en CRT-packages van Microsoft moeten installeren op elke computer waar je TortoiseSVN per gebruiker wilt installeren

C.2. De update controle omleiden

TortoiseSVN controleert om de paar dagen of er een nieuwe versie beschikbaar is. Is er een nieuwe versie beschikbaar, dan wordt een dialoog getoond welke de gebruiker hierover informeert.



Figuur C.1. De upgrade-dialoog

Wanneer je verantwoordelijk bent voor een grote hoeveelheid gebruikers in je domein, kan het zijn dat je wilt dat je gebruikers alleen door jou goedgekeurde versies gebruiken en dat ze niet altijd de nieuwste versies installeren. Je wil dan waarschijnlijk niet dat het update dialoogvenster getoond wordt, zodat je gebruikers gelijk de update gebruiken.

Versies 1.4.0 en hoger van TortoiseSVN staan toe dat je de update controle omleid naar je intranet server. Je kan de register sleutel `HKCU\Software\TortoiseSVN\UpdateCheckURL` (tekst waarde) laten wijzen naar een URL welke verwijst naar een tekstbestand op je intranet. Dat tekstbestand moet het volgende formaat hebben:

1.4.1.6000

Een nieuwere versie van TortoiseSVN is beschikbaar voor je om te downloaden!
<http://192.168.2.1/downloads/TortoiseSVN-1.4.1.6000-svn-1.4.0.msi>

De eerste regel in dat bestand is de versie tekst. Je moet ervoor zorgen dat deze exact overeenkomt met de versie tekst van het TortoiseSVN installatie pakket. De tweede regel is een eigen tekst welke getoond wordt in het upgrade dialoogvenster. Je kan hierin zetten wat je wilt, maar let er op dat de ruimte in het upgrade dialoog venster beperkt is. Te lange berichten worden afgekapt! De derde regel is de URL waar het nieuwe installatie pakket te vinden is. Deze URL wordt geopend wanneer de gebruiker klikt op het label met de eigen tekst in het upgrade dialoog venster. Je kan de gebruiker ook eenvoudig naar een web pagina sturen in plaats van direct naar het MSI bestand. De URL wordt geopend in de standaard web browser, dus als je een web pagina specificeert, wordt deze geopend en getoond aan de gebruiker. Als je het MSI bestand specificeert, zal de browser de gebruiker vragen om het MSI bestand lokaal op te slaan.

C.3. Zetten van de SVN_ASP_DOT_NET_HACK omgevingsvariable

Vanaf versie 1.4.0 en hoger, verschaft de TortoiseSVN installatie niet meer de optie aan de gebruiker om de SVN_ASP_DOT_NET_HACK omgevingsvariabele te zetten, omdat deze vele problemen opleverde en verwarring zaaide bij gebruikers die altijd *alles* installeren ongeacht of ze weten waar het voor is.

Maar die optie is alleen verborgen voor de gebruiker. Je kan de TortoiseSVN installatie nog steeds forceren de omgevingsvariabele te zetten door de eigenschap ASPDOTNETHACK op TRUE te zetten. Je kan de installatie bijvoorbeeld starten op de volgende manier:

```
msiexec /i TortoiseSVN-1.4.0.msi ASPDOTNETHACK=TRUE
```

C.4. Opties in het contextmenu uitschakelen

Vanaf versie 1.5.0 en hoger, staat TortoiseSvn je toe om opties van het context menu uit te schakelen (eigenlijk verbergen). Aangezien dit een eigenschap is dat niet lichtvaardig zou moeten worden gebruikt, maar alleen als er een dwingende reden voor is, is er geen GUI voor aanwezig en moet dit rechtstreeks in het register worden gedaan. Dit kan gebruikt worden om bepaalde commando's uit te schakelen voor gebruikers die deze niet zouden moeten gebruiken. Maar let er wel op dat alleen de context menu opties in de *Verkenner* zijn verborgen. De commando's zijn nog steeds beschikbaar via andere middelen, zoals bijvoorbeeld de opdrachtregel of zelfs andere dialogen in TortoiseSVN zelf!

De registersleutels die de informatie bevatten over welke contextmenus getoond moeten worden zijn HKEY_CURRENT_USER\Software\TortoiseSVN\ContextMenuEntriesMaskLow en HKEY_CURRENT_USER\Software\TortoiseSVN\ContextMenuEntriesMaskHigh.

Elk van deze registersleutels is een DWORD waarde, waarvan elke bit correspondeert met een specifieke menuoptie. Een bit dat gezet is, betekent dat de bijbehorende menuoptie is uitgeschakeld.

Waarde	Menuoptie
0x0000000000000001	Ophalen
0x0000000000000002	Verversen
0x0000000000000004	Vastleggen
0x0000000000000008	Toevoegen
0x0000000000000010	Ongedaan maken
0x0000000000000020	Schonen
0x0000000000000040	Oplossen
0x0000000000000080	Optie
0x0000000000000100	Importereren

Waarde	Menuoptie
0x0000000000000200	Exporteren
0x0000000000000400	Maak hier archief aan
0x0000000000000800	Tak/Label
0x0000000000001000	Samenvoegen
0x0000000000002000	Verwijderen
0x0000000000004000	Hernoemen
0x0000000000008000	Verversen naar revisie
0x0000000000010000	Vershil
0x0000000000020000	Toon logboek
0x0000000000040000	Bewerk conflicten
0x0000000000080000	Locatie wijzigen
0x0000000000100000	Controleer op wijzigingen
0x0000000000200000	Negeer
0x0000000000400000	Archief-verkenner
0x0000000000800000	Verklaren
0x0000000001000000	Maak patch
0x0000000002000000	Patch toepassen
0x0000000004000000	Revisie schema
0x0000000008000000	Blokkade
0x0000000010000000	Blokkade opheffen
0x0000000020000000	Eigenschappen
0x0000000040000000	Vershil met URL
0x0000000080000000	Verwijder niet-geversioneerde items
0x2000000000000000	Instellingen
0x4000000000000000	Help
0x8000000000000000	Over

Tabel C.1. Menuopties en hun waarden

Voorbeeld: om de “Verhuizen”, de “Verwijder objecten zonder versiebeheer” en de “Instellingen” menuopties uit te schakelen, voeg de waarden behorende bij deze opties op de volgende manier toe:

```

0x0000000000008000
+ 0x0000000080000000
+ 0x2000000000000000
= 0x2000000080080000

```

De lagere DWORD waarde (0x80080000) moet dan opgeslagen worden in HKEY_CURRENT_USER\Software\TortoiseSVN\ContextMenuEntriesMaskLow, de hogere DWORD waarde (0x20000000) in HKEY_CURRENT_USER\Software\TortoiseSVN\ContextMenuEntriesMaskHigh.

Voor het opnieuw inschakelen van de menuopties, kun je eenvoudig deze twee registersleutels verwijderen.

Bijlage D. Automatisering van TortoiseSVN

Omdat alle commando's voor TortoiseSVN worden bestuurd via commandoregel parameters kunt u dit automatiseren door middel van batch scripts of specifieke commando's of schermen van andere programma's opstarten. (bijvoorbeeld uw favoriete tekstverwerker).



Belangrijk

Onthoudt dat TortoiseSVN een GUI client is en deze automatiserings handleiding aangeeft hoe de TortoiseSVN schermen te tonen welke gebruikersinvoer verzamelen. Als u een script wilt schrijven dat geen enkele invoer van de gebruiker vereist, dient u de officiële Subversion commando-regel client gebruiken.

D.1. TortoiseSVN commando's

Het TortoiseSVN GUI programma wordt `TortoiseProc.exe` genoemd. Alle commando's worden opgegeven met de parameter `/command:abcd` waarbij `abcd` de naam is van het vereiste commando. De meeste van deze commando's hebben ten minste één pad-argument nodig, welke wordt gegeven met `/path:"some\path"`. In de volgende tabel verwijst het commando naar de `/command:abcd` parameter en het pad verwijst naar de `/path:"some\path"` parameter.

Omdat sommige commando's een lijst van doelpaden kunnen aannemen (bijv. een aantal specifieke bestanden vastleggen), kan de `/path` parameter meerdere paden, gescheiden door een `*` teken, bevatten.

TortoiseSVN gebruikt tijdelijke bestanden om meerdere argumenten uit te wisselen tussen de shell extensie en het hoofdprogramma. Vanaf TortoiseSVN 1.5.0 en hoger is de `/notempfile` parameter overbodig en hoeft deze niet meer te worden toegevoegd.

Het voortgangsvenster, dat wordt gebruikt voor vastleggingen, bijwerkingen en vele andere commando's, blijft gewoonlijk geopend nadat het commando is beëindigd totdat de gebruiker op de OK knop klikt. Dit kan worden gewijzigd door de corresponderende optie in het instellingenvenster aan te vinken. Maar als die instelling wordt gebruikt, zal het voortgangsvenster sluiten hoe het commando ook vanuit het batchbestand of vanuit de TortoiseSVN contextmenu wordt opgestart.

Om een andere locatie van het configuratiebestand aan te geven gebruikt u de parameter `/configdir:"path\to\config\directory"`. Dit zal het standaardpad, alsook elke registry instelling overschrijven.

Om het voortgangsscherm aan het eind van een commando automatisch te sluiten zonder gebruik te maken van de permanente instelling, kun je de parameter `/closeonend` meegeven.

- `/closeonend:0` het venster niet automatisch sluiten
- `/closeonend:1` automatisch sluiten indien er geen fouten zijn
- `/closeonend:2` automatisch sluiten indien er geen fouten of conflicten zijn
- `/closeonend:3` automatisch sluiten indien er geen fouten, conflicten of samenvoegingen zijn
- Om het voortgangsscherm aan het einde van een commando automatisch te sluiten zonder de permanente instelling te gebruiken, kunt u de `/closeonend` parameter opgeven.
`<placeholder-1></placeholder-1>`
Om het voortgangsvenster voor locale operaties te sluiten als er geen fouten of conflicten zijn, geeft u de `/closeforlocal` parameter op.

De onderstaande tabel toont alle commando's welke kunnen worden benaderd met de TortoiseProc.exe commando regel. Zoals hierboven is beschreven, dienen deze in de vorm `/command:abcd` te worden gebruikt. In de tabel wordt het voorvoegsel `/command` weggelaten om ruimte te sparen.

Commando	Beschrijving
:about	Toont het 'over' scherm. Dit wordt ook getoond als geen enkel commando is opgegeven.
:log	Opent het log venster. Het <code>/path</code> geeft het bestand of map aan waarvan de log moet worden getoond. Er kunnen zes extra opties worden ingesteld: <code>/startrev:xxx</code> , <code>/endrev:xxx</code> , <code>/strict</code> maakt de selectievak 'stoppen-bij-kopie' benaderbaar, <code><placeholder-5></placeholder-5></code> vult de filtertekst, <code><placeholder-6></placeholder-6></code> dwingt het filter om tekst in plaats van regex te gebruiken, of <code><placeholder-7></placeholder-7></code> dwingt het filter om een regex in plaats van een simpele tekst te gebruiken om te zoeken, en <code><placeholder-8></placeholder-8></code> waarbij X een getal is tussen 1 en 7. Corresponderen de nummers met: <code><placeholder-9></placeholder-9></code>
:checkout	Opent het ophaal venster. De <code>/path</code> geeft de doelmap aan en de <code>/url</code> geeft de URL aan waarvan wordt opgehaald.
:import	Opent het import venster. De <code>/path</code> geeft de map aan met de gegevens om te importeren.
:update	Werkt de werkkopie in <code>/path</code> bij naar HEAD. Als de optie <code>/rev</code> is gegeven wordt een venster getoond dat de gebruiker vraagt naar welke revisie moet worden bijgewerkt. Als u dit venster niet wilt zien, geeft dan een revisienummer op <code>/rev:1234</code> . Andere opties zijn <code>/nonrecursive</code> , <code>/ignoreexternals</code> en <code><placeholder-6></placeholder-6></code> .
:commit	Opent het vastleggen venster. De <code>/path</code> geeft de doelmap of de lijst met bestanden op om vast te leggen. U kunt ook de <code>/logmsg</code> optie aangeven om een voorgedefiniëerd logbericht aan het vastleggen venster door te geven. Of, als u geen logbericht op de commandoregel wilt doorgeven gebruikt u dan <code>/logmsgfile:path</code> , waarbij <code>pad</code> een bestand aangeeft dat het logbericht bevat. Om het bug ID veld van tevoren te vullen (voor het geval dat u integratie met bug trackers naar behoren heeft ingesteld) kunt u de <code>/bugid:"het id van de bug hier"</code> gebruiken.
:add	Voegt de bestanden in <code>/path</code> toe aan versiebeheer.
:revert	Draait alle locale aanpassingen aan de werkkopie terug. De <code>/path</code> geeft aan welke items terug te draaien.
:cleanup	Schoont onderbroken of afgebroken activiteiten op en heft de blokkade van de werkkopie op in <code>/path</code> . Gebruik <code><placeholder-2></placeholder-2></code> om te voorkomen dat het resultaten scherm wordt getoond (of het nou gaat over de beëindiging van de opschoning of het tonen van een foutmelding).
:resolve	Markeert een conflicterend bestand dat wordt aangegeven in <code>/path</code> als opgelost. Als <code>/noquestion</code> is ingesteld, dan wordt het probleem opgelost zonder de gebruiker van tevoren te raadplegen of het daadwerkelijk moet worden uitgevoerd.
:repopcreate	Maakt een archief aan in <code>/path</code>

Commando	Beschrijving
:switch	Opent het wissel venster. De /path geeft de doelmap aan.
:export	Exporteert de werkkopie vanuit /path naar een andere map. Als de /path naar een map wijst welke niet onder versiebeheer is, zal een venster vragen naar een URL om naar de map in /path te exporteren.
:merge	Opent het samenvoeg venster. De /path geeft de doelmap aan. Om een revisie bereik samen te voegen zijn de volgende opties beschikbaar: /fromurl:URL, /revrange:string. Om twee archiefmap boomstructuren samen te voegen zijn de volgende opties beschikbaar: /fromurl:URL, /tourl:URL, /fromrev:xxx en /torev:xxx. Deze voeren de relevante velden in het samenvoeg venster van tevoren in.
:mergeall	Opent het samenvoeg scherm. De /path geeft de doeldirectory aan.
:copy	Toont het tak/label scherm. De /path is de werkkopie waar de tak/label van kopieëerd moet worden en de /url is de doel URL. U kunt ook de /logmsg opgeven om een vooraf ingesteld logbericht aan de tak/label door te geven. Of, als u geen logbericht via de commandoregel wilt doorgeven, /logmsgfile:path gebruiken waarbij pad een bestand aangeeft dat het logbericht bevat.
:settings	Opent het instellingen scherm.
:remove	Verwijdert één of meerdere bestanden van versiebeheer.
:rename	Hernoemt het bestand in /path. De nieuwe naam voor het bestand wordt in een venster gevraagd. Om de vraag om gelijksoortige bestanden in dezelfde handeling te hernoemen te voorkomen, geeft u /noquestion op.
:diff	Start het externe vergelijkingsprogramma dat in de TortoiseSVN instellingen is opgegeven. De /path geeft het eerste bestand aan. Als de optie /path2 is opgegeven, wordt het vergelijkingsprogramma met die twee bestanden opgestart. Als /path2 is weggelaten wordt de vergelijking uitgevoerd tussen het bestand gegeven in /path en zijn BASE. Om de revisie nummers expliciet op te geven gebruikt u /startrev:xxx en /endrev:xxx. Als /blame is opgegeven en /path2 niet is opgegeven wordt de vergelijking uitgevoerd nadat de bestanden eerst zijn verklaard met de gegeven revisie nummers. De parameter <code><placeholder-9></placeholder-9></code> geeft de regel aan waarnaar wordt gesprongen nadat de vergelijking is getoond.
:showcompare	Afhankelijk van de URL's en de revisie nummers welke moeten worden vergeleken, toont dit of een unified diff (als de optie unified is ingesteld), of een lijst met bestanden welke zijn gewijzigd, of start de vergelijkingskijker als de URL's verwijzen naar twee bestanden. De opties url1, url2, revision1 en revision2 dienen te zijn opgegeven. De opties pegrevision, ignoreancestry, verklaren en unified zijn optioneel.
:conflicteditor	Start de conflict bewerker, welke is ingesteld in de TortoiseSVN instellingen, met de correcte bestanden voor het conflicterende bestand in /path.
:relocate	Opent het verhuis scherm. De /path geeft de werkkopie aan om te verhuizen.
:help	Opent het help bestand.
:repostatus	Opent het scherm voor controle op wijzigingen. De /path geeft de map van de werkkopie aan. Als <code><placeholder-2></placeholder-2></code>

Commando	Beschrijving
	is opgegeven maakt het scherm bij opstarten direct contact met het archief alsof de gebruiker op de <code><placeholder-3></placeholder-3></code> knop heeft geklikt.
<code>:repobrowser</code>	Start het archief-verkenner scherm welke verwijst naar de URL van de werkkopie die is opgegeven in <code>/path</code> of <code>/path</code> verwijst direct naar een URL. Een extra optie <code>/rev:xxx</code> kan worden gebruikt om de revisie aan te geven welke de archief-verkenner moet tonen. Als de <code>/rev:xxx</code> wordt weggelaten wordt teruggevallen op HEAD. Als <code>/path</code> naar een URL wijst, geeft <code>/projectpropertiespath:path/to/wc</code> het pad aan waarvan de project eigenschappen moeten worden gelezen en gebruikt.
<code>:ignore</code>	Voegt alle doelen in <code>/path</code> toe aan de negeren lijst. Oftewel, voegt de <code>svn:ignore</code> eigenschap toe aan die bestanden.
<code>:blame</code>	Opent het verklaar scherm voor het bestand dat wordt gegeven in <code>/path</code> . Als de opties <code>/startrev</code> en <code>/endrev</code> zijn ingesteld wordt niet het scherm dat het te verklaren bereik vraagt getoond, maar worden de waarden van deze opties in de revisie gebruikt. Als de optie <code>/line:nnn</code> is ingesteld zal TortoiseBlame worden geopend met het aangegeven regelnummer in beeld. De opties <code>/ignoreeol</code> , <code>/ignorespaces</code> en <code>/ignoreallspaces</code> worden ook ondersteund.
<code>:cat</code>	Slaat een bestand op vanuit een URL of werkkopiepad gegeven in <code>/path</code> naar de locatie gegeven in <code>/savepath:path</code> . De revisie wordt gegeven in <code>/revision:xxx</code> . Dit kan worden gebruikt om een bestand met van een bepaalde revisie op te halen.
<code>:createpatch</code>	Maakt een patch bestand aan voor het in <code>/path</code> gegeven pad.
<code>:revisiongraph</code>	Toont het revisie schema voor het in <code>/path</code> gegeven pad.
<code>:lock</code>	Blokkeert één of meerdere bestanden in de in <code>/path</code> gegeven map. Het 'blokkeer' venster wordt getoond zodat de gebruiker commentaar op de blokkade kan ingeven.
<code>:unlock</code>	Verwijdert de blokkering van één of meerdere bestanden in de in <code>/path</code> gegeven map.
<code>:rebuildiconcache</code>	Bouwt de windows pictogrammen cache opnieuw op. Gebruik dit alleen voor het geval dat de windows pictogrammen corrupt zijn geraakt. Een bijwerking hiervan is (wat niet kan worden voorkomen) dat de pictogrammen op het werkblad worden herschikt. Geef <code>/noquestion</code> mee om het meldingsscherm te onderdrukken.
<code>:properties</code>	Toont het eigenschappen scherm voor het pad dat is gegeven in <code>/path</code> .

Tabel D.1. Lijst van alle beschikbare commando's en opties

Voorbeelden (welke op één regel dienen te worden ingegeven):

```
TortoiseProc.exe /command:commit
                 /path:"c:\svn_wc\file1.txt*c:\svn_wc\file2.txt"
                 /logmsg:"test logbericht" /closeonend:0
```

```
TortoiseProc.exe /command:update /path:"c:\svn_wc\" /closeonend:0
```

```
TortoiseProc.exe /command:log /path:"c:\svn_wc\file1.txt"
                /startrev:50 /endrev:60 /closeonend:0
```

D.2. TortoiseIDiff Commando's

Het vergelijkingshulpprogramma voor afbeeldingen heeft een aantal commando-regel opties welke kunnen worden gebruikt om te bepalen hoe het hulpprogramma wordt opgestart. Het programma wordt TortoiseIDiff.exe genoemd.

De onderstaande tabel toont alle opties welke op de commando-regel kunnen worden doorgegeven aan het vergelijkingshulpprogramma voor afbeeldingen.

Optie	Beschrijving
:left	Pad naar het bestand dat links wordt getoond.
:lefttitle	Een titel tekst. Deze tekst wordt gebruikt in de titel van de afbeeldingsweergave in plaats van het volledige pad naar het afbeeldingsbestand.
:right	Pad naar het bestand dat rechts wordt getoond.
:righttitle	Een titel tekst. Deze tekst wordt gebruikt in de titel van de afbeeldingsweergave in plaats van het volledige pad naar het afbeeldingsbestand.
:overlay	Indien opgegeven schakelt het vergelijkingshulpprogramma voor afbeeldingen over naar de overlap modus (alfa mengen).
:fit	Indien opgegeven maakt het vergelijkingshulpprogramma voor afbeeldingen beide afbeeldingen aan elkaar gelijk qua grootte.
:showinfo	Toont het informatieblok voor de afbeelding.

Tabel D.2. Lijst van beschikbare opties.

Voorbeeld (welke op één regel dient te worden ingegeven):

```
TortoiseIDiff.exe /left:"c:\images\img1.jpg" /lefttitle:"image 1"
                 /right:"c:\images\img2.jpg" /righttitle:"image 2"
                 /fit /overlay
```

Bijlage E. Opdrachtprompt Interface Cross Reference

Deze handleiding verwijst soms naar de Subversion documentatie, die de Subversion opdrachtprompt interface (Command Line Interface, CLI) beschrijft. Om een indruk te geven van wat TortoiseSVN op de achtergrond doet, hebben we een lijst gemaakt van de equivalente CLI commando's voor elke TortoiseSVN GUI actie.

Opmerking

Onthoudt dat, ondanks dat er CLI equivalenten zijn voor wat TortoiseSVN doet, TortoiseSVN de bibliotheken direct benadert en *niet* de CLI commando's gebruikt.

Als je denkt dat je een bug in TortoiseSVN hebt gevonden, kan het zijn dat we je vragen om het probleem te reproduceren met CLI commando's. We kunnen dan makkelijk een onderscheid maken tussen TortoiseSVN en Subversion problemen. Dit overzicht verteld je welke commando's je moet gebruiken.

E.1. Conventies en Basis Regels

In de volgende beschrijvingen wordt de URL van een archief eenvoudig weergegeven als URL, met als voorbeeld `http://tortoisesvn.googlecode.com/svn/trunk`. Het pad van de werkkopie wordt aangegeven als PATH, met als voorbeeld `C:\TortoiseSVN\trunk`.



Belangrijk

Omdat TortoiseSVN een Windows Shell uitbreiding is, heeft het geen idee van de huidige werkkopie map. Alle paden van werkkopieën moeten als absolute paden opgegeven worden, niet als relatieve paden.

Bepaalde onderdelen zijn optioneel. Deze zijn dan ook meestal als selectievak en radiobutton in TortoiseSVN beschikbaar. Deze opties worden weergegeven tussen [blokhaken] in de definitie van de opdrachtregel.

E.2. TortoiseSVN commando's

E.2.1. Ophalen

```
svn checkout [-N] [--ignore-externals] [-r rev] URL PATH
```

Als Alleen de bovenste map ophalen is aangevinkt, gebruik dan de `-N` optie.

Als Externen uitsluiten aangevinkt is, gebruik dan de `--ignore-externals` optie.

Als je een specifieke revisie ophaalt, specificeer deze dan na de URL met de `-r` optie.

E.2.2. Verversen

```
svn info URL_of_WC
svn update [-r rev] PATH
```

Het verversen van meerdere objecten is op dit moment niet een enkele actie in Subversion. TortoiseSVN zoekt eerst de HEAD revisie van het archief en ververst dan alle objecten naar die specifieke revisie om te voorkomen dat er een werkkopie ontstaat met verschillende revisies.

Als maar één object geselecteerd is voor het verversen of de geselecteerde objecten zijn niet allemaal uit hetzelfde archief, dan zal TortoiseSVN gewoon naar de HEAD revisie verversen.

Er worden geen opdrachtregel opties gebruikt hierbij. Verversen naar revisie gebruikt ook het verversen commando, maar biedt meer opties.

E.2.3. Verversen naar Revisie

```
svn info URL_of_WC
svn update [-r rev] [-N] [--ignore-externals] PATH
```

Als Alleen de bovenste map verversen is aangevinkt, gebruik dan de `-N` optie.

Als Externen uitsluiten aangevinkt is, gebruik dan de `--ignore-externals` optie.

E.2.4. Vastleggen

Het vastleg venster van TortoiseSVN gebruikt meerdere Subversion commando's. In de eerste fase wordt een status controle uitgevoerd om te bepalen welke objecten in je werkkopie mogelijk vastgelegd moeten worden. Je kunt de lijst bekijken, bestanden vergelijken ten opzichte van de BASE en de objecten selecteren die je mee wilt nemen bij het vastleggen.

```
svn status -v PATH
```

Als Toon bestanden zonder versiebeheer aangevinkt is, zal TortoiseSVN alle bestanden en mappen zonder versiebeheer in je werkkopie laten zien, rekening houdend met de neger regels. Deze functie heeft geen directe equivalent in Subversion, omdat het `svn status` commando niet afdaalt in mappen zonder versiebeheer.

Als je bestanden en mappen zonder versiebeheer controleert, dan zullen die objecten eerst aan je werkkopie worden toegevoegd.

```
svn add PATH...
```

Als je op OK klikt, zal de Subversion vastlegging plaatsvinden. Als je alle bestandsselectievakken in de standaard stand hebt laten staan, zal TortoiseSVN een enkele recursieve vastlegging van je werkkopie uitvoeren. Als je enkele bestanden de-selecteert, dan moet een niet-recursieve vastlegging (`-N`) gebruikt worden en moet elke pad apart gespecificeerd worden in de vastlegging opdrachtregel.

```
svn commit -m "LogMessage" [-N] [--no-unlock] PATH...
```

`LogMessage` vertegenwoordigt hier de inhoud van het logboek boodschap invoerveld. Deze kan leeg gelaten worden.

Als Blokkade handhaven aangevinkt is, gebruik dan de `--no-unlock` optie.

E.2.5. Verschil

```
svn diff PATH
```

Als je Verschil gebruikt vanaf het hoofd context menu, dan ben je een gewijzigd bestand aan het vergelijken met de BASE revisie. De uitkomst van bovenstaande CLI commando voert ook deze actie uit en levert het resultaat op in een unified-diff formaat. Dit is echter niet wat TortoiseSVN gebruikt. TortoiseSVN gebruikt TortoiseMerge (of een ander vergelijk programma waarvoor jij gekozen hebt) voor

het visueel weergeven van verschillen tussen volledig op tekst gebaseerde bestanden. Er is hier dus geen CLI equivalent voor.

Je kunt ook 2 willekeurige bestanden vergelijken met TortoiseSVN, onafhankelijk of deze versiebeheer hebben. TortoiseSVN geeft de twee bestanden door aan het gekozen vergelijkingsprogramma en laat dat programma de verschillen uitzoeken.

E.2.6. Toon logboek

```
svn log -v -r 0:N --limit 100 [--stop-on-copy] PATH
  or
svn log -v -r M:N [--stop-on-copy] PATH
```

Standaard beperkt TortoiseSVN het aantal logboek boodschappen tot 100 met de `--limit` methode. Als de instellingen aangeven om oude API's te gebruiken, dan wordt de tweede indeling gebruikt voor het ophalen van de logboek boodschappen voor 100 archief revisies.

Als Stoppen bij kopie/hernoemd bestand is aangevinkt, gebruik dan de `--stop-on-copy` optie.

E.2.7. Controleer op Wijzigingen

```
svn status -v PATH
  or
svn status -u -v PATH
```

De initiële controle van de statussen kijkt alleen naar je werkkopie. Als je op **Controleer archief** klikt, dan wordt er ook in het archief gekeken welke bestanden er veranderen zullen bij het verversen. Hiervoor wordt de `-u` optie gebruikt.

Als **Toon bestanden zonder versiebeheer** aangevinkt is, zal TortoiseSVN alle bestanden en mappen zonder versiebeheer in je werkkopie laten zien, rekening houdend met de neger regels. Deze functie heeft geen directe equivalent in Subversion, omdat het `svn status` commando niet afdaald in mappen zonder versiebeheer.

E.2.8. Revisie schema

Het revisie schema is een functie van TortoiseSVN. Er is een CLI equivalent.

TortoiseSVN voert hierbij

```
svn info URL_of_WC
svn log -v URL
```

uit en analyseert het resultaat. Hierbij is de URL de *basis map* van het archief.

E.2.9. Archief Verkenner

```
svn info URL_of_WC
svn list [-r rev] -v URL
```

Je kunt `svn info` gebruiken om de basis map van het archief te vinden, dit is de bovenste map die in de archief-verkenner wordt weergegeven. Je kunt niet Hoger dan deze map komen in de mapstructuur. Dit commando geeft ook alle blokkade informatie weer die in de archief-verkenner wordt weergegeven.

De `svn list` opdracht zal een lijst met de inhoud van een map opleveren, van de ingegeven URL en revisie.

E.2.10. Bewerk conflicten

Dit commando heeft geen CLI equivalent. Het roept TortoiseMerge of een ander extern 3-weg vergelijkingsprogramma om de naar de betrokken bestanden te kunnen kijken en te bepalen welke regels er gebruikt moeten worden.

E.2.11. Opgelost

```
svn resolved PATH
```

E.2.12. Hernoemen

```
svn rename CURR_PATH NEW_PATH
```

E.2.13. Verwijderen

```
svn delete PATH
```

E.2.14. Ongedaan maken

```
svn status -v PATH
```

De eerste fase is een status controle, welke de objecten zoekt die mogelijk ongedaan gemaakt kunnen worden. Je kunt de lijst bekijken, bestanden vergelijken ten opzichte van de BASE en de objecten selecteren die je ongedaan wilt maken.

Als je op OK klikt, zal Subversion de actie uitvoeren. Als je alle bestandsselectievakken in de standaard stand hebt laten staan, zal TortoiseSVN een enkele recursieve (-R) ongedaanmaking van je werkkopie uitvoeren. Als je enkele bestanden de-selecteert, dan moet elk pad apart gespecificeerd worden in de opdrachtregel.

```
svn revert [-R] PATH...
```

E.2.15. Schonen

```
svn cleanup PATH
```

E.2.16. Plaats Blokkade

```
svn status -v PATH
```

De eerste fase is een status controle, welke de objecten zoekt in je werkkopie die mogelijk geblokkeerd kunnen worden. Je kunt de objecten die je wilt blokkeren selecteren.

```
svn lock -m "LockMessage" [--force] PATH...
```

Hierbij is LockMessage het bericht in het invoerveld. Dit kan leeg gelaten worden.

Als Steel de blokkades is aangevinkt, gebruik dan de --force optie.

E.2.17. Blokkade Opheffen

```
svn unlock PATH
```

E.2.18. Tak/Label

```
svn copy -m "LogMessage" URL URL
  or
svn copy -m "LogMessage" URL@rev URL@rev
  or
svn copy -m "LogMessage" PATH URL
```

Het Tak/Label scherm voert een kopieer actie uit op het archief. Er zijn drie radioknop opties:

- HEAD revisie in het archief
- Specifieke revisie in het archief
- Werkkopie

welke overeenkomen met de 3 eerder genoemde commando varianten.

LogMessage vertegenwoordigt hier de inhoud van het logboek boodschap invoerveld. Deze kan leeg gelaten worden.

E.2.19. Optie

```
svn info URL_of_WC
svn switch [-r rev] URL PATH
```

E.2.20. Samenvoegen

```
svn merge [--dry-run] --force From_URL@revN To_URL@revM PATH
```

De Test samenvoegen knop voert dezelfde samenvoeging uit, maar dan met de `--dry-run` optie.

```
svn diff From_URL@revN To_URL@revM
```

De Voer Unified Diff knop toont de verschil actie, welke gebruikt wordt om de samenvoeging uit te voeren.

E.2.21. Exporteren

```
svn export [-r rev] [--ignore-externals] URL Export_PATH
```

Deze formattering wordt gebruikt als de actie op een map zonder versiebeheer wordt opgestart en de map dus als doel map gebruikt wordt.

Het exporteren van een werkkopie naar een andere locatie wordt zonder de Subversion bibliotheek te gebruiken uitgevoerd. Er is dus geen CLI equivalent.

TortoiseSVN kopieert alle bestanden naar de nieuwe locatie terwijl de voortgang wordt weergegeven. Bestanden/mappen zonder versiebeheer kunnen optioneel ook geëxporteerd worden.

Als in één van beide gevallen de Externen uitsluiten aangevinkt is, gebruik dan de `--ignore-externals` optie.

E.2.22. Locatie wijzigen

```
svn switch --relocate From_URL To_URL
```

E.2.23. Maak Archief Hier

```
svnadmin create --fs-type fsfs PATH
```

E.2.24. Toevoegen

```
svn add PATH...
```

Als je een map geselecteerd hebt, zal TortoiseSVN deze eerst recursief controleren op objecten die toegevoegd kunnen worden.

E.2.25. Importeren

```
svn import -m LogMessage PATH URL
```

LogMessage vertegenwoordigt hier de inhoud van het logboek boodschap invoerveld. Deze kan leeg gelaten worden.

E.2.26. Verklaren

```
svn blame -r N:M -v PATH  
svn log -r N:M PATH
```

Als je TortoiseBlame gebruikt om de Verklaren informatie te bekijken, dan is het bestand logboek ook nodig om de logboek boodschappen in een tootip te laten zien. Als je de Verklaren informatie als een tekstbestand bekijkt, dan is deze informatie niet nodig.

E.2.27. Voeg aan Negeerlijst toe

```
svn propget svn:ignore PATH > tempfile  
{edit new ignore item into tempfile}  
svn propset svn:ignore -F tempfile PATH
```

Omdat de `svn:ignore` eigenschap vaak meerdere regels heeft, wordt het hier weergegeven alsof dit gewijzigd is met een tekstbestand in plaats van direct mee te geven in de opdrachtregel.

E.2.28. Maak patch

```
svn diff PATH > patch-file
```

TortoiseSVN maakt een patch bestand aan in unified diff indeling door de werkkopie met de BASE revisie te vergelijken.

E.2.29. Patch toepassen

Het toepassen van patches is een riskante activiteit, tenzij de patch en de werkkopie van dezelfde revisie zijn. Gelukkig kun je TortoiseMerge gebruiken, waarvoor geen direct equivalent beschikbaar is in Subversion.

Bijlage F. Implementatie details

Deze bijlage bevat een gedetailleerde verhandeling over de implementatie van enkele mogelijkheden van TortoiseSVN.

F.1. Overlappende Pictogrammen

Ieder bestand en map heeft een Subversion statuswaarde zoals gerapporteerd door de Subversion bibliotheek. In de commandoregel client worden deze vertegenwoordigd door enkelvoudige karakter codes, maar in TortoiseSVN worden ze grafisch als een overlappend pictogram weergegeven. Omdat het aantal overlappende pictogrammen erg beperkt is, vertegenwoordigd elk overlappend pictogram één van meerdere statuswaarden.



Het *Conflicterend* overlappende pictogram vertegenwoordigt de `conflict` status, waar een bijwerking of wisseling leidt tot conflicten tussen lokale wijzigingen en wijzigingen welke vanuit het archief zijn binnen gehaald. Het wordt ook gebruikt om de *belemmering* status aan te duiden, welke zich kan voordoen wanneer een activiteit niet kan worden voltooid.



Het *Aangepast* overlappende pictogram vertegenwoordigt de *aangepast* status, waar u lokale wijzigingen heeft aangebracht, de *samengevoegd* status, waar wijzigingen vanuit het archief zijn samengevoegd met de lokale wijzigingen en de *vervange* status, waar een bestand is verwijderd en vervangen door een ander bestand met dezelfde naam.



Het *Verwijderd* overlappende pictogram vertegenwoordigt de *verwijderd* status, waar een item is klaargezet voor verwijdering, of de *ontbrekend* status, waar een item niet aanwezig is. Een item dat ontbreekt kan zelf natuurlijk geen overlappend pictogram bezitten, maar de bovenliggende map kan worden gemarkeerd wanneer een van de onderliggende items ontbreekt.



Het *Toegevoegd* overlappende pictogram vertegenwoordigt de *toegevoegd* status wanneer een item is toegevoegd aan versiebeheer.



Het *In Subversion* overlappende pictogram wordt gebruikt om een item te vertegenwoordigen dat zich in de *normaal* status bevindt, of een item onder versiebeheer waarvan de status nog niet bekend is. Omdat TortoiseSVN een cacheprocess op de achtergrond gebruikt om de status te vergaren, kan het enige seconden in beslag nemen voordat de overlappende pictogrammen worden bijgewerkt.



Het *Benodigd vergrendeling* overlappende pictogram wordt gebruikt om aan te geven dat op een bestand de eigenschap `svn:needs-lock` is ingesteld. Voor werkkopieën welke zijn aangemaakt met gebruik van Subversion 1.4.0 en hoger wordt de `svn:needs-lock` status door Subversion in de lokale cache geplaatst, wat wordt gebruikt om te bepalen wanneer dit overlappende pictogram moet worden getoond. Voor werkkopieën welke zijn aangemaakt met voorlopers van 1.4.x toont TortoiseSVN dit overlappende pictogram wanneer het bestand de *alleen-lezen* status heeft. Houdt er rekening mee dat Subversion

werkkopieën automatisch opwaardeert wanneer u deze bijwerkt, ondanks dat de `svn:needs-lock` eigenschap niet in de cache zal worden geplaatst totdat het bestand zelf is bijgewerkt.



Het *Vergrendeld* overlappende pictogram wordt gebruikt wanneer de lokale werkkopie een vergrendeling op het betreffende bestand heeft.



Het *Genegeerd* overlappende pictogram wordt gebruikt om een item te vertegenwoordigen dat zich in de `genegeerd` status bevindt door een globaal ingesteld negeren patroon, of door de `svn:ignore` eigenschap van de bovenliggende map. Dit overlappende pictogram is optioneel.



Het *Niet onder versiebeheer* overlappende pictogram wordt gebruikt om een item te vertegenwoordigen dat zich in de `niet onder versiebeheer` status bevindt. Dit is een item in een map onder versiebeheer, maar zelf niet onder versiebeheer valt. Dit overlappende pictogram is optioneel.

Als een item de subversion status `geen` heeft (het item bevindt zich niet in een werkkopie) wordt er geen overlappend pictogram getoond. Als u ervoor heeft gekozen om de overlappende pictogrammen voor *Genegeerd* en *Niet onder versiebeheer* niet actief te maken, worden voor deze bestanden ook geen overlappende pictogrammen getoond.

Een item kan slechts één Subversion statuswaarde hebben. Een bestand kan bijvoorbeeld lokaal zijn aangepast en op het zelfde moment zijn gemarkeerd voor verwijdering. Subversion geeft één enkele statuswaarde - in dit geval `verwijderd`. Deze prioriteiten worden in Subversion zelf gedefinieerd.

Als TortoiseSVN de status recursief toont (standaard instelling), toont elke map een overlappend pictogram dat zijn eigen status en dat van de onderliggende sub-mappen weergeeft. Om een enkel *samenvattend* overlappende pictogram te tonen gebruiken we bovenstaande volgorde in prioriteit om te bepalen welk overlappend pictogram moet worden gebruikt, waarbij het *Conflicterend* overlappende pictogram de hoogste prioriteit heeft.

Het zou kunnen zijn dat u bemerkt dat niet al deze pictogrammen op uw systeem worden gebruikt. Dat is omdat het aantal overlappende pictogrammen wat door Windows wordt toegestaan begrenst is op 15. Windows gebruikt 4 hiervan en de overige 11 kunnen door andere applicaties worden gebruikt. Als er niet voldoende ingangen voor overlappende pictogrammen aanwezig zijn, probeert TortoiseSVN een “Good Citizen (TM)” te zijn en begrenst het zijn eigen gebruik van overlappende pictogrammen om andere applicaties een kans te geven.

- *Normaal*, *Aangepast* en *Conflicterend* worden altijd geladen en zijn zijn altijd zichtbaar.
- *Verwijderd* wordt indien mogelijk geladen, maar valt terug naar *Aangepast* als er niet genoeg ingangen zijn.
- *Alleen-lezen* wordt indien mogelijk geladen, maar valt terug op *Normaal* als er niet genoeg ingangen zijn.
- *Geblokkeerd* wordt alleen geladen indien er minder dan 13 overlappende pictogrammen zijn geladen. Het valt terug op *Normaal* als er niet genoeg ingangen zijn.
- *Toegevoegd* wordt alleen geladen indien er minder dan 14 overlappende pictogrammen zijn geladen. Het valt terug op *Aangepast* als er niet genoeg ingangen zijn.

Bijlage G. Svnselve beveiligen met SSH

In deze sectie wordt stap voor stap beschreven hoe je Subversion en TortoiseSVN moet instellen om gebruik te gaan maken van het `svn+ssh` protocol. Als je al gebruik maakt van SSH geauthenticeerde connecties bij het inloggen op je server, dan ben er eigenlijk al en vind je meer informatie in het Subversion boek. Als je SSH nog niet gebruikt, maar het wel wilt gaan gebruiken om je Subversion installatie te beveiligen, dan geeft deze handleiding de mogelijkheid om dit te realiseren zonder dat je voor elke Subversion gebruiker apart een SSH account hoeft aan te maken.

Bij deze eenvoudige implementatie maken we een enkele SSH gebruikersnaam aan voor alle subversion gebruikers en gebruiken verschillende authenticatie sleutels om een onderscheid te maken tussen de echte Subversion gebruikers.

In deze appendix gaan we er van uit dat je de subversion programma's al geïnstalleerd hebt en je al een archief hebt aangemaakt, zoals elders in deze handleiding beschreven. Merk op dat je svnselve *niet* moet starten als service of daemon als je SSH gebruikt.

Veel van de hier beschreven informatie komt uit de tutorial van Marc Logemann, www.logemann.org [<http://www.logemann.org/>]. Additionele informatie voor het opzetten van een Windows server is beschikbaar gesteld door Thorsten Müller. Bedankt jongens!

G.1. Een Linux server opzetten

Je moet SSH ingeschakeld hebben op je server en we gaan er hier van uit dat je OpenSSH zult gebruiken. Bij de meeste distributies is dit al geïnstalleerd. Om hier zeker van te zijn, type:

```
ps xa | grep sshd
```

en zoek naar ssh processen.

Een punt dat opgemerkt moet worden is dat als je zelf Subversion bouwt van de broncode en geen argument opgeeft voor `./configure`, dan zal Subversion een `bin` map aanmaken onder `/usr/local` en de binaries zullen dan daarin opgeslagen worden. Als je tunnels wilt gebruiken met SSH, dan moet je in de gaten houden dat de gebruiker die via SSH inlogt svnselve en enkele andere binaries moet kunnen opstarten. Neem daarom `/usr/local/bin` op in de `PATH` variabele of maak symbolic links aan voor je binaries naar de `/usr/sbin` map of elke andere map die gemeenschappelijk is in `PATH`.

Om te controleren of alles goed staat, log dan in als de eindgebruiker met SSH en type:

```
which svnselve
```

. Dit commando zal je laten weten of svnselve te bereiken is.

Maak een nieuwe gebruiker aan, die we gaan gebruiken om het svn archief te benaderen:

```
useradd -m svnuser
```

Zorg er voor dat deze gebruiker alle rechten heeft op het archief.

G.2. Een Windows server opzetten

Installeer de Cygwin SSH daemon zoals hier beschreven: <http://pigtail.net/LRP/printsrv/cygwin-sshd.html>

Maak een nieuw Windows gebruikers account aan `svnuser`, welke we gaan gebruiken om het archief te benaderen. Zorg er voor dat deze gebruiker alle rechten voor toegang tot het archief heeft.

Als er nog geen wachtwoord ingesteld is, maak er dan één aan vanuit de Cygwin console met:

```
mkpasswd -l > /etc/passwd
```

G.3. SSH Client Tools voor gebruik met TortoiseSVN

Haal de programma's die we nodig hebben voor SSH op windows van deze site: <http://www.chiark.greenend.org.uk/~sgtatham/putty/> Ga naar de download scitie en haal Putty, Plink, Pageant en Puttygen op.

G.4. Maken van OpenSSH Certificaten

De volgende stap is het aanmaken van een sleutel paar voor authenticatie. Er zijn twee manieren mogelijk om sleutels aan te maken. De eerste is door gebruik te maken van de OpenSSH tool `ssh-keygen`. Download een privé sleutel naar je client en converteer de privé sleutel naar een PuTTY-stijl privé sleutel. De tweede is de sleutels aan te maken met PuTTYgen op de client, de publieke sleutel naar je server te sturen en je privé sleutel met PuTTY te gebruiken.

G.4.1. Sleutels aanmaken met `ssh-keygen`

Log in op de server als root of `svnuser` en type:

```
ssh-keygen -b 1024 -t dsa -N passphrase -f keyfile
```

en een echt wachtwoord (die jij alleen weet) en een sleutel bestand te gebruiken. We hebben nu net een SSH2 DSA sleutel met een 1024 bits sleutel-zin gemaakt. Als je nu

```
ls -l keyfile*
```

typt zie je twee bestanden, `keyfile` en `keyfile.pub`. Zoals je misschien kunt raden is de `.pub` het publieke sleutel bestand en de andere de privé.

Voeg de publieke sleutel toe in de `.ssh` map met de `svnuser` home map.

```
cat keyfile.pub >> /home/svnuser/.ssh/authorized_keys
```

Om de privé sleutel te kunnen gebruiken, moeten we deze converteren naar de putty indeling. Dit komt doordat het bestand met de privé sleutel niet gespecificeerd is een standaard autoriteit. Nadat je de privé sleutel naar je client PC gehaald hebt, start PuTTYgen en gebruik Conversions → Import key. Blader naar je `keyfile` bestand die je van de server hebt gehaald en voer je wachtwoord (of de zin) in die je hebt gebruikt toe je het bestand aanmaakte. Klik vervolgens op Save private key en sla het bestand op als `keyfile.PPK`.

G.4.2. Sleutels aanmaken met PuTTYgen

Gebruik PuTTYgen voor het genereren van een publieke sleutel/ privé sleutel paar en sla deze op. Kopieer de publieke sleutel naar de server en voeg deze toe aan `.ssh` map in de home map van de `svnuser`:

```
cat keyfile.pub >> /home/svnuser/.ssh/authorized_keys
```

G.5. Testen met PuTTY

Om de verbinding te testen gebruiken we PuTTY. Start het programma en stel in de Sessions of Sessies tab de hostnaam of het IP address van je server in, het protocol op SSH en sla de sessie op als

SvnConnectie of elke willekeurige andere naam die je wilt gebruiken. In de SSH tab stel je het SSH protocol in op versie 2 en bij Auth geef je het hele pad naar het .PPK bestand met de privé sleutel, welke je eerder geconverteerd hebt. Ga terug naar de Sessions of Sessies tab en druk op de Save of Opslaan knop. Nu zul je SvnConnectie zien staan in de lijst met opgeslagen sessies.

Klik op Open en je zou een telnet-achtig loginscherm moeten zien. Gebruik svnuser als gebruikersnaam en als alles goed is zou er direct een verbinding opgestart moeten worden, zonder dat er om een wachtwoord wordt gevraagd.

Misschien moet je /etc/sshd_config op de server wijzigen. Bewerk de regels als volgt en herstart de SSH service daarna.

```
PubkeyAuthentication yes
PasswordAuthentication no
PermitEmptyPasswords no
ChallengeResponseAuthentication no
```

G.6. SSH testen met TortoiseSVN

Tot op dit moment hebben we alleen getest of je in kunt loggen met SSH. Nu moeten we kijken of via de SSH verbinding svnserve ook opgestart kan worden. Verander op de server /home/svnuser/.ssh/authorized_keys als volgt om er voor te zorgen dat meerdere subversion gebruikers het systeem account svnuser kunnen gebruiken. Merk op dat elke subversion auteur dezelfde gebruikersnaam/login gebruikt, maar wel met ieder een aparte authenticatie sleutel. Je hoeft dus maar één regel per auteur toe te voegen.

Noot: Het volgende staat allemaal op één lange regel.

```
command="svnserve -t -r <ReposRootPath> --tunnel-user=<author> ",
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,
no-pty ssh-rsa <PublicKey> <Comment>
```

Je moet verschillende waardes instellen afhankelijk van jouw opstelling.

<ReposRootPath> moet vervangen worden door het pad van de map waarin je archieven staan. Hiermee wordt voorkomen dat je volledige paden van servers in je URL op moet nemen. Merk op dat je schuine strepen naar rechts (forward slash) moet gebruiken, zelfs op Windows servers, bijv. c:/svn/reposroot. In het volgende voorbeeld gaan we er van uit dat je een archief map hebt in de basismap van het archief met de naam repos.

<author> moet vervangen worden door de svn auteur die je wilt opslaan bij een vastlegging. Dit zorgt er ook voor dat svnserve zijn eigen toegangsrechten kan gebruiken voor svnserve.conf.

<PublicKey> moet vervangen worden door de publieke sleutel die je eerder hebt gegenereerd.

<Comment> kan elke commentaar bevatten dat je wil gebruiken, maar is vooral handig voor het koppelen van een svn auteursnaam aan de echte naam van een persoon.

Klik met recht op een map in de Windows Verkenner en selecteer TortoiseSVN → Archief-verkenner. Type achter de dubbele schuine streep naar rechts de gebruikersnaam voor de verbinding naar de server, in ons geval svnuser. Na het @ teken geven we de PuTTY sessie naam. Deze sessie heeft alle details over waar de privé sleutel gevonden moet worden en het IP adres of de DNS van de server. Als laatste moeten we het pad van het archief relatief ten opzicht van hoofdmap van de archief van de server, zoals gespecificeerd in het authorized_keys bestand.

Klik op de OK knop en het zou mogelijk moeten zijn om door het archief te bladeren. Als dat zo is, dan heb je nu een SSH tunnel samenwerken met TortoiseSVN.

Merk op dat TortoiseSVN standaard zijn eigen versie van Plink gebruikt om de verbinding op te zetten. Hiermee wordt voorkomen dat er elke keer een console scherm te voorschijn komt bij een authenticatie poging, maar dit houdt ook in dat foutmelding nergens getoond kunnen worden. Als je de foutmelding “Unable to write to standard output” krijgt, kun je proberen om Plink als in te stellen als client bij de netwerk instellingen van TortoiseSVN. Hiermee kun je dan wel de echte foutmelding zien die door Plink gegenereerd wordt.

G.7. SSH Configuration Varianten

Een methode om de URL in TortoiseSVN te vereenvoudigen is door de gebruiker in de PuTTY sessie op te nemen. Hiervoor moet je de reeds gedefinieerde sessie `SvnConnectie` in PuTTY laden en in de `Connection` tab de `Auto login user` instellen op de gebruikersnaam, bijv. `svnuser`. Sla de PuTTY sessie op zoals eerder gedaan en probeer de volgende URL uit binnen TortoiseSVN:

```
svn+ssh://SvnConnection/repos
```

Dit keer geven we alleen de PuTTY sessie `SvnConnectie` door aan de SSH client die TortoiseSVN gebruikt (TortoisePlink.exe). Deze client controleert de sessie op alle benodigde details.

Op het moment dat deze tekst geschreven is, controleert PuTTY niet alle opgeslagen configuraties. Als je dus meerdere configuraties hebt met dezelfde servernaam, dan zal het de eerste overeenkomende testen. Bovendien, als je de standaard configuratie bewerkt en opslaat, dan wordt de login naam van automatisch inloggen *not* opgeslagen.

Veel mensen gebruiken graag Pageant voor het opslaan van al hun sleutels. Omdat een PuTTY sessie in staat is om sleutels op te slaan, heb je Pageant niet altijd nodig. Maar stel je voor dat je sleutels voor verschillende servers op wilt slaan; in dat geval zal je de PuTTY sessie telkens weer moeten bewerken, afhankelijk van de server waarmee je verbinding wilt maken. In deze situatie zal het handig zijn Pageant te gebruiken, omdat zodra PuTTY, Plink, TortoisePlink of een andere PuTTY-gebaseerd tool een verbinding probeert te maken met een SSH server, deze tools alle privé sleutels van Pageant nagaat om de verbinding op te zetten.

Start hiervoor de Pageant en voeg de privé sleutel toe. Dit zou dezelfde sleutel moeten zijn als die je gedefinieerd hebt in de eerder beschreven PuTTY sessie. Als je Pageant gebruikt voor het opslaan van privé sleutels, dan kun je de referentie naar het bestand met de privé sleutel die je opgeslagen hebt in de PuTTY sessie verwijderen. Je kunt natuurlijk meerdere sleutels voor andere servers of andere gebruikers opslaan.

Als je deze procedure niet telkens wilt herhalen bij het opstarten van je client, dan moet je Pageant in de auto-start groep plaatsen van je Windows installatie. Je kunt de sleutels aan de volledige paden toevoegen als opdrachtregel optie bij Pageant.exe.

De laatste methode om een verbinding met een SSH server op te zetten is door eenvoudigweg deze URL in TortoiseSVN in te voeren:

```
svn+ssh://svnuser@100.101.102.103/repos  
svn+ssh://svnuser@mydomain.com/repos
```

Zoals je kunt zien, gebruiken we geen opgeslagen PuTTY sessie, maar een IP adres (of domeinnaam) als doel van de verbinding. We geven ook de gebruikersnaam op, maar je vraagt je misschien af hoe het bestand met de privé sleutel gevonden wordt. Omdat TortoisePlink.exe gewoon een gemodificeerde versie is van de standaard Plink tool uit de PuTTY suite, zal TortoiseSVN ook alle sleutels opgeslagen in Pageant proberen.

Als je deze laatste methode gebruikt, zorg er dan voor dat je niet de standaard gebruikersnaam hebt ingesteld in PuTTY. We hebben melding gehad van een bug waarbij PuTTY de verbinding afsluit in zulke situaties. Wis `HKEY_CURRENT_USER\Software\SimonTatham\Putty\Sessions\Default%20Settings\HostName` om de standaard gebruikersnaam te verwijderen.

Verklarende woordenlijst

Archief	Een archief is een centrale plaats waar data opgeslagen en beheerd wordt. Een archief kan een plaats zijn waar meerdere databases of bestanden zijn opgeslagen om gedistribueerd te worden over een netwerk. Een archief kan ook een locatie zijn die direct toegankelijk is voor een gebruiker, zonder dat er gegevens over een netwerk verstuurd hoeven te worden (oftewel een locatie op een lokale harde schijf).
BASE revisie	De huidige basis revisie van een bestand of map in je <i>werkkopie</i> . Dit is de revisie van een bestand of map van de laatste ophaal actie, verversing of vastlegging. De BASE revisie is normaal gesproken niet gelijk aan de HEAD revisie (er kunnen in de tussentijd alweer wijzigingen in het archief vastgelegd zijn, waardoor de lokale bestanden achter lopen op de HEAD revisie in het archief).
BDB	Berkeley DB. Een grondig geteste database backend voorarchieven, welke niet gebruikt kan worden op netwerk schijven. Standaard voorarchieven van voor versie 1.2.
Blokkade	Als je een blokkade plaatst op een object met versiebeheer, dan markeer je dit object in het archief als niet-vast-te-leggen, behalve vanuit de werkkopie waar vandaan de blokkade geplaatst is.
Conflict	Als wijzigingen in het archief worden samengevoegd met je lokale wijzigingen, dan kan het zijn dat er wijzigingen op dezelfde regels zijn doorgevoerd. In zo'n situatie kan Subversion niet automatisch beslissen welke versie gebruikt moet worden en markeert het bestand als in conflict zijnde. Je moet het conflict dan handmatig oplossen (eventueel met behulp van TortoiseMerge), voordat je wijzigingen weer kunt vastleggen.
Eigenschap	Naast het toepassen van versiebeheer voor je mappen en bestanden, geeft Subversion je ook de mogelijkheid om metadata met versiebeheer te gebruiken - aangeduid als "eigenschappen" voor elke map en bestand onder versiebeheer. Elke eigenschap heeft een naam en waarde, net zoals een register sleutel. Subversion maakt intern gebruik van enkele speciale eigenschappen, zoals <code>svn:eol-style</code> . TortoiseSVN heeft er ook enkele, zoals bijvoorbeeld <code>tsvn:logminsize</code> . Je kunt ook zelf eigenschappen toevoegen, waarvoor je zelf de naam en waarde kunt kiezen.
Exporteren	Dit commando levert een kopie op van een map onder versiebeheer, net zoals een werkkopie, maar dan zonder de lokale <code>.svn</code> mappen.
FSFS	Een Subversion eigen bestandssysteem voor de backend vanarchieven. Kan gebruikt worden op netwerkschijven. Standaard voor versie 1.2 en nieuwerearchieven.
Geschiedenis	Toont de revisiegeschiedenis van een bestand of map. Ook bekend als "Log"
GPO	Groepsolicyobject
HEAD revisie	De meest recente revisie van een bestand of map in het <i>archief</i> . Ook bekend als HEAD revisie.

Importeren	Het Subversion commando voor het in het archief importeren van een complete mapstructuur in één enkele revisie.
Kopie	In een Subversion archief kun je een kopie aanmaken van een enkel bestand of een hele mapstructuur. Deze kopieën worden gemaakt als “goedkope kopieën” wat eigenlijk niet meer is dan het aanmaken van een link naar het origineel, waardoor ze bijna geen extra ruimte in beslag nemen. Bij het maken van een kopie wordt de geschiedenis ook meegenomen, waardoor je de geschiedenis kunt achterhalen van voordat de kopie werd gemaakt.
Locatie wijzigen	<p>Als je archief verhuist, bijvoorbeeld doordat je het verplaatst hebt naar een andere map op je server of het domein van de server is veranderd, moet je je werkkopie de “locatie wijzigen” van je werkkopie. De URL's voor het archief van je werkkopie moeten gaan wijzen naar de nieuwe locatie.</p> <p>Noot: je moet dit commando alleen gebruiken als het hele archief verhuist is, terwijl jouw werkkopie nog steeds wijst naar dezelfde locatie in dat archief. In elke andere situatie moet je in plaats hiervan waarschijnlijk het “Wisselen” commando gebruiken.</p>
Log	Toont de revisiegeschiedenis van een bestand of map. Ook bekend als “Geschiedenis”
Ongedaan maken	Subversion bewaart een lokale “ongewijzigde” kopie van elk bestand, zoals het was op het moment dat je je werkkopie voor het laatst ververste. Als je wijzigingen hebt aangebracht en besluit deze terug te draaien, dan kun je met het commando “Ongedaan maken” gebruiken om de ongewijzigde kopie weer terug te krijgen.
Ophalen	Een Subversion commando waarmee een lokale werkkopie wordt gemaakt in een lege (lokale) map door de bestanden met versiebeheer uit het archief op te halen.
Oplossen	Als bestanden in een werkkopie in een conflict situatie zijn achtergebleven na een samenvoegactie, dan moeten die conflicten door een mens opgelost worden door gebruik te maken van een tekstverwerker (of misschien TortoiseMerge). Dit proces wordt “Conflicten Oplossen” genoemd. Als je de conflicten hebt opgelost, dan moet je het conflicterende bestand markeren als Opgelost, waarna de wijzigingen vastgelegd kunnen worden.
Optie	Met “Verversen-naar-revisie” kun je je werkkopie de status/inhoud geven zoals de informatie beschikbaar was op een ander moment in de tijd, zo kun je met “Wisselen” je werkkopie naar een ander deel van het archief laten kijken wat inhoud betreft. Dit is vooral handig als je aan de basislijn en takken werkt, waarbij er maar enkele bestanden verschillend zijn. Je kunt dan je werkkopie wisselen tussen de twee, terwijl alleen de gewijzigde bestanden verstuurd worden.
Patch	Als er in een werkkopie alleen wijzigingen in tekstbestanden zijn doorgevoerd, dan is het mogelijk om met het Diff commando van Subversion een enkel bestand te genereren met de wijzigingen in Unified Diff formaat. Een bestand met deze indeling wordt meestal een “Patch” genoemd, kan naar iemand anders (of een maillijst) gestuurd worden en worden toegepast op een werkkopie. Iemand die geen rechten voor het vastleggen heeft, kan zo toch wijzigingen maken en deze in een patch bestand naar iemand sturen die wel

	<p>rechten heeft. Je kunt de patch ook gebruiken als je je wijzigingen wilt laten nakijken door iemand anders.</p>
Revisie	<p>Elke keer dat je een set van wijzigingen vastlegt, maak je een nieuwe “revisie” aan in het archief. Elke revisie geeft de status van de structuur van het archief weer op een zeker moment in de tijd. Als je terug wilt gaan in de tijd, kun je het archief bekijken zoals het was ten tijde van revisie N.</p> <p>Op een andere manier gezegd, een revisie kan wijzen naar een set van wijzigingen die zijn doorgevoerd toen de revisie aangemaakt is.</p>
Revisie eigenschap (revprop)	<p>Net zoals bestanden eigenschappen kunnen hebben, kan elke revisie in het archief ook eigenschappen hebben. Enkele speciale revprops worden automatisch toegevoegd als de revisie gemaakt wordt, namelijk: <code>svn:date</code> <code>svn:author</code> <code>svn:log</code> respectievelijk de datum en tijd van de vastlegging, degene die de vastlegging uitvoerde en het logboek bericht. Deze eigenschappen kunnen bewerkt worden, maar deze wijzigingen worden niet met versiebeheer gevolgd. Elke wijziging aan zo'n eigenschap is dus permanent en kan niet ongedaan gemaakt worden.</p>
Samenvoegen	<p>Dit is het proces waarmee je wijzigingen vanuit het archief in jouw lokale werkkopie opneemt, zonder dat de wijzigingen die jij hebt toegepast verstoord worden. Soms kunnen de wijzigingen niet helemaal automatisch samengevoegd worden, waardoor je werkkopie in een zogenoemde conflict status terecht komt.</p> <p>Samenvoegen vindt automatisch plaats als je je werkkopie ververst. Je kunt ook specifieke wijzigingen samenvoegen vanuit een andere tak met het TortoiseSVN Samenvoeg commando.</p>
Schonen	<p>Om het Subversion boek aan te halen: “Recursief opschonen van de werkkopie, blokkades verwijderen en het afronden van niet afgeronde acties. Als je ooit een <i>werkkopie geblokkeerd</i> foutmelding hebt gehad, gebruik dan dit commando om achtergebleven blokkades te verwijderen en je werkkopie weer in een bruikbare situatie te brengen.” Merk op dat in deze context <i>geblokkeerd</i> refereert naar een blokkade van het lokale bestandssysteem, niet het plaatsen van blokkades op een archief.</p>
SVN	<p>Een veelgebruikte afkorting voor Subversion</p> <p>De naam van het Subversion eigen protocol wat gebruikt wordt door “svnserve”, de archief server.</p>
Tak	<p>Een term die regelmatig gebruikt wordt bij versiebeheer systemen om aan te geven wat er gebeurt als een ontwikkeling zich op een bepaald moment opsplijt en 2 aparte paden volgt. Je kunt een vertakking maken van de basislijn voor het ontwikkelen van een nieuwe functie, zonder dat de basislijn onstabiel wordt. Of je kunt een stabiele vrijgave vertakken om alleen bug op te lossen, terwijl nieuwe ontwikkelingen plaatsvinden op de onstabiele basislijn (trunk). In Subversion worden takken gemaakt als “goedkope kopieën”.</p>
Toevoegen	<p>Een Subversion commando die gebruikt wordt om bestanden en mappen toe te voegen aan je werkkopie. De nieuwe objecten worden pas aan het archief toegevoegd als je je wijzigingen vastlegt.</p>

Vastleggen	Met dit Subversion commando worden de wijzigingen in je lokale werkkopie naar het archief gestuurd, waardoor een nieuwe revisie in het archief wordt aangemaakt.
Verklaren	Dit commando is alleen voor tekstbestanden bedoeld en analyseert elke regel, zodat het aangegeven kan worden in welke revisie van het archief welke auteur welke regel gewijzigd heeft. Onze GUI implementatie heeft TortoiseBlame en deze toont ook de vastleg datum/tijd en het logboek bericht als je je muis boven het revisienummer houdt.
Verschil	Toont de “Verschillen”, ook bekend als Diff. Erg handig als je wilt zien welke wijzigingen er precies zijn doorgevoerd.
Verversen	Dit Subversion commando haalt de laatste wijzigingen op uit het archief naar je werkkopie, terwijl het wijzigingen van anderen samenvoegt met jouw lokale wijzigingen in je werkkopie.
Verwijderen	Als je een object met versiebeheer verwijderd (en deze wijziging vastlegt) zal het object niet langer aanwezig zijn in het archief vanaf deze vastgelegde revisie. Het object is natuurlijk nog wel beschikbaar in het archief voor de oudere revisies, je kunt het dan dus nog steeds terugvinden. Indien nodig, kun je een verwijderd object kopiëren en weer “terughalen” met de complete geschiedenis.
Werkkopie	Dit is je lokale “zandbak”, de omgeving waar je aan je bestanden met versiebeheer werkt en welke normaal gesproken op je lokale harde schijf is opgeslagen. Je maakt een werkkopie aan door het “Ophalen” van de bestanden uit een archief. Je wijzigingen sla je ook weer in het archief op door het “Vastleggen” van je gewijzigde bestanden.

Register

Symbolen

'werkkopie' zonder versiebeheer, 130
.svn map, 168
_svn map, 168

A

Aanmaken
 Command Line Client, 16
 TortoiseSVN, 16
alleen-lezen, 116
annoteren, 122
Apache, 26
archief, 5, 41
archief URL gewijzigd, 132
archief venster, 137
archief verkenner, 125
archief-verkenner, 125
ASP projecten, 186
authenticatie, 40
Authorisatie, 31
auto-props, 98
automatisering, 188, 192

B

bestanden toevoegen aan archief, 41
bestanden vergelijken, 182
bewerk log/auteur, 74
blokkeren, 116
bug tracker, 133
bug tracking, 133
bugtracker, 133

C

check in, 46
CLI, 193
client hooks, 162
COM, 169, 175
COM SubWCRev interface, 172
command line, 188, 192
command line client, 193
commit bericht, 180
commit ongedaan maken, 181
conflict, 9, 54
contekstmenu, 38
context menu ingangen, 186
controleer op nieuwe versie, 185

D

diff, 80, 120
domeincontroller, 31, 185
Drag and drop, 40, 40

E

eigenschappen, 94

explorer, 1
exporteer veranderingen, 82
exporteren, 130
Externe archieven, 101
externen, 101, 182

F

filteren, 75
functies uitschakelen, 186

G

geluiden, 138
geschiedenis, 65
globaal negeren, 139
globbing, 89
GPO, 185
grafiek, 127
groepsbeleid, 185, 186

H

hernoem bestanden, 86
hernoemen, 91, 125, 180
herorganiseren, 180
hook scripts, 20, 162
hooks, 20

I

IBugtraqProvider, 175
importeren, 41
Index van projecten, 30
installeer, 3
instellingen, 138
invoeren op zijn plaats, 43
issue tracker, 133, 175

K

kopieer bestanden, 86
kopiëren, 104, 125

L

laag prioriteit, 199
label, 86, 104
leeg bericht, 180
link, 20
Link voor Ophalen, 20
log, 65
log cache, 159
logbericht, 180
logboekboodschappen, 65

M

maak een werkkopie, 44
mappen vergelijken, 182
mapstructuur conflicten, 54
markeer vrijgave, 104
maximaliseer, 41

Meervoudige authenticatie, 33
merge tools, 85
mod_authz_svn, 28, 31
msi, 185

N

negeer, 88
Netwerk schijf, 17
Niet geversioneerde bestanden, 88
NTLM, 32

O

ongedaan maken, 93
ophalen, 44
oplossen, 54
opschonen, 94
overlays, 58, 199

P

patch, 120
patroonherkenning, 89
pictogrammen, 58
plugin, 175
prijzen, 122
project eigenschappen, 99
proxy server, 152

R

rechterklik, 38
rechts slepen, 40
register, 166
repository aanmaken, 16
reservekopie, 19
revisie, 12, 127
revisie eigenschappen, 74
revisie schema, 127
revprops, 74

S

samenvoeg conflicten, 114
samenvoegen, 107
 herintegreren, 110
 revisiebereik, 108
 twee takken, 111
samenvoegen herintegratie, 115
samenvoegen traceren, 114
SASL, 25
scheiden van archief, 184
server gebaseerde hook scripts, 20
server verplaatst, 132
server-side acties, 125
snelkoppeling, 184
speciale bestanden, 43
spellingscontrole, 3
SSL, 34
SSPI, 32
statistieken, 76

status, 58, 60
SUBST schijven, 151
Subversion boek, 5
Subversion eigenschappen, 95
SubWCRev, 169
SVNParentPath, 29, 30
SVNPath, 29
svnserve, 21, 23
SVN_ASP_DOT_NET_HACK, 186

T

taalpakketen, 3
tak, 86, 104
terugdraaiing, 181
terugzetten, 93, 181
tijdelijke bestanden, 42
Toegang, 17
toevoegen, 85
TortoiseIDiff, 84
TortoiseSVN eigenschappen, 99
TortoiseSVN koppeling, 20
trefwoorden, 96

U

uitbreiden van trefwoorden, 96
uitsluiten patroon, 139
UNC paden, 17
unified diff, 120
upgrade controle, 185
URL veranderd, 132

V

vastleggen, 46
Veelgestelde vragen, 179
veelvoorkomende projecten, 182
vendor projecten, 182
veranderingen, 182
vergelijken, 62, 80
vergelijken van revisies, 82
verhuizen, 132
Verkenner Kolommen, 60
verklaren, 122
verplaats bestanden, 86
verplaatsen, 91, 180
verplaatste server, 132
verschil gereedschappen, 85
verschil tussen beelden, 84
versie, 185
versie extractie, 169
versiebeheer, 1
versiebeheer verwijder, 184
versiebeheer verwijderen, 132, 184
versienummering in bestanden, 169
versioneer nieuwe bestanden, 85
verspreiden, 185
verstuur wijzigingen, 46
vertalingen, 3

verversen, 52, 181
verwijderen, 90
vestleggen berichten, 65
ViewVC, 137
Voeg trackinglog samen, 73
VS2003, 186

W

web weergave, 137
WebDAV, 27
website, 20
WebSVN, 137
weghalen, 90
werkkopie, 10
werkkopie status, 58
werkset, 62
wijziging in hoofd-/kleine letters, 92
wijziging ongedaan maken, 181
wijzigingen, 60
wijzigingen ophalen, 52
wijzigingen weergeven, 58
Windows domein, 31
Windows shell, 1
wisselen, 106
woordenboek, 3