

TortoiseSVN

**Stav všech verzí souborů a složek je
indikován malými překryvnými ikonami.
Tímto způsobem můžete ihned vidět,
v jakém stavu je Vaše pracovní kopie**

Verze 1.6.16

**Stefan Küng
Lübbe Onken
Simon Large**

TortoiseSVN: Stav všech verzí souborů a složek je indikován malými překryvnými ikonami. Tímto způsobem můžete ihned vidět, v jakém stavu je Vaše pracovní kopie: Verze 1.6.16

Stefan Küng, Lübke Onken a Simon Large

Překlad: Zbyněk Schwarz (zbynek.schwarz@gmail.com)

Vydáno 2011/01/21 21:21:17 (r20750)

Obsah

Předmluva	xi
1. Diváci	xi
2. Průvodce Čtením	xi
3. TortoiseSVN je zadarmo!	xii
4. Komunita	xii
5. Poděkování	xii
6. Terminologie použitá v tomto dokumentu	xii
1. Úvod	1
1.1. Co je TortoiseSVN?	1
1.2. Historie TortoiseSVN	1
1.3. Funkce TortoiseSVN	1
1.4. Instalace TortoiseSVN	2
1.4.1. Systémové požadavky	2
1.4.2. Instalace	3
1.4.3. Jazykové balíčky	3
1.4.4. Kontrola pravopisu	3
2. Základní koncepty Kontroly Verzí	5
2.1. Úložiště	5
2.2. Modely Verzování	5
2.2.1. Problém sdílení souborů	6
2.2.2. Řešení pomocí modelu Zamknout-Změnit-Odemknout	6
2.2.3. Řešení pomocí modelu Zkopírovat-Změnit-Sloučit	7
2.2.4. Co Subversion dělá?	10
2.3. Subversion v akci	10
2.3.1. Pracovní kopie	10
2.3.2. URL úložiště	12
2.3.3. Revize	12
2.3.4. Jak Pracovní Kopie Sledují Úložiště	14
2.4. Shrnutí	14
3. Úložiště	15
3.1. Vytváření úložiště	15
3.1.1. Vytvoření repozitáře z příkazového řádku klienta	15
3.1.2. Vytváření úložiště pomocí TortoiseSVN	15
3.1.3. Lokální Přístup do Úložiště	16
3.1.4. Přístup k úložišti na síťovém disku	16
3.1.5. Rozložení úložiště	17
3.2. Záloha Úložiště	18
3.3. Hook skripty na straně serveru	19
3.4. Odkazy pro Získání	19
3.5. Přístup do úložiště	20
3.6. Server Svnserve	20
3.6.1. Úvod	20
3.6.2. Instalace svnserve	20
3.6.3. Spuštění svnserve	21
3.6.4. Základní ověření pomocí svnserve	22
3.6.5. Lepší zabezpečení se SASL	23
3.6.6. Ověření pomocí svn+ssh	25
3.6.7. Pověření podle cesty v svnserve	25
3.7. Apache Server	25
3.7.1. Úvod	25
3.7.2. Instalace Apache	26
3.7.3. Instalace Subversion	26
3.7.4. Konfigurace	27
3.7.5. Více úložišť	29
3.7.6. Pověření podle cesty	29

3.7.7. Ověření pomocí Domény Windows	30
3.7.8. Více zdrojů ověření	31
3.7.9. Zabezpečení serveru pomocí SSL	32
3.7.10. Použití klientských certifikátů s virtuálními hostiteli SSL	34
4. Příručka Každodenního Použití	36
4.1. Začínáme	36
4.1.1. Překryv Ikon	36
4.1.2. Kontextové menu	36
4.1.3. Táhnout a Pustit	38
4.1.4. Běžné zkratky	39
4.1.5. Ověření	39
4.1.6. Maximalizace Windows	40
4.2. Import dat do úložiště	40
4.2.1. Import	40
4.2.2. Import na místě	42
4.2.3. Mimořádné soubory	42
4.3. Získání Pracovní Kopie	43
4.3.1. Hloubka Získání	43
4.4. Odevzdání Vašich změn do úložiště	45
4.4.1. Dialogové okno Odevzdání	45
4.4.2. Seznam Změn	48
4.4.3. Vyjmout položky ze seznamu odevzdání	48
4.4.4. Zprávy Záznamu Odevzdání	48
4.4.5. Průběh Odevzdání	50
4.5. Aktualizovat pracovní kopii změnami od ostatních	51
4.6. Řešení Konfliktů	53
4.6.1. Konflikty souboru	53
4.6.2. Konflikty Stromu	54
4.7. Získávání informací o stavu	57
4.7.1. Překryv Ikon	57
4.7.2. Sloupce TortoiseSVN v Průzkumníku Windows	58
4.7.3. Místní a Vzdálený Stav	59
4.7.4. Zobrazení rozdílů	60
4.8. Seznam Změn	61
4.9. Dialogové Okno Záznamu Revizí	63
4.9.1. Vyvolání Dialogového Okna Záznamu Revizí	64
4.9.2. Činnosti Záznamu Revizí	64
4.9.3. Získání dodatečných informací	65
4.9.4. Zobrazení více zpráv záznamu	69
4.9.5. Aktuální Revize Pracovní Kopie	70
4.9.6. Funkce Sledování Sloučení	70
4.9.7. Změna Zpráv Záznamu a Autora	71
4.9.8. Filtrování Zpráv Záznamu	72
4.9.9. Statistické informace	72
4.9.10. Offline režim	76
4.9.11. Obnovení zobrazení	76
4.10. Zobrazit Rozdíly	76
4.10.1. Rozdíly souboru	77
4.10.2. Možnosti konců řádků a mezer	78
4.10.3. Porovnávání adresářů	78
4.10.4. Porovnávání snímků pomocí TortoiseIDiff	80
4.10.5. Externí Nástroje Porovnání/Sloučení	81
4.11. Přidání nových souborů a adresářů	81
4.12. Kopírování/Přesun/Přejmenování Souborů a Adresářů	82
4.13. Ignorování Souborů a Adresářů	83
4.13.1. Shoda vzorů v seznamech k Ignorování	84
4.14. Mazání, Přesunutí a Přejmenování	85
4.14.1. Smazání souborů a adresářů	86

4.14.2. Přesouvání souborů a adresářů	87
4.14.3. Změna velikosti písmen v souboru	87
4.14.4. Řešení konfliktů v názvech souborů	88
4.14.5. Oprava Přejmenování Souborů	88
4.14.6. Smazání souborů bez verze	88
4.15. Vrátit změny zpět	89
4.16. Vyčistit	90
4.17. Nastavení projektu	90
4.17.1. Subversion Vlastnosti	91
4.17.2. TortoiseSVN Vlastnosti projektu	94
4.18. Externí položky	96
4.18.1. Externí adresáře	96
4.18.2. Externí soubory	99
4.19. Větvení/Značkování	99
4.19.1. Vytvoření Větvě nebo Značky	99
4.19.2. Získat nebo Překlopit...	101
4.20. Sloučení	102
4.20.1. Sloučení Rozsahu Revizí	103
4.20.2. Znovu začlenit větev	105
4.20.3. Sloučení Dvou Různých Stromů	106
4.20.4. Nastavení Sloučení	107
4.20.5. Zhodnocení výsledků sloučení	108
4.20.6. Sledování Sloučení	109
4.20.7. Řešení Konfliktů při Sloučení	109
4.20.8. Sloučení Dokončenou větev	110
4.20.9. Údržba Větvě Funkcí	110
4.21. Zamykání	111
4.21.1. Jak Zamykání v Subversion Funguje	111
4.21.2. Získání zámku	112
4.21.3. Uvolnění zámku	112
4.21.4. Kontrola stavu zámku	113
4.21.5. Udělat z Neuzamčených Souborů Soubory Pouze Pro Čtení	113
4.21.6. Hákové Skripty Uzamknutí	114
4.22. Vytváření a Používání Záplat	114
4.22.1. Vytvoření Záplatového souboru	114
4.22.2. Použití Záplatového souboru	115
4.23. Kdo Změnil Který Řádek?	115
4.23.1. Obviňovat ze souborů	116
4.23.2. Obvinít Rozdíly	118
4.24. Průzkumník Úložiště	118
4.25. Grafy Revizí	121
4.25.1. Uzly Grafu Revizí	122
4.25.2. Změna zobrazení	122
4.25.3. Použití Grafu	124
4.25.4. Obnovení zobrazení	125
4.25.5. Čištění stromů	125
4.26. Export Pracovní Kopie Subversion	125
4.26.1. Smazání pracovní kopie ze správy verzí	127
4.27. Přemístit pracovní kopii	127
4.28. Integrace se Systémy pro Sledování Chyb/Problémů	128
4.28.1. Přidání Čísel Problémů do zpráv záznamu	128
4.28.2. Získávání informací od Systému pro Sledování Problémů	131
4.29. Integrace s Internetovými Prohlížeči úložiště	132
4.30. Nastavení TortoiseSVN	133
4.30.1. Obecná nastavení	133
4.30.2. Nastavení Grafu Revizí	141
4.30.3. Nastavení Překryvu Ikon	143
4.30.4. Nastavení sítě	146

4.30.5. Nastavení Externího Programu	148
4.30.6. Nastavení Uložených dat	151
4.30.7. Uchovávání záznamu	152
4.30.8. Hákové Skripty na Straně Klienta	155
4.30.9. Nastavení TortoiseBlame	159
4.30.10. Registry Settings	159
4.30.11. Pracovní adresáře Subversion	161
4.31. Poslední krok	161
5. Program SubWCRev	162
5.1. Příkazový řádek SubWCRev	162
5.2. Nahrazení Klíčového Slova	162
5.3. Příklad klíčového slova	163
5.4. Rozhraní COM	164
6. Rozhraní IBUGTRAQProvider	167
6.1. Rozhraní IBUGTRAQProvider	167
6.2. Rozhraní IBUGTRAQProvider2	168
A. Často kladené otázky (FAQ)	171
B. Jak	172
B.1. Přesunout/zkopírovat větší množství souborů najednou	172
B.2. Přinutit uživatele zadat zprávu záznamu	172
B.2.1. Hákový skript na serveru	172
B.2.2. Vlastnosti projektu	172
B.3. Aktualizovat vybrané soubory z úložiště	173
B.4. Vrátit zpět revize v úložišti	173
B.4.1. Pomocí dialogového okna záznamu revizí	173
B.4.2. Pomocí dialogového okna sloučení	173
B.4.3. Pomocí svndumpfilter	174
B.5. Porovnat dvě revize souboru nebo adresáře	174
B.6. Zahrnout společný podprojekt	174
B.6.1. Pomocí svn:externals	174
B.6.2. Pomocí vnořené pracovní kopie	175
B.6.3. Pomocí relativního umístění	175
B.7. Vytvořit zkratku úložiště	175
B.8. Ignorovat soubory, které už mají verzi	175
B.9. Odstranit verzi pracovní kopie	176
B.10. Odebrat pracovní kopii	176
C. Užitečné tipy pro administrátory	177
C.1. Nasazení TortoiseSVN pomocí zásad skupiny	177
C.2. Přesměrovat kontrolu nové verze	177
C.3. Nastavení proměnné prostředí SVN_ASP_DOT_NET_HACK	178
C.4. Zakázání položek kontextového menu	178
D. Automatizace TortoiseSVN	180
D.1. Příkazy TortoiseSVN	180
D.2. Příkazy TortoiseIDiff	183
E. Odkazy Rozhraní Příkazové Řádky	184
E.1. Úmluvy a Základní pravidla	184
E.2. Příkazy TortoiseSVN	184
E.2.1. Získat	184
E.2.2. Aktualizovat	184
E.2.3. Aktualizovat na Revizi	185
E.2.4. Odevzdat	185
E.2.5. Porovnat	185
E.2.6. Zobrazit Záznam	186
E.2.7. Zjistit změny	186
E.2.8. Graf Revizí	186
E.2.9. Průzkumník Úložiště	186
E.2.10. Upravit konflikty	186
E.2.11. Vyřešeno	187

E.2.12. Přejmenovat	187
E.2.13. Smazat	187
E.2.14. Vrátit	187
E.2.15. Vyčistit	187
E.2.16. Uzamknout	187
E.2.17. Uvolnit zámek	187
E.2.18. Větev/značka	187
E.2.19. Překlopit	188
E.2.20. Sloučit	188
E.2.21. Exportovat	188
E.2.22. Přemístit	188
E.2.23. Vytvořit Úložiště Zde	188
E.2.24. Přidat	189
E.2.25. Import	189
E.2.26. Obvinit	189
E.2.27. Přidat do seznamu k ignorování	189
E.2.28. Vytvořit Záplatu	189
E.2.29. Použít Záplatu	189
F. Detaily zavedení	190
F.1. Překryv Ikon	190
G. Zabezpečení svnservice pomocí SSH	192
G.1. Nastavení Serveru Linux	192
G.2. Nastavení Windows Serveru	192
G.3. Nástroje klienta SSH pro použití s TortoiseSVN	193
G.4. Vytváření Certifikátů OpenSSH	193
G.4.1. Vytvoření Klíčů pomocí ssh-keygen	193
G.4.2. Vytvoření klíčů pomocí PuTTYgen	193
G.5. Test pomocí PuTTY	193
G.6. Testování SSH s TortoiseSVN	194
G.7. Konfigurační varianty SSH	195
Slovník pojmů	196
Rejstřík	199

Seznam obrázků

2.1. Typický klient/server systém	5
2.2. Problém, kterému je třeba se vyvarovat.	6
2.3. Řešení pomocí modelu Zamknout-Změnit-Odemknout	7
2.4. Řešení pomocí modelu Zkopírovat-Změnit-Sloučit	8
2.5. ... Pokračování Kopírovat-Změnit-Sloučit	9
2.6. Systém souborů uložště	11
2.7. Úložiště	13
3.1. Menu TortoiseSVN pro složky bez verzí	15
4.1. Průzkumník ukazující překryv ikon	36
4.2. Kontextové menu pro adresář pod správou verzí	37
4.3. Menu Soubor v Průzkumníku pro zkratky v adresáři s verzí	38
4.4. Menu táhnutí pravým tlačítkem pro adresář pod správou verzí	39
4.5. Dialogové okno Ověření	40
4.6. Dialogové okno importu	41
4.7. Dialogové okno Získání	43
4.8. Dialogové okno odevzdání	46
4.9. Kontrola pravopisu v Dialogovém oknu Odevzdání	49
4.10. Dialogové okno průběhu zobrazující probíhající odevzdání	50
4.11. Dialogové okno průběhu zobrazující dokončení aktualizace	51
4.12. Průzkumník ukazující překryv ikon	57
4.13. Zjistit změny	59
4.14. Dialogové okno odevzdání se Seznamem Změn	62
4.15. Dialogové Okno Záznamu Revizí	64
4.16. Horní Panel Dialogového okna Záznamu revizí s Kontextovým menu	65
4.17. Horní panel Kontextového Menu pro 2 vybrané Revize	67
4.18. Dolní panel dialogového okna Záznamů s Kontextovým Menu	68
4.19. Dialogové Okno Záznamu Zobrazující Sledování Sloučení Revizí	71
4.20. Sloupcový graf Odevzdání-podle-Autora	73
4.21. Koláčový graf Odevzdání-podle-Autora	74
4.22. Graf odevzdání podle data	75
4.23. Dialogové okno Jít Offline	76
4.24. Dialogové okno Porovnání Revizí	79
4.25. Prohlížeč porovnání obrázku	80
4.26. Kontextové menu průzkumníka pro soubory bez verze	82
4.27. Menu táhnutí pravým tlačítkem pro adresář pod správou verzí	83
4.28. Kontextové menu průzkumníka pro soubory bez verze	84
4.29. Kontextové menu průzkumníka pro soubory s verzí	86
4.30. Dialogové okno Vrátit	89
4.31. Stránka vlastností v Průzkumníku, karta Subversion	91
4.32. Stránka Subversion vlastností	92
4.33. Přidání vlastností	93
4.34. Dialogové okno Větev/Značka	100
4.35. Dialogové okno Překlopení	102
4.36. Průvodce sloučením - Vyberte Rozsah Revizí	104
4.37. Průvodce Sloučením - Znovu začleňovací Sloučení	106
4.38. Průvodce Sloučením - Sloučení Stromu	107
4.39. Dialogové Okno Zpětného Volání Sloučení při Konfliktu	109
4.40. Dialogové okno znovu začlenění sloučení	110
4.41. Dialogové okno Uzamknutí	112
4.42. Dialogové okno Zjistit Změny	113
4.43. Dialogové okno Vytvořit Záplatu	114
4.44. Dialogové okno Okomentování / Obvinění	116
4.45. TortoiseBlame	117
4.46. Průzkumník Úložiště	119
4.47. Graf Revize	121

4.48. Dialogové okno Export z URL	126
4.49. Dialogové okno přemístění	127
4.50. Ukázka dialogového okna dotazu na systém pro sledování problémů	132
4.51. Dialogové okno nastavení, stránka Obecné	134
4.52. Dialogové okno Nastavení, Stránka Kontextového menu	136
4.53. Dialogové Okno Nastavení, Dialogová okna 1 Stránka	137
4.54. Dialogové okno Nastavení, Dialogová okna Stránka 2	139
4.55. Dialogové okno Nastavení, Strana Barvy	140
4.56. Dialogové okno nastavení, Stránka Graf Revizí	141
4.57. Dialogové okno nastavení, Stránka Barvy Grafu Revizí	142
4.58. Dialogové okno nastavení, Stránka Překryv Ikon	143
4.59. Dialogové okno nastavení, Stránka Sady Ikon	146
4.60. Dialogové okno nastavení, Stránka Síť	147
4.61. Dialogové okno Nastavení, Stránka Prohlížeč Rozdílů	148
4.62. Dialogové okno Nastavení, Pokročilé Dialogové Okno Rozdílů/Sloučení	150
4.63. Dialogové okno nastavení, Stránka Uložená Data	151
4.64. Dialogové okno Nastavení, Stránka Úschova Záznamu	152
4.65. Dialogové okno Nastavení, Statistika Úschovy Záznamu	154
4.66. Dialogové okno nastavení, Stránka Hákové Skripty	155
4.67. Dialogové okno Nastavení, Konfigurace Hákových Skriptů	156
4.68. Dialogové okno nastavení, Stránka Integrace se Systémem pro Sledování Problémů	158
4.69. Dialogové okno nastavení, Stránka TortoiseBlame	159
C.1. The upgrade dialog	177

Seznam tabulek

2.1. Přístupová URL úložiště	12
3.1. Nastavení Apache <code>httpd.conf</code>	28
5.1. Seznam dostupných přepínačů v příkazovém řádku	162
5.2. Seznam dostupných přepínačů v příkazovém řádku	163
5.3. Podporované metody COM/automatizace	164
C.1. Položky menu a jejich hodnoty	178
D.1. Seznam dostupných příkazů a možností	180
D.2. Seznam dostupných možností	183

Předmluva



TortoiseSVN

- Pracujete v týmu?
- Stalo se Vám někdy, že jste pracovali na souboru, a někdo jiný pracuje na stejném souboru ve stejnou dobu? Ztráily se kvůli tomu Vaše změny toho souboru?
- Už jste někdy uložili soubor, a pak jste chtěli vrátit provedené změny? Přáli jste si někdy, abyste mohli vidět, jak vypadal soubor před nějakou dobou?
- Už jste někdy narazili na chybu v projektu a chtěli jste vědět, kdy že se tato chyba dostala do Vašich souborů?

Pokud jste odpověděli „ano“ na jednu z těchto otázek, pak TortoiseSVN je právě pro Vás! Jen si přečtete abyste zjistili, jak Vám TortoiseSVN může pomoci ve Vaší práci. Není to tak těžké.

1. Diváci

Tato kniha je napsána pro znalce počítačů, kteří chtějí používat Subversion pro správu svých dat, ale nechtějí používat příkazový řádek klienta. Vzhledem k tomu, že TortoiseSVN je rozšíření shellu Windows předpokládá se, že uživatel je obeznámen s Průzkumníkem Windows a ví, jak ho používat.

2. Průvodce Čtením

Tento **Předmluva** Vám trochu poví o projektu TortoiseSVN, komunitě lidí, kteří na něm pracují, a licenční podmínky pro jeho použití a distribuci.

1 – „Úvod“ vysvětluje, co TortoiseSVN je, co dělá, odkud pochází a základy pro instalaci na vašem počítači.

V 2 – „Základní koncepty Kontroly Verzí“ dáváme krátký úvod do *Subversion* systému pro správu revizí, z něhož vychází TortoiseSVN. Toto je převzato z dokumentace pro projekt Subversion a vysvětluje různé přístupy k řízení verzí, a jak funguje Subversion.

V kapitole o **3 – „Úložiště“** se vysvětluje, jak nastavit místní úložiště, které je užitečné pro testování Subversion a TortoiseSVN pomocí jediného PC. Také je vysvětleno něco o správě úložiště, která je rovněž relevantní pro úložiště umístěné na serveru. K dispozici je zde také sekce jak nastavit server, pokud ho budete potřebovat.

4 – „Příručka Každodenního Použití“ je nejdůležitější část, protože vysvětluje všechny hlavní rysy TortoiseSVN a jak je používat. Má podobu výuky, začíná se získáním pracovní kopie, její změnou, odevzdáním Vašich změn, atd. Pak přikročí k pokročilejším tématům.

5 – „Program SubWCRev“ je samostatný program, který je součástí TortoiseSVN. Extrahuje informace z vaší pracovní kopie a zapisuje je do souboru. To je užitečné pro připojení informací o sestvení ve Vašich projektech.

B – „Jak...“ tato část odpovídá na některé běžné otázky o plnění úkolů, které nejsou výslovně uvedeny jinde.

Část o **D – „Automatizace TortoiseSVN“** ukazuje, jak můžou dialogy TortoiseSVN GUI být volány z příkazového řádku. To je užitečné pro skriptování, kde ještě potřebujete interakci s uživatelem.

E – „Odkazy Rozhraní Příkazové Řádky“ zprostředkuje vztah mezi příkazy TortoiseSVN a jejich ekvivalenty v příkazovém řádku klienta Subversion `svn.exe`.

3. TortoiseSVN je zadarmo!

TortoiseSVN je zadarmo. Pro používání tohoto programu nemusíte platit a můžete ho používat jak chcete. Je vyvíjen pod licencí GNU General Public License (GPL).

TortoiseSVN je Open Source projekt. To znamená, že máte plný přístup ke zdrojovému kódu tohoto programu. Můžete ho procházet na této adrese <http://code.google.com/p/tortoisesvn/source/browse/>. Budete vyzváni k zadání uživatelského jména a hesla. Uživatelské jméno je `host`, heslo musí být prázdné. Nejnovější verzi (na které se v současnosti pracuje) se nachází pod `/Trunk/`, vydané verze jsou umístěny pod `/tags/`.

4. Komunita

TortoiseSVN i Subversion jsou vyvíjeny komunitou lidí, kteří pracují na těchto projektech. Pocházejí z různých zemí celého světa a společně pracují na vytvoření skvělých programů.

5. Poděkování

Tim Kemp
pro založení projektu TortoiseSVN

Stefan Küng
za tvrdou práci, aby dostal TortoiseSVN tam, kde je teď

Lübbe Onken
za nádherné ikony, logo, odchyt chyb, překládání a řízení překladů

Simon Large
za pomoc s dokumentací a odchytem chyb

Knize Subversion
za skvělý úvod do Subversion a její 2. kapitolu, kterou jsme zde zkopírovali

Projektu Tigris Style
za některé styly, které byly použity v této dokumentaci

Naším Příspěvatelům
za opravy, hlášení chyb a nové nápady, a za pomoc druhým odpovídáním na otázky týkající se našeho mailing listu.

Naším Dárcům
za mnoho hodin radosti s hudbou, kterou nám poslali

6. Terminologie použitá v tomto dokumentu

Pro snazší čtení těchto dokumentů, všechny jména obrazovek a menu z TortoiseSVN jsou označeny jiným písmem. Například Dialog Logu.

Výběr z menu je označen šipkou. TortoiseSVN → Zobrazit záznam znamená: vybrat *Zobrazit log záznamy* z kontextového menu *TortoiseSVN*.

Tam, kde se objevuje místní kontextové menu v jednom z dialogů TortoiseSVN, je zobrazeno takto: Kontextové menu → Uložit jako ...

Tlačítka uživatelského rozhraní jsou označena takto: Stiskněte OK pro pokračování.

Akce uživatelů jsou vyznačeny pomocí tučného písma. **Alt + A** : Stiskněte klávesu **Alt** na Vaší klávesnici a při držení zároveň stiskněte klávesu **A**. Táhnutí pravým tlačítkem : Stiskněte pravé tlačítko myši a při držení *táhněte* položky na nové umístění.

Systémový výstup a vstup z klávesnice je také indikován j i n ý m písmem.



Důležité

Důležité poznámky jsou označeny ikonou.



Tip

Tipy, které vám usnadní život.



Výstraha

Místa, kde musíte být opatrní, co děláte.



Varování

Tam, kde je vyžadováno extrémní opatrnosti, může dojít k poškození dat či jiným ošklivým věcem, pokud jsou tato varování ignorována.



Kapitola 1. Úvod

Správa verzí je umění řízení změn informací. To dlouho byl zásadní nástroj pro programátory, kteří obvykle tráví svůj čas prováděním malých změn v softwaru a pak další den vracejí nebo odevzdají některou z těchto změn. Představte si, že tým takovýchto vývojářů pracují současně - a možná i současně na stejném souboru! - a můžete vidět, proč je zapotřebí systém, pro *správu potencionálního chaosu*.

1.1. Co je TortoiseSVN?

TortoiseSVN je bezplatný open-source klient pro Subversion systém správy verzí. To znamená, že TortoiseSVN spravuje soubory a adresáře v průběhu času. Soubory jsou uloženy v centrálním úložišti. Úložiště je podobné jako běžný server se soubory, až na to, že si pamatuje každou změnu kdo kdy udělal ve Vašich souborech a adresářích. To vám umožní obnovit starší verze souborů a zkoumat historii, jak a kdy se Vaše data změnila, a kdo je změnil. To je důvod, proč si mnoho lidí myslí, že Subversion, a systém pro správu verzí vůbec, je jako „stroj času“.

Některé systémy pro správu verzí jsou také software pro řízení konfigurace (SCM). Tyto systémy jsou speciálně přizpůsobeny pro správu stromů zdrojového kódu, a mají mnoho funkcí, které jsou specifické pro vývoj software - jako je nativní podpora programovacích jazyků, nebo dodávají nástroje pro vytváření softwaru. Subversion, nicméně, není jeden z těchto systémů, je to všeobecný systém, který lze použít ke správě všech sbírek souborů, včetně zdrojového kódu.

1.2. Historie TortoiseSVN

V roce 2002 Tim Kemp zjistil, že Subversion je velmi dobrý systém pro správu verzí, ale postrádá dobrého GUI klienta. Myšlenka pro klienta Subversion jako integraci do rozhraní Windows brala inspiraci z podobného klienta pro CVS jménem TortoiseCVS.

Tim zkoumal zdrojový kód TortoiseCVS a použil ho jako základ pro TortoiseSVN. Pak založil projekt, zaregistroval doménu `tortoisesvn.org` a dal zdrojový kód online. Během té doby, Stefan Küng hledal dobrý a svobodný systém správy verzí a našel Subversion a zdroj pro TortoiseSVN. Vzhledem k tomu, že TortoiseSVN nebyl v té době ještě připraven k použití, se brzy připojil k projektu a začal programovat. Zanedlouho přepsal většinu stávajícího kódu a začal přidávat příkazy a funkce, až do bodu, kdy nic z původního kódu nezůstalo.

Jak se Subversion stával více stabilnějším, přitahoval více a více uživatelů, kteří také začali používat TortoiseSVN jako jejich klienta Subversion. Uživatelská základna rychle rostla (a stále roste každý den). V tu dobu Lubbe Onken nabídl svoji pomoc a vytvořil krásné ikony a logo pro TortoiseSVN. Dále se stará o webové stránky a spravuje překlady.

1.3. Funkce TortoiseSVN

Co dělá z TortoiseSVN takového dobrého klienta Subversion? Zde je krátký seznam vlastností.

Integrace do Rozhraní

TortoiseSVN se hladce integruje do prostředí Windows (tj. průzkumník). To znamená, že můžete pokračovat v práci s nástroji, které už znáte. A nemusíte přepínat do jiné aplikace vždy, když potřebujete funkce správy verzí!

A nemusíte ani používat Průzkumník Windows. Kontextové menu TortoiseSVN fungují i v mnoha jiných manažerech souborů, a v dialogovém okně Soubor/Otevřít, které je společné pro většinu standardních aplikací Windows. Měli byste však mít na paměti, že TortoiseSVN je záměrně vyvinut jako rozšíření pro Průzkumník Windows. Může se stát, že v jiných aplikacích integrace není tak kompletní a např. překryvy ikon nemusí být zobrazeny.

Překryv ikon

Stav všech verzí souborů a složek je indikován malými překryvnými ikonami. Tímto způsobem můžete ihned vidět, jaký je stav vaší pracovní kopie.

Snadný přístup k příkazům Subversion

Veškeré příkazy Subversion jsou dostupné z kontextového menu průzkumníka. TortoiseSVN do něho přidá své vlastní podmenu.

Vzhledem k tomu, že TortoiseSVN je klient Subversion, bychom Vám také rádi ukázali i některé funkce Subversion:

Verze adresáře

CVS sleduje pouze historii jednotlivých souborů, ale Subversion zavedlo „virtuální“ souborový systém verzí, který sleduje změny celého adresářového stromu v průběhu času. Soubory a adresáře jsou opatřeny verzí. Výsledkem je existence opravdových příkazů **přesunout** a **kopírovat** na straně klienta, které pracují se soubory a adresáři.

Atomová odevzdání

Odevzdání buď jde do úložiště úplně, nebo vůbec ne. To umožňuje vývojářům vytvářet a odevzdávat změny změny jako logické kousky.

Metadata s verzí

Každý soubor a adresář má přiložen neviditelný soubor „vlastností“. Můžete si vymyslet a ukládat libovolný pár klíčů/hodnot, které chcete. Vlastnosti jsou opatřeny verzí v průběhu času, tak jako obsah souborů.

Výběr síťových vrstev

Subversion má abstraktní pojem o přístupu k úložišti, takže je snadné pro lidi si zavést nový síťový mechanismus. „Vyspělý“ síťový server Subversion je modul pro webový server Apache, který mluví variantou HTTP nazývanou WebDAV/DeltaV. To dává Subversion velkou výhodu v oblasti stability a interoperability, a poskytuje zdarma různé klíčové vlastnosti např.: autentizace, autorizace, komprese po síti a prohlížení úložiště. Menší, samostatný server Subversion je také k dispozici. Tento server mluví vlastním protokol, který lze snadno tunelovat přes ssh.

Konzistentní zpracování dat

Subversion vyjadřuje rozdíl souborů za použití binárního odlišovacího algoritmu, který pracuje shodně na textových (čitelný člověkem) i binárních (nečitelný člověkem) souborech. Oba typy souborů jsou uloženy stejně komprimované v úložišti, a rozdíly jsou přenášeny v obou směrech přes síť.

Efektivní větvení a značkování

Cena za větvení a značkování nemusí být úměrná velikosti projektu. Subversion vytváří větve a značky tím, že zkopíruje projekt, pomocí mechanismu podobný tvrdému odkazu. Proto tato operace zabírá pouze velmi malé, neměnné množství času, a velmi málo místa v úložišti.

Hackabilita

Subversion nemá žádnou dávnou historii, je implementován jako kolekce sdílených knihoven jazyka C s dobře definovaným rozhraním API. Tím je Subversion dobře udržitelná a použitelná v jiných aplikacích a jazycích.

1.4. Instalace TortoiseSVN

1.4.1. Systémové požadavky

TortoiseSVN funguje na Windows 2000 SP2, Windows XP nebo vyšší. Windows 98, Windows ME a Windows NT4 již nejsou podporovány od TortoiseSVN 1.2.0, ale stále si můžete stáhnout starší verze, jestli je opravdu potřebujete.

Pokud narazíte na nějaké problémy během nebo po instalaci TortoiseSVN nejdříve si prosím prohlédněte [A – „Často kladené otázky \(FAQ\)“](#).

1.4.2. Instalace

TortoiseSVN přichází se snadno použitelným instalátorem. Dvakrát klikněte na instalační soubor a postupujte podle pokynů. Instalátor se postará o zbytek.



Důležité

K instalaci TortoiseSVN musíte mít správcovská oprávnění.

1.4.3. Jazykové balíčky

Uživatelské rozhraní TortoiseSVN bylo přeloženo do mnoha různých jazyků, takže máte možnost stáhnout si jazykový balíček, který vyhovuje Vaším potřebám. Jazykové balíčky najdete na naší stránce [translation - stav překladů](http://tortoisesvn.net/translation_status) [http://tortoisesvn.net/translation_status]. A pokud není Váš jazykový balíček prozatím k dispozici, proč se nepřipojit k týmu a neposlat svůj vlastní překlad ;-)

Každý jazykový balíček je zabalen jako EXE instalátor. Stačí spustit instalační program a postupujte podle pokynů. V následujícím spuštění bude překlad k dispozici.

1.4.4. Kontrola pravopisu

TortoiseSVN obsahuje i kontrolu pravopisu, která Vám umožní kontrolovat vaše záznamy logu odevzdání. To je užitečné zejména v případě, když jazyk projektu není váš rodný jazyk. Kontrola pravopisu používá stejné soubory slovníku jako [OpenOffice](http://openoffice.org) [http://openoffice.org] a [Mozilla](http://mozilla.org) [http://mozilla.org].

Instalátor automaticky přidá anlické slovníky USA a Velké Británie. Pokud chcete další jazyky, nejjednodušší možností je si nainstalovat jeden z jazykových balíčků TortoiseSVN. Tímto se nainstaluje příslušné soubory slovníku, stejně jako lokalizované uživatelské rozhraní TortoiseSVN. Po restartování, bude slovník také k dispozici.

Nebo si můžete nainstalovat slovníky sami. Pokud máte nainstalovaný OpenOffice nebo Mozillu, můžete zkopírovat tyto slovníky, které jsou umístěny v instalační složce těchto aplikací. V opačném případě je potřeba stáhnout potřebné soubory slovníku z <http://wiki.services.openoffice.org/wiki/Dictionaries>

Jakmile máte soubory se slovníkem, pravděpodobně je budete muset přejmenovat tak, aby názvy souborů měli pouze název jazyka. Příklad:

- en_US.aff
- en_US.dic

Pak je stačí zkopírovat do podadresáře bin instalace TortoiseSVN. Běžně to je C:\Program Files\TortoiseSVN\bin. Pokud si nechcete, zaneřadit podadresář bin, můžete místo toho umístit vaše soubory pro kontrolu pravopisu do C:\Program Files\TortoiseSVN\Languages. Pokud tato složka neexistuje, musíte ji nejdřív vytvořit. Při příštím spuštění TortoiseSVN, bude kontrola pravopisu k dispozici.

Pokud nainstalujete více slovníků, TortoiseSVN používá tato pravidla pro výběr, které z nich použít.

1. Podívejte se na nastavení `tsvn:projectlanguage`. Podívejte se na [4.17 – „Nastavení projektu“](#) pro informace o nastavení vlastností projektu.
2. Není-li nastaven žádný jazyk projektu, nebo tento jazyk není nainstalován, zkuste jazyk odpovídající locale Windows.

3. Pokud stejný jazyk jako jazyk Windows nefunguje, zkuste „Základní“ jazyk, např. de_CH (Švýcarsko-německý) spadá do de_DE (Německý).
4. Pokud nic z výše uvedeného nefunguje, pak je výchozím jazykem angličtina, která je součástí standardní instalace.

Kapitola 2. Základní koncepty Kontroly Verzí

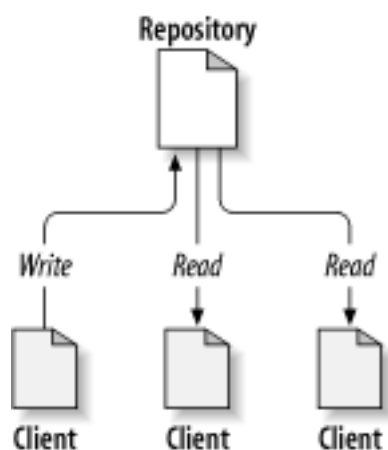
Tato kapitola je mírně upravená verze téže kapitoly v knize Subversion. On-line verze knihy Subversion je k dispozici zde: <http://svnbook.red-bean.com/>.

Tato kapitola je krátký, neformální úvod do Subversion. Pokud nejste obeznámeni se správou verzí, tato kapitola je určitě pro Vás. Začneme diskusí o obecných představách o správě verzí, propracujeme se ke konkrétním myšlenkám Subversion, a ukážeme si některé jednoduché příklady použití Subversion.

I přesto, že příklady uvedené v této kapitole ukazují lidi sdílející sbírky zdrojového kódu programu, mějte na paměti, že Subversion může spravovat jakýkoli druh souborů - není omezen na pomoc programátorům.

2.1. Úložiště

Subversion je centralizovaný systém pro sdílení informací. Jejím jádrem je *úložiště*, které je centrálním skladištěm dat. Úložiště uchovává informace ve formě *stromu souborového systému* - typická hierarchie souborů a adresářů. *Libovolný počet klientů* se připojí k úložišti, a pak te nebo zapisuje do těchto souborů. Zapisováním dat klient zpřístupňuje informaci ostatním, čtením dat, klient od ostatních informace obdrží.



Obrázek 2.1. Typický klient/server systém

Tak proč je to zajímavé? Zatím to zní jako definice typického souborového serveru. A skutečně, úložiště je typ souborového serveru, ale není to jen tak nějaký souborový server. Co dělá úložiště Subversion zvláštní je, že *si pamatuje každou změnu* která kdy byla na něj vložena: každou změnu každého souboru, a dokonce i změny v samotném stromovém adresáři, jako je přidávání, mazání a přeskupování souborů a adresářů.

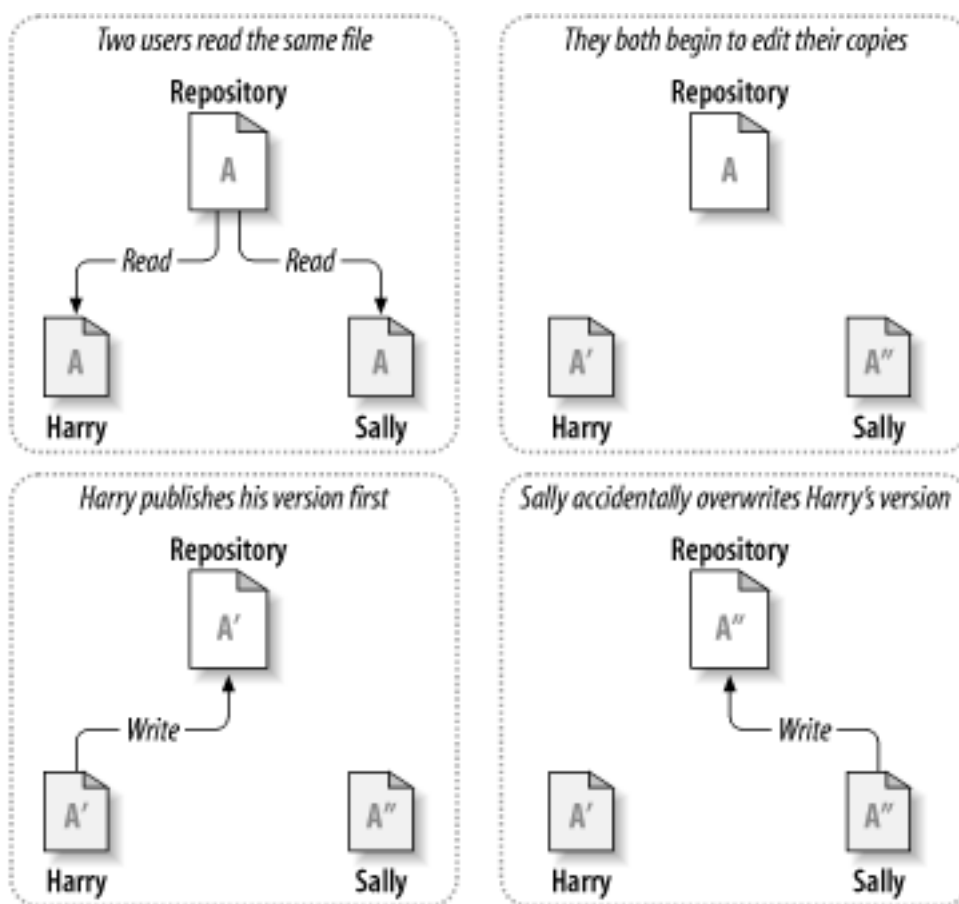
Když klient čte data z úložiště, obvykle vidí pouze poslední verzi stromu souborového systému. Ale klient má také možnost zobrazit *předchozí stavy souborového systému*. Například, klient může položit historickou otázku jako: „co tento adresář obsahoval minulou středu?“, nebo „kdo byl poslední, kdo změnil tento soubor, a jaké změny udělali?“ To jsou druhy otázek, které jsou srdcem každého *systému pro správu verzí*: systémy, které jsou navrženy tak, aby uchovávaly a sledovaly změny údajů v čase.

2.2. Modely Verzování

Všechny systémy pro správu verzí musí řešit stejný základní problém: jak se systém, který umožňuje uživatelům sdílet informace, bude bránit proti nechtěnému šlápnutí na nohu toho druhého? Je až příliš snadné, aby si uživatelé navzájem přepsali změny v úložišti.

2.2.1. Problém sdílení souborů

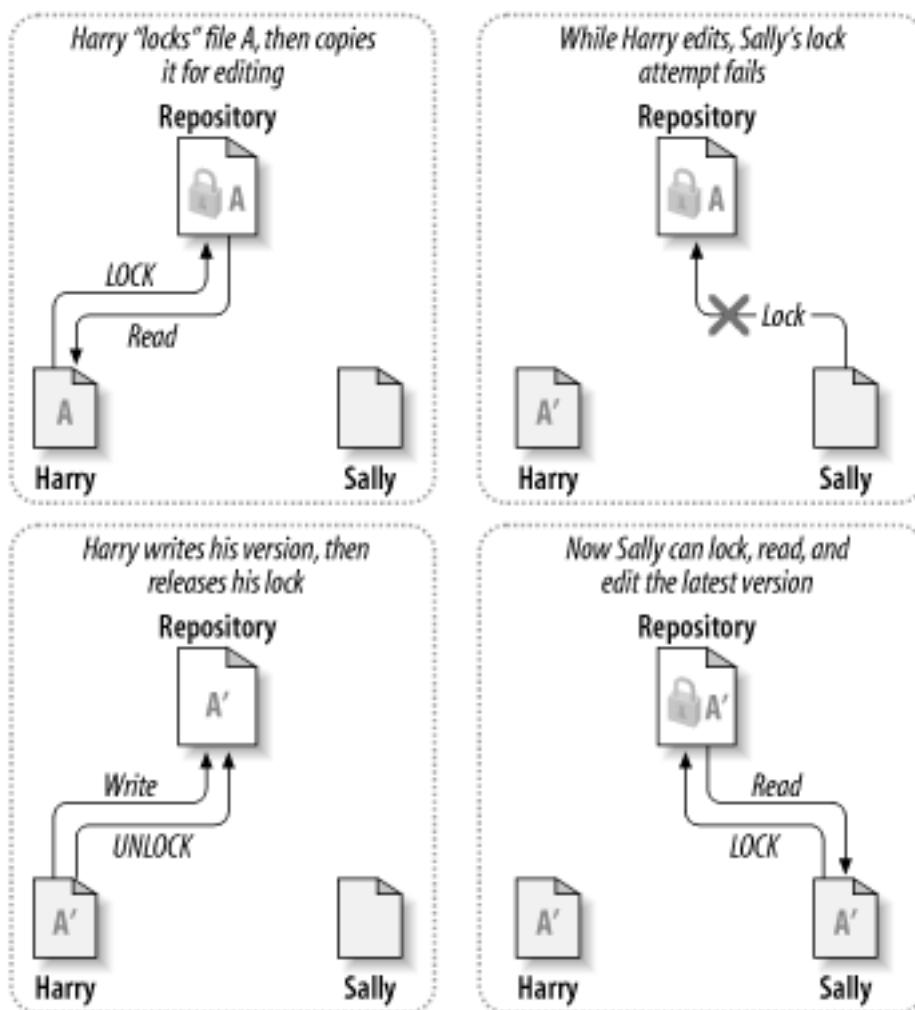
Vezměme si tento scénář: Předpokládejme, že máme dva spolupracovníky, Lojzu a Lízu. Každý z nich se rozhodne upravit stejné úložiště souborů současně. Pokud si Lojza uloží své změny úložiště první, pak je možné, že (o několik okamžiků později) Líza je mohla omylem přepsat svoji vlastní novou verzí souboru. Zatímco Lojzova verze souboru nebude navždy ztracena (protože systém si pamatuje každou změnu), všechny změny, které provedl Lojza *nebudou* přítomny v Lízině novější verzi souboru, protože nikdy neviděla Lojzovy změny. Lojzova práce je stále ztracena - nebo alespoň chybí v nejnovější verzi souboru - a pravděpodobně omylem. To je určitě situace, které se chceme vyhnout!



Obrázek 2.2. Problém, kterému je třeba se vyvarovat.

2.2.2. Řešení pomocí modelu Zamknout-Změnit-Odemknout

Mnoho systémů pro správu verzí používá model *Zamknout-Upravit-Odemknout* pro řešení tohoto problému, což je velmi jednoduché řešení. V takovém systému úložiště umožňuje pouze jedné osobě provést změnu jednoho souboru najednou. Nejdříve Lojza musí *zamknout* soubor dříve, než do něho může začít dělat změny. Zamykání souboru je podobné jako vypůjčit si knihu z knihovny, když Lojza uzamkl soubor, Líza nemůže provádět žádné změny. Pokud se pokusí zamknout soubor, úložiště požadavek zamítne. Jediné, co může udělat, je otevřít si soubor pro čtení, a čekat až Lojza dokončí změny a zámek odemkne. Poté, co Lojza odemkne soubor, je na řadě Líza, která si může soubor zamknout a upravit ho.



Obrázek 2.3. Řešení pomocí modelu Zamknout-Změnit-Odemknout

Problém s modelem zamknout-změnit-odemknout je, že je trochu omezující, a často se stává překážkou pro uživatele:

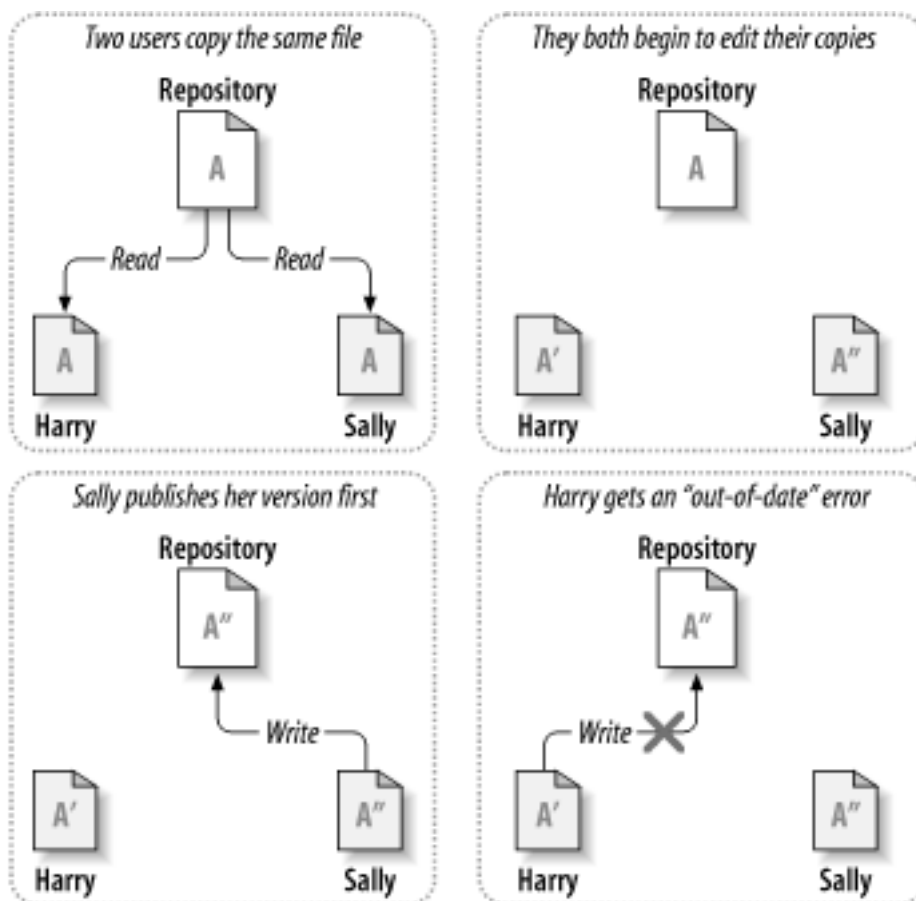
- *Uzamknutí může způsobit administrativní problémy.* Někdy Lojza zamkne soubor a pak na něj zapomene. Mezitím Líza, protože stále čeká na příležitost upravit soubor, má svázané ruce. A pak Lojza odjede na dovolenou. Nyní Líza musí požádat správce k odemknutí Lojzova zámku. Situace tak způsobila spoustu zbytečných průtahů a ztracený čas.
- *Zamykání může způsobit zbytečné serializace.* Co když Lojza upravuje začátek textového souboru, a Sally chce upravovat pouze konec stejného souboru? Tyto změny se vůbec nepřekrývají. Snadno by mohli upravovat soubor současně a žádné velké škody by nenastaly, za předpokladu, že změny byly řádně sloučeny. Není třeba, aby se v této situaci střídali.
- *Zamykání může vytvořit falešný pocit bezpečí.* Předstírejme, že Lojza uzamkl a upravuje soubor A, a současně Líza zamkla a upravuje soubor B. Předpokládejme, že A i B jsou na sobě závislé a změny provedené v obou souborech jsou sémanticky neslučitelné. Najednou A i B už spolu nefungují. Systém zámků nedokázal zabránit problému - přesto nějak poskytl pocit falešného bezpečí. Je snadné pro Lojzu a Lízu si představit, že zamknutím souborů, každý začíná bezpečný, izolovaný úkol, a tím jim už od začátku bylo zabráněno prodiskutovat jejich nekompatibilní změny.

2.2.3. Řešení pomocí modelu Zkopírovat-Změnit-Sloučit

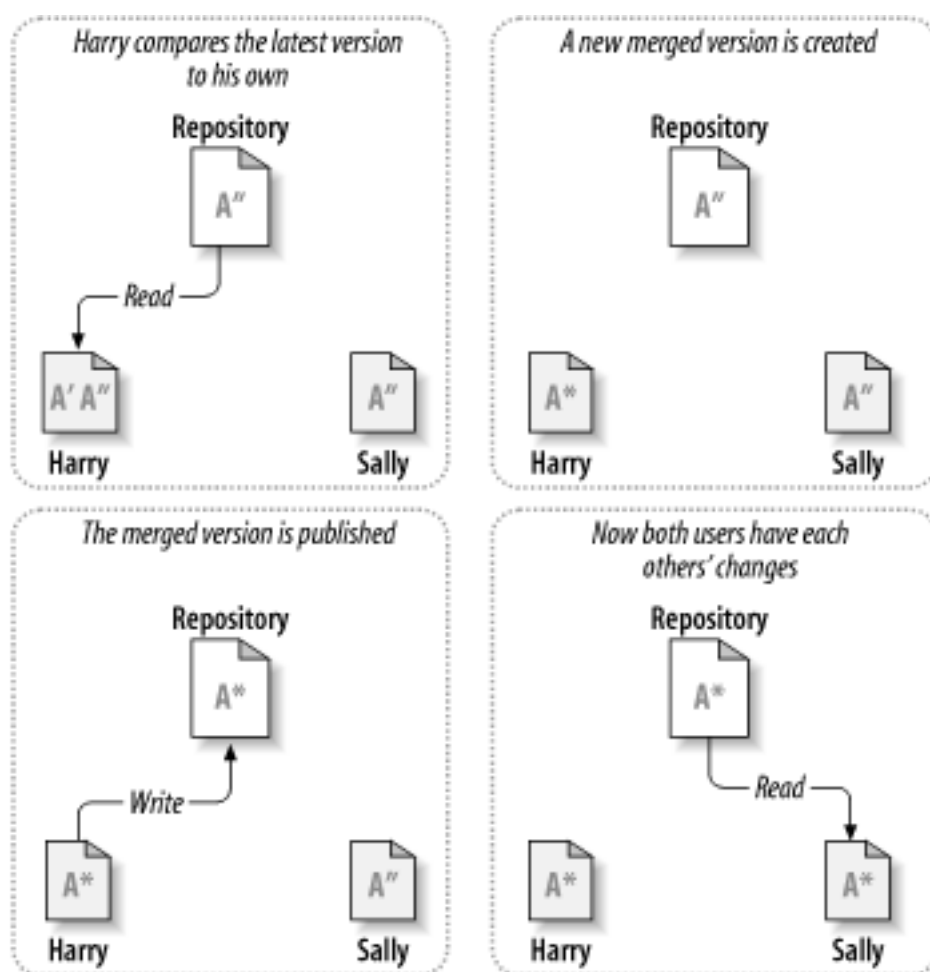
Subversion, CVS, a další systémy pro správu verzí používají model *Kopírovat-Upravit-sloučit* jako alternativa k zamykání. V tomto modelu, každý klient uživatele čte úložiště a vytvoří osobní pracovní

kopii ze souboru nebo projektu. Uživatelé pak pracují paralelně, a mění svoje vlastní soukromé kopie. Nakonec soukromé kopie jsou sloučeny do nové, konečné verze. Systém pro správu verzí často pomáhá se slučováním, ale v konečném důsledku je člověk odpovědný za to, aby vše proběhlo v pořádku.

Zde je příklad. Řekněme, že Lojza a Lída si každý vytvořil pracovní kopii stejného projektu, kopírovanou z úložiště. Pracují současně, a provádějí změny ve stejném souboru A v rámci jejich kopií. Lída uloží změny do úložiště jako první. Když se Lojza pokusí uložit své změny později, úložiště ho informuje, že jeho soubor A je *zastaralý*. Jinými slovy, tento soubor v repozitáři se od posledního zkopírování nějak změnil. Takže Lojza požádá svého klienta, aby *sloučil* všechny nové změny z úložiště do jeho pracovní kopii souboru A. Je zde šance, že Líziny změny se nepřekrývají s jeho, takže jakmile má obě sady změn integrovány, si uloží svou pracovní kopii zpět do repozitáře.



Obrázek 2.4. Řešení pomocí modelu Zkopírovat-Změnit-Sloučit



Obrázek 2.5. ... Pokračování Kopírovat-Změnit-Sloučit

Ale co když Líziny změny se *překrývají* s Lojzovými? Co pak? Tato situace se nazývá *konflikt*, A obvykle to není velký problém. Když Lojza požádá svého klienta o sloučení poslední změny úložiště do jeho pracovní kopie, jeho kopie souboru A je nějakým způsobem označena, že je ve stavu konfliktu: bude moci vidět obě sady konfliktních změn, a ručně si mezi nimi vybrat. Všimněte si, že software nemůže automaticky vyřešit konflikty, pouze lidé jsou schopni porozumět a učinit nezbytné inteligentní volby. Poté, co Lojza ručně vyřešil překrývající se změny (možná po diskuzi o konfliktu s Lízou!), může bezpečně uložit sloučený soubor zpět do repozitáře.

Model Kopírovat-Upravit-Sloučit může znít trochu chaoticky, ale v praxi je to velmi lehké. Uživatelé mohou pracovat paralelně, nikdo nečeká na někoho jiného. Když se pracuje na stejném souboru, nakonec se ukáže, že většina jejich souběžných změn se vůbec nepřekrývá; konflikty se dějí vzácně. A doba potřebná k řešení těchto konfliktů, je mnohem menší, než doba ztracená při zamknutí systému.

Na konec všechno záleží na jednom zásadním faktoru: komunikace mezi uživateli. **Když uživatelé špatně komunikují, jak syntaktické tak sémantické konflikty se zvyšují.** Žádný systém nemůže nutit uživatele komunikovat dokonale a žádný systém nemůže odhalit sémantické konflikty. Takže nemá smysl ukolébat se falešným slibem, že zamykání bude nějak předcházet konfliktům; v praxi, zamykání brzdí produktivitu víc, než cokoliv jiného.

Je jedna častá situace, kdy model Uzamknout-Změnit-Odemknout je lepší a to, když máte neslučitelné soubory. Například, pokud vaše úložiště obsahuje některé grafické obrázky a dva lidé změní obrázek ve stejnou chvíli, neexistuje žádný způsob, jak tyto změny mohou být sloučeny. Buď Lojza nebo Liza ztratí svoji změnu.

2.2.4. Co Subversion dělá?

Subversion používá model Kopírovat-Upravit-Sloučit jako výchozí nastavení a v mnoha případech to je vše, co budete potřebovat. Nicméně, od verze 1.2, Subversion také podporuje zamykání souborů, takže pokud máte neslučitelné soubory, nebo pokud jste prostě nuceni do zásady řízení zamykání, Subversion Vám stále bude poskytovat funkce, které potřebujete.

2.3. Subversion v akci

2.3.1. Pracovní kopie

Už jste četli o pracovních kopiích, nyní budeme demonstrovat, jak je klient Subversion vytváří a využívá.

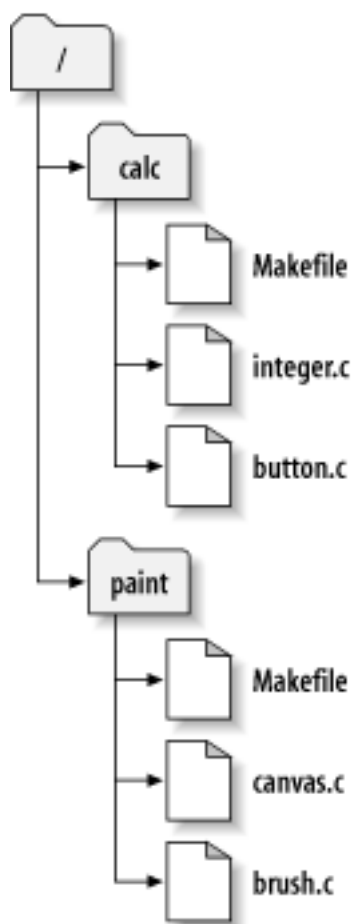
Pracovní kopie Subversion je obyčejný stromový adresářů na vašem lokálním systému, který obsahuje sadu souborů. Tyto soubory můžete upravovat, jak chcete a pokud jsou to soubory zdrojového kódu, je můžete z nich i kompilovat svůj program obvyklým způsobem. Vaše pracovní kopie je Váš vlastní soukromý pracovní prostor: Subversion nikdy nevčlení změny jiných lidí, ani nezpřístupní Vaše vlastní změny ostatním, pokud si to výslovně nepřejete.

Poté, co jste provedli nějaké změny v souborech v pracovní kopii a ověřili jste, že správně fungují, Subversion vám poskytuje příkazy jak *zveřejnit* Vaše změny pro ostatní, kteří s Vámi spolupracují na tomto projektu (zápisem do úložiště). Pokud ostatní zveřejní své vlastní změny, Subversion Vám nabízí příkazy sloučit tyto změny do Vašeho pracovního adresáře (čtením z úložiště).

Pracovní kopie také obsahuje i některé další soubory, vytvořené a udržované Subversion, na pomoc při provádění těchto příkazů. Zejména každý adresář v pracovní kopii obsahuje podadresář `.svn`, také známý jako pracovní kopie *správcovského adresáře*. Soubory v každém správčovském adresáři pomáhají Subversion poznat, které soubory obsahují nepublikované změny a soubory, které jsou zastaralé s ohledem na práci ostatních.

Typické úložiště Subversion často obsahuje soubory (nebo zdrojový kód) několika projektů; obvykle, každý projekt je podadresář v souborovém systému stromu úložiště. V tomto uspořádání bude pracovní kopie uživatele obvykle odpovídat konkrétnímu podstromu úložiště.

Předpokládejme například, že máte úložiště, které obsahuje dva softwarové projekty.



Obrázek 2.6. Systém souborů uložště

Jinými slovy, kořen adresáře úložiště má dva podadresáře: *barva* a *kalkul*.

To get a working copy, you must *check out* some subtree of the repository. (The term *check out* may sound like it has something to do with locking or reserving resources, but it doesn't; it simply creates a private copy of the project for you).

Předpokládejme, že provedete změny v `tlacitko.c`. Vzhledem k tomu, že adresář `.svn` si pamatuje datum změny souboru a původní obsah, Subversion může zjistit, že jste změnili soubor. Nicméně, Subversion změny veřejnými, dokud mu to výslovně neřeknete. Akt publikování změn je více znám jako *odevzdání* (nebo *zapsání*) změn do úložiště.

Chcete-li publikovat své změny pro ostatní, můžete použít Subversion příkaz **odevzdat**.

Nyní Vaše změny v `tlacitko.c` byly odevzdány do úložiště, pokud jiný uživatel získá pracovní kopii `/Kalkul`, uvidí Vaše změny v nejnovější verzi souboru.

Předpokládejme, že máte spolupracovníka, Lízu, která získala pracovní kopii `/Kalk` ve stejnou dobu jako Vy. Když jste odevzdal své změny `tlacitko.c`, Lízina pracovní kopie je ponechána beze změny; Subversion mění pracovní kopie pouze na žádost uživatele.

Aby její projekt byl aktuální, Líza může požádat Subversion o *aktualizaci* její pracovní kopie, pomocí Subversion příkazu **aktualizovat**. To zahrne Vaše změny do její pracovní kopie, jakož i změny všech ostatních, které byly odevzdány, od doby získání kopie.

Všimněte si, že Líza nemusela určit, které soubory se mají aktualizovat; Subversion používá informace v adresáři `.svn` a další informace v úložišti, k určení, které soubory je třeba aktualizovat.

2.3.2. URL úložiště

K úložištím Subversion lze přistupovat pomocí mnoha různých metod - na lokálním disku, nebo prostřednictvím různých síťových protokolů. Avšak umístění úložiště je vždy URL. Schéma URL udává přístupové metody:

Schéma	Metoda přístupu
file://	Přímý přístup k úložišti na místním nebo síťovém disku.
http://	Přístup přes protokol WebDAV na Apache server se Subversion
https://	Stejně jako http://, ale s šifrováním pomocí SSL.
svn://	Neautorizovaný přístup TCP/IP přes vlastní protokol na server svnserve.
svn+ssh://	ověřený, zašifrovaný přístup TCP/IP přes vlastní protokol na server svnserve.

Tabulka 2.1. Přístupová URL úložiště

Většinou, Subversion URL používají standardní syntaxi, která umožňuje jména serverů a čísla portů být zadána jako součást adresy URL. Přístupová metoda `file://` se běžně používá pro lokální přístup, ačkoli může být použita s cestami UNC k síťovému hostiteli. URL tedy nabývá formy `file://hostname/path/to/repos`. Pro místní počítač, je část URL `hostname` buď vynechána, nebo je `localhost`. Z tohoto důvodu místní cesty mají obvykle tři lomítka, `file:///path/to/repos`.

Rovněž uživatelé protokolu `file://` na platformách Windows, budou muset používat neoficiálně „standardní“ syntaxi pro přístup k úložištím, které jsou na stejném počítači, ale na jiném disku, než je klientova aktuální pracovní jednotka. Jeden ze dvou následujících syntaxí cesty URL bude fungovat, když `X` je jednotka, na které je úložiště:

```
file:///X:/path/to/repos - cesta k úložišti
...
file:///X|/path/to/repos - cesta k úložišti
...
```

Všimněte si, že URL používá obyčejná lomítka, i když nativní (ne URL) forma cesty v systému Windows používá zpětná lomítka.

Můžete bezpečně vstupovat do FSFS úložiště přes sdílenou síť, ale stejným způsobem *nemůžete* vstupovat do BDB úložiště.



Varování

Nevytvářejte ani nevstupujte do Berkeley DB úložiště přes sdílenou síť. Úložiště *nemůže* existovat ve vzdáleném systému souborů. Ani když máte mapovanou síťovou jednotku na písmeno jednotky. Pokud se pokusíte použít Berkeley DB v sdílené síti, výsledky jsou nepředvídatelné - divné chyby se mohou objevit ihned, nebo to také může trvat měsíce, než zjistíte, že Vaše databáze úložiště je nepatrně poškozená.

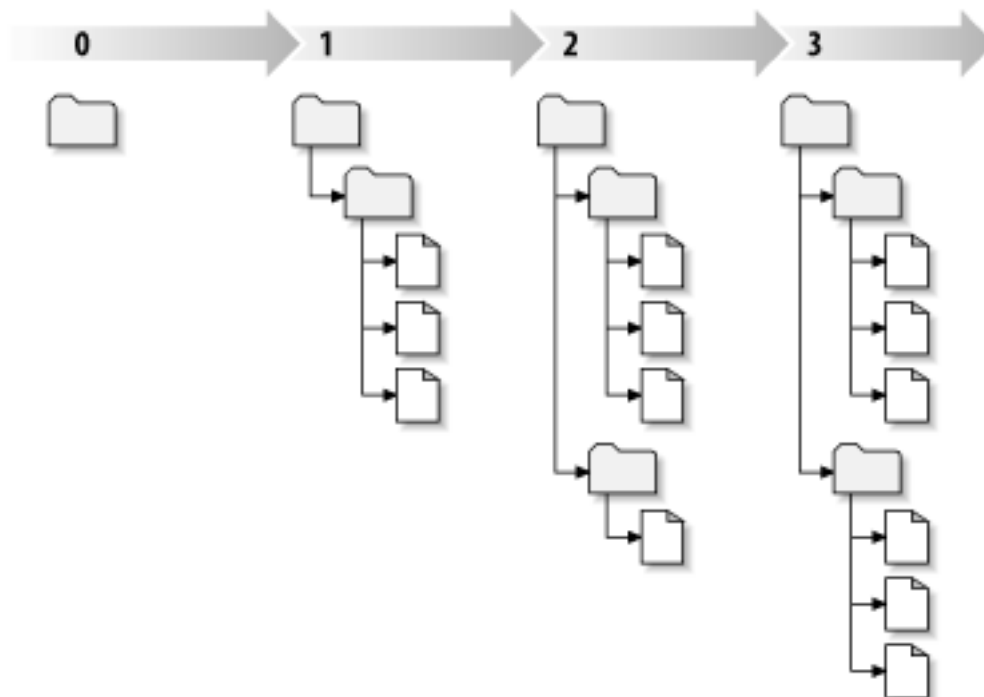
2.3.3. Revize

Operace **odevzdání** může publikovat změny v libovolném počtu souborů a adresářů jako jediný atomový přenos. Ve vaší pracovní kopie, můžete měnit obsah souborů, vytvořit, smazat, přejmenovávat a kopírovat soubory a adresáře a pak odevzdat kompletní sadu změn jako celek.

V úložišti, každé odevzdání se považuje za atomové transakce: buď dojde ke všem změnám v odevzdání, nebo nedojde k žádné. Subversion zachovává tuto atomizaci navzdory pádům programu, zhroucení systému, problémy se sítí, a činnostmi jiných uživatelů.

Pokaždé, když úložiště přijme odevzdání, vytvoří se nový stav stromu souborového systému, který se nazývá *revize*. Ke každé revizi je přiřazeno unikátní přirozené číslo, o jedno větší než předchozí revize. Výchozí revize z čerstvě vytvořeného úložiště je očíslována nulou a neobsahuje nic kromě kořenového adresáře.

Hezký způsob, jak si úložiště představit je jako řadu stromů. Představte si řadu revizí čísel od 0, rozprostírající se zleva doprava. Každá revize má číslo stromu souborového systému visící pod ním a každý strom je „snímek“ úložiště jak vypadalo po každém odevzdání.



Obrázek 2.7. Úložiště

Globální Číslo Revizí

Na rozdíl od těch mnoha jiných systémů pro správu verzí, revizní čísla Subversion se vztahují na *celé stromy*, ne na jednotlivé soubory. Každé revizní číslo vybere celý strom, konkrétní stav úložiště po odevzdání nějaké změny. Další způsob, jak si to představit je, že revize N reprezentuje stav souborového systému úložiště po Ntém odevzdání. Když uživatel Subversion mluví o ``5. revizi bla.c'', myslí tím jak vypadá ``bla.c v 5. revizi.'' Nezapomeňte, že obvykle revize N a M souboru se *nemusí* nutně lišit!

Je důležité si uvědomit, že pracovní kopie ne vždy odpovídají každé jednotlivé revizi v úložišti; mohou obsahovat soubory z několika různých revizí. Například předpokládejme, že byste získali pracovní kopii z úložiště, jejíž nejnovější revize je 4:

```
kalkul/Makefile:4
    integer.c:4
    tlacitko.c:4
```

V současné době tento pracovní adresář, přesně odpovídá revizi 4 v úložišti. Předpokládejme však, že uděláte změnu v `tlacitko.c` a odevzdáte tuto změnu. Za předpokladu, že k žádným jiným odevzdáním nedošlo, Vaše odevzdání vytvoří 5. revizi úložiště, a vaše pracovní kopie bude nyní vypadat takto:

```
kalkul/Makefile:4
integer.c:4
tlacitko.c:5
```

Předpokládejme, že v tomto okamžiku, Líza odevzdá změnu `integer.c` a tím vytvoří revizi 6. Pokud používáte **svn update**, aby vaše pracovní kopie zůstala aktuální, pak to bude vypadat takto:

```
kalkul/Makefile:6
integer.c:6
tlacitko.c:6
```

Líziny změny `integer.c` se objeví ve Vaší pracovní kopii a vaše změny stále budou přítomny v `tlacitko.c`. V tomto příkladu, text `Makefile` je totožný s revizí 4, 5 a 6, ale Subversion označí vaši pracovní kopii `Makefile` s revizí 6 aby naznačil, že je stále aktuální. Takže poté, co provedete čistou aktualizaci v nejvyšší části Vaší pracovní kopie, bude zpravidla odpovídat přesně jedné revizi v úložišti.

2.3.4. Jak Pracovní Kopie Sledují Úložiště

U každého souboru v pracovním adresáři, Subversion zaznamenává dva základní druhy informací ve správních oblastech `.svn/`:

- na jaké revizi je Váš pracovní soubor založen (toto se nazývá *pracovní revize* souboru) a
- časové razítko ukazující, kdy byla místní kopie naposledy aktualizována úložištěm.

Vzhledem k této informaci, mluvením s úložištěm, může Subversion říct, ve kterém z těchto čtyř stavů se pracovní soubor nachází:

Beze změny, a aktuální

Soubor je v pracovním adresáři nezměněn a žádné změny tohoto souboru nebyly odevzdány do úložiště od jeho pracovní verze. **Odevzdáním** tohoto souboru nebo **aktualizací** tohoto souboru se nic nestane.

Změněn lokálně a aktuální

Soubor byl změněn v pracovním adresáři a žádné změny do tohoto souboru nebyly odevzdány do úložiště od jeho počáteční revize. Existují lokální změny, které nebyly odevzdány do úložiště, tudíž **odevzdání** souboru úspěšně publikuje Vaše změny a **aktualizace** tohoto souboru nic neprovede.

Beze změny a zastaralé

Soubor nebyl změněn v pracovním adresáři, ale byl změněn v úložišti. Soubor by měl být časem aktualizován, aby byl aktuální s veřejnou revizí. **Odevzdání** tohoto souboru nic neprovede a **aktualizace** tohoto souboru zahrne poslední změny do Vaší pracovní kopie.

Změněn lokálně a zastaralý

Soubor byl změněn a to jak v pracovním adresáři tak i v úložišti. **Odevzdání** tohoto souboru bude hlásit chybu *zastaralé*. Soubor by měl být nejdříve aktualizován; Příkaz **aktualizovat** se pokusí sloučit veřejné změny se změnami místními. Pokud Subversion nemůže dokončit sloučení přijatelným způsobem automaticky, je ponecháno na uživateli, aby vyřešil konflikt.

2.4. Shrnutí

V této kapitole jsme probrali řadu základních pojmů Subversion:

- Zavedli jsme pojmy centrálního úložiště, pracovní kopie klienta, a pole stromových revizí úložiště.
- Viděli jsme několik jednoduchých příkladů, jak dva spolupracovníci mohou využít Subversion pro publikaci a přijetí změn, jeden od druhého, pomocí modelu 'Kopírovat-Upravit-Sloučit'.
- Mluvili jsme o způsobu, jakým Subversion sleduje a spravuje informace v pracovní kopii.

Kapitola 3. Úložiště

Ať používáte jakýkoli protokol pro přístup do úložiště, vždy musíte alespoň jedno vytvořit. To může být provedeno buď z příkazové řádky klienta Subversion nebo z TortoiseSVN.

Pokud jste ještě nevytvořili úložiště Subversion, teď je na to vhodná doba.

3.1. Vytváření úložiště

Můžete vytvořit úložiště s FSFS backendem nebo starší formát Berkeley databáze (BDB). Formát FSFS je obecně rychlejší a snadnější pro správu a funguje jak na sdílené síti tak i na Windows 98 bez problémů. Formát BDB byl kdysi považován za stabilnější prostě proto, že se používal déle, ale protože FSFS se nyní používal v této oblasti již několik let, tento argument je nyní spíše výmluva. Pro více informací si přečtěte [Choosing Výběr datového úložiště](http://svnbook.red-bean.com/en/1.5/svn.reposadmin.planning.html#svn.reposadmin.basics.backends) [http://svnbook.red-bean.com/en/1.5/svn.reposadmin.planning.html#svn.reposadmin.basics.backends] v knize Subversion.

3.1.1. Vytvoření repozitáře z příkazového řádku klienta

1. Vytvořte si prázdnou složku s názvem SVN (např. D:\SVN\), která bude použita jako kořen pro všechny vaše úložiště.
2. Vytvořte si další složku MojeNoveUloziste v D:\SVN\.
3. Spust'ete příkazový řádek (nebo DOS-Box), změny v D:\SVN\ a napište

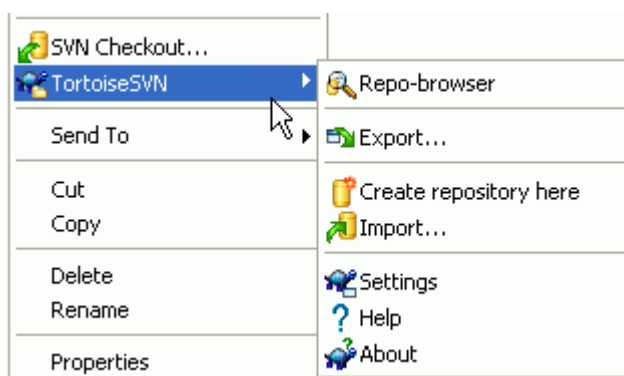
```
svnadmin create --fs-type bdb MojeNoveUloziste
```

nebo

```
svnadmin create --fs-type fsfs MojeNoveUloziste
```

Vaše nové úložiště se teď nachází v D:\SVN\MojeNoveUloziste.

3.1.2. Vytváření úložiště pomocí TortoiseSVN



Obrázek 3.1. Menu TortoiseSVN pro složky bez verzí

1. Otevřete Průzkumníka Windows
2. Vytvořte si novou složku a pojmenujte ji např. SVNUloziste

3. Klikněte pravým tlačítkem na nově vytvořenou složku a vyberte TortoiseSVN → Zde vytvořit úložiště....

Úložiště je poté vytvořeno uvnitř nové složky. *Neupravujte tyto soubory sami!*. Pokud se zobrazí jakékoliv chyby, ujistěte se, že složka je prázdná a není chráněná proti zápisu.



Tip

TortoiseSVN už nenabízí možnost vytvořit BDB úložiště, ale stále můžete použít příkazový řádek klienta k jejich vytvoření. FSFS úložiště jsou pro Vás zpravidla jednodušší na údržbu a také je pro nás snadnější udržovat TortoiseSVN kvůli problémům s kompatibilitou mezi různými verzemi BDB.

Future versions of TortoiseSVN will not support `file://` access to BDB repositories due to these compatibility issues, although it will of course always support this repository format when accessed via a server through the `svn://`, `http://` or `https://` protocols. For this reason, we strongly recommend that any new repository which must be accessed using `file://` protocol is created as FSFS.

Samozřejmě také vůbec nedoporučujeme používat pro přístup `file://`, kromě pro lokální účely testování. Použití serveru je mnohem bezpečnější a spolehlivější pro všechny, až na práci jednoho vývojáře.

3.1.3. Lokální Přístup do Úložiště

Pro přístup do lokálního úložiště je třeba znát cestu k tomuto adresáři. Jen nezapomeňte, že Subversion očekává, že všechny cesty úložiště budou ve formě `file:///C:/SVNUloziste/`. Nezapomeňte všude používat lomítka.

Pro přístup k úložišti nacházející se ve sdílené síti, buď můžete použít mapování disku, nebo UNC cestu. Cesty UNC mají formu `file://JmenoServeru/cesta/do/uloziste/`. Všimněte si, že se zde používají pouze 2 přední lomítka.

Před SVN 1.2, musely být UNC cesty uvedeny v nepřehledné podobě `file:///file:///JmenoServeru/cesta/do/uloziste/`. Tato forma je stále podporována, ale nedoporučuje se.



Varování

Nevytvářejte ani nevstupujte do Berkeley DB úložiště ve sdílené síti. *Nemůže existovat ve vzdáleném systému souborů*, ani když máte síťovou jednotku mapovanou na písmeno jednotky. Pokud se pokusíte použít Berkeley DB ve sdílené síti, výsledky jsou nepředvídatelné - zvláštní chyby se mohou objevit ihned, nebo to může trvat měsíce předtím, než zjistíte, že vaše databáze úložiště je lehce poškozená.

3.1.4. Přístup k úložišti na síťovém disku

I když je teoreticky možné mít FSFS úložiště v sdílené síti a více uživatelů by k němu přistupovalo pomocí protokolu `file://`, toto rozhodně *není* doporučeno. Ve skutečnosti *silně* odrazujeme a takové použití nepodporujeme.

Zaprvé: Každému uživateli dáváte přímý přístup k zápisu do úložiště, takže jakýkoliv uživatel může omylem vymazat celé úložiště nebo nějakým jiným způsobem ho poškodit.

Za druhé, ne všechny síťové protokoly pro sdílení souborů podporují zamykání, které Subversion potřebuje, takže Vaše úložiště může být poškozeno. Nemusí se to stát hned, ale jednoho dne se dva uživatelé pokusí o přístup do úložiště ve stejnou dobu.

Zatřetí: oprávnění k souboru musí být na to nastavena. Možná Vám to projde na sdíleném disku Windows, ale SAMBA je obzvlášť obtížná.

Přístup přes `file://` je určen pouze pro místní přístup jednoho uživatele a to zejména pro účely testování a ladění. Když chcete sdílet úložiště je *opravdu* třeba zřídit si řádný server a vůbec to není tak složité, jak si možná myslíte. Přečtěte si [3.5 – „Přístup do úložiště“](#) pro pokyny k výběru a nastavení serveru.

3.1.5. Rozložení úložiště

Než začnete importovat data do úložiště, měli byste se nejprve zamyslet nad tím, jak chcete Vaše data uspořádat. Pokud použijete jeden z doporučených rozložení, budete to později mít mnohem jednodušší.

Existují některé standardní, doporučené způsoby, jak organizovat úložiště. Většina lidí si vytvoří kmen adresář, který obsahuje „hlavní linku“ vývoje, adresář větve obsahuje kopie větví a adresář značky obsahuje kopie značek. Pokud úložiště obsahuje pouze jeden projekt, pak se často vytváří tyto adresáře nejvyšší úrovně:

```
/kmen  
/větve  
/značky
```

Pokud úložiště obsahuje více projektů, lidé často indexují jejich rozložení podle větve:

```
/kmen/barva  
/trunk/kalkul  
/větve/barva  
/větve/kalkul  
/značky/barva  
/značky/kalkul
```

... nebo podle projektu:

```
/barva/kmen  
/barva/větve  
/barva/značky  
/kalkul/kmen  
/kalkul/větve  
/kalkul/značky
```

Indexování podle projektu má smysl v případě, že projekty spolu úzce nesouvisí a každý z nich je získáván individuálně. Pro související projekty, kde se budete chtít podívat na všechny projekty najednou, nebo tam, kde projekty jsou všechny spojeny v jediném distribučním balíčku, je často lepší indexovat podle větví. Tímto způsobem máte jen jeden kmen pro získání a vztahy mezi podprojekty jsou lépe viditelné.

Pokud přijmete model nejvyšších úrovní `/kmen` `/značky` `/větve`, nemusíme říkat, že musíte zkopírovat celý kmen pro každou větev a značku, a v některých ohledech tato struktura nabízí největší flexibilitu.

Pro nesouvisející projekty byste raději měli použít samostatné úložiště. Když se odevzdají změny, je to číslo revize celého úložiště, které se mění a ne číslo revize projektu. Mít 2 nesouvisející projekty sdílející 1 úložiště může znamenat velké rozdíly v číslech revizí. Projekty Subversion a TortoiseSVN se objevují na stejné adrese hostitele, ale jsou to zcela oddělená úložiště umožňující nezávislý vývoj a žádný zmatek nad čísly sestavení.

Samozřejmě, máte možnost tyto běžná rozložení ignorovat. Můžete si vytvořit jakýkoli druh variace, co bude pro Vás a Váš tým nejlepší. Pamatuje si, že to, co si zvolíte, není trvalý závazek. Svoje úložiště

si můžete uspořádat kdykoliv. Vzhledem k tomu, že větve a značky jsou obyčejné adresáře, může je TortoiseSVN přesunout nebo přejmenovat jak budete chtít.

Přechod z jednoho rozložení na druhé je jen otázkou vydání řady přesunů na straně serveru; Pokud se vám nelíbí, jak jsou věci v úložišti organizovány, stačí jen zahýbat kolem adresáři.

Takže pokud jste si ještě nevytvořili základní adresářovou strukturu uvnitř úložiště, měli byste to hned udělat. Existují dva způsoby, jak toho dosáhnout. Pokud si chcete jednoduše vytvořit strukturu /kmen /značky /větve, můžete použít prohlížeč úložiště pro vytvoření těchto tří adresářů (ve třech samostatných odevzdání). Pokud chcete vytvořit hlubší hierarchii, pak je jednodušší si nejdříve vytvořit adresářovou strukturu na disku a pak importovat v jediném odevzdání takto:

1. vytvořte si novou prázdnou složku na Vašem pevném disku
2. vytvořte si požadovanou strukturu adresáře nejvyšší úrovně v daném adresáři - nevkládejte, prozatím, do něho žádné soubory!
3. import this structure into the repository via a right click on the folder and selecting TortoiseSVN → Import... This will import your temp folder into the repository root to create the basic repository layout.

Nezapomeňte, že název adresáře, který importujete, se nezobrazí v úložišti, pouze jeho obsah. Například, vytvořte následující strukturu adresáře:

```
C:\Temp\Novy\kmen
C:\Temp\Novy\vetve
C:\Temp\Novy\štítky
```

Import C:\Temp\Novy do kořene úložiště, pak bude vypadat takto:

```
/kmen
/větve
/značky
```

3.2. Zálaha Úložiště

Ať používáte jakýkoliv typ úložiště, je životně důležité, abyste prováděli pravidelné zálohy a pak je ověřovali. Pokud server selže, máte možnost přístupu k poslední verzi souborů, ale bez úložiště všechna vaše historie je navždy ztracena.

Nejjednodušší způsob (ale nedoporučuje se) jak zkopírovat adresář úložiště na záložní médium. Nicméně, musíte si být naprosto jisti, že žádný proces, nepřistupuje k datům. V této souvislosti přístup znamená *jakýkoliv* přístup. Do BDB úložiště se zapisuje i když se zdá, že je pouze vyžadováno čtení např. získání stavu. Pokud je na Vaše úložiště vstoupeno kdykoli v průběhu kopírování, (webový prohlížeč zůstane otevřený, WebSVN, atd.), záloha bude bezcenná.

Doporučená metoda je spustit

```
svnadmin hotcopy cesta/k/úložišti cesta/k/záloze --clean-logs
```

k vytvoření kopie Vašeho úložiště bezpečným způsobem. Pak zálohujte kopii. Příkaz `--clean-logs` není nutný, ale odstraní všechny nadbytečné záznamy souborů při zálohování BDB úložiště, které můžou ušetřit kapacitu.

svnadmin je nástroj automaticky nainstalovaný při instalaci příkazového řádku klienta Subversion. Pokud instalujete nástroje příkazového řádku na Windows PC, nejlepší způsob je stáhnout si verzi

Windows Installer. Je lépe komprimován než verze .zip, takže má menší velikost a postará se o nastavení cest. Nejnovější verzi si můžete stáhnout pomocí příkazové řádky klienta Subversion <http://subversion.apache.org/getting.html>.

3.3. Hook skripty na straně serveru

Hákový skript je program spouštěný při nějaké události úložiště, jako je například vytvoření nové revize nebo změna vlastnosti bez verze. Každému háku je předán dostatek informací, pro zjištění, co je to za událost, jaké jsou cíle() operace a uživatelské jméno osoby, která událost spustila. V závislosti na výstupu háku či návratovém kódu, může hákový program pokračovat v činnosti, zastavit nebo ji nějakým způsobem pozastavit. Více informací naleznete v kapitole týkající se *Hákové skripty* [<http://svnbook.red-bean.com/en/1.5/svn.reposadmin.create.html#svn.reposadmin.create.hooks>] v Subversion Book pro podrobnosti o hácích, které jsou implementovány.

Tyto hákové skripty jsou spouštěny serverem, který je hostitelem úložiště. TortoiseSVN také umožňuje nastavit hákové skripty na straně klienta, které jsou spouštěny lokálně během určitých událostí. Navštivte [4.30.8 – „Hákové Skripty na Straně Klienta“](#) pro více informací.

Sample hook scripts can be found in the `hooks` directory of the repository. These sample scripts are suitable for Unix/Linux servers but need to be modified if your server is Windows based. The hook can be a batch file or an executable. The sample below shows a batch file which might be used to implement a pre-revprop-change hook.

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

Note that anything sent to stdout is discarded. if you want a message to appear in the Commit Reject dialog you must send it to stderr. In a batch file this is achieved using `>&2`

3.4. Odkazy pro Získání

Pokud chcete, aby Vaše úložiště Subversion bylo k dispozici ostatním, asi byste měli vložit na něj odkaz z Vašich internetových stránek. Jeden způsob, jak toto lépe zpřístupnit je začlenit *odkaz pro získání* ostatním uživatelům TortoiseSVN.

Při instalaci TortoiseSVN se zaregistruje nový protokol `tsvn:`. Když uživatel TortoiseSVN na takový odkaz klikne, automaticky se otevře dialogové okno získání s URL adresou úložiště již vyplněnou.

Chcete-li zahrnout takovéto spojení do Vaší vlastní html stránky, budete muset přidat kód, který vypadá nějak takto:

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

Of course it would look even better if you included a suitable picture. You can use the *TortoiseSVN logo* [<http://tortoisesvn.tigris.org/images/TortoiseCheckout.png>] or you can provide your own image.

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

Můžete také odkázat na konkrétní revizi, například

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


3.5. Přístup do úložiště

Chcete-li používat TortoiseSVN (nebo jakéhokoliv jiného klienta Subversion), budete potřebovat místo, kde se budou nacházet Vaše úložiště. Můžete buď Vaše úložiště mít uložená lokálně a přistupovat k nim pomocí protokolu `file://` nebo je můžete umístit na server a přistupovat k nim pomocí protokolů `http://` nebo `svn://`. Tyto dva serverové protokoly mohou být také zašifrovány. Používáte k tomu `https://` nebo `svn+ssh://`, nebo můžete použít `svn://` a SASL.

Pokud používáte veřejnou hostingovou službu, jako je *Google Code* [<http://code.google.com/hosting/>], nebo Váš server byl již nastaven někým jiným pak už nemusíte dělat nic jiného. Pokračujte dál na **4 – „Příručka Každodenního Použití“**.

Pokud nemáte server a budete pracovat sami, nebo právě hodnotíte Subversion a TortoiseSVN v soukromí, pak místní úložiště je pro Vás asi nejlepší volba. Stačí vytvořit úložiště na vlastním PC, jak bylo předtím popsáno v **3 – „Úložiště“**. Můžete přeskočit zbytek této kapitoly a hned jít na **4 – „Příručka Každodenního Použití“** abyste zjistili, jak úložiště začít používat.

Pokud jste uvažovali o vytvoření úložiště ve sdílené síti s více uživateli, pořádně se zamyslete. Přečtěte si **3.1.4 – „Přístup k úložišti na síťovém disku“** abyste zjistili, proč si myslíme, že je to špatný nápad. Nastavit server není tak těžké, jak se zdá a poskytne Vám lepší spolehlivost a asi i rychlost.

Další sekce jsou průvodci krok za krokem, jak si můžete nastavit takový server na počítači se systémem Windows. Samozřejmě také můžete nastavit server na počítači Linux, ale to je nad rámec této příručky. Podrobnější informace o možnostech serveru Subversion a jak si vybrat ty nejlepší architektury pro vaši situaci, lze nalézt v knize Subversion v *Server - Konfigurace serveru* [<http://svnbook.red-bean.com/en/1.5/svn.serverconfig.html>].

3.6. Server Svnserve

3.6.1. Úvod

Subversion obsahuje svnserve - lehký samostatný server, který používá vlastní protokol nad běžným TCP/IP spojením. Je ideální pro menší zařízení, nebo v případě kdy plnohodnotný Apache server nemůže být použit.

Ve většině případů je svnserve snadnější nastavit a běží rychleji než Apache server, ačkoli nemá některé pokročilé funkce. A teď, když je jeho součástí i podpora SASL je snadné ho i zabezpečit.

3.6.2. Instalace svnserve

1. Získejte nejnovější verzi Subversion z <http://subversion.apache.org/getting.html>. Jinak také můžete získat instalační balíčky z CollabNet na <http://www.collab.net/downloads/subversion>. Tento instalátor nastaví svnserve jako službu Windows a také zahrnuje některé nástroje, které potřebujete, pokud budete používat SASL pro zabezpečení.
2. Pokud již máte nainstalovanou verzi Subversion a svnserve běží, budete ho muset před pokračováním instalace zastavit.
3. Spusťte instalační program Subversion. Spouštíte-li instalační program z Vašeho serveru (doporučeno) můžete přeskočit krok 4.
4. Open the windows-explorer, go to the installation directory of Subversion (usually `C:\Program Files\Subversion`) and in the bin directory, find the files `svnserve.exe`, `intl3_svn.dll`, `libapr.dll`, `libapriconv.dll`, `libapriutil.dll`, `libdb*.dll`, `libeay32.dll` and `ssleay32.dll` - copy these files, or just copy all of the bin directory, into a directory on your server e.g. `c:\svnserve`

3.6.3. Spuštění svnservice

Ted', když je svnservice nainstalován, budete potřebovat, aby byl na serveru spuštěn. Nejjednodušší možností je spustit následující z rozhraní DOS nebo vytvořit zástupce:

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

svnservice nyní začne čekat na příchozí požadavky na portu 3690. Příkaz --daemon říká svnservice, že má běžet jako daemon proces, což znamená, že bude vždycky existovat, dokud není ručně vypnut.

Pokud jste ještě nevytvořili úložiště, postupujte podle pokynů nastavení serveru Apache [3.7.4 – „Konfigurace“](#).

Chcete-li testovat, zda svnservice pracuje, použijte TortoiseSVN → Průzkumník úložiště k zobrazení úložiště.

Předpokládejme, že Vaše úložiště se nachází v c:\uloz\TestUloz a Váš server se nazývá localhost, zadejte:

```
svn://localhost/uloz/TestUloz
```

po výzvě průzkumníka úložiště.

Můžete také zvýšit bezpečnost a ušetřit čas zadáváním adresy URL pomocí svnservice a přepínač --root pro nastavení kořenového umístění a omezit přístup do určeného adresáře na serveru:

```
svnservice.exe --daemon --root jednotka:\cesta\do\úložiště\kořen
```

Za použití předchozích testů jako průvodce, svnservice by teď běžel jako:

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

a v TortoiseSVN je naše adresa URL průzkumníka úložiště nyní zkrácena na:

```
svn://localhost/TestUloz
```

Všimněte si, že přepínač --root je také zapotřebí, pokud se Vaše úložiště nachází na jiném oddíle nebo jednotce, než je umístění svnservice na Vašem serveru.

Svnservice může obsluhovat libovolný počet úložišť. Pouze je stačí najít někde pod kořenovým adresářem, který jste právě definovali a vstoupit do nich pomocí adresy URL relativní k tomuto kořenu.



Varování

Nevytvářejte ani nevstupujte do Berkeley DB úložiště přes sdílenou síť. Úložiště *nemůže* existovat ve vzdáleném systému souborů. Ani když máte mapovanou síťovou jednotku na písmeno jednotky. Pokud se pokusíte použít Berkeley DB v sdílené síti, výsledky jsou nepředvídatelné - divné chyby se můžou objevit ihned, nebo to také může trvat měsíce, než zjistíte, že Vaše databáze úložiště je nepatrně poškozená.

3.6.3.1. Spustit svnservice jako Službu

Spouštění svnservice jako uživatele obvykle není nejlepším způsobem. Znamená to, že budete mít na serveru vždy přihlášeného uživatele a pamatovat si ho znovu po restartu spustit po. Lepší způsob jak spustit

svnserve je jako Službu Windows. Počínaje Subversion 1.4, svnserve může být nainstalován jako nativní Služba Windows.

To install svnserve as a native windows service, execute the following command all on one line to create a service which is automatically started when windows starts.

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

If any of the paths include spaces, you have to use (escaped) quotes around the path, like this:

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

You can also add a description after creating the service. This will show up in the Windows Services Manager.

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

Povšimněte si poněkud neobvyklého formátu příkazového řádku používající sc. V páru klíč=hodnota nesmí být žádná mezera mezi klíčem a = ale před hodnotou mezera být musí.



Tip

Microsoft nyní doporučuje služby spustit přes účet Místní nebo Síťové služby. Viz [Průvodce Plánováním Bezpečnosti Služeb a Účtů Služeb](http://www.microsoft.com/technet/security/Plánováním%20Bezpečnosti%20Služeb%20a%20Účtů%20Služeb) [http://www.microsoft.com/technet/security/topics/serversecurity/serviceaccount/default.mspx]. Chcete-li vytvořit službu pod účtem Místní Služby, přidejte následující do výše uvedeného příkladu.

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

Nezapomeňte, že budete muset dát účtu Místní služby příslušná práva pro Subversion i pro Vaše úložiště, stejně jako pro všechny aplikace, které jsou používány háčkovými skripty. Vestavěná skupina pro toto se nazývá "LOCAL SERVICE".

Po instalaci služby, musíte jít do správce služeb k jejímu spuštění (pouze tentokrát; jinak se spouští automaticky při restartu serveru).

Podrobnější informace najdete na [Podpora Služeb Windows ve svnserve](http://svn.collab.net/repos/svn/trunk/notes/windows-service.txt) [http://svn.collab.net/repos/svn/trunk/notes/windows-service.txt].

Pokud jste nainstalovali předchozí verzi svnserve pomocí wrapperu SVNService a nyní místo toho chcete použít nativní podporu budete muset odstranit wrapper jako službu (nezapomeňte nejdřív službu zastavit!). Stačí použít příkaz

```
svnservice -remove
```

k odstranění položky služby z registru.

3.6.4. Základní ověření pomocí svnserve

Výchozí nastavení svnserve poskytuje anonymní přístup pro čtení. To znamená, že můžete použít URL `svn://` pro získání a aktualizaci, nebo pomocí průzkumníka úložiště v TortoiseSVN zobrazit úložiště, ale nebudete schopni odevzdat žádné změny.

Chcete-li povolit přístup do úložiště pro zápis, je třeba upravit soubor `conf/svnserve.conf` v adresáři úložiště. Tento soubor řídí konfiguraci `svnserve` daemonu a také obsahuje užitečnou dokumentaci.

Můžete jednoduše povolit anonymní přístup pro zápis nastavením:

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

Nicméně, nebudete vědět, kdo v úložišti provedl změny, protože vlastnost `svn:author` bude prázdná. Také nebudete moci kontrolovat, kdo dělá změny v úložišti. Toto je poněkud riskantní nastavení!

Jedním způsobem, jak toto překonat je vytvořit databázi hesel:

```
[General]
anon-access = none
auth-access = write
heslo-db = userfile
```

Kde `userfile` je soubor, který existuje ve stejném adresáři jako `svnserve.conf`. Tento soubor může být i jinde ve Vašem souborovém systému (to je užitečné, když máte více úložišť, která vyžadují stejná přístupová práva) a můžete na něj odkazovat pomocí absolutní cesty, nebo cesty relativní vzhledem k adresáři `conf`. Pokud zahrnete cestu, musí být napsána `/Unixovským/Způsobem/`. Používání `\` nebo písmena jednotky nebude fungovat. `userfile` by měl mít strukturu:

```
[users]
uživatelské jméno = heslo
...
```

Tento příklad by zamítnul přístup všem neověřeným (anonymním) uživatelům a dá čist-psát přístup k uživatelům uvedeným v `userfile`.



Tip

If you maintain multiple repositories using the same password database, the use of an authentication realm will make life easier for users, as TortoiseSVN can cache your credentials so that you only have to enter them once. More information can be found in the Subversion book, specifically in the sections [Create a 'users' file and realm](http://svnbook.red-bean.com/en/1.5/svn.serverconfig.svnserve.html#svn.serverconfig.svnserve.auth.users) [http://svnbook.red-bean.com/en/1.5/svn.serverconfig.svnserve.html#svn.serverconfig.svnserve.auth.users] and [Client Credentials Caching](http://svnbook.red-bean.com/en/1.5/svn.serverconfig.netmodel.html#svn.serverconfig.netmodel.credcache) [http://svnbook.red-bean.com/en/1.5/svn.serverconfig.netmodel.html#svn.serverconfig.netmodel.credcache]

3.6.5. Lepší zabezpečení se SASL

3.6.5.1. Co je SASL?

Cyrus Simple Authentication and Security Layer je open source software, který vytvořila universita Carnegie Mellon. Přidává všeobecně použitelné ověřovací a šifrovací funkce pro všechny síťové protokoly a od Subversion 1.5 a novější, jak `svnserve` server tak i klient TortoiseSVN vi, jak použít tuto knihovnu.

Pro obsáhlejší diskuzi o dostupných možnostech, byste se měli podívat v knize Subversion na sekci [Používání svnserve spolu se SASL](http://svnbook.red-bean.com/en/1.5/svn.serverconfig.svnserve.html#svn.serverconfig.svnserve.sasl) [http://svnbook.red-bean.com/en/1.5/svn.serverconfig.svnserve.html#svn.serverconfig.svnserve.sasl]. Pokud jste právě hledali jednoduchý

způsob, jak vytvořit bezpečné ověřování a šifrování na serveru Windows tak, aby přístup do Vašeho úložiště z toho velkého zlého internetu byl bezpečný, tak čtěte dál.

3.6.5.2. Autentizace SASL

Chcete-li aktivovat konkrétní SASL mechanismus na serveru, budete muset udělat tři věci. Nejprve vytvořte sekci [SASL] v souboru `svnserve.conf` ve Vašem úložišti, s tímto párem klíč-hodnota:

```
use-SASL = true
```

Za druhé, vytvořte soubor nazvaný `svn.conf` na vhodném místě - typicky v adresáři, kde je subversion nainstalován.

Za třetí, vytvořte dvě nové položky v registru, aby SASL věděl, kde věci najít. Vytvořte klíč registru s názvem [HKEY_LOCAL_MACHINE\SOFTWARE\Carnegie Mellon\Project Cyrus\SASL Library] a uvnitř umístěte dvě nové řetězcové hodnoty: `SearchPath` nastavte na cestu k adresáři obsahující zásuvné moduly `SASL*.dll` (obvykle v adresáři instalace Subversion) a `ConfFile` nastaven na adresář obsahující soubor `svn.conf`. Pokud jste použili instalátor CollabNet, budou tyto klíče registru již pro Vás vytvořeny.

Upravte soubor `svn.conf`, aby obsahoval následující:

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

Poslední řádek odkazuje na umístění databáze ověření, což je soubor s názvem `sasldb`. Může být kdekoliv, ale nejlepší asi bude hlavní cesta úložiště. Ujistěte se, že služba `svnserve` má povolení číst tento soubor.

Pokud `svnserve` byl již spuštěn, bude ho třeba restartovat, aby se zajistilo, že načte aktualizovanou konfiguraci.

Nyní, když je vše nastaveno, vše, co potřebujete udělat, je vytvořit některé uživatele a hesla. K tomu budete potřebovat program `saslpasswd2`. Pokud jste použili instalátor CollabNet, bude tento program v instalačním adresáři. Jako příkaz použijte například:

```
saslpasswd2 -c -f C:\TortoiseSVN\sasldb -u sféra uživatelské jméno
```

Přepínač `-f` předává umístění databáze, `sféra` musí být stejná jako hodnota, kterou jste definovali v souboru `svnserve.conf` ve Vašem úložišti a uživatelské jméno je přesně to, co čekáte. Nezapomeňte, že `sféra` nesmí obsahovat mezery.

Můžete zobrazit seznam uživatelských jmen uložených v databázi pomocí programu `sasldblistusers2`.

3.6.5.3. SASL Šifrování

Chcete-li povolit nebo zakázat různé úrovně šifrování, můžete nastavit dvě hodnoty v souboru `svnserve.conf` ve Vašem úložišti:

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

Proměnné `min-encryption` a `max-encryption` kontrolují úroveň šifrování požadovaná serverem. Chcete-li šifrování zakázat úplně, nastavte obě hodnoty na 0. Chcete-li povolit jednoduchou kontrolu součtu dat (tj. zabránit manipulaci a zaručit integritu dat bez šifrování), nastavte obě hodnoty na 1. Pokud chcete povolit (ale nepožadovat) šifrování, nastavte minimální hodnotu na 0 a maximální hodnoty na nějakou délku bitu. Chcete-li požadovat šifrování bezpodmínečně, nastavte obě hodnoty na čísla větší než 1. V našem předchozím příkladu, požadujeme, aby klienti měli alespoň 128-bitové šifrování, ale ne více než 256-bitové šifrování.

3.6.6. Ověření pomocí `svn+ssh`

Dalším způsobem, jak ověřit uživatele na `svnserve` serveru je použít Secure Shell (SSH) jehož prostřednictvím požadavky tunelujete. Nastavit SSH je těžší nastavit SASL, ale v některých případech to může být užitečné.

Tímto způsobem `svnserve` neběží jako daemon proces, spíše Secure Shell spustí `svnserve` pro Vás jako ověřeného uživatele SSH. Aby toto fungovalo, je třeba mít na serveru secure shell daemona.

Základní metoda pro nastavení vašeho serveru, je uvedena v **G – „Zabezpečení `svnserve` pomocí SSH“**. Můžete také najít jiná témata SSH v nejčastějších otázkách hledáním „SSH“.

Další informace o `svnserve` lze nalézt v ***[Řízení verzí pomocí Subversion](http://svnbook.red-bean.com)*** [<http://svnbook.red-bean.com>].

3.6.7. Pověření podle cesty v `svnserve`

Počínaje Subversion 1.3, `svnserve` podporuje stejné `mod_authz_svn` schéma pověření podle cesty, které je k dispozici na serveru Apache. Musíte upravit soubor `conf/svnserve.conf` v adresáři úložiště a přidat řádek s odkazem na Váš soubor pověření.

```
[General]
authz-db = authz
```

`Authz` je zde soubor, který vytvoříte k udělení přístupových oprávnění. Každé úložiště může mít svůj vlastní soubor, nebo můžete použít stejný soubor na několik úložišť. Přečtěte si **3.7.6 – „Pověření podle cesty“** pro popis formátu souboru.

3.7. Apache Server

3.7.1. Úvod

Nejflexibilnější ze všech možných nastavení pro Subversion server je server s Apache. I když je trochu složitější na nastavení, nabízí výhody, které jiné servery nenabízejí:

WebDAV

Subversion server s Apache používá protokol WebDAV, který je také podporován mnoha jinými programy. Mohli byste např. připojit takové úložiště jako „Webovou složku“ průzkumníka Windows a pak se do ní vstupovat jako do všech ostatních složek v systému souborů.

Procházení Úložiště

Můžete zadat adresu URL úložiště do Vašeho prohlížeče a procházet obsah bez nutnosti instalovat Subversion klienta. To dává mnohem širšímu okruhu uživatelů přístup k Vaším datům.

Ověření

Můžete použít jakýkoli mechanismus ověření, který Apache podporuje, včetně SSPI a LDAP.

Zabezpečení

Vzhledem k tomu, že Apache je velmi stabilní a bezpečný, Vaše úložiště automaticky získá stejné zabezpečení. To zahrnuje podporu SSL šifrování.

3.7.2. Instalace Apache

První, co musíte před instalací Apache mít je počítač s Windows 2000, Windows XP+SP1, Windows 2003, Vista nebo Server 2008.



Varování

Upozorňujeme, že Windows XP bez aktualizace Service Pack 1 bude vést k falešným síťovým datům a tudíž by mohlo poškodit Vaše úložiště!

1. Stáhněte si nejnovější verzi webového serveru Apache z <http://httpd.apache.org/download.cgi>. Ujistěte se, že si stáhnete verzi 2.2.x - verze 1.3.xx nebude fungovat!

MSI instalátor pro Apache lze nalézt kliknutím na další soubory, pak přejděte do `binaries/win32`. Možná si budete chtít vybrat MSI soubor `apache-2.2.x-win32-x86-openssl-0.9.x.msi` (To je ten, který obsahuje OpenSSL).

2. Jakmile budete mít instalátor Apache2, můžete na něj poklepat a průvodce Vás provede celým instalačním procesem. Ujistěte se, že jste zadali správnou URL serveru (pokud nemáte název domény DNS pro server stačí zadat IP adresu). Doporučuji nainstalovat Apache *pro všechny uživatele na portu 80, jako službu*. Poznámka: Pokud již máte IIS nebo jakýkoli jiný program běžící na portu 80, instalace může selhat. Pokud se tak stane, jděte do adresáře programů, `\Apache Group\ Apache2\conf` a vyhledejte soubor `httpd.conf`. Upravte tento soubor tak, aby `Listen 80` byl změněn na volný port, např. `Listen 81`. Restartujte instalaci - tentokrát by mělo vše proběhnout bez problémů.
3. Now test if the Apache web server is running correctly by pointing your web browser to `http://localhost/` - a preconfigured Website should show up.



Výstraha

Pokud se rozhodnete instalovat Apache jako službu, je třeba upozornit, že ve výchozím nastavení bude spuštěn pod místním systémovým účtem. Bylo by pro Vás bezpečnější vytvořit samostatný účet a pod ním Apache spouštět.

Make sure that the account on the server that Apache is running as has an explicit entry in the repository directory's access control list (right-click directory | properties | security), with full control. Otherwise, users will not be able to commit their changes.

I když Apache běží jako lokální systém, budete stále potřebovat takový záznam (v tomto případě to bude účet SYSTÉM).

Pokud Apache nemá toto oprávnění nastavené, Vaši uživatelé budou dostávat chybou zprávu „Přístup zamítnut“, které se v záznamu protokolu chyb Apache zobrazí jako chyba 500.

3.7.3. Instalace Subversion

1. Download the latest version of the Subversion Win32 binaries for Apache. Be sure to get the right version to integrate with your version of Apache, otherwise you will get an obscure error message when you try to restart. If you have Apache 2.2.x go to <http://subversion.tigris.org/servlets/ProjectDocumentList?folderID=8100>.
2. Spusťte instalaci Subversion a postupujte podle pokynů. Pokud instalátor Subversion zjistil, že máte nainstalovaný Apache, pak jste skoro hotovi. Pokud nemohl najít server Apache, pak budete muset udělat ještě další kroky.
- 3.

Pomocí programu Průzkumník Windows přejděte do instalačního adresáře Subversion (obvykle `c:\Program Files\Subversion`) A najděte soubory `/httpd/mod_dav_svn.so` a `mod_authz_svn.so`. Tyto soubory zkopírujte do adresáře modulů Apache (obvykle `c:\Program Files\Apache Group\apache2\modules`).

4. Zkopírujte soubor `/bin/libdb*.dll` a `/bin/intl3_svn.dll` z instalačního adresáře Subversion do bin adresáře Apache.
5. Upravte konfigurační soubor Apache (obvykle je v `C:\Program Files\Apache Group\Apache2\conf\httpd.conf`) textovým editorem, jako je například Poznámkový blok a proveďte následující změny:

Odkomentujte (odstraňte znak '#') následující řádky:

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

A na konec části LoadModule přidejte následující dva řádky.

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

3.7.4. Konfigurace

Ted', když jste nastavili Apache i Subversion, Apache stále ještě neví, jak zacházet s klienty Subversion jako je TortoiseSVN. Chcete-li dát Apache vědět, které URL budou použity pro úložiště Subversion, musíte upravit konfigurační soubor Apache (obvykle se nachází v `c:\Program Files\Apache Group\apache2\conf\httpd.conf`) jakýmkoli textovým editorem (např. Poznámkový blok):

1. Na konci konfiguračního souboru přidejte následující řádky:

```
<Location /svn>
  DAV svn
  SVNListParentPath on
  SVNParentPath D:\SVN
  #SVNIndexXSLT "/ svnindex.xsl"
  AuthType Basic
  AuthName "Subversion uloziste"
  AuthUserFile passwd
  #AuthzSVNAccessFile svnaccessfile
  Require valid-user
</ Location>
```

Takto konfiguruje Apache tak, aby všechny vaše úložiště Subversion byly fyzicky umístěny pod `D:\SVN`. Repozitáře jsou k dispozici vnějšímu světu z URL: `http://MujServer/svn/`. Přístup je omezen na známé uživatele/hesla, které jsou uvedeny v souboru `passwd`.

2. Chcete-li vytvořit soubor `passwd`, otevřete znovu příkazový řádek (DOS) přejděte do adresáře `apache2` (obvykle v `c:\Program Files\Apache Group\apache2`) a vytvořte soubor zadáním

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

Tím vytvoříte soubor s názvem `passwd` který je použit pro ověření. Další uživatelé mohou být přidány příkazem


```
bin\htpasswd passwd <uzivatelske_jmeno>
```

3. Znovu restartujte službu Apache.
4. Zadejte do Vašeho prohlížeče `http://MujServer/svn/MojeNoveUloziste` (Kde `MojeNoveUloziste` je název úložiště Subversion, které jste předtím vytvořili). Pokud šlo všechno dobře, měli byste být vyzváni k zadání uživatelského jména a hesla, pak si můžete prohlédnout obsah Vašeho úložiště.

Krátké vysvětlení toho, co jste zadali:

Nastavení	Vysvětlení
<Location /svn>	znamená, že úložiště Subversion jsou dostupné z URL <code>http://MujServer/svn/</code>
DAV svn	řekne Apache serveru, který modul bude odpovědný za URL adresu - v tomto případě modul Subversion.
SVNListParentPath on	Pro verzi Subversion 1.3 a vyšší, tento příkaz povoluje seznam všech dostupných repozitářů v <code>SVNParentPath</code> .
SVNParentPath D:\SVN	sděluje Subversion, že úložiště má hledat níže <code>D:\SVN</code>
SVNIndexXSLT svnindex.xsl"	"/ Aby prohlížení pomocí webového prohlížeče bylo hezčí.
AuthType Basic	je pro aktivaci základního ověření, tj. uživatelské jméno/heslo
AuthName "Subversion repositories"	se používá jako informace, kdykoliv se objeví dialogové okno ověření, aby sdělilo uživateli, na co je to ověření
AuthUserFile passwd	určuje, který soubor hesel má být použit pro ověření
AuthzSVNAccessFile	Umístění souboru aplikace Access pro cesty uvnitř Subversion úložiště
Require valid-user	určuje, že pouze uživatelé, kteří zadali správné uživatelské jméno/heslo mají na URL povolen přístup

Tabulka 3.1. Nastavení Apache `httpd.conf`

Ale to je jen příklad. Existuje mnoho, mnoho dalších možností, co můžete udělat s webovým serverem Apache.

- Pokud chcete, aby Vaše úložiště povolilo každému přístup pro čtení, ale přístup pro zápis pouze určitým uživatelům, můžete změnit řádek

```
Require valid-user
```

```
na
```

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

- Using a `passwd` file limits and grants access to all of your repositories as a unit. If you want more control over which users have access to each folder inside a repository you can uncomment the line

```
#AuthzSVNAccessFile svnaccessfile
```

and create a Subversion access file. Apache will make sure that only valid users are able to access your /svn location, and will then pass the username to Subversion's AuthzSVNAccessFile module so that it can enforce more granular access based upon rules listed in the Subversion access file. Note that paths are specified either as `repos:path` or simply `path`. If you don't specify a particular repository, that access rule will apply to all repositories under `SVNParentPath`. The format of the authorization-policy file used by `mod_authz_svn` is described in [3.7.6 – „Pověření podle cesty“](#)

- Chcete-li udělat procházení úložiště pomocí webového prohlížeče 'hezčí', vložte řádek

```
#SVNIndexXSLT "/ svnindex.xml"
```

a umístěte soubory `svnindex.xml`, `svnindex.css` a `menucheckout.ico` v kořenovém adresáři dokumentů (obvykle `C:/Program Files/Apache Group/Apache2/htdocs`). Adresář se nastavuje směrnicí `DocumentRoot` ve Vašem konfiguračním souboru Apache.

Tyto tři soubory můžete získat přímo z našeho zdrojového úložiště na <http://tortoisesvn.googlecode.com/svn/trunk/contrib/svnindex>. (3 – „TortoiseSVN je zadarmo!“ vysvětluje, jak získat přístup do zdrojového úložiště TortoiseSVN).

Soubor XSL z úložiště TortoiseSVN dělá pěkný trik: pokud procházíte úložiště pomocí Vašeho webového prohlížeče, pak každý adresář v úložišti má vpravo zobrazenou ikonu. Pokud na tuto ikonu kliknete, dialogové okno získání TortoiseSVN je z této URL spuštěno.

3.7.5. Více úložišť

Pokud jste použili směrnicí `SVNParentPath` pak nemusíte měnit konfigurační soubor Apache po každém přidání nového úložiště Subversion. Stačí vytvořit nové úložiště na stejném místě jako první úložiště a jste hotovi! V mé firmě mám přímý přístup do tohoto konkrétního adresáře na serveru pomocí protokolu SMB (normální přístup k souborům ve Windows). Takže jenom tam vytvořím nový adresář, spustím TortoiseSVN příkaz `TortoiseSVN → Zde vytvořit úložiště...` a nový projekt je doma...

Pokud používáte Subversion 1.3 nebo novější, můžete použít směrnicí `SVNListParentPath on`, k povolení Apache vytvářet seznam všech dostupných projektů, pokud zadáte do prohlížeče hlavní cestu a ne konkrétní úložiště.

3.7.6. Pověření podle cesty

Modul `mod_authz_svn` umožňuje úplnou kontrolu oprávnění k přístupu na základě uživatelských jmen a cest k úložištím. Toto je k dispozici na serveru Apache, a od Subversion 1.3 je to i k dispozici na `svnserve`.

Vzorový soubor by vypadal takto:

```
[groups]
admin = marek, klara
devteam1 = marek, lucie, veronika
devteam2 = klara, petr, tomas
dok = pavel, martina, michal
učeň = zak
# Výchozí pravidlo pro přístup ke VŠEM úložištím
# Každý může číst, pouze administrátoři zapisovat, Danu Germanovi je přístup zamí
```

```
[/]
* = r
@admin = rw
dangerman =
# umožňuje vývojářům úplný přístup k jejich projektům
[proj1:/]
@vyvtym1 = rw
[proj2:/]
@vyvtym2 = rw
[velproj:/]
@vyvtym1 = rw
@vyvtym2 = rw
trevor = rw
# Dát lidem z dokumentace přístup k zápisu do všech adresářů dokumentace
[/kmen/dok]
@dok = rw
# Dát učňům povolení k zápisu pouze do školících úložišť
[SkoliciUloziste:/]
@ucen = rw
```

Nezapomeňte, že kontrola každé cesty může být náročná operace, zvláště v případě záznamu revizí. Server kontroluje každou změnu cesty v každé revizi a kontroluje čitelnost, což může být časově náročné v revizích, které zahrnují velké množství souborů.

Ověření a pověření jsou oddělené procesy. Pokud chce uživatel získat přístup k cestě úložiště, musí splňovat *oba* běžné požadavky ověření a schválení v přístupovém souboru.

3.7.7. Ověření pomocí Domény Windows

Jak jste si možná všimli, je třeba vytvořit záznam uživatelského jména/hesla v souboru `passwd` pro každého uživatele zvlášť. A pokud (z bezpečnostních důvodů), chcete, aby si uživatelé pravidelně měnili svá hesla, musíte provést změnu ručně.

Ale pro tento problém je možné řešení - alespoň pokud přistupujete k úložišti zevnitř LAN se správcem domény windows: `mod_auth_sspi`!

Původní modul SSPI byl nabízen společností Syneapps včetně zdrojového kódu. Ale vývoj byl zastaven. Nemusíte si zoufat, komunita ho vzala a vylepšila. Má nový domov na [SourceForge](http://sourceforge.net/projects/mod-auth-sspi/) [http://sourceforge.net/projects/mod-auth-sspi/].

- Stáhněte si modul, který odpovídá Vaší verzi apache, pak zkopírujte soubor `mod_auth_sspi.so` do adresáře modulů Apache.
- Upravte konfigurační soubor Apache: přidejte řádek

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

do části `LoadModule`. Ujistěte se, že vložíte tento řádek *před* řádkem

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

- Chcete-li, aby umístění Subversion použilo tento typ ověření, je třeba změnit řádek

```
AuthType Basic
```

na

AuthType SSPI

Také je třeba přidat

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

v rámci bloku <Location/svn>. Pokud nemáte správce domény, nechte název správce domény <domaincontroller>.

Nezapomeňte, že pokud ověřujete pomocí SSPI, pak už nepotřebujete řádek AuthUserFile pro určení souboru s hesly. Apache ověřuje Vaše uživatelské jméno a heslo přes Vaši doménu Windows. Budete muset aktualizovat seznam uživatelů ve Vašem svnaccessfile abyste také odkázali na DOMÉNA \uživatelskéjméno.



Důležité

Ověření pomocí SSPI je k dispozici pouze pro zabezpečená připojení SSL (https). Pokud používáte normální http připojení k serveru, nebude to fungovat.

To enable SSL on your server, see the chapter: [3.7.9 – „Zabezpečení serveru pomocí SSL“](#)



Tip

Subversion soubory AuthzSVNAccessFile jsou citlivé na velikost písmen, pokud jde o uživatelská jména (JUser se liší od juser).

Ve světě Microsoft, domény Windows a uživatelská jména nejsou citlivá na velikost písmen. Přesto někteří správci sítě rádi vytvářejí uživatelské účty v CamelCase (např. JUser).

Tento rozdíl Vás může potrápit, když používáte ověření pomocí SSPI, protože doména Windows a uživatelská jména jsou předávány Subversion ve stejné velikosti jak je uživatel zadá. Internet Explorer často předává uživatelské jméno Apache automaticky v té velikosti, v jaké byl účet vytvořen.

V konečném výsledku budete potřebovat alespoň dvě položky ve Vašem AuthzSVNAccessFile pro každého uživatele - malým písmem a ve stejné velikosti, kterou Internet Explorer předává Apache. Budete také muset naučit uživatele, aby zadávali svá ověření malými písmeny při vstupu do úložišť přes TortoiseSVN.

Záznamy Apache o přístupech a chybách jsou Vaši nejlepší přátelé při řešení problémů, jako jsou tyto, protože Vám pomůžou určit řetězec uživatelského jména, který byl předán Subversion modulu AuthzSVNAccessFile. Možná budete muset experimentovat s přesným formátem uživatelského řetězce v svnaccessfile (Např. DOMÉNA \uživatel oproti DOMÉNA//uživatel), aby všechno fungovalo.

3.7.8. Více zdrojů ověření

Je také možné mít více než jeden zdroj ověření pro Vaše úložiště Subversion. Pro to je třeba, aby žádný typ ověření nebyl ten hlavní, takže Apache bude kontrolovat více zdrojů pro odpovídající uživatelské jméno a heslo.

Běžný scénář je používat ověření pomocí domény Windows i souboru `passwd`, takže můžete poskytnout přístup do SVN uživatelům, kteří nemají přihlášení v doméně Windows.

- Chcete-li povolit ověřování podle domény Windows a souboru `passwd`, přidejte následující položky do bloku `<location>` Vašeho konfiguračního souboru Apache:

```
AuthBasicAuthoritative Off
SSPIAuthoritative Off
```

Zde je příklad úplné konfigurace Apache pro ověření pomocí Windows domény a souboru `passwd`:

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

  AuthName "Uloziste Subversion"
  AuthzSVNAccessFile svnaccessfile.txt

# Přihlášení přes domény.
AuthType SSPI
SSPIAuth On
SSPIAuthoritative Off
SSPIDomain <domaincontroller>
SSPIOfferBasic On

# Přihlášení pomocí Htpasswd.
AuthType Basic
AuthBasicAuthoritative Off
AuthUserFile passwd

  Require valid-user
</Location>
```

3.7.9. Zabezpečení serveru pomocí SSL

I když Apache 2.2.x má podporu OpenSSL, není ve výchozím nastavení zapnuto. Musíte ji aktivovat ručně.

1. V konfiguračním souboru Apache, odkomentujte tyto řádky:

```
#LoadModule ssl_module modules/mod_ssl.so
```

a na konci

```
#Include conf/extra/httpd-ssl.conf
```

změňte řádek (vše na jednom řádku)

```
SSLMutex "file:C:/Program Files/Apache Software Foundation/\
```

```
Apache2.2/logs/ssl_mutex"
```

na

```
SSLMutex default
```

2. Dále potřebujete vytvořit SSL certifikát. K tomu je třeba otevřít okno příkazového řádku (DOS) a přejít do adresáře Apache (např. C:\Program Files\Apache Group\apache2) a zadejte následující příkaz:

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

Budete vyzváni k zadání hesla. Prosíme, nepoužívejte jednoduchá slova, ale celé věty, např. část básně. Čím delší je fráze, tím lépe. Také je třeba zadat URL adresu Vašeho serveru. Všechny ostatní otázky jsou nepovinné, ale doporučujeme je také vyplnit.

Normálně je soubor `privkey.pem` vytvořen automaticky, ale pokud není je nutné zadat tento následující příkaz ke generování:

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

Next type the commands

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

and (on one line)

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

and then (on one line)

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

This will create a certificate which will expire in 4000 days. And finally enter (on one line):

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

These commands created some files in the Apache conf folder (`server.der.crt`, `server.csr`, `server.key`, `.rnd`, `privkey.pem`, `server.cert`).

3. Restartujte službu Apache.
4. Zadejte do Vašeho prohlížeče `https://jmenoserveru/svn/projekt ...`



SSL a Internet Explorer

Pokud jste zabezpečili Váš server pomocí SSL a používáte ověřování pomocí domény Windows, zjistíte, že procházení úložiště s Internet Explorer již nefunguje. Nebojte se - to jen Internet Explorer není schopen ověření. Ostatní prohlížeče tento problém nemají a TortoiseSVN i jiní klienti Subversion jsou stále schopni ověření.

Pokud přesto chcete používat IE k procházení úložiště můžete například:

- definujte samostatnou směrnici `<Location /cesta>` v konfiguračním souboru Apache a přidejte `SSPIBasicPreferred On`. To umožní IE znovu se ověřit, ale jiné prohlížeče a Subversion se nebudou schopni v tomto umístění ověřit.
- Nabídněte také prohlížení bez šifrovaného ověření (bez SSL). Kupodivu IE nemá žádné problémy s ověřením, pokud spojení není zabezpečené pomocí SSL.
- Ve "standardním" nastavení SSL je často následující prohlášení ve virtuálním hostiteli Apache SSL:

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

Pro tuto konfiguraci jsou (byly?) dobré důvody, viz http://www.modssl.org/docs/2.8/ssl_faq.html#ToC49 Ale pokud chcete ověření NTLM budete muset použít `keepalive`. Pokud odkomentujete celý `SetEnvIf` měli byste být schopni ověřit IE s ověřením Windows přes SSL pod Apache na Win32 spolu s `mod_auth_sspi`.



Donutit k přístupu pomocí SSL

Když jste nastavili SSL, aby Vaše úložiště bylo bezpečnější, možná budete chtít zakázat normální přístup bez SSL (http) a umožnit přístup pouze přes https. Chcete-li to tak, budete muset přidat další směrnici do Subversion bloku `<location>`: `SSLRequireSSL`.

Příklad bloku `<Location>` by mohl vypadat nějak takto:

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

3.7.10. Použití klientských certifikátů s virtuálními hostiteli SSL

Posláno Nigelem Greenem do poštovního adresáře TortoiseSVN. Díky!

V některých konfiguracích serveru byste mohli potřebovat nastavit jeden server obsahující 2 virtuálních hostitele SSL: první pro veřejný přístup na webové stránky, bez požadavku na klientský certifikát. Druhý bezpečný s požadavkem na klientský certifikát, obsahující Subversion server

Přidáním směrnice `SSLVerifyClient Optional` do části *server* konfigurace Apache (tj. mimo bloky `VirtualHost` a `Directory` bloků) donutí Apache požádat o certifikát klienta v počátečním zahájení spojení přes SSL. Kvůli chybě v `mod_ssl` je důležité, aby byl certifikát vyžadován právě v tuto chvíli, je-li SSL spojení později znovu vyžádáno tak nefunguje.

Řešením je přidat následující směrnici do virtuálního adresáře hostitele, který chcete pro Subversion uzamknout:

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

Tato směrnice poskytuje přístup k adresáři pouze tehdy, pokud klientský certifikát byl úspěšně získán a ověřen.

To summarise, the relevant lines of the Apache configuration are:

```
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>
```

Kapitola 4. Příručka Každodenního Použití

Tento dokument popisuje každodenní používání klienta TortoiseSVN. Toto *není* úvod do systémů pro správu verzí *ani* úvod do Subversion (SVN). Je to spíš místo, kam se můžete obrátit, když přibližně víte, co chcete dělat, ale nemůžete si vzpomenout, jak na to.

Pokud potřebujete úvod do správy verzí pomocí Subversion, pak doporučujeme přečíst si tyto fantastické knihy: *Správa verzí pomocí Subversion* [<http://svnbook.red-bean.com/>].

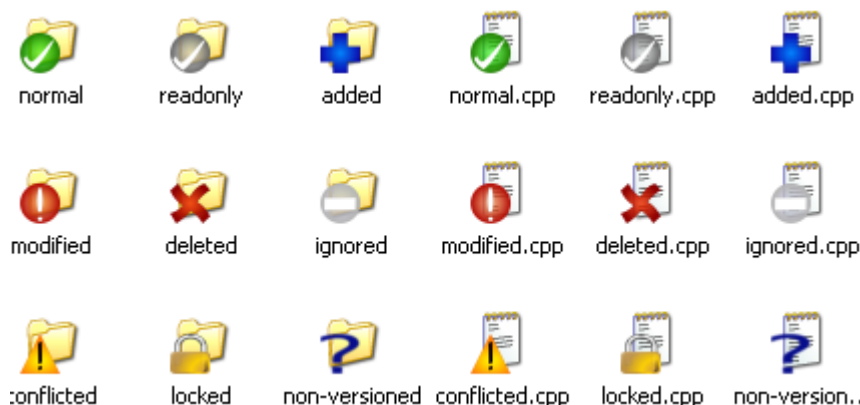
Tento dokument je také nedokončená práce, stejně jako jsou TortoiseSVN a Subversion. Pokud najdete nějaké chyby, nahlašte je, prosím, do mailing listu, abychom mohli aktualizovat dokumentaci. Některé snímky obrazovek v průvodci každodenním použitím (PKP) nemusí odrážet současný stav softwaru. Prosím, odpusťte nám. Na TortoiseSVN pracujeme v našem volném čase.

Abyste získali co nejvíce z průvodce každodenním použitím:

- Měli byste již mít nainstalován TortoiseSVN.
- Měli byste být seznámeni se systémy pro správu verzí.
- Měli byste znát základy Subversion.
- Měli byste mít nastaven server a/nebo mít přístup k úložišti Subversion.

4.1. Začínáme

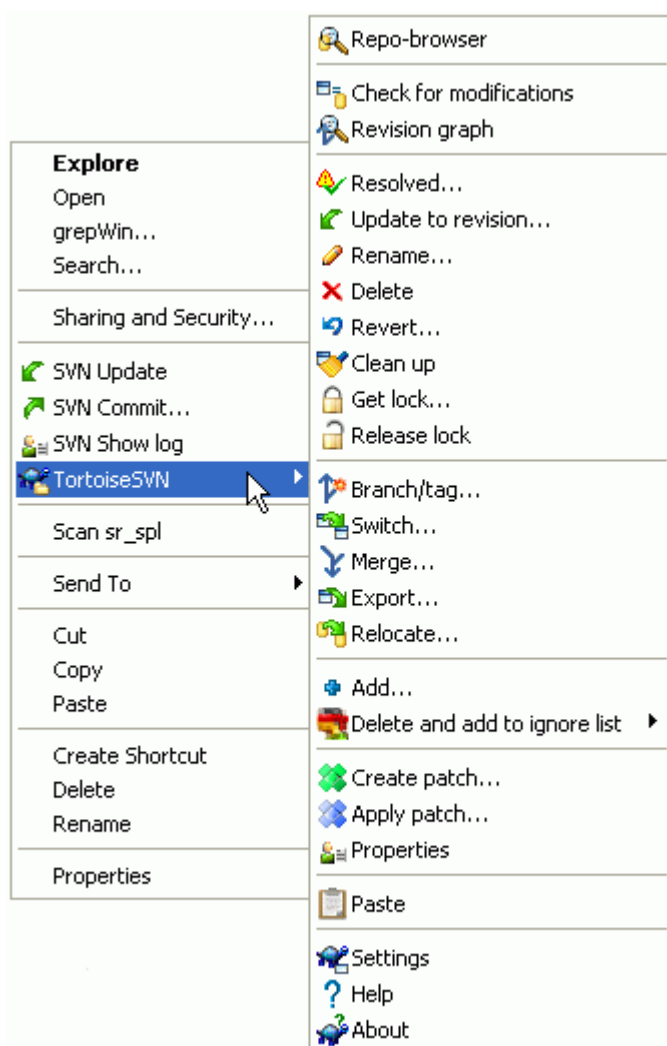
4.1.1. Překryv Ikon



Obrázek 4.1. Průzkumník ukazující překryv ikon

Jedním z nejviditelnějších funkcí TortoiseSVN je překryv ikon, které se objevují na soubory v pracovní kopii. Tyto ikony Vám na první pohled ukáží, které z Vašich souborů byly změněny. Prohlédněte si [4.7.1 – „Překryv Ikon“](#) abyste zjistili, co různé překryvy znamenají.

4.1.2. Kontextové menu



Obrázek 4.2. Kontextové menu pro adresář pod správou verzí

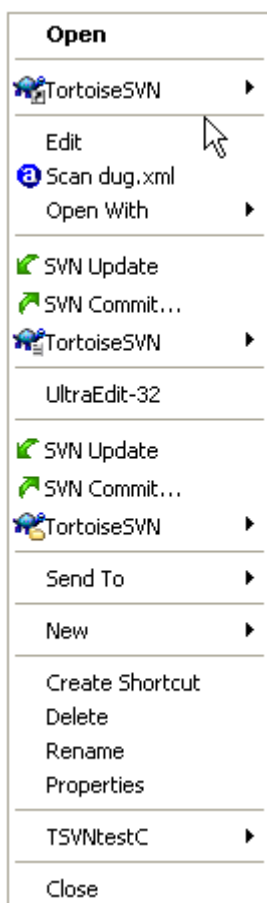
Všechny příkazy TortoiseSVN jsou volány z kontextového menu Průzkumníka Windows. Většina z nich je hned viditelná, když kliknete pravým tlačítkem myši na soubor nebo adresář. Příkazy, které jsou k dispozici, závisí na tom, zda soubor nebo adresář, nebo jeho nadřazený adresář je pod správou verzí, nebo ne. Můžete si také zobrazit menu TortoiseSVN jako součást menu Soubor v Průzkumníku.



Tip

Some commands which are very rarely used are only available in the extended context menu. To bring up the extended context menu, hold down the **Shift** key when you right-click.

V některých případech se může zobrazit několik položek TortoiseSVN. To není chyba!

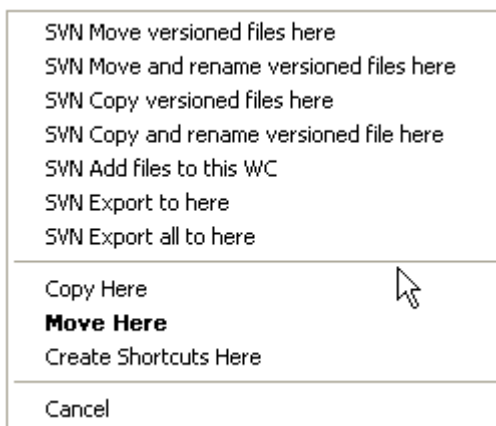


Obrázek 4.3. Menu Soubor v Průzkumníku pro zkratky v adresáři s verzí

Tento příklad je pro zkratky bez verze v adresáři s verzí a v menu Soubor v Průzkumníku existují *tři* položky TortoiseSVN. Jedna z nich je pro adresář, jedna pro tuto zkratku a třetí pro objekt, na který zkratka ukazuje. Abychom Vám pomohli se mezi nimi vyznat, ikony mají v pravém dolním rohu ukazatel, který zobrazí, zda položka v menu je pro soubor, adresář, zkratku, nebo pro více vybraných položek.

Pokud používáte Windows 2000, tak zjistíte, že kontextová menu jsou zobrazeny jako prostý text, aniž by se v nabídce zobrazili ikony, jak je uvedeno výše. Jsme si vědomi, že toto v předchozích verzích fungovalo, ale Microsoft změnil ve Windows Vista způsob, jak jsou ikony ovládány, což nás nutí použít jiný způsob zobrazení, který bohužel ve Windows 2000 nefunguje.

4.1.3. Táhnout a Pustit



Obrázek 4.4. Menu táhnutí pravým tlačítkem pro adresář pod správou verzí

Ostatní příkazy jsou dostupné jako ovladače táhnutí, když soubory nebo adresáře táhnete pravým tlačítkem do nového umístění uvnitř pracovní kopie, nebo když soubor nebo adresář bez verze táhnete pravým tlačítkem do adresáře, který je pod správou verzí.

4.1.4. Běžné zkratky

Některé běžné operace ve Windows mají známé klávesové zkratky, ale nejsou uvedeny na tlačítkách, ani v menu. Pokud nemůžete přijít na to, jak udělat něco zřejmého, jako obnovit zobrazení, podívejte se sem.

F1

To je samozřejmě nápověda

F5

Obnoví současné zobrazení. Toto je možná jeden z nejužitečnějších jedno klávesových příkazů. Například ... V Průzkumníku obnoví překryv ikon ve Vaši pracovní kopii. V dialogovém okně odevzdání znovu prohlédne pracovní kopii, jestli tam je něco, co může být odevzdáno. V dialogovém okně Záznamu Revizí znovu zkontaktuje úložiště ke kontrole posledních změn.

Ctrl-A

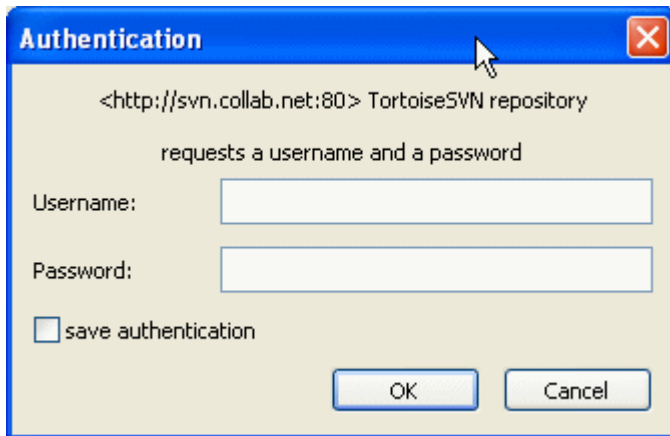
Vybrat vše. Toto může být použito, pokud dostanete chybovou zprávu a chcete ji celou zkopírovat a vložit do e-mailu. Použijte Ctrl-A pro výběr chybové zprávy a poté ...

Ctrl-C

... Zkopírujte vybraný text.

4.1.5. Ověření

Pokud je úložiště, do kterého se pokoušíte vstoupit chráněno heslem, zobrazí se dialogové okno ověření.



Obrázek 4.5. Dialogové okno Ověření

Zadejte své uživatelské jméno a heslo. Zaškrtnutí políčko uloží Vaše pověření ve výchozím adresáři Subversion: %APPDATA%\Subversion\auth ve třech podadresářích:

- `svn.simple` contains credentials for basic authentication (username/password).
- `svn.ssl.server` obsahuje SSL certifikáty serveru.
- `svn.username` obsahuje pověření pro ověření pouze uživatelským jménem (není třeba heslo).

Pokud chcete vymazat uchovaná ověření na Všech serverech, můžete tak učinit ze stránky dialogového okna **Uložená data** v nastavení TortoiseSVN. Toto tlačítko vymaže všechny uložené údaje ověření z `auth` adresáře Subversion i všechna ověřovací data uložená v registrech z předchozích verzí TortoiseSVN. Viz [4.30.6 – „Nastavení Uložených dat“](#).

Some people like to have the authentication data deleted when they log off Windows, or on shutdown. The way to do that is to use a shutdown script to delete the %APPDATA%\Subversion\auth directory, e.g.

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

You can find a description of how to install such scripts at [windows-help-central.com](http://www.windows-help-central.com/windows-shutdown-script.html) [http://www.windows-help-central.com/windows-shutdown-script.html].

For more information on how to set up your server for authentication and access control, refer to [3.5 – „Přístup do úložiště“](#)

4.1.6. Maximalizace Windows

Mnoho dialogových oken TortoiseSVN zobrazuje spousty informací, ale je často lepší raději maximalizovat pouze výšku, nebo šířku, než maximalizovat na celou obrazovku. Pro Vaše pohodlí, existují pro tuto funkci zkratky v tlačítku **Maximalizovat**. Použijte prostřední tlačítko myši pro vertikální maximalizaci a pravé tlačítko myši pro horizontální maximalizaci.

4.2. Import dat do úložiště

4.2.1. Import

If you are importing into an existing repository which already contains some projects, then the repository structure will already have been decided. If are importing data into a new repository then it is worth taking the time to think about how it will be organised. Read 3.1.5 – „Rozložení úložiště“ for further advice.

Tato část popisuje Subversion příkaz Importovat, který je určen pro import kompletní hierarchie adresáře do úložiště najednou. I když funguje, má několik nedostatků:

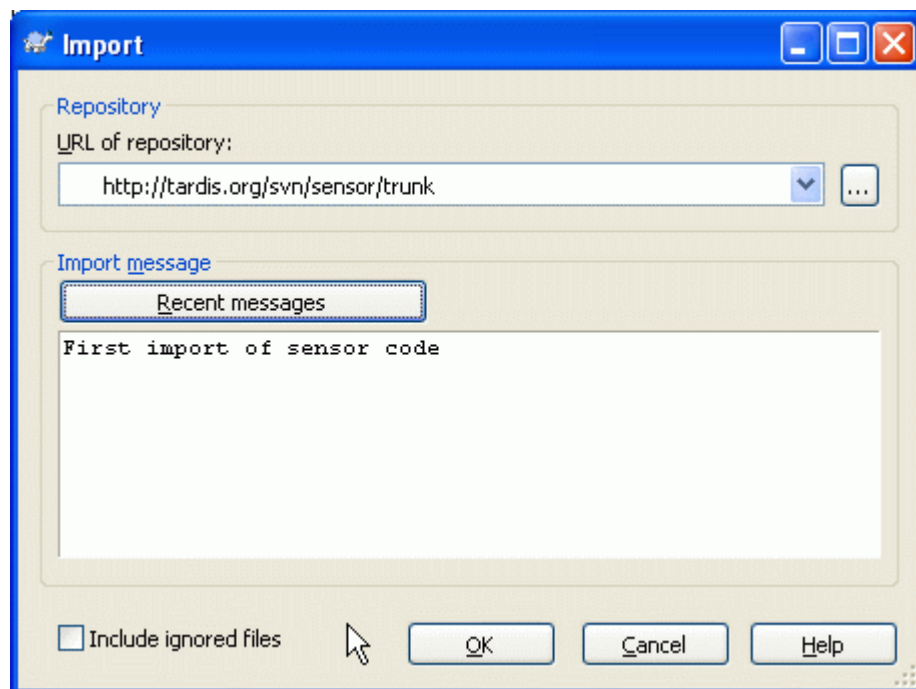
- Neexistuje žádný způsob, jak vybrat k zahrnutí soubory a adresáře, kromě použití globálního nastavení ignorování.
- Importovaný adresář se nestane pracovní kopií. Musíte ji získat, pro zkopírování souborů zpět ze serveru.
- Je snadné v úložišti importovat do špatné úrovně adresáře.

Z těchto důvodů vůbec nedoporučujeme používat příkaz Importovat, ale spíše postupovat podle metody dvou kroků popsané v 4.2.2 – „Import na místě“. Ale protože už jste tady, takhle funguje jednoduchý import ...

Před importem projektu do úložiště, byste měli:

1. Odstraňte všechny soubory, které nejsou potřebné k sestavení projektu (dočasné soubory, soubory, které jsou vytvořeny kompilátorem např. *.obj, zkompileované spustitelné soubory, ...)
2. Zorganizujte soubory v adresářích a podadresářích. Ačkoli je možné soubory přejmenovat/přesunout později je výrazně doporučováno, abyste si strukturu Vašeho projektu ujasnili před importem!

Nyní vyberte nejvyšší úroveň struktury adresáře projektu v programu Průzkumník Windows a kliknutím pravým tlačítkem myši otevřete kontextové menu. Vyberte příkaz TortoiseSVN → Importovat... , který vyvolá dialogové okno:



Obrázek 4.6. Dialogové okno importu

V tomto dialogovém oknu je třeba zadat URL adresu místa v úložišti, kam chcete Váš projekt importovat. Je velmi důležité si uvědomit, že místní adresář, který importujete, se nezobrazí v úložišti, pouze jeho obsah. Například pokud máte strukturu:

```
C:\Projekty\Widget\zdroj
C:\Projekty\Widget\dok
C:\Projekty\Widget\obrázky
```

a importujete C:\Projekty\Widget do `http://mojedomena.cz/svn/kmen` pak můžete být překvapeni, když zjistíte, že vaše podadresáře byly umístěny rovnou do kmene a ne do podadresáře Widget. Musíte zadat podadresáře jako součást URL, `http://mojedomena.cz/svn/kmen/Widget-X`. Nezapomeňte, že příkaz Importovat automaticky vytvoří v úložišti podadresáře, pokud neexistují.

Zpráva Importu se použije jako zpráva záznamu.

Ve výchozím nastavení, soubory a adresáře, které se shodují se vzory globálního ignorování, *nejsou* importovány. Chcete-li toto chování potlačit, můžete zaškrtnout políčko Zahrnout ignorované soubory. Pro více informací o formátech nastavení globálního ignorování navštivte [4.30.1 – „Obecná nastavení“](#).

Jakmile stisknete tlačítko OK TortoiseSVN importuje celý adresář stromu, včetně všech souborů do úložiště. Projekt je nyní uložen v úložišti pod správou verzí. Uvědomte si, prosím, že importované adresáře *NEJSOU* pod správou verzí! Pro získání *pracovní kopie* pod správou verzí musíte Získat verzi, kterou jste právě importovali. Nebo čtěte dále, abyste zjistili, jak importovat adresáře na místě.

4.2.2. Import na místě

Za předpokladu, že již máte úložiště a chcete přidat do něho novou strukturu adresáře, postupujte podle následujících kroků:

1. Použitím Průzkumníka úložiště vytvořte nový adresář projektu přímo v úložišti.
2. Získejte nový adresář, který bude o úroveň výše než Váš adresářem, který chcete importovat. Budete varováni, že místní adresář není prázdný. Nyní máte adresář nejvyšší úrovně s verzí s obsahem bez verze.
3. Použitím TortoiseSVN → Přidat... na tento adresář s verzí přidejte některý nebo všechny jeho obsah. Můžete přidat a odstranit soubory, nastavit vlastnosti adresáře `svn:ignore` a provádět další změny, které potřebujete.
4. Odevzdejte nejvyšší úroveň adresáře a máte nový strom s verzí a místní pracovní kopii, vytvořenou z Vašeho stávajícího adresáře.

4.2.3. Mimořádné soubory

Někdy musíte mít soubor pod kontrolou verzí, který obsahuje konkrétní údaje uživatele. To znamená, že máte soubor, který každý vývojář/uživatel potřebuje změnit, aby vyhovovaly jeho/jejím místním nastavení. Ale verzování takového souboru, je obtížné, protože každý uživatel by pokaždé odevzdal jeho/její změny do úložiště.

V těchto případech doporučujeme použít *šablonový* soubor. Vytvoříte soubor, který obsahuje všechna data Vaši vývojáři budou potřebovat, dát tento soubor pod kontrolu verzí a nechat vývojáře získat tento soubor. Potom, každý vývojář si *vytvoří kopii* tohoto souboru a přejmenuje si ji. Poté změna kopie už není problémem.

Jako příklad se můžete podívat na sestavovací skript TortoiseSVN. Volá souboru jménem `TortoiseVars.bat`, který v úložišti neexistuje. Pouze soubor `TortoiseVars.tmpl`. `TortoiseVars.tmpl` je šablonový soubor a každý vývojář si musí vytvořit kopii tohoto souboru a přejmenovat ho na `TortoiseVars.bat`. Uvnitř tohoto souboru jsme přidali komentáře, takže

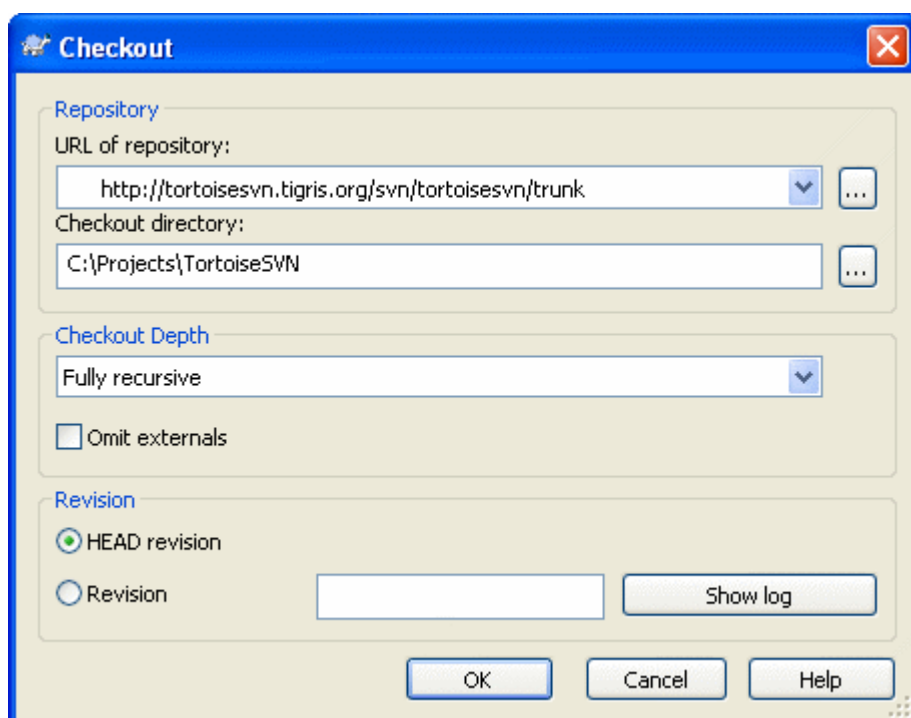
uživatelé mohou vidět, které řádky budou muset upravit a měnit v závislosti na jejich místním nastavení, aby všechno fungovalo.

Aby toto uživatele nerušilo, jsme také přidali soubor `TortoiseVars.bat` do seznamu k ignorování v jeho nadřazené složce, tedy jsme nastavili Subversion vlastnost `svn:ignore`, aby zahrnovala tento název souboru. Takto se v každém odevzdání neukáže jako bez verze.

4.3. Získání Pracovní Kopie

Chcete-li získat pracovní kopii, musíte ji nejdříve z úložiště *získat*.

Vyberte adresář v Průzkumníku Windows, kde chcete umístit Vaši pracovní kopie. Klikněte pravým tlačítkem myši pro zobrazení kontextového menu a vyberte volbu **TortoiseSVN → Získat...**, která zobrazí následující dialogové okno:



Obrázek 4.7. Dialogové okno Získání

Pokud zadáte název adresáře, který ještě neexistuje, pak bude s tímto názvem vytvořen.

4.3.1. Hloubka Získání

Můžete si vybrat *hloubku*, kterou chcete získat, což Vám umožňuje určit hloubku rekurze v podřízených adresářích. Pokud chcete pouze několik částí velkého stromu, můžete získat nejvyšší úroveň adresáře a pak aktualizovat vybrané adresáře rekurzivně.

Plně rekurzivní

Získat celý strom, včetně všechny podřízených adresářů a podadresářů.

Bezprostředně podřazené prvky, včetně adresářů

Získat zadaný adresář včetně všech souborů a podřazených složek, ale podřízené adresáře nezaplnňovat.

Pouze podřízené soubory

Získat zadaný adresář včetně všech souborů, kromě podřízených adresářů.

Pouze tuto položku

Získat pouze tento adresář. Nezaplnovat ho soubory nebo podřízenými adresáři.

Pracovní kopie

Ponechat hloubku uvedenou v pracovní kopii. Tato volba není použita v dialogovém okně získání, ale je výchozí ve všech ostatních dialogích, které mají nastavení hloubky.

Vynechat

Používá se při snížení hloubky pracovní kopii poté, co adresář byl již naplněn. Tato volba je dostupná pouze v dialogovém oknu **Aktualizovat na revizi**.

If you check out a sparse working copy (i.e., by choosing something other than `fully recursive` for the checkout depth), you can fetch additional sub-folders by using the repository browser (4.24 – „Průzkumník Úložiště“) or the check for modifications dialog (4.7.3 – „Místní a Vzdálený Stav“).

In the repository browser, Right click on the checked out folder, then use **TortoiseSVN → Repo-Browser** to bring up the repository browser. Find the sub-folder you would like to add to your working copy, then use **Context menu → Update item to revision...** That menu will only be visible if the selected item does not exist yet in your working copy, but the parent item does exist.

V dialogovém okně **Zjistit Změny**, nejprve klikněte na tlačítko **Zkontrolovat Úložiště**. Dialogové okno zobrazí všechny soubory a adresáře, které jsou v úložišti, ale nemáte je získané jako vzdáleně přidané. Klikněte pravým tlačítkem myši na adresář(e), které byste chtěli přidat do Vaší pracovní kopie, pak zvolte **Kontextové menu → Aktualizovat**.

Tato funkce je velmi užitečná, když chcete získat části velkého stromu, ale chcete mít pohodlí aktualizace jediné pracovní kopie. Předpokládejme, že máte velký strom, který má podadresáře **Projekt01** až **Projekt99** a chcete získat pouze **Projekt03**, **Projekt25** a **Projekt76/PodProj**. Postupujte podle těchto kroků:

1. Získejte nadřazený adresář s hloubkou „Jen tato položka“ Nyní máte prázdný adresář nejvyšší úrovně.
2. Vyberte novou složku a zvolte **TortoiseSVN → Průzkumník úložiště** k zobrazení obsahu úložiště.
3. Pravým tlačítkem myši klikněte na **Projekt03** a **Kontextové menu → Aktualizovat položku na revizi...** . Ponechejte výchozí nastavení a klikněte na **OK**. Nyní máte tento adresář zcela zaplněn.

Opakujte stejný postup pro **Projekt25**.

4. Přejděte na **Projekt76/PodProj** a udělejte to samé. Uvědomte si ale, že adresář **Projekt76** nemá žádný obsah, kromě **PodProj**, který je zcela zaplněn. Subversion pro Vás vytvořil přechodné adresáře jejich bez naplnění.



Změna hloubky pracovní kopie

Once you have checked out a working copy to a particular depth you can change that depth later to get more or less content using **Context menu → Update item to revision...**



Používání staršího serveru

Pre-1.5 servers do not understand the working copy depth request, so they cannot always deal with requests efficiently. The command will still work, but an older server may send

all the data, leaving the client to filter out what is not required, which may mean a lot of network traffic. If possible you should upgrade your server to 1.5.

Pokud projekt obsahuje odkazy na externí projekty, které *nechcete* získat ve stejnou dobu, použijte políčko **Vynechat externí**.



Důležité

Pokud **Vynechat externí** je zaškrtnuto, nebo pokud chcete zvýšit hodnotu hloubky, budete muset provést aktualizaci Vaší pracovní kopie pomocí **TortoiseSVN → Aktualizovat na revizi...** místo **TortoiseSVN → Aktualizace**. Standardní aktualizace bude obsahovat všechny externí a zachová stávající hloubku.

Doporučuje se, abyste získali pouze část `kmén` adresářového stromu, nebo nižší. Zadáte-li hlavní cestu k adresáři stromu v URL pak byste mohli skončit s plným pevným diskem, protože budete mít kopie celého stromu úložiště, včetně všech větví a značek Vašeho projektu!



Exportování

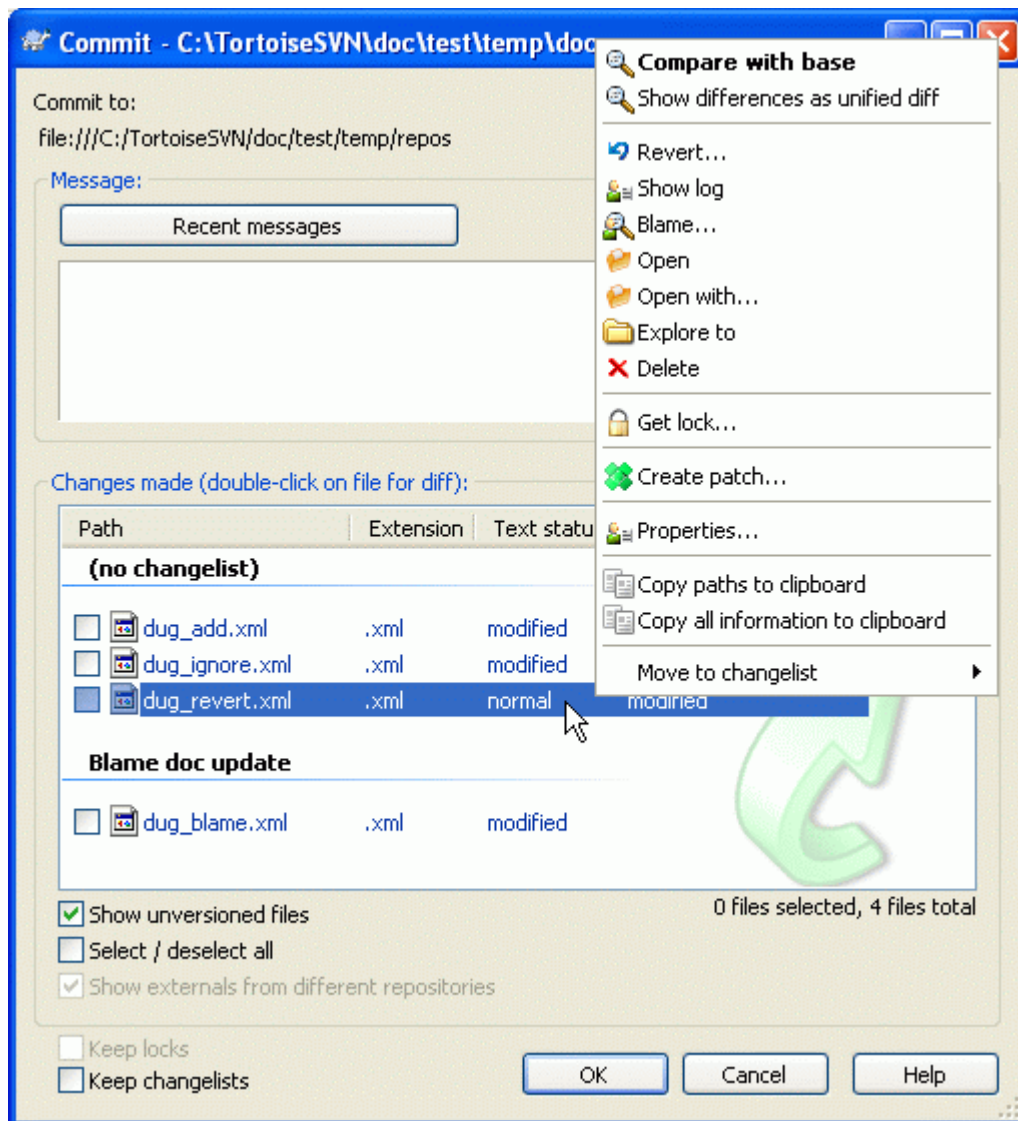
Někdy možná budete chtít vytvořit lokální kopii, bez těch všech `.svn` adresářů, např. vytvořit tarem zazipovaný archiv Vašeho zdroje. Přečtěte si [4.26 – „Export Pracovní Kopie Subversion“](#) abyste zjistili, jak na to.

4.4. Odevzdání Vašich změn do úložiště

Odeslání změn provedených ve Vaší pracovní kopii se nazývá *odevzdání* změn. Ale dříve, než je můžete odevzdat, se musíte ujistit, že Vaše pracovní kopie je aktuální. Můžete použít buď přímo **TortoiseSVN → Aktualizovat**, nebo můžete nejdříve použít **TortoiseSVN → Zkontrolovat změny** k zjištění, které soubory se lokálně nebo na serveru změnily.

4.4.1. Dialogové okno Odevzdání

Pokud vaše pracovní kopie je aktuální a neexistují žádné konflikty, jste připraveni odevzdat změny. Vyberte jakýkoli soubor a/nebo adresáře, které chcete odevzdat, pak **TortoiseSVN → Odevzdat...**



Obrázek 4.8. Dialogové okno odevzdání

Dialogové okno odevzdání Vám ukáže každý změněný soubor, včetně přidáných, smazaných a bez verze. Pokud si nepřejete, aby změněný soubor byl odevzdán, stačí zrušit zaškrtnutí tohoto souboru. Pokud chcete zařadit soubor bez verze, stačí zaškrtnout tento soubor k přidání do odevzdání.

Položky, které byly překlopeny do jiné cesty úložiště, jsou rovněž uvedeny pomocí značky (s). Mohli jste něco překlomit při práci na větvi a zapomněli jste překlomit zpět do kmene. Toto je Váš varovný signál!



Odevzdat soubory nebo adresáře?

Když odevzdáváte soubory, dialogové okno odevzdání zobrazuje pouze soubory, které jste vybrali. Když odevzdáváte adresáře, dialogové okno odevzdání vybere změněné soubory automaticky. Pokud zapomenete na nový soubor, který jste vytvořili, odevzdáním adresáře se stejně najde. Odevzdání adresáře *neznamená*, že každý soubor je označen jako změněný, jen to ulehčuje Váš život tím, že za Vás udělá více práce.

If you have modified files which have been included from a different repository using `svn:externals`, those changes cannot be included in the same atomic commit. A warning symbol below the file list tells you if this has happened, and the tooltip explains that those external files have to be committed separately.



Mnoho souborů bez verze v dialogovém okně odevzdání

Pokud si myslíte, že dialogové okno odevzdání zobrazuje příliš mnoho souborů bez verze (např. vytvořené kompilátorem nebo záloha editoru), existuje několik způsobů, jak toto vyřešit. Můžete:

- přidejte soubor (nebo zástupný znak výjimky) do seznamu souborů k vynechání na stránce nastavení. To bude mít vliv na každou pracovní kopii, kterou máte.
- přidat soubor do seznamu `svn:ignore` pomocí TortoiseSVN → Přidat do seznamu k ignorování. To bude mít vliv pouze na adresář, na kterém nastavíte vlastnost `svn:ignore`. Použitím Dialogového okna Vlastností SVN, můžete měnit vlastnost `svn:ignore` pro adresář.

Přečtěte si [4.13 – „Ignorování Souborů a Adresářů“](#) pro více informací.

Dvojitým kliknutím na každý modifikovaný soubor v dialogovém okně odevzdání zahájí externí nástroj rozdílů pro zobrazení změn. Kontextové menu Vám dá více možností, jak je na obrázku znázorněno. Můžete také přetáhnout soubory odtud do jiné aplikace, jako je textový editor nebo IDE.

Můžete označit nebo odznačit výběr položky klepnutím na zaškrtačací políčko vlevo od položky. Pro adresáře můžete využít **Shift**-vybrat, aby se akce stala rekurzivní.

Sloupce zobrazené v dolní části panelu jsou přizpůsobitelné. Pokud kliknete pravým tlačítkem myši na jakékoli záhlaví sloupce, zobrazí se kontextové menu, umožňující vybrat, které sloupce se mají zobrazit. Můžete také změnit šířku sloupce pomocí tažení rukojeti, která se objeví při pohybu myši nad hranici sloupce. Tyto úpravy jsou zachovány, takže i příště uvidíte stejné položky.

Ve výchozím nastavení, když odevzdáváte změny, všechny zámky, které na souborech držíte, jsou automaticky uvolněny po úspěšném odevzdání. Pokud si chcete tyto zámky ponechat, ujistěte se, že volba **Udržet zámky** je zaškrtnuta. Výchozí stav tohoto políčka je převzat z možnosti `no_unlock` v konfiguračním souboru Subversion. Pro více informací jak upravit konfigurační soubor Subversion navštivte [4.30.1 – „Obecná nastavení“](#).



Táhnout a Pustit

Můžete přetáhnout soubory do dialogového okna odevzdání odjinud, pokud byly pracovní kopie získány ze stejného úložiště. Například můžete mít obrovskou pracovní kopii s několika otevřenými okny Průzkumníka k zobrazení vzdálených adresářů hierarchie. Pokud chcete, aby se zabránilo odevzdání nejvyšší úrovně adresáře (se zdlouhavým procházením adresářů ke kontrole změn) můžete otevřít dialogové okno odevzdání jednoho adresáře a přetáhnout do položky z ostatních oken, k zahrnutí do stejného atomového odevzdání.

Můžete přetáhnout soubory bez verze, které jsou v pracovní kopii do dialogového okna odevzdání a budou automaticky SVN přidány.



Oprava Přejmenovaných Externích

Někdy jsou soubory mimo Subversion přejmenovány a pak se objeví v seznamu souborů jako chybějící a soubor bez verze. Aby nedošlo ke ztrátě historie je třeba Subversion o připojení oznámit. Stačí jen vybrat starý (chybějící) a nový (bez verze) název (chybí) a použít Kontextové menu → Opravit přesun pro spárování těchto souborů jako přejmenování.

4.4.2. Seznam Změn

Dialogové okno odevzdání podporuje funkce seznamu změn Subversion, které pomáhají seskupit dohromady související soubory. Více informací o této funkci naleznete v [4.8 – „Seznam Změn“](#).

4.4.3. Vyjmout položky ze seznamu odevzdání

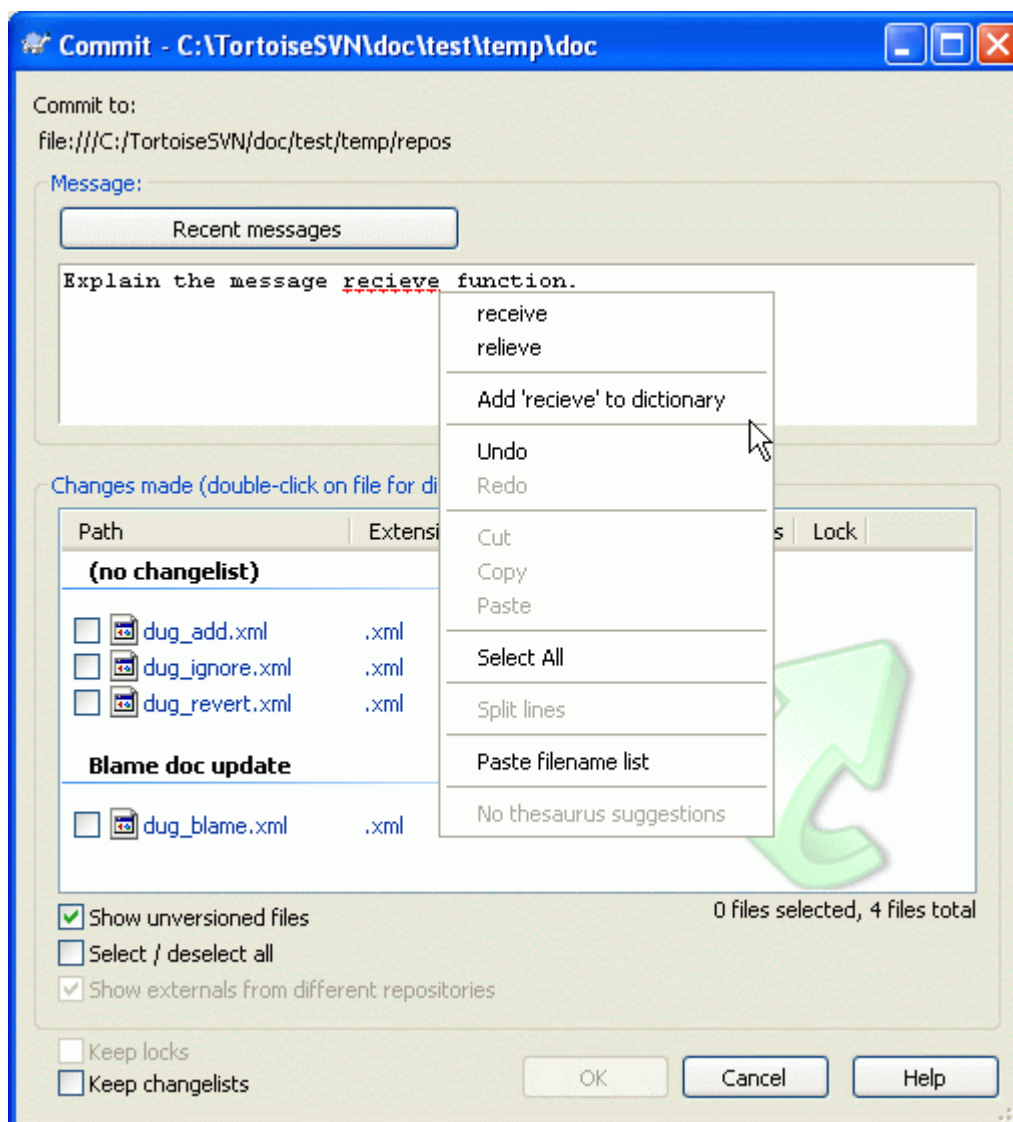
Někdy máte soubory v adresářích, které se často mění, ale které nechcete odevzdat. Někdy to znamená chybu v sestavovacím procesu - Proč mají ty soubory verzi? Máte používat šablonové soubory? Ale občas je to nevyhnutelné. Klasický důvodem je, že vaše IDE změnil časové razítko v souboru projektu pokaždé, když sestavuje. Soubor projektu musí mít verzi, protože zahrnuje všechny nastavení sestavení, ale nemusí být odevzdán jen proto, že časové razítko se změnilo.

Chcete-li pomoci v nepříjemných případech jako je tento, vyhradili jsme seznam změn s názvem `ignore-on-commit`. Jakýkoli soubor přidáný do tohoto seznamu bude automaticky odškrtnut v dialogovém oknu odevzdání. Stále můžete pořád odevzdávat změny, ale musíte je v dialogovém oknu odevzdání zvolit ručně.

4.4.4. Zprávy Záznamu Odevzdání

Ujistěte se, že zadáte zprávu záznamu, která popisuje změny, které odevzdáváte. To vám pomůže zjistit, co se stalo a kdy, pokud později budete procházet zprávy záznamu projektu. Zpráva může být tak dlouhá nebo krátká jak chcete; mnoho projektů má návody, co by mělo být zahrnuto, jaký jazyk používat a někdy dokonce i přísný formát.

Můžete použít jednoduché formátování zprávy záznamu pomocí dohody podobné té používané v rámci e-mailu. Chcete-li použít styl na `text`, použijte `*text*` pro tučné, `_text_` pro podtržení a `^text^` pro kurzívu.



Obrázek 4.9. Kontrola pravopisu v Dialogovém oknu Odevzdání

TortoiseSVN obsahuje kontrolu pravopisu, které Vám pomůže opravit Vaše zprávy záznamu. Bude upozorňovat na jakékoliv pravopisné chyby. Použijte kontextové menu pro zobrazení navrhaných oprav. Samozřejmě neví *každý* technický termín, který Vy znáte, takže pravopisně správná slova se někdy zobrazí jako chyby. Ale nebojte se. Stačí je jen přidat do Vašeho osobního slovníku pomocí kontextového menu.

Okno zprávy záznamu také zobrazuje název souboru a funkce automatického dokončování. Používá regulární výrazy pro extrakci tříd a názvů funkcí z (textových) souborů, které odevzdáváte, stejně jako názvy souborů samotných. Pokud slovo, které píšete, se shoduje se slovem v seznamu (poté, co jste napsali alespoň 3 znaky, nebo jste stiskli **Ctrl+Mezerník**), zobrazí se rozbalovací seznam, který umožňuje vybrat celé jméno. Regulární výrazy dodávané v TortoiseSVN jsou uloženy v instalaci TortoiseSVN v adresáři bin. Můžete také definovat vlastní regulární výrazy a uložit je do %APPDATA%\TortoiseSVN\autolist.txt. Samozřejmě Váš soukromý autoseznam nebude přepsán při aktualizaci instalace TortoiseSVN. Pokud nejste seznámeni s regulárními výrazy, podívejte se na úvod do http://cs.wikipedia.org/wiki/Regulární_výraz a on-line dokumentace a návody na <http://www.regularnivrazy.info/>.

Můžete znovu použít dříve zadanou zprávu záznamu. Stačí kliknout na **Poslední zprávy** k zobrazení seznamu několika posledních zpráv, které jste zadali pro tuto pracovní kopii. Počet uložených zpráv lze přizpůsobit v dialogovém okně nastavení TortoiseSVN.

Můžete vymazat všechny uložené zprávy odevzdání ze stránky **Uložená data** v nastavení TortoiseSVN, nebo můžete jednotlivé zprávy vymazat v rámci dialogového okna **Poslední zprávy** s použitím klávesy **Delete**.

Pokud chcete zahrnout zkontrolované cesty ve své zprávě záznamu, můžete použít příkaz **Kontextové menu** → **Vložit seznam souborů** v kontrole úprav.

Dalším způsobem, jak vložit cesty do zpráv záznamů je jednoduše přetáhnout soubory ze seznamu souborů na kontrolu úprav.



Vlastnosti Mimořádného Adresáře

Existuje několik speciálních vlastností adresáře, které mohou být použity k poskytnutí větší kontroly nad formátováním zpráv záznamu odevzdání a jazyk používaný modulem kontroly pravopisu. Přečtěte si **4.17 – „Nastavení projektu“** pro další informace.

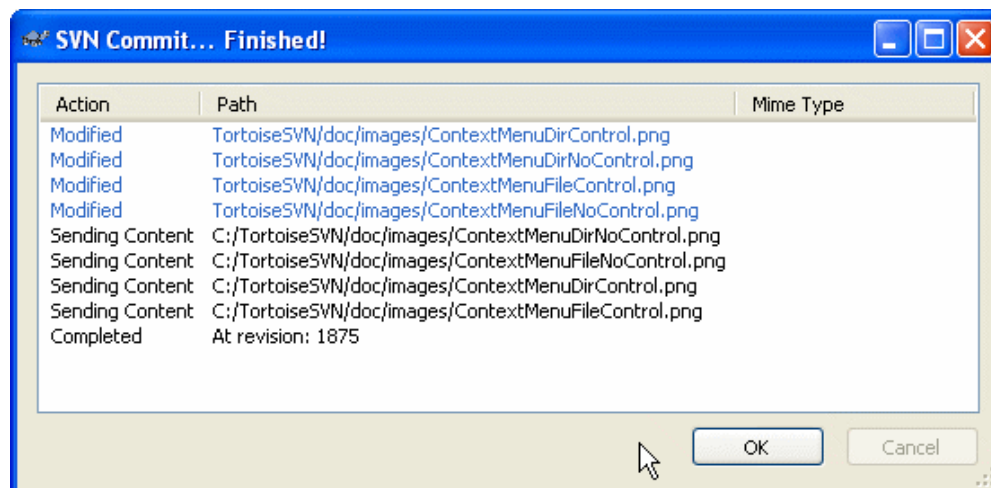


Integrace s Nástroji Hledání Chyb

Pokud jste aktivovali systém pro sledování chyb, můžete nastavit jeden nebo více chyb v textovém poli **ID chyby / Č. problému**. Více chyb by mělo být odděleno čárkou. Případně, pokud používáte podporu regulárních výrazů ve sledování chyb, stačí přidat Váš odkaz na chybu jako součást zpráv záznamu. Více se dozvíte na **4.28 – „Integrace se Systémy pro Sledování Chyb/Problémů“**.

4.4.5. Průběh Odevzdání

Po stisknutí tlačítka **OK** se objeví dialogové okno zobrazující průběh odevzdání.



Obrázek 4.10. Dialogové okno průběhu zobrazující probíhající odevzdání

Dialogové okno průběhu používá barevné kódování pro zvýraznění různých činností odevzdání

Modrá

Odevzdání změny.

Fialová

Odevzdání nové položky.

Tmavě červená

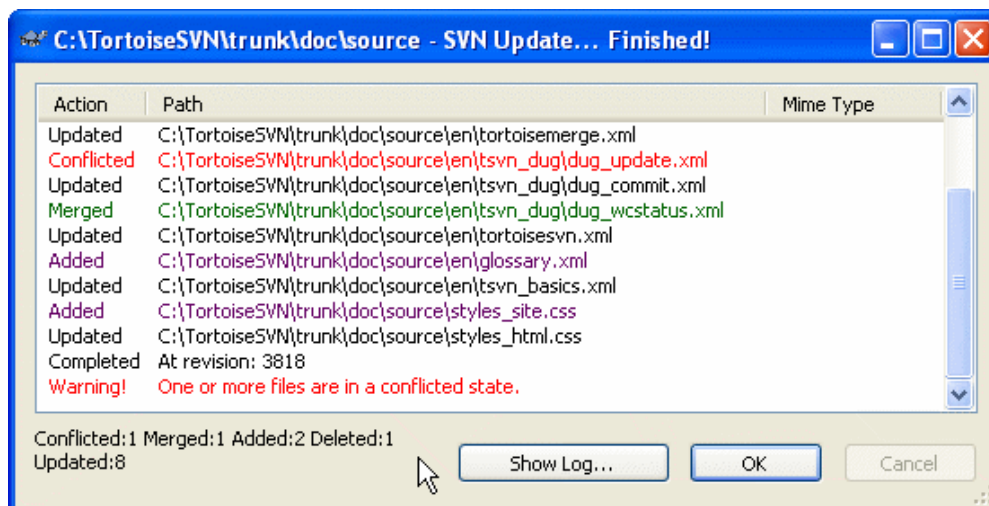
Odevzdání vymazání nebo nahrazení.

Černá

Všechny ostatní položky.

Toto je výchozí barevné schéma, ale můžete si ho přizpůsobit pomocí dialogového okna nastavení. Přečtěte si 4.30.1.4 – „Nastavení Barev TortoiseSVN“ pro více informací.

4.5. Aktualizovat pracovní kopii změnami od ostatních



Obrázek 4.11. Dialogové okno průběhu zobrazující dokončení aktualizace

Měli byste pravidelně zajistit, že změny provedené ostatními jsou začleněny do Vaší místní pracovní kopie. Proces získávání změn ze serveru do Vaší místní kopie je známý jako *aktualizace*. Aktualizace může být provedena pro jednotlivé soubory, sadu vybraných souborů, nebo rekurzivně na celou hierarchii adresáře. Chcete-li aktualizovat, vyberte soubory a / nebo adresáře, které chcete, klikněte pravým tlačítkem myši a vyberte TortoiseSVN → Aktualizovat v kontextovém menu Průzkumníka. Objeví se okno zobrazující průběh aktualizace, jak postupuje. Změny provedené ostatními budou sloučeny do Vašich souborů. Veškeré změny, aktualizovaných souborů, které jste provedli jsou zachovány. Úložiště *není* aktualizací ovlivněno.

Dialogové okno průběhu používá barevné kódování pro zvýraznění různých činností aktualizace

Fialová

Do Vaší pracovní kopie byla přidána nová položka.

Tmavě červená

Nadbytečná položka byla smazána z pracovní kopie, nebo byla nahrazena chybějící položka ve Vaší pracovní kopii.

Zelená

Změny z úložiště byly úspěšně sloučeny se změnami místními.

Jasně červená

Změny z úložiště byly sloučeny se změnami místními. Vznikly konflikty, které je třeba vyřešit.

Černá

Nezměněné položky ve Vaší pracovní kopii byly aktualizovány novější verzí z úložiště.

Toto je výchozí barevné schéma, ale můžete si ho přizpůsobit pomocí dialogového okna nastavení. Přečtěte si 4.30.1.4 – „Nastavení Barev TortoiseSVN“ pro více informací.

Pokud máte při aktualizaci jakékoli *konflikty* (to se může stát, když ostatní změnili stejné řádky ve stejném souboru jako Vy a tyto změny si neodpovídají), pak tyto konflikty mají v dialogovém okně červenou barvu. Můžete dvojklikem na tyto řádky spustit externí nástroj sloučení pro vyřešení konfliktů.

Když je aktualizace dokončena, dialogové okno průběhu udává přehled o počtu aktualizovaných, přidaných, odstraněných, konfliktních, atd. položek pod seznamem souborů. Tyto souhrnné informace lze zkopírovat do schránky pomocí **Ctrl+C**.

Standardní příkaz Update nemá žádné možnosti a pouze aktualizuje pracovní kopii na revizi HEAD z úložiště, což je nejčastější případ použití. Pokud chcete větší kontrolu nad procesem aktualizace, můžete použít TortoiseSVN → Aktualizovat na revizi.... To vám umožní aktualizovat pracovní kopii na konkrétní revizi, ne pouze na poslední. Předpokládejme, že vaše pracovní kopie je v revizi 100, ale chcete, aby odrážela stav, který měla v revizi 50 - pak jednoduše aktualizujte na revizi 50. Ve stejném dialogovém okně můžete také vybrat *hloubku*, ve které se má stávající adresář aktualizovat. Použité pojmy jsou popsány v 4.3.1 – „*Hloubka Získání*“. Výchozí hloubka je **Pracovní kopie**, která zachovává stávající nastavení hloubky. Můžete také zvolit, zda chcete v aktualizaci ignorovat všechny vnější projekty (tj. projekty, odkazující pomocí `svn:externals`).



Výstraha

Pokud aktualizujete soubor nebo adresář na určitou revizi, neměli byste provádět v těchto souborech změny. Získáte chybové hlášení „zastaralé“ při pokusu o jejich odevzdání! Pokud chcete vrátit zpět změny v souboru a začít znovu od dřívější revize, můžete se vrátit zpět k předchozí revizi z dialogového okna záznamu revize. Podívejte se na B.4 – „*Vrátit zpět revize v úložišti*“ pro další instrukce a alternativní metody.

Aktualizovat na revizi... někdy může být užitečná pro zobrazení, jak Váš projekt vypadal v nějakém dřívějším bodu své historie. Ale obvykle aktualizace jednotlivých souborů starší revizí není dobrý nápad, protože to zanechá Vaši pracovní kopii v nekonzistentním stavu. Pokud soubor, který chcete aktualizovat, změnil jméno, můžete dokonce zjistit, že tento soubor prostě zmizí z Vaší pracovní kopie, protože žádný soubor s daným názvem v dřívější revizi neexistoval. Měli byste také vzít na vědomí, že položka se zobrazí s normálním zeleným překryvem, takže je k nerozeznání od souborů, které jsou aktuální.

Chcete-li jenom místní kopii staré verze souboru, je lepší použít příkaz Kontextové menu → Uložit revizi do... z dialogového okna záznamu pro tento soubor.



Vícero Souborů/Adresářů

Pokud vyberete více souborů a adresářů v Průzkumníku a poté zvolíte Aktualizovat, všechny tyto soubory/adresáře jsou aktualizovány jeden po druhém. TortoiseSVN se ujistí, že všechny soubory/adresáře, které jsou ze stejného úložiště, jsou aktualizovány na přesně stejnou revizi! I když mezi těmito aktualizacemi došlo k dalšímu odevzdání.



Místní Soubor Již Existuje

Někdy, při pokusu o aktualizaci, aktualizace selže se zprávou, že místní soubor se stejným názvem již existuje. Toto obvykle nastane, když se Subversion pokusí získat nový soubor s verzí a zjistí, že uvedený soubor bez verze se stejným názvem již existuje ve vašem pracovním adresáři. Subversion nikdy nepřepisuje soubor bez verze - mohlo by totiž obsahovat něco, na čem pracujete a shodou okolností má stejný název, jako jiný vývojář použil pro svůj nově odevzdaný soubor.

Pokud dostanete tuto chybovou zprávu, řešením je místní soubor bez verze jednoduše přejmenovat. Po dokončení aktualizace můžete zkontrolovat, zda je přejmenovaný soubor stále ještě zapotřebí.

Pokud budete stále dostávat chybové zprávy, použijte TortoiseSVN → Zkontrolovat změny místo zobrazení seznamu všech problémových souborů. Tímto způsobem se můžete se všemi vypořádat najednou.

4.6. Řešení Konfliktů

Jednou za čas, dostanete *konflikt*, když při aktualizaci/sloučení souborů z úložiště, nebo při překlopení Vaší pracovní kopie na jinou adresu URL. Jsou dva druhy konfliktů:

konflikty souboru

Konflikt souboru nastane, pokud dva (nebo více) vývojářů změní stejné řádky v souboru.

konflikty stromu

Konflikt stromu nastane, když vývojář přesune/přejmenuje/vymaže nějaký soubor nebo adresář, který jiný vývojář buď také přesunul/přejmenoval/ smazat, nebo právě upravil.

4.6.1. Konflikty souboru

A file conflict occurs when two or more developers have changed the same few lines of a file. As Subversion knows nothing of your project, it leaves resolving the conflicts to the developers. Whenever a conflict is reported, you should open the file in question, and search for lines starting with the string <<<<<<. The conflicting area is marked like this:

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

Also, for every conflicted file Subversion places three additional files in your directory:

filename.ext.mine

Toto je Váš soubor, jak vypadal v pracovní kopii, než jste ji aktualizovali - to je, bez značek konfliktu. Tento soubor má v něm Vaše poslední změny a nic jiného.

filename.ext.OLDREV

Jedná se o soubor, který byl základní revizí před aktualizací Vaší pracovní kopie. To znamená, soubor, který jste získali předtím, než jste provedli Vaše poslední úpravy.

filename.ext.rNEWREV

Jedná se o soubor, který váš klient Subversion právě obdržel ze serveru při aktualizaci Vaší pracovní kopie. Tento soubor odpovídá revizi HEAD v úložišti.

Můžete buď spustit externí nástroj sloučení / editor konfliktů s TortoiseSVN → Upravit Konflikty nebo můžete použít jakýkoli jiný editor pro ruční vyřešení konfliktu. Měli byste se rozhodnout, jak má kód vypadat, udělat potřebné změny a soubor uložit.

Poté spusťte příkaz TortoiseSVN → Vyřešeno a odevzdejte Vaše změny do úložiště. Vezměte prosím na vědomí, že příkaz Vyřešeno ve skutečnosti konfliktu nevyřeší. Pouze odstraní soubory filename.ext.mine a filename.ext.r*, které Vám umožní změny odevzdat.

Máte-li konflikt ve spustitelných souborech, Subversion se nepokusí soubory sloučit. Místní soubor zůstane beze změny (přesně tak, jak jste ho naposledy změnili) a máte soubory filename.ext.r*. Pokud chcete změny zahodit a nechat si verzi z úložiště, stačí použít příkaz Vrátil. Pokud si chcete nechat Vaši verzi a přepsat verzi z úložiště, použijte příkaz Vyřešeno, pak odevzdejte svoji verzi.

Můžete použít příkaz Vyřešeno na více souborů, pokud kliknete pravým tlačítkem na nadřazený adresář a zvolíte TortoiseSVN → Vyřešeno... zobrazí se dialogové okno se seznamem všech souborů v konfliktu v tomto adresáři a můžete si vybrat, které z nich označit jako vyřešené.

4.6.2. Konflikty Stromu

Konflikt stromu nastane, když vývojář přesune/přejmenuje/vymaže nějaký soubor nebo adresář, který jiný vývojář buď také přesunul/přejmenoval/ smazal, nebo právě upravil. Existuje mnoho různých situací, které mohou vyústit v konflikt stromu a všechny z nich vyžadují různé kroky k jeho vyřešení.

Když je soubor v Subversion smazán lokálně, je také smazán z lokálního souborového systému, takže i když je součástí konfliktu stromu nemůže ukázat překrytí konfliktu a nemůžete na něj kliknout pravým tlačítkem myši, abyste ho vyřešili. Použijte dialogové okno Zkontrolovat změny místo volby Upravit konflikt.

TortoiseSVN Vám může pomoci najít správné místo ke sloučení změn, ale možná bude za potřebí další práce potřebná k vyřešení konfliktů. Pamatujte si, že po aktualizaci pracovního originálu bude vždy obsahovat revizi jednotlivých položek, tak jak byly v úložišti v době aktualizace. Pokud vrátíte změny po aktualizaci, vrátí se zpátky do stavu úložiště, ne jak byla, když jste začali provádět své vlastní místní změny.

4.6.2.1. Místní smazání, příchozí úpravy po aktualizaci

1. Developer A modifies Foo.c and commits it to the repository
2. Vývojář B ve stejnou dobu přesune Bla.c do Alb.c v jeho pracovní kopii, nebo jednoduše smaže Bla.c nebo jeho nadřazený adresář.

Aktualizace pracovní kopie vývojáře B má za následek konflikt stromu:

- Bla.c byl smazán z pracovní kopie, ale je označen konfliktem stromu.
- Pokud konflikt vyplývá z přejmenování spíše než ze smazání pak Alb.c je označen jako přidaný, ale neobsahuje změny vývojáře A.

Vývojář B musí nyní rozhodnout, zda zachovat změny vývojáře A. V případě přejmenování souboru, může sloučit změny Bla.c do přejmenovaného souboru Alb.c. V případě prostého odstranění souboru nebo adresáře si může vybrat, zda si nechat položku se změnami vývojáře A a odstranění zruší. Nebo tím, že označení konfliktu jako vyřešený, aniž by něco udělal, se zbaví změn vývojáře A.

Editační dialogové okno konfliktu nabízí sloučit změny, jestliže lze najít původní soubor přejmenovaného Alb.c. V závislosti na tom, kde byla aktualizace vyvolána, nemusí být možné najít zdrojový soubor.

4.6.2.2. Místní upravení, po aktualizaci příchozí smazání

1. Vývojář A přesune Bla.c do Alb.c a odevzdá ho do úložiště.
2. Vývojář B změní B1s.c v jeho pracovní kopii.

Nebo v případě přesunutí adresáře...

1. Vývojář A přesune nadřazený adresář AdresářBla do AdresářAlb a odevzdá ho do úložiště.
2. Vývojář B změní B1s.c v jeho pracovní kopii.

Aktualizace pracovní kopie vývojáře B má za následek konflikt stromu. Pro jednoduchý konflikt souboru:

- Alb.c je přidán do pracovní kopie jako normální soubor.
- Bla.c je označen jako přidaný (s historií) a má konflikt stromu.

Pro konflikt adresářů:

- AdresářAlb je přidán do pracovní kopie jako normální adresář.
- AdresářBla je označen jako přidán (s historií) a má konflikt stromu.

Bla.c je označen jako upravený.

Vývojář B musí nyní rozhodnout, zda souhlasit s reorganizací vývojáře A a sloučit její změny do odpovídajícího souboru v nové struktuře, nebo jednoduše vrátit změny vývojáře A a ponechat si místní soubor.

Chcete-li sloučit lokální změny reorganizace, musí vývojář B nejprve zjistit, jak byl konfliktní soubor Bla.c v úložišti přejmenován/přesunut. To lze provést pomocí dialogového okna záznamů. Tyto změny pak musí být ručně sloučeny, protože v současnosti neexistuje způsob, jak automatizovat nebo dokonce ani jak zjednodušit tento proces. Jakmile jsou změny přeneseny, konfliktní cesta je zbytečná a může být smazána. V tomto případě použijte tlačítko **Odstranit** v dialogovém okně editoru konfliktů pro vyčištění a označení konfliktu jako vyřešený.

Pokud Vývojář B rozhodne, že změny Vývojáře A byly špatné, pak musí zvolit volbu **Ponechat** v dialogovém okně editoru konfliktů. Toto označí se konflikt souboru/adresáře jako vyřešený, ale změny vývojáře A musí být odstraněny ručně. Opět dialogové okno záznamů pomáhá vypátrat, co bylo přesunuto.

4.6.2.3. Místní smazání, po aktualizaci příchozí smazání

1. Developer A moves Foo.c to Bar.c and commits it to the repository
2. Developer B moves Foo.c to Bix.c

Aktualizace pracovní kopie vývojáře B má za následek konflikt stromu:

- Elb.c je označeno jako přidáný s historií.
- Alb.c je přidán do pracovní kopie se stavem 'normální'.
- Bla.c je označen jako smazaný a má konflikt stromu.

Pro vyřešení tohoto konfliktu, Vývojář B musí zjistit, jak byl konfliktní soubor Bla.c v úložišti přejmenován/přesunut. To lze provést pomocí dialogového okna záznamů.

Pak vývojář B musí rozhodnout, jaké nové jméno souboru Bla.c si ponechat - to od vývojáře A nebo to jeho.

Poté, co vývojář B ručně konflikt vyřešil, konfliktu stromu musí být označen jako vyřešený pomocí tlačítka v dialogovém okně editoru konfliktu.

4.6.2.4. Místní chybí, po sloučení příchozí úprava

1. Vývojář A pracující v kmeni změní Bla.c a odevzdá ho do úložiště
2. Vývojář B pracující na větvi přesune Bla.c do Alb.c a odevzdá ho do úložiště

Sloučení změn kmene vývojáře A do pracovní kopie větve vývojáře B vyústí v konflikt stromu:

- Alb.c je již v pracovní kopii se stavem 'normální'.
- Bla.c je označen jako chybějící s konfliktem stromu.

Pro vyřešení tohoto konfliktu Vývojář B musí označit soubor jako vyřešený v dialogovém okně editoru konfliktu, což ho ze seznamu konfliktu odstraní. Pak musí rozhodnout, zda zkopírovat chybějící soubor Bla.c z úložiště do pracovní kopie, zda sloučit změny Bla.c Vývojáře A do přejmenovaného Alb.c nebo zda ignorovat změny označením konfliktu za vyřešený a nedělat nic.

Nezapomeňte, že pokud zkopírujete chybějící soubor z úložiště a pak označíte konflikt jako vyřešený, Vaše kopie bude znovu odstraněna. Nejdříve musíte konflikt vyřešit.

4.6.2.5. Místní úprava, po sloučení příchozí smazání

1. Developer A working on trunk moves Foo.c to Bar.c and commits it to the repository

2. Vývojář B pracující na větvi změní Bla.c a odevzdá ho do úložiště.

Existuje obdobný případ u přesunu adresářů se, ale Subversion 1.6 to stále ještě nedokáže odhalit ...

1. Vývojář A pracující na kmeni přesune nadřazený adresář AdresářBla do AdresářAlb a odevzdá ho do úložiště.

2. Vývojář B pracující na větvi změní Bla.c v jeho pracovní kopii.

Sloučení změn kmene vývojáře A do pracovní kopie větve vývojáře B vyústí v konflikt stromu:

- Alb.c je označen jako přidaný.
- Bla.c je označen jako upravený s konfliktem stromu.

Vývojář B musí nyní rozhodnout, zda souhlasit s reorganizací vývojáře A a sloučit její změny do odpovídajícího souboru v nové struktuře, nebo jednoduše vrátit změny vývojáře A a ponechat si místní soubor.

Ke sloučení místních změn s reorganizací, musí Vývojář B nejprve zjistit, jak byl konfliktní soubor Bla.c v úložišti přejmenován/přesunut. To lze provést pomocí dialogového okna záznamu pro zdroj sloučení. Editor konfliktů zobrazuje pouze záznam pracovní kopie, protože neví, která cesta byla ve sloučení použita, to budete muset najít sami. Tyto změny pak musí být ručně sloučeny, protože v současnosti neexistuje způsob, jak automatizovat nebo dokonce tento proces zjednodušit. Jakmile jsou změny přeneseny, konfliktní cesta je zbýtečná a může být smazána. V tomto případě použijte tlačítko **Odstranit** v dialogovém okně editoru konfliktů pro vyčištění a označení konfliktu jako vyřešený.

Pokud Vývojář B rozhodne, že změny Vývojáře A byly špatné pak musí zvolit tlačítko **Ponechat** v dialogovém okně editoru konfliktů. Toto označí konfliktní soubor/adresář jako vyřešený, ale změny vývojáře A musí být odstraněny ručně. Opět dialogové okno záznamů pro zdroj sloučení pomáhá vypátrat, co bylo přesunuto.

4.6.2.6. Místní smazání, po sloučení příchozí smazání

1. Developer A working on trunk moves Foo.c to Bar.c and commits it to the repository

2. Developer B working on a branch moves Foo.c to Bix.c and commits it to the repository

Sloučení změn kmene vývojáře A do pracovní kopie větve vývojáře B vyústí v konflikt stromu:

- Elb.c je označen normálním (bez úprav) stavem.
- Alb.c je označen jako přidaný s historií.
- Bla.c je označen jako chybějící a má konfliktu stromu.

Chcete-li vyřešit tento konflikt, Vývojář B musí zjistit, jak byl konfliktní soubor Bla.c v úložišti přejmenován/přesunut. To lze provést pomocí dialogového okna záznamů pro zdroj sloučení. Editor konfliktů zobrazuje pouze záznam o pracovní kopii, protože neví, která cesta byla při sloučení použita, na to budete muset přijít sami.

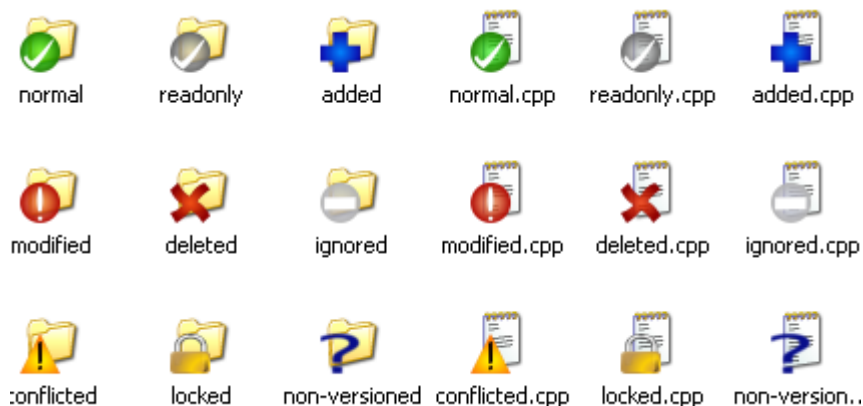
Pak vývojář B musí rozhodnout, jaké nové jméno souboru Bla.c si ponechat - to od vývojáře A nebo to jeho.

Poté, co vývojář B ručně konflikt vyřešil, konfliktu stromu musí být označen jako vyřešený pomocí tlačítka v dialogovém okně editoru konfliktu.

4.7. Získávání informací o stavu

Při práci ve Vaší pracovní kopii si často musíte pamatovat, které soubory jste změnili/přidali/ odstranili nebo přejmenovali, či dokonce, které soubory byly změněny a odevzdány ostatními.

4.7.1. Překryv ikon



Obrázek 4.12. Průzkumník ukazující překryv ikon

Nyní, když jste získali pracovní kopii z úložiště Subversion můžete vidět, jak se v Průzkumníku Windows souborům změnila ikona. To je jeden z důvodů, proč je TortoiseSVN tak populární. TortoiseSVN přidává tzv. překryvné ikony ke každé ikoně souboru, která překrývá původní ikonu. V závislosti na Subversion stav souboru je každá překryvná ikona jiná.



Čerstvě získaná pracovní kopie má jako překryv zelené zaškrtnutí. To znamená, že Subversion stav je *normální*.



Jakmile začnete upravovat soubor, stav se změní na *změněno* a překryvná ikona se pak změní na červený vykřičník. Tímto způsobem můžete snadno zjistit, které soubory byly změněny od poslední aktualizace Vaší pracovní kopie a musí být odevzdány.



Pokud dojde během aktualizace ke *konfliktu*, pak se ikona změní na žlutý vykřičník.



Pokud jste nastavili v souboru vlastnost `svn:needs-lock`, Subversion soubor zpřístupní pouze pro čtení, dokud tento soubor nezamknete. Takové soubory mají tento překryv oznamující, že nejdříve musíte soubor uzamknout, než ho můžete začít upravovat.



Pokud nad souborem držíte zámek a Subversion stav je *normální*, toto překrytí ikony Vám připomene, že byste měli zámek uvolnit, pokud ho nepoužíváte, aby ostatní mohli odevzdat své změny do souboru.



Tato ikona ukazuje, že některé soubory nebo adresáře v aktuálním adresáři byly naplánovány ke *smazání* z kontroly verzí, nebo soubor pod kontrolou verzí v adresáři chybí.



Znaménko plus Vám říká, že soubor nebo adresář byly naplánovány k *přidání* do kontroly verzí.



Znamínko mínus Vám napovídá, že soubor nebo adresář je *ignorován* pro účely kontroly verzí. Tento překryv je volitelný.



Tato ikona reprezentuje soubory a adresáře, které nejsou pod kontrolou verzí, ale nebyly ignorovány. Tento překryv je volitelný.

Ve skutečnosti, možná zjistíte, že ne všechny tyto ikony jsou ve Vašem systému používány. Je to proto, že počet překrytí povoleným systémem Windows je velmi omezený a pokud také používáte starou verzi TortoiseCVS, pak není dostatek volných míst pro překryvy. TortoiseSVN se snaží být „Dobrým Občanem (TM)“ a omezuje své používání překryvů, aby i jiné aplikace dostali šanci.

Protože v současnosti existuje více klientů Tortoise (TortoiseCVS, TortoiseHg,...) limit ikon se stává skutečným problémem. Projekt TortoiseSVN proto představil společně sdílenou sadu ikon, načtenou jako DLL, které lze použít pro všechny klienty Tortoise. Poradte se se svým poskytovatelem klienta, zda toto už bylo integrováno :-)

Pro popis toho, kterému stavu Subversion překryvy ikon odpovídají a další technické podrobnosti si přečtěte [F.1 – „Překryv Ikon“](#).

4.7.2. Sloupce TortoiseSVN v Průzkumníku Windows

Stejně informace, které jsou k dispozici v překryvech ikon (a ještě mnohem víc), mohou být zobrazeny jako další sloupce v okně podrobností Průzkumníka Windows.

Jednoduše klikněte pravým tlačítkem myši na jeden ze záhlaví sloupce, zvolte Více... ze zobrazeného kontextového menu. Objeví se dialogové okno, kde můžete zadat sloupce a jejich pořadí, který je zobrazen v „Detailním zobrazení“. Přejděte dolů, dokud se položky začínající SVN nezobrazí. Zaškrtněte ty, které byste chtěli mít zobrazeny a zavřete dialogové okno stisknutím OK. Sloupce budou připojen napravo od momentálně zobrazených. Můžete změnit jejich pořadí přetažením, nebo měnit jejich velikost, aby odpovídali vašim potřebám.



Důležité

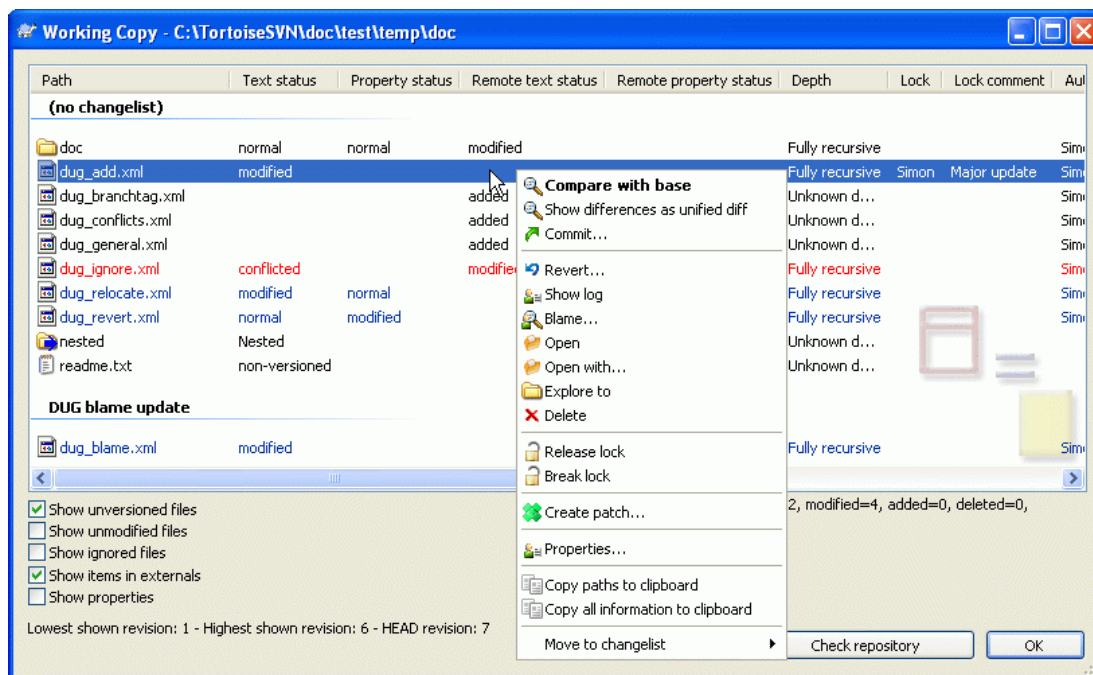
Další sloupce v Průzkumníku Windows nejsou ve Windows Vista k dispozici, protože Microsoft rozhodl, nepovolovat takové sloupce pro *všechny* soubory ale jen pro konkrétní typy souborů.



Tip

Pokud chcete, aby stávající uspořádání, které je zobrazeno bylo ve všech Vašich pracovních kopiích, můžete zobrazení nastavit jako výchozí zobrazení.

4.7.3. Místní a Vzdálený Stav



Obrázek 4.13. Zjistit změny

Je často velmi užitečné vědět, které soubory jste změnili, a také soubory byly změněny od ostatních. Zde je užitečný příkaz TortoiseSVN → Zjistit změny. Toto dialogové okno Vám ukáže každý soubor, který se jakýmkoli způsobem v pracovní kopii změnil, jakož i všechny soubory bez verze, které můžete mít.

Pokud kliknete na Zkontrolovat Úložiště můžete se také podívat na změny v úložišti. Tímto způsobem si můžete před aktualizací zkontrolovat, jestli existuje možnost konfliktu. Můžete také aktualizovat vybrané soubory z úložiště bez aktualizace celého adresáře. Ve výchozím nastavení tlačítko Zkontrolovat Úložiště pouze přinese vzdálený stav s hloubka získání pracovní kopie. Pokud chcete zobrazit všechny soubory a adresáře v úložišti a to i těch, které jste nezískali, pak budete muset držet klávesu **Shift**, když klikáte na tlačítko kontrolovat Úložiště.

Dialogové okno používá barevné označení pro zvýraznění stavu.

Modrá

Lokálně změněné položky.

Fialová

Přidané položky. Položky, které byly přidány s historií mají znamínko + ve sloupci Stavů obsahu a popisek Vám ukáže, odkud byla položka zkopírována.

Tmavě červená

Smazané nebo chybějící položky.

Zelená

Položky upravené lokálně a v úložišti. Změny budou v aktualizaci sloučeny. *Můžou* vyvolat při aktualizaci konflikt.

Jasně červená

Položky upravené lokálně a smazané v úložišti, nebo v úložišti změněné a lokálně smazané. Tyto *vyvolají* konflikt při aktualizaci.

Černá

Položky nezměněné a bez verze.

Toto je výchozí barevné schéma, ale můžete si ho přizpůsobit pomocí dialogového okna nastavení. Přečtěte si [4.30.1.4 – „Nastavení Barev TortoiseSVN“](#) pro více informací.

Položky, které byly překlopeny do jiné cesty úložiště, jsou rovněž uvedeny pomocí značky (s). Mohli jste něco překlomit při práci na větvi a zapomněli jste překlomit zpět do kmene. Toto je Váš varovný signál!

Z kontextového menu v dialogovém okně můžete zobrazit porovnání změn. Zkontrolujte lokální změny, které jste *l*y provedli s použitím Kontextové menu → Porovnat s originálem. Zkontrolujte změny, které v úložišti udělali jiní pomocí Kontextové menu → Zobrazit Rozdíly jako Sjedený rozdíl.

Můžete také vrátit změny v jednotlivých souborech. Pokud jste omylem smazali soubor, zobrazí se jako *Chybějící* a můžete použít *Vrátit* pro jeho obnovení.

Soubory ignorované a bez verze mohou být přesunuty odsud do koše s pomocí Kontextové menu → Smazat. Pokud chcete smazat soubory trvale (přeskočit koš) držte klávesu **Shift**, při klikání na Smazat.

Pokud chcete prozkoumat soubor podrobně, můžete jej odtud přetáhnout do jiné aplikace, třeba textového editoru nebo IDE.

Sloupce jsou přizpůsobitelné. Pokud kliknete pravým tlačítkem myši na libovolné záhlaví sloupce, zobrazí se kontextové menu umožňující vybrat, které sloupce se mají zobrazit. Můžete také změnit šířku sloupce pomocí tažení rukojeti, která se objeví při pohybu myši nad hranici sloupce. Tyto úpravy jsou zachovány, takže stejné záhlaví uvidíte příště.

Pokud pracujete na několika nesouvisejících úkolech najednou, můžete seskupit soubory do seznamu změn. Přečtěte si [4.4.2 – „Seznam Změn“](#) pro více informací.

Ve spodní části dialogového okna si můžete prohlédnout přehled rozsahu revizí úložiště používaných v pracovní kopii. Jedná se o *odevzdané* revize, ne o *aktualizované* revize; představují řadu revizí, kde byly tyto soubory naposledy odevzdány, nikoliv revize, kde byly aktualizovány. Všimněte si, že zobrazený rozsah revizí se vztahuje pouze na zobrazené položky, nikoli na celou pracovní kopii. Pokud chcete vidět tuto informaci pro celou pracovní kopii, musíte zaškrtnout políčko **Zobrazit nezměněné soubory**.



Tip

Pokud chcete mít přehledný pohled ve Vaší pracovní kopii, tj. zobrazení všech souborů a adresářů na všech úrovních hierarchie adresáře, pak dialogové okno **Zkontrolovat změny** je nejjednodušší způsob, jak toho dosáhnout. Stačí zaškrtnout **Zobrazit nezměněné soubory** k zobrazení všech souborů v pracovní kopii.



Oprava Přejmenovaných Externích

Někdy jsou soubory mimo Subversion přejmenovány a pak se objeví v seznamu souborů jako *chybějící* a soubor bez verze. Aby nedošlo ke ztrátě historie je třeba Subversion o připojení oznámit. Stačí jen vybrat starý (*chybějící*) a nový (*bez verze*) název (*chybí*) a použít Kontextové menu → **Opravit přesun** pro spárování těchto souborů jako přejmenování.

4.7.4. Zobrazení rozdílů

Často se chcete podívat dovnitř souborů, podíval se na to, co jste změnili. Tuto operaci lze provést výběrem souboru, který byl změněn a vybrat **Porovnat** z kontextového menu TortoiseSVN. Spustí se externí prohlížeč pro porovnání, který pak porovná aktuální soubor s původní kopií (*Původní revize*), který byl uložen po posledním získání nebo aktualizaci.



Tip

I když nejste v pracovní kopii nebo pokud se Vám více verzí souboru okolo povaluje, stále můžete zobrazit porovnání:

Vyberte dva soubory, které chcete v průzkumníkovy porovnat (např. pomocí **Ctrl** a myši) a zvolte Porovnat z kontextového menu TortoiseSVN. Soubor, na který bylo kliknuto naposled (ten se zaměřením, tj. tečkovaným obdélníkem), bude považován za ten pozdější.

4.8. Seznam Změn

V ideálním světě, vždycky pracujete pouze na jedné věci najednou a Vaše pracovní kopie obsahuje pouze jednu sadu logických změn. OK, zpátky do reality. Často se stává, že musíte pracovat na několika nesouvisejících úkolech najednou a když se podíváte do dialogového okna odevzdání, všechny změny jsou zamotané dohromady. Funkce *Seznam Změn* Vám pomůže seskupit soubory dohromady, takže je lépe vidět, co děláte. Samozřejmě to může fungovat pouze v případě, že se změny nepřekrývají. Pokud mají dva odlišné úkoly vliv na stejný soubor, neexistuje žádný způsob, jak změny oddělit.



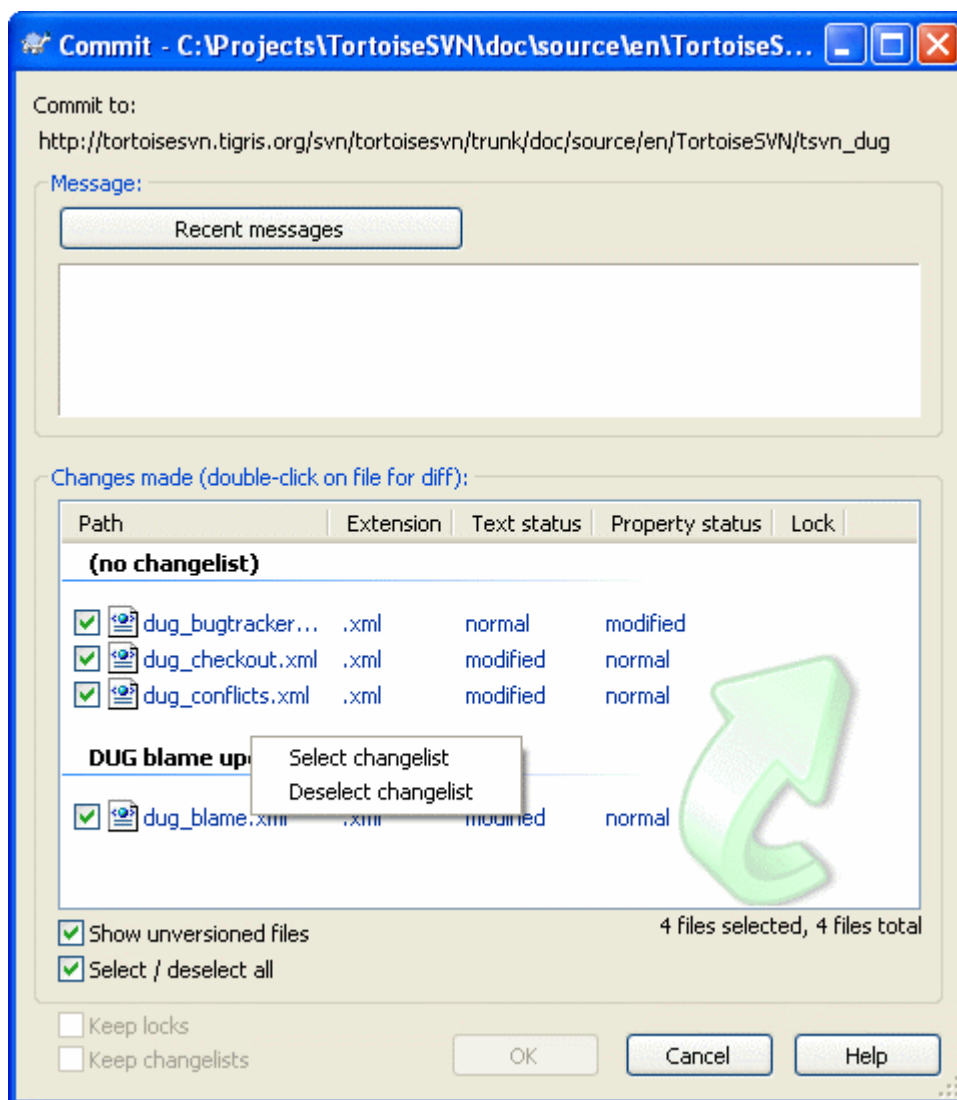
Důležité

The changelist feature in TortoiseSVN is only available in Windows XP and later, as it depends on a shell capability which is not present in Windows 2000. Sorry, but Win2K is really quite old now, so please don't complain.

Seznam změn můžete vidět na několika místech, ale nejdůležitější je v dialogovém okně odevzdání a Zjistit změny. Začneme v dialogovém okně Zjistit změny poté, co jste pracovali na několika funkcích a mnoho souborech. Když poprvé otevřete dialogové okno, všechny změněné soubory jsou uvedeny společně. Představte si, že teď chcete uspořádat věci a seskupit tyto soubory podle funkce.

Vyberte jeden nebo více souborů a použijte Kontextové menu → Přesunout do seznamu změn k přidání položky do seznamu změn. Zpočátku nebude žádný vytvořen, takže při zvolení této volby vytvoříte nový seznam změn. Dejte mu jméno, které popisuje, na co ho používáte a klepněte na tlačítko OK. Dialogové okno se nyní změní na zobrazení skupiny položek.

Jakmile jste vytvořili seznam změn, můžete do něho přetáhnout položky, ať už z jiného seznamu změn, nebo z Průzkumníka Windows. Přetažení z Průzkumníka může být užitečné, protože vám umožní přidat položky do seznamu změn před jejich úpravami. Dalo by se to provádět i z dialogového okna Zjistit změny, ale jen zobrazením všech nezměněných souborů.



Obrázek 4.14. Dialogové okno odevzdání se Seznamem Změn

V dialogovém oknu odevzdání můžete vidět ty stejné soubory, seskupené podle seznamu změn. Kromě toho, že dává bezprostřední vizuální indikaci seskupení, můžete také použít záhlaví skupin pro výběr, které soubory chcete odevzdat.

Na XP existuje kontextové menu po klepnutí pravým tlačítkem myši na záhlaví skupin, které Vám dává na výběr označit či odznačit všechny položky skupiny. Na Windows Vista však kontextové menu není nutné. Kliknutím na záhlaví skupiny vyberte všechny položky, pak zaškrtněte jednu z vybraných položek k označení všech.

TortoiseSVN rezervuje jeden jméno seznamu změn pro svoji vlastní potřebu a to `ignore-on-commit`. Používá se pro označení souborů v adresářích, které téměř nikdy nechcete odevzdat i když mají lokální změny. Tato funkce je popsána v [4.4.3 – „Vyjmout položky ze seznamu odevzdání“](#).

Když odevzdáte soubory patřící do seznamu změn, pak byste normálně očekávali, že členství v seznamu změn již není potřeba. Takže ve výchozím nastavení jsou soubory odstraněny ze seznamu změn automaticky po odevzdání. Pokud chcete zachovat soubor v jeho seznamu změn, použijte políčko **Ponechat seznamy změn** v dolní části dialogového okna odevzdání.



Tip

Seznam změn je čistě funkce místního klienta. Vytváření a odstraňování seznamů změn nebude mít vliv na úložiště, ani na pracovní kopii někoho jiného. Jsou jenom pohodlný způsob, jak organizovat soubory.

4.9. Dialogové Okno Záznamu Revizí

Pro každou změnu, kterou uděláte a odevzdáte, byste měli poskytnout záznam této změny. Tímto způsobem můžete později zjistit, jaké změny jste udělali a proč a máte podrobný záznam svého vývojového procesu.

Dialogové okno Záznamu Revizí načte pro Vás všechny ty zprávy záznamů a zobrazí je Vám. Okno je rozděleno do 3 panelů.

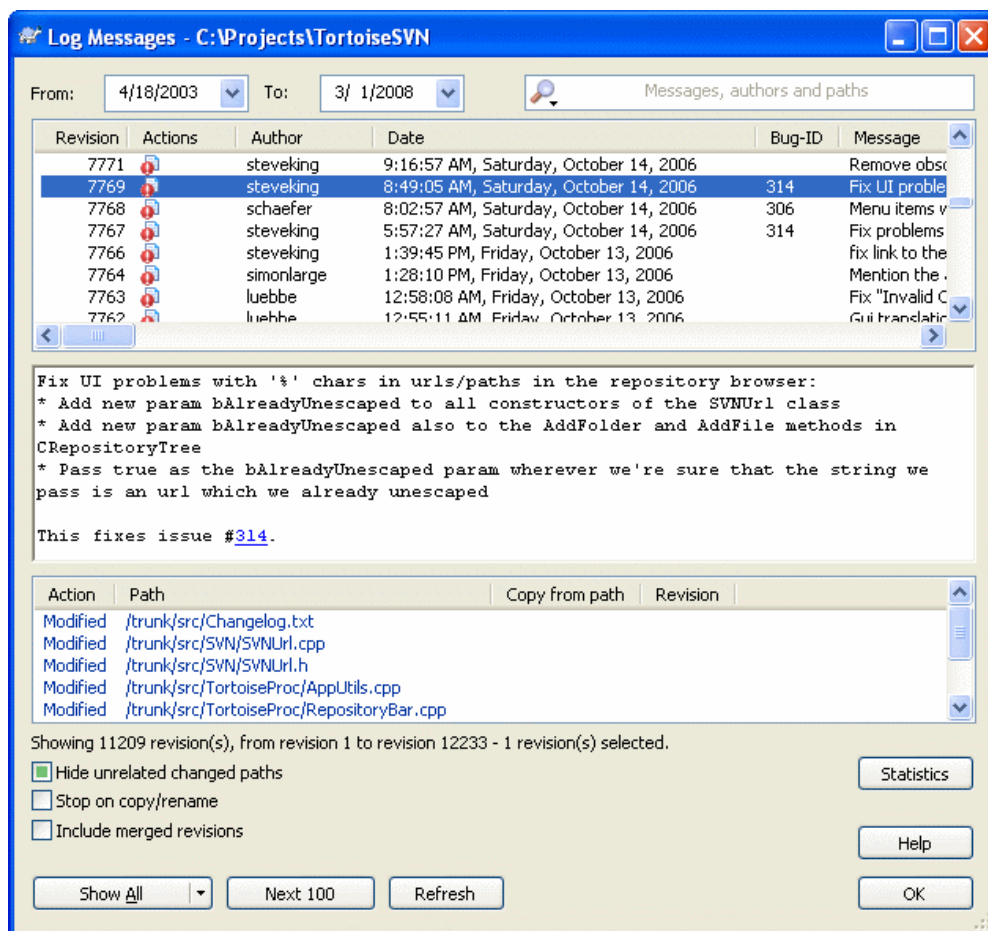
- Horní panel zobrazuje seznam revizí, kde změny v souboru/adresáři byly odevzdány. Tento přehled obsahuje datum a čas, osobu, která odevzdala revizi a začátek zprávy záznamu.

Modré řádky naznačují, že něco bylo zkopírováno do této vývojové linie (možná z větve).

- Prostřední panel zobrazuje úplnou zprávu záznamu pro vybranou revizi.
- Dolní panel zobrazuje seznam všech souborů a adresářů, které byly v rámci vybrané revize změněny.

Ale dělá mnohem víc než to - poskytuje příkazy kontextového menu, které můžete použít k získání ještě více informací o historii projektu.

4.9.1. Vyvolání Dialogového Okna Záznamu Revizí



Obrázek 4.15. Dialogové Okno Záznamu Revizí

Existuje několik míst, z nichž můžete zobrazit dialogové okno Záznamů:

- Z kontextového menu TortoiseSVN
- Ze stránky vlastností
- Z dialogového okna Průběhu po dokončení aktualizace. Pak dialogové okno Záznamů zobrazuje pouze ty revize, které byly změněny od Vaší poslední aktualizace

Pokud úložiště není k dispozici, uvidíte dialogové okno *Chcete jít offline?*, je popsáno v 4.9.10 – „Offline režim“.

4.9.2. Činnosti Záznamu Revizí

Horní panelu má sloupec Činnosti, který obsahuje ikony, které shrnují, co bylo učiněno v této revizi. Existují čtyři různé ikony, z nichž každá je uvedena v jejím vlastním sloupci.



Pokud revize změnila soubor nebo adresář, ikona *změněno* je zobrazena v prvním sloupci.



Pokud revizi přidala soubor nebo adresář, ikona *přidáno* je zobrazena ve druhém sloupci.

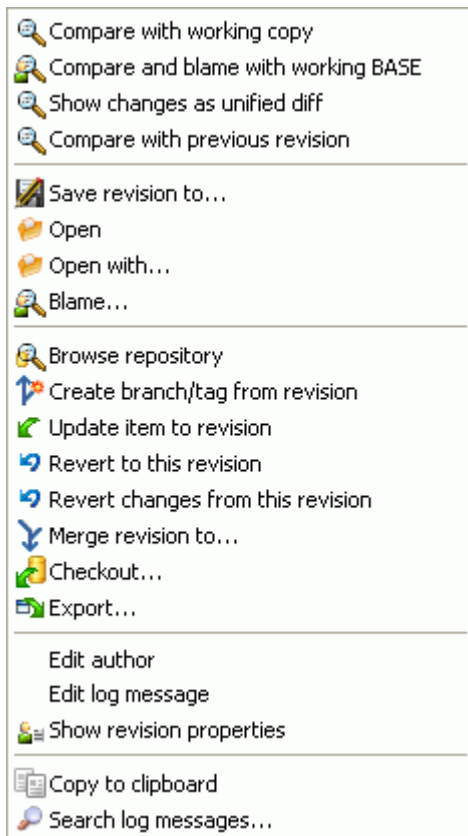


Pokud revize vymazala nějaký soubor nebo adresář, ikona *smazáno* je zobrazena ve třetím sloupci.



Pokud revize nahradila soubor nebo adresář, ikona *nahrazeno* je zobrazena ve čtvrtém sloupci.

4.9.3. Získání dodatečných informací



Obrázek 4.16. Horní Panel Dialogového okna Záznamu revizí s Kontextovým menu

Horní panel v dialogovém oknu Záznamů má kontextové menu, které umožňuje přístup k mnohem více informacím. Některé z těchto položek menu se zobrazí pouze tehdy, pokud je záznam zobrazen pro soubor a některé, jen když je záznam zobrazen pro adresář.

Porovnat s pracovní kopií

Porovnat vybranou revizi s Vaší pracovní kopií. Výchozí nástroj pro porovnávání rozdílů je TortoiseMerge, který je dodáván s TortoiseSVN. Pokud je dialogové okno Záznamů pro adresář, zobrazí Vám seznam změněných souborů a umožní Vám zkontrolovat změny každého souboru jednotlivě.

Porovnejte a obviňte s pracovním originálem

Blame the selected revision, and the file in your working BASE and compare the blame reports using a visual diff tool. Read [4.23.2 – „Obvinít Rozdíly“](#) for more detail. (files only).

Zobrazit změny jako sjednocený rozdíl

Zobrazit změny provedené ve vybrané revizi jako sjednocený rozdíl (formát GNU záplaty). Ukazuje pouze rozdíly v několika řádcích kontextu. Je těžší číst než vizuální porovnání souborů, ale zobrazí všechny změny souborů dohromady v jednodušším formátu.

Porovnat s předchozí revizí

Porovnat vybranou verzi s předchozí revizí. Funguje to podobným způsobem jako porovnání Vaší pracovní kopie. U složek tato možnost nejdříve zobrazí dialogové okno změnéné soubory, umožňující vybrat soubory k porovnání.

Porovnat a obvinít s předchozí revizí

Show the changed files dialog allowing you to select files. Blame the selected revision, and the previous revision, and compare the results using a visual diff tool. (folders only).

Uložit revize do...

Save the selected revision to a file so you have an older version of that file. (files only).

Otevřít / Otevřít v...

Open the selected file, either with the default viewer for that file type, or with a program you choose. (files only).

Obvinít...

Blame the file up to the selected revision. (files only).

Procházet úložiště

Otevřete Průzkumníka úložiště, pro náhled vybraného souboru nebo adresáře v úložišti tak, jak byl ve vybrané revizi.

Vytvořit větev/značku z revize

Vytvořit větev nebo značku z vybrané revize. To je užitečné např., pokud jste zapomněli vytvořit značku a již jste odevzdali nějaké změny, které se neměli dostat do onoho vydání.

Aktualizovat položku na revizi

Aktualizujte Vaši pracovní kopii na vybranou revizi. Užitečné, pokud chcete, aby Vaše pracovní kopii odrážela stav v minulosti, nebo v případě, že došlo k dalšímu odevzdání do úložiště a Vy chcete aktualizovat pracovní kopii postupně. Je nejlepší aktualizovat celý adresář ve Vaší pracovní kopii, ne jen jeden soubor, jinak Vaše pracovní kopie mohla být v nekonzistentním stavu.

Pokud chcete vrátit zpět předchozí změny trvale, použijte raději **Vrátit se k této revizi**.

Vrátit se k této revizi

Vrátit se k dřívější revizi. Pokud jste provedli několik změn a pak se rozhodnete, že raději se chcete vrátit tak, jak věci byly v revizi N, toto je příkaz, který potřebujete. Změny jsou vráceny zpět ve Vaší pracovní kopii, takže tato operace *nemá* vliv na úložiště, dokud neodevzdáte změny. Nezapomeňte, že toto vrátí zpět *všechny* změny provedené po vybranou revizi, nahradí tak soubor/adresář starší verzí.

Pokud Vaše pracovní kopie je v nezměněném stavu, po provedení této činnosti se Vaše pracovní kopie bude zobrazovat jako upravená. Pokud již máte místní změny, tento příkaz sloučí *zpětné* změny do pracovní kopie.

Co se děje uvnitř, je, že Subversion provádí zpětné sloučení všech změn provedených po vybranou revizi, rušící účinek předchozích odevzdání.

Pokud se po provedení této akce rozhodnete, že chcete *vzít zpět zrušení* a dostat Vaši pracovní kopii zpět na předchozí neupravený stav, měli byste použít TortoiseSVN → **Vrátit** v Průzkumníku Windows, což zruší místní úpravy provedené tímto obráceným sloučením.

Pokud chcete jednoduše zjistit, jak soubor nebo adresář vypadal v dřívější revizi, použijte **Aktualizovat na revizi** nebo místo toho **Uložit revizi jako....**

Vrátit změny z této revize

Vrátit zpět změny tak jak byly ve vybrané revizi provedeny. Tyto změny jsou vráceny zpět ve Vaší pracovní kopii, takže tato operace *nemá* vůbec vliv na úložiště! Nezapomeňte, že toto vrátí zpět změny provedené pouze v této revizi; nenahradí Vaši pracovní kopii celým souborem se starší revizí.

To je velmi užitečné pro vrácení dřívější změny, když jiné nesouvisející změny už bylo od té doby provedeny.

Pokud Vaše pracovní kopie je v nezměněném stavu, po provedení této činnosti se Vaše pracovní kopie bude zobrazovat jako upravená. Pokud již máte místní změny, tento příkaz sloučí *zpětné* změny do pracovní kopie.

Co se děje uvnitř, je, že Subversion provádí zpětné sloučení konkrétní revize, vrací její změny od předchozího odevzdání.

Můžete vzít *zpět vrácení* jak je popsáno výše v *Vrátit se k této revizi*.

Sloučit revize do...

Sloučit vybranou revizi(e) do jiné pracovní kopie. Dialogové okno pro výběr adresáře umožňuje zvolit pracovní kopii ke sloučení, ale poté už není žádné dialogové okno pro potvrzení, ani příležitost sloučení vyzkoušet. Je dobrý nápad sloučit do nezměněné pracovní kopie, takže můžete vrátit změny, pokud to nevyjde! To je užitečná vlastnost, pokud chcete sloučit vybrané revize z jedné větve do druhé.

Získání...

Provést nové získání vybraného adresáře ve vybrané revizi. Toto otevře dialogové okno k potvrzení URL a revize a výběr umístění pro získání.

Exportovat...

Exportovat vybraný soubor/adresář na vybranou revizi. Toto otevře dialogové okno k potvrzení URL a revize a výběr umístění pro export.

Upravit autora / záznam zprávy

Upravit záznam zpráv, nebo autora připojeného k předchozímu odevzdání. Přečtěte si [4.9.7 – „Změna Zpráv Záznamu a Autora“](#) abyste zjistili, jak to funguje.

Zobrazit vlastnosti revizí

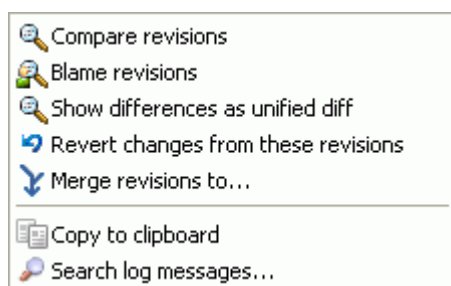
Prohlížet a upravovat jakékoli vlastnosti revizí, nejenom záznam zprávy a autora. Viz [4.9.7 – „Změna Zpráv Záznamu a Autora“](#).

Zkopírovat do schránky

Kopírovat podrobnosti záznamu o vybraných revizích do schránky. Toto zkopíruje číslo revize, autora, datum, záznam zprávy a seznam změněných položek pro každou revizi.

Prohledávat záznamy zpráv...

Hledat záznam zprávy zadaným textem. Toto prohledá záznam zpráv, které jste zadali a také souhrn činností vytvořených Subversion (uvedeno v dolní části panelu). Vyhledávání není citlivé na velikost písmen.



Obrázek 4.17. Horní panel Kontextového Menu pro 2 vybrané Revize

Zvolíte-li dvě revize najednou (s použitím běžného **Ctrl**-modifikátor), kontextové menu se změní a dá Vám méně možností:

Porovnat revize

Porovnejte dvě vybrané revize pomocí nástroje pro porovnání rozdílů. Výchozím takovýto nástrojem je TortoiseMerge, který je dodáván spolu s TortoiseSVN.

Pokud zvolíte tuto možnost pro adresář, další dialogové okno zobrazí seznam změněných souborů a nabídne Vám další možnosti porovnání. Přečtěte si více o dialogovém okně Porovnání Revizí v [4.10.3 – „Porovnávání adresářů“](#).

Obvinít revize

Obvinít dvě revize a porovnat hlášení o obvinění pomocí nástroje pro porovnání rozdílů. Přečtěte si [4.23.2 – „Obvinít Rozdíly“](#) pro další podrobnosti.

Zobrazit rozdíly jako sjednocený rozdíl

Zobrazit rozdíly mezi dvěma vybranými revizemi jako soubor sjednoceného rozdílu. Toto funguje pro soubory i adresáře.

Zkopírovat do schránky

Zkopírovat zprávy záznamu do schránky, jak je popsáno výše.

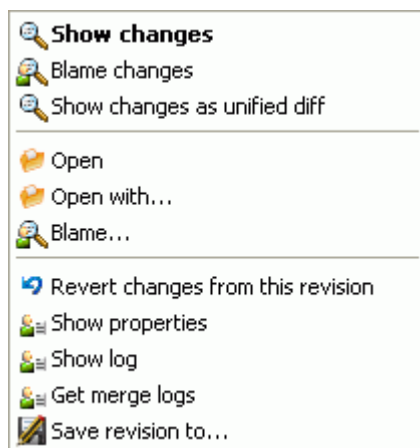
Prohledávat záznamy zpráv...

Vyhledat zprávy záznamu jak je popsáno výše.

Vyberete-li dva nebo více revizí (s použitím běžného **Ctrl** nebo **Shift** modifikátorů), kontextové menu zahrne položku Vrátit všechny změny, které byly provedeny ve vybraných revizích. To je nejjednodušší způsob, jak vrátit zpět skupinu revizí na jednu.

Můžete také zvolit sloučení vybraných revizí do jiné pracovní kopie, jak je popsáno výše.

Pokud všechny vybrané revize mají téhož autora, můžete upravit autora všech těchto revizí na jednu.



Obrázek 4.18. Dolní panel dialogového okna Záznamů s Kontextovým Menu

Dolní panel dialogového okna Záznamu má také kontextové menu, které Vám umožní

Zobrazit změny

Show changes made in the selected revision for the selected file. This context menu is only available for files shown as *modified*.

Obvinít změny

Obvinít vybranou revizi a předchozí revizi ve vybraném souboru a porovnat hlášení o obvinění pomocí nástroje pro zobrazení. Přečtěte si [4.23.2 – „Obvinít Rozdíly“](#) pro další podrobnosti.

Zobrazit jako sjednocený rozdíl

Zobrazit soubor změn ve formátu sjednoceného rozdílu. Toto kontextové menu je k dispozici pouze pro soubory zobrazené jako *změněné*.

Otevřít / Otevřít v...

Otevře zvolený soubor a to buď ve výchozím prohlížeči pro tento typ souboru, nebo s programem, který vyberete.

Obvinit...

Otevře Dialogové okno Obvinění, což vám umožní obvinit až po vybranou revizi.

Vrátit změny z této revize

Vrátit změny vybraného souboru v té revizi.

Zobrazit vlastnosti

Zobrazit Subversion vlastnosti vybrané položky.

Zobrazit záznam

Zobrazit záznam revize pro jeden vybraný soubor.

Získat záznamy o sloučení

Zobrazit záznam revize pro jeden vybraný soubor, včetně sloučených změn. Více se dozvíte na [4.9.6 – „Funkce Sledování Sloučení“](#).

Uložit revize do...

Uloží vybranou revizi do souboru, takže máte starší verzi tohoto souboru.



Tip

Můžete si všimnout, že někdy budeme odkazovat na změny a jindy na rozdíly. V čem je to jiné?

Subversion uses revision numbers to mean 2 different things. A revision generally represents the state of the repository at a point in time, but it can also be used to represent the changeset which created that revision, eg. „Done in r1234“ means that the changes committed in r1234 implement feature X. To make it clearer which sense is being used, we use two different terms.

Zvolíte-li dvě revize N a M, kontextového menu Vám nabídne zobrazení *Rozdílu* mezi těmito dvěma revizemi. V jazyce Subversion to je `diff -r M:N`.

Pokud vyberete jednu revize N kontextové menu, Vám nabídne zobrazení *změn* provedených v této revizi. V jazyce Subversion to je `diff -r N-1:N` nebo `diff -c N`.

Spodní panel zobrazuje soubory změněné ve všech vybraných revizích, takže kontextové menu Vám vždy nabídne ukázat *změny*.

4.9.4. Zobrazení více zpráv záznamu

Dialogové okno Záznamů ne vždy zobrazí všechny změny, je to z několika důvodů:

- Ve Velkém úložišti může existovat stovky nebo tisíce změn a všechny je získat může trvat delší dobu. Běžně Vás budou zajímat pouze nedávné změny. Ve výchozím nastavení je počet zpráv záznamů omezen na 100, ale můžete tuto hodnotu změnit v TortoiseSVN → Nastavení ([4.30.1.2 – „Dialogové Okno Nastavení TortoiseSVN 1“](#)),
- Když je políčko Zastavit na kopírování/přejmenování zaškrtnuto, Zobrazit záznamy se zastaví na místě, kde vybraný soubor nebo adresář byl zkopírován z jiného místa v rámci úložiště. To může být užitečné, když prohlížíte větve (nebo značky), protože se zastaví u kořene té větve a poskytuje rychlou indikaci změn provedených pouze v té větvi.

Za normálních okolností budete chtít nechat tuto možnost odškrtnutou. TortoiseSVN si pamatuje stav tohoto políčka, takže bude Vaše preference respektovat.

Je-li dialogové okno Zobrazit záznamy spuštěno z dialogového okna Sloučení, je políčko vždy ve výchozím nastavení zaškrtnuto. To proto, že sloučení se nejčastěji dívá na změny ve větvích a jít za kořen větve nemá v této situaci smysl.

Všimněte si, že Subversion v současné době zavádí přejmenování jako pár kopírování/smazání, takže přejmenování souboru nebo adresáře také způsobí, že displej záznamů se zastaví, pokud je tato volba zaškrtnuta.

Pokud chcete vidět více zpráv záznamů, klepněte na tlačítko **Dalších 100** pro získání dalších 100 zpráv záznamů. Toto můžete opakovat, kolikrát chcete.

Vedle tohoto tlačítka je tlačítko s mnoha funkcemi, které si pamatuje poslední možnost, kterou jste v něm použili. Klikněte na šipku pro zobrazení dalších nabízených možností.

Použijte **Zobrazit Rozsah...** pokud chcete zobrazit konkrétní rozsah revizí. Dialogové okno Vás pak vyzve k zadání začátku a konce revize.

Použijte **Zobrazit všechny**, pokud chcete vidět *všechny* zprávy záznamu z revize HEAD zpátky k revizi 1.

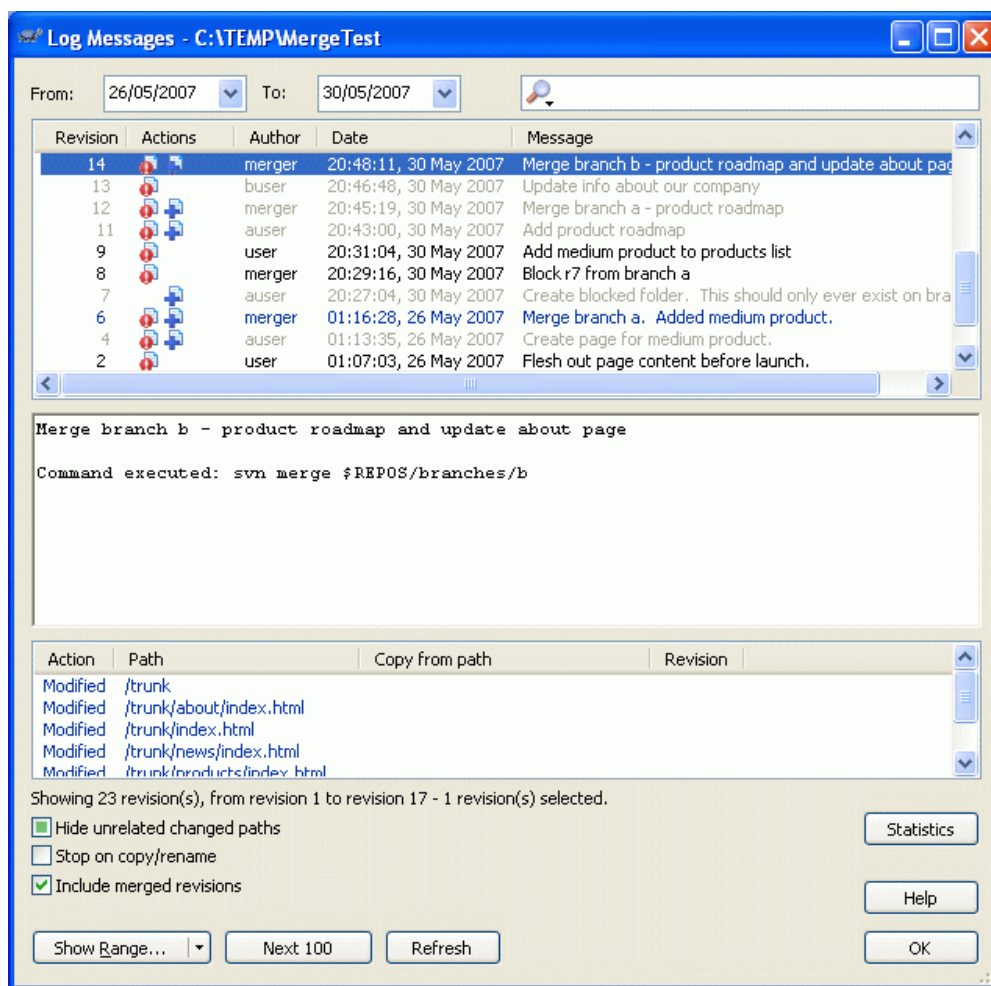
4.9.5. Aktuální Revize Pracovní Kopie

Vzhledem k tomu, že Dialogové okno Záznamů zobrazuje záznamy z HEAD, ne z aktuální revize pracovní kopie, často se stává, že v záznamu jsou zprávy vypovídající o obsahu, který dosud nebyl aktualizován ve Vaší pracovní kopii. Pro jasnější představu je zpráva odevzdání, která odpovídá revizi ve Vaší pracovní kopii, zobrazena tučně.

When you show the log for a folder the revision highlighted is the highest revision found anywhere within that folder, which requires a crawl of the working copy. This can be a slow operation for large working copies, and the log messages are not displayed until the crawl completes. If you want to disable or limit this feature you need to set a registry key `HKCU\Software\TortoiseSVN\RecursiveLogRev` as described in [4.30.10 – „Registry Settings“](#).

4.9.6. Funkce Sledování Sloučení

Subversion 1.5 a novější uchovává záznamy o sloučení pomocí vlastností. To nám umožňuje získat podrobnější historii sloučených změn. Například, pokud jste vyvinuli novou funkci ve větví a pak sloučili tuto větev zpět do kmene, tak se vývoj funkce objeví v záznamu kmene jako jediné odevzdání pro sloučení i když tam mohlo být 1000 odevzdání v průběhu vývoje větve.



Obrázek 4.19. Dialogové Okno Záznamu Zobrazující Sledování Sloučení Revizí

Pokud chcete vidět detaily, které revize byly sloučeny jako součást tohoto odevzdání, použijte políčko **Zahrnout sloučené revize**. To znovu stáhne zprávy záznamu, ale také proloží zprávy záznamů z revize, které byly sloučeny. Sloučené revize jsou zobrazeny v šedé barvě, protože představují změny provedené v jiné části stromu.

Samozřejmě, slučování není nikdy jednoduché! Během vývoje funkce na větvi, budou pravděpodobně i občasné sloučení zpět do kmene, aby větve byly v synchronizaci s hlavní linií kódu. Takže historii sloučení větve bude také zahrnovat další vrstvu historie sloučení. Tyto různé vrstvy jsou uvedeny v dialogovém okně záznamu pomocí úrovní odsazení.

4.9.7. Změna Zpráv Záznamu a Autora

Vlastnosti revize jsou zcela odlišné od Subversion vlastností každé položky. vlastrev jsou popisné položky, které jsou spojeny s jedním konkrétním číslem revize v úložišti, jako třeba zpráva záznamu, datum odevzdání a jméno vývojáře (autor).

Někdy můžete chtít změnit zprávu záznamu, kterou jste jednou zadali, možná proto, že je v ní pravopisná chyba, nebo chcete zprávu zlepšit nebo z jiných důvodů změnit. Nebo chcete změnit autora odevzdání, protože jste zapomněli nastavit ověřování nebo...

Subversion umožňuje měnit vlastnosti revizí kdykoliv budete chtít. Ale protože takové změny nelze vrátit zpět (tyto změny nemají verzi), tato funkce je ve výchozím nastavení zakázána. Pro povolení, musíte nastavit hák před-vlastrev-změnou. Více informací je v kapitole týkající se [Hákové skripty](http://svnbook.red-bean.com/en/1.5/svn.reposadmin.create.html#svn.reposadmin.create.hooks) [http://svnbook.red-bean.com/en/1.5/svn.reposadmin.create.html#svn.reposadmin.create.hooks] v knize

Subversion pro podrobnosti o tom, jak to provést. Přečtěte si 3.3 – „Hook skripty na straně serveru“ pro některé další poznámky o zavedení háků na počítači s Windows.

Jakmile jste nastavili váš server s požadovanými háky, můžete změnit autora a zprávu záznamu (nebo jakoukoli jinou vlastev) jakékoli revize, pomocí kontextového menu v horním panelu dialogového okna Záznamu. Můžete také upravit zprávu záznamu pomocí kontextového menu ve středním panelu.



Varování

Vzhledem k tomu, že vlastnosti revizí Subversion nemají verzi, změny těchto vlastností (např. vlastnost `svn:log` zprávy odevzdání) *navždy* přepíše předchozí hodnotu této vlastnosti.

4.9.8. Filtrování Zpráv Záznamu

Pokud chcete omezit zprávy záznamu na ty, o které máte zájem, tak spíše než procházet seznam stovek zpráv, můžete použít kontroly filtru v horní části dialogového okna záznamů. Kontrola počátečního a konečného data umožňuje omezit výstup na známé časové období. Pole vyhledávání umožňuje zobrazit pouze ty zprávy, které obsahují konkrétní frázi.

Click on the search icon to select which information you want to search in, and to choose *regex* mode. Normally you will only need a simple text search, but if you need to more flexible search terms, you can use regular expressions. If you hover the mouse over the box, a tooltip will give hints on how to use the regex functions. You can also find online documentation and a tutorial at <http://www.regular-expressions.info/>. The filter works by checking whether your filter string matches the log entries, and then only those entries which *match* the filter string are shown.

To make the filter show all log entries that do *not* match the filter string, start the string with an exclamation mark (!). For example, a filter string `!username` will only show those entries which were not committed by `username`.

Upozorňujeme, že tyto filtry působí pouze na již stáhnuté zprávy. Nemají žádný vliv na stahování zpráv z úložiště.

You can also filter the path names in the bottom pane using the **Hide unrelated changed paths** checkbox. Related paths are those which contain the path used to display the log. If you fetch the log for a folder, that means anything in that folder or below it. For a file it means just that one file. The checkbox is tristate: you can show all paths, grey out the unrelated ones, or hide the unrelated paths completely.

Někdy Vaše pracovní postupy budou vyžadovat, aby zprávy záznamů měly určitý formát, což znamená, že text popisující změny, není vidět ve zkráceném shrnutí uvedený v horním panelu. Vlastnost `tsvn:logsummary` může být použita k extrakci části zprávy záznamu k zobrazení v horním panelu. Přečtěte si 4.17.2 – „TortoiseSVN Vlastnosti projektu“ pro více informací, jak použít tuto vlastnost.



Žádné formátování záznamu z Průzkumníka úložiště

Vzhledem k tomu, že formátování závisí na přístupu k vlastnostem Subversion, výsledky uvidíte pouze při použití získané pracovní kopie. Vzdálené získávání vlastnosti je pomalá operace, takže tuto funkci v akci neuvídíte v Průzkumníku úložiště.

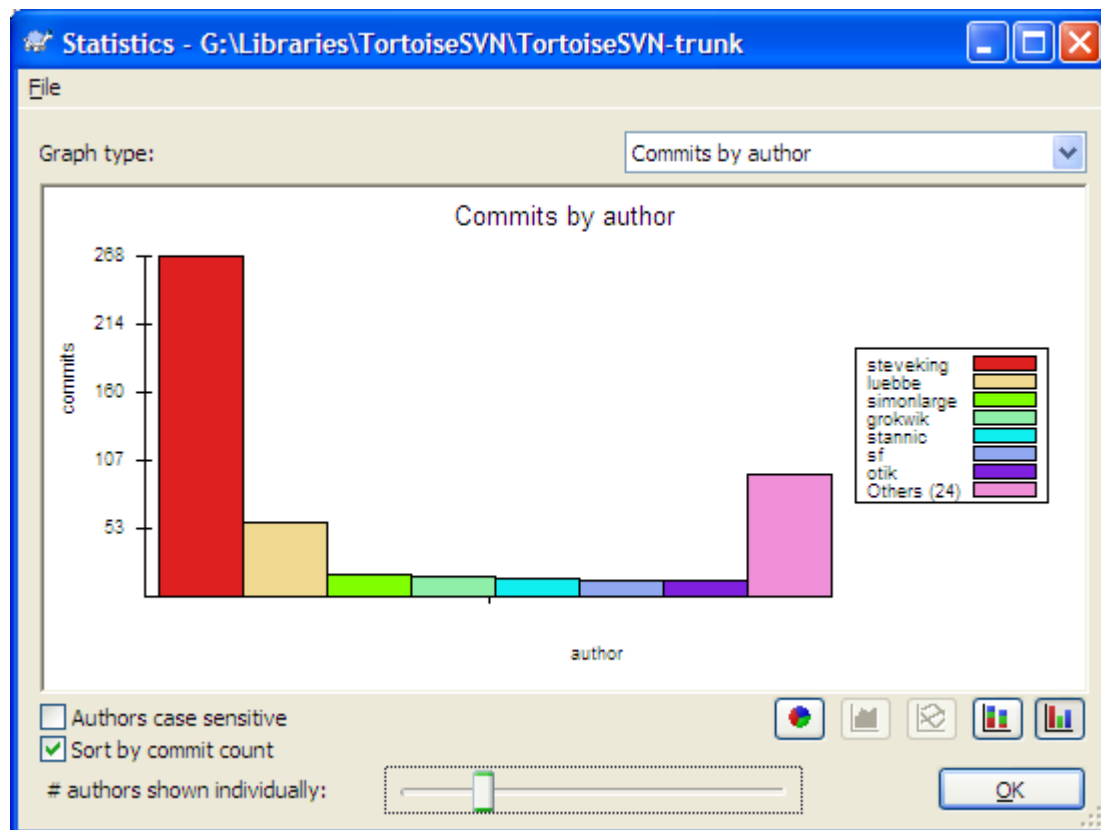
4.9.9. Statistické informace

Tlačítkem **Statistika** vyvoláte okno zobrazující některé zajímavé informace o revizích zobrazených v dialogovém okně Záznamu. Ukazuje, kolik autorů pracovalo, kolik odevzdání udělali, pokrok v týdnů a ještě mnohem více. Nyní můžete na první pohled vidět, kdo pracoval nejtvrději a kdo se flákal ;-)

4.9.9.1. Stránka Statistik

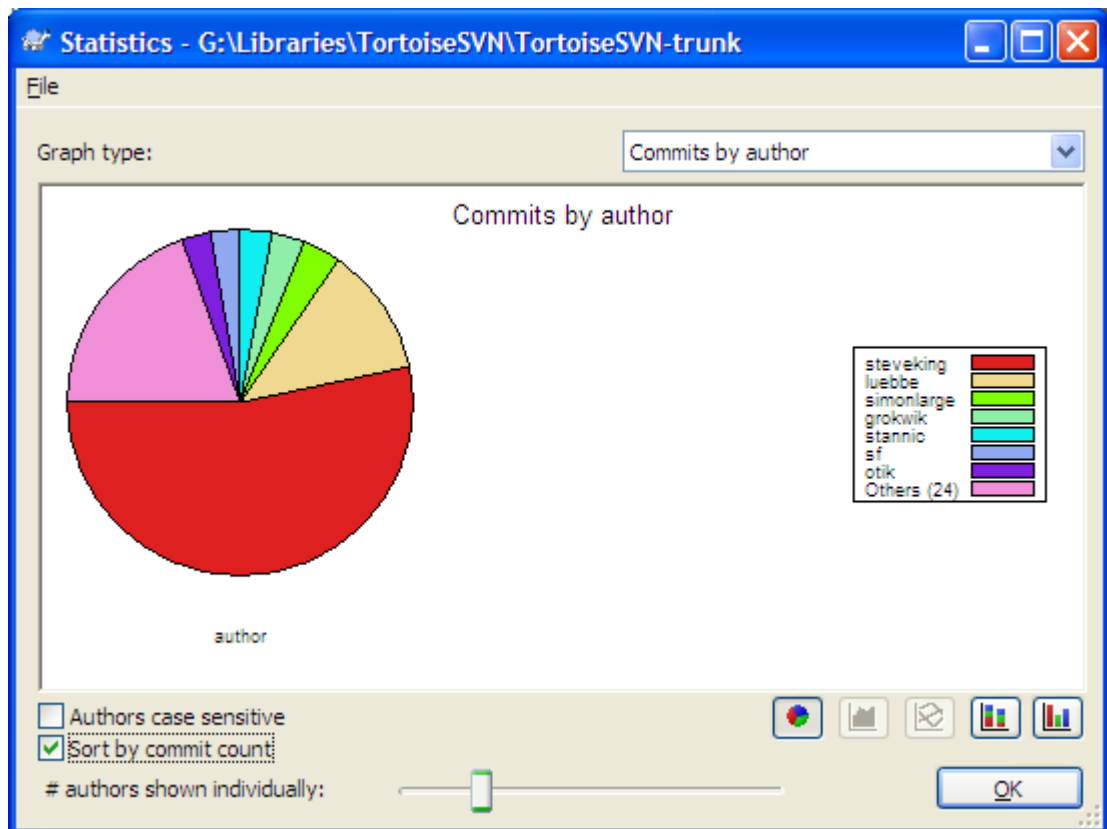
Tato stránka Vám dá všechna čísla, na která můžete pomyslet, zejména období a počet pokrytých revizí a některé min/max/průměrné hodnoty.

4.9.9.2. Stránka Odevzdání podle Autora



Obrázek 4.20. Sloupcový graf Odevzdání-podle-Autora

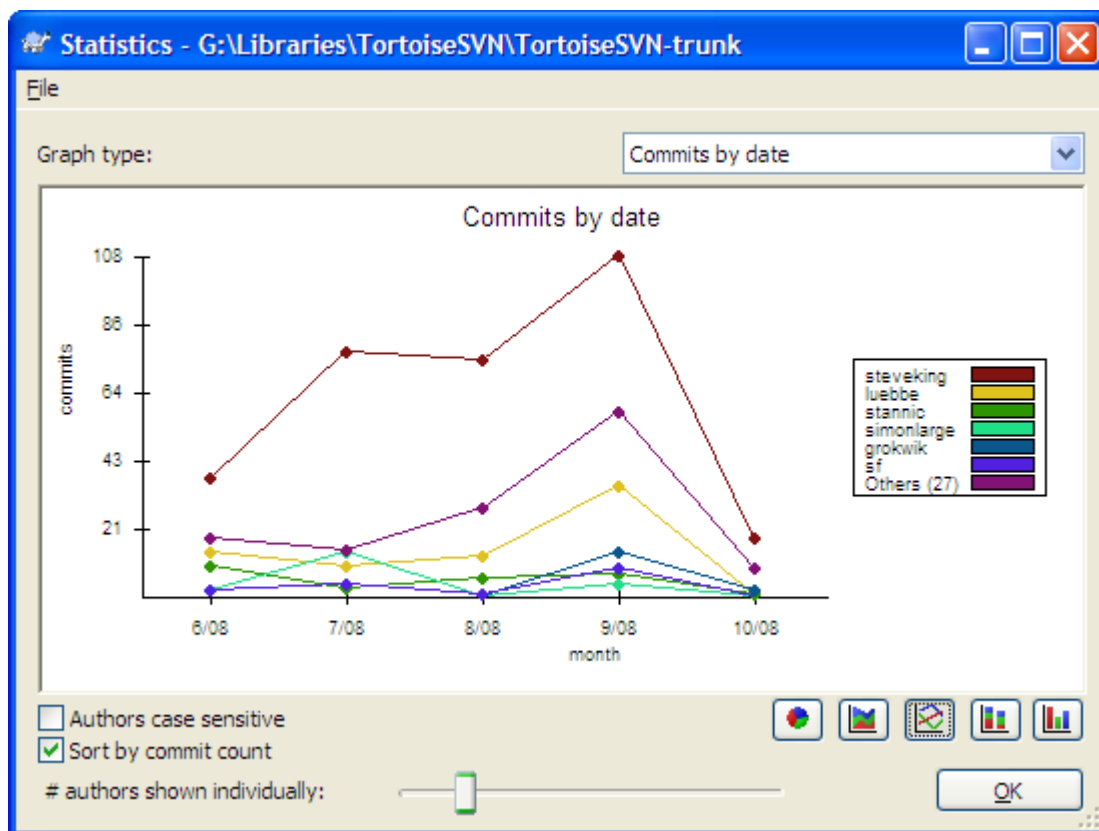
Tento graf ukazuje, kteří autoři byli v projektu aktivní jako jednoduchý sloupcový graf, skládaný sloupcový graf nebo koláčový graf.



Obrázek 4.21. Koláčový graf Odevzdání-podle-Autora

Pokud existuje několik hlavních autorů a mnoho menších přispěvatelů, množství malých segmentů může způsobit, že graf je obtížnější číst. Posuvník v dolní části umožňuje nastavit prahové hodnoty (jako procento z celkových odevzdání), pod níž je jakákoli aktivita seskupena do kategorie *Ostatní*.

4.9.9.3. Stránka odevzdání podle data



Obrázek 4.22. Graf odevzdání podle data

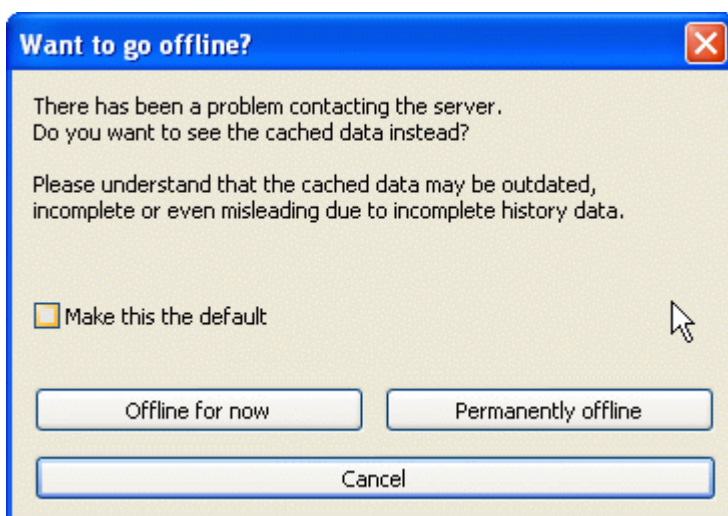
Tato stránka Vám dává grafické znázornění činnosti projektu, pokud jde o počet odevzdání a autorů. To dává určitou představu o tom, kdy se na projektu pracuje a kdo v které době pracoval.

Když je několik autorů, budete mít v grafu mnoho řádků. Zde jsou k dispozici dva pohledy: *normální*, kde každá aktivita autora je vzhledem k základní linii, a *skládání*, kde každá aktivita autora je vzhledem k řádku pod ním. Druhá z možností zabraňuje překřížení linky, což může udělat graf čitelnějším, ale je méně snadné vidět výkon jednoho autora.

Ve výchozím nastavení je analýza citlivá na velikost písmen, takže uživatelé JosefNovák a JosefNovák jsou považováni za různé autory. Nicméně, v mnoha případech uživatelská jména nejsou na velikost písmen citlivá a někdy jsou zadávány nedůsledně, takže možná budete chtít, aby PetrMilý a petrMilý byli považováni za stejnou osobu. Zaškrtněte políčko **Rozlišovat velikost písmen autorů**, ke kontrole, jak je toto řešeno.

Nezapomeňte, že statistiky se týkají stejného období jako dialogové okno Záznamu. Pokud okno zobrazuje pouze jednu revizi pak Vám statistiky moc nepoví.

4.9.10. Offline režim



Obrázek 4.23. Dialogové okno Jít Offline

Pokud je server nedostupný a Vy máte zapnuto uchování záznamů, můžete použít dialogové okno záznamů a graf revizí v režimu offline. Využívá se dat z úschovy, což Vám umožní pokračovat v práci, i když informace nemusí aktuální, nebo dokonce úplná.

Zde máte tři možnosti:

Pro teď offline

Dokončit aktuální operaci v režimu offline, ale zkusit úložiště znovu v okamžiku, kdy je požadován záznam dat.

Trvale offline

Zůstat v režimu offline, dokud není o kontrolu úložiště výslovně požádáno. Viz [4.9.11 – „Obnovení zobrazení“](#).

Zrušit

Pokud nechcete pokračovat v operaci s možností zastaralých dat, stačí ji zrušit.

Políčko **Nastavit jako výchozí** zabraňuje tomuto dialogovému oknu se znovu objevit a vždy vybere tu možnost, kterou Vy zvolíte jako další. Můžete stále měnit (nebo odebrat) výchozí nastavení po zaškrtnutí v TortoiseSVN → Nastavení.

4.9.11. Obnovení zobrazení

Pokud chcete opět zkontrolovat server pro novější zprávy záznamů, můžete jednoduše aktualizovat zobrazení pomocí **F5**. Pokud používáte úschovu záznamu (standardně zapnuto), tuto zkontroluje úložiště pro novější zprávy a načte pouze nové. Pokud je úschova záznamů v režimu offline, tímto se také pokusí vrátit zpět do režimu online.

Pokud používáte úschovu záznamů a myslíte si, že obsah zprávy, nebo autor se mohli změnit, můžete použít **Shift-F5** nebo **Ctrl-F5** pro znovu načtení zobrazených zpráv ze serveru a aktualizovat úschovu záznamů. Nezapomeňte, že toto se týká pouze právě zobrazených zpráv a nenaruší celou úschovu pro toto úložiště.

4.10. Zobrazit Rozdíly

Jeden z nejčastějších požadavků ve vývoji projektu je vidět, co se změnilo. Možná se budete chtít podívat na rozdíly mezi dvěma revizemi stejného souboru, nebo rozdíly mezi dvouma samostatnými soubory.

TortoiseSVN nabízí vestavěný nástroj s názvem TortoiseMerge k prohlížení rozdílů textových souborů. K prohlížení rozdílů obrázků, TortoiseSVN má také nástroj s názvem TortoiseIDiff. Samozřejmě, že můžete používat svůj vlastní oblíbený program pro porovnání, pokud se Vám líbí.

4.10.1. Rozdíly souboru

Místní změny

Pokud chcete vidět, jaké změny jste *ly* provedli ve Vaší pracovní kopii, stačí použít kontextové menu průzkumníka a zvolte TortoiseSVN → Porovnat.

Porovnat s jinou větví/značkou

Pokud chcete vidět, co se změnilo v kmeni (pokud pracujete ve větvi), nebo v konkrétní větvi (pokud pracujete v kmeni), můžete použít kontextového menu Průzkumníka. Podržte klávesu **Shift**, při klepnutí pravým tlačítkem myši na soubor. Pak vyberte TortoiseSVN → Porovnat s URL. V následujícím dialogovém okně zadejte adresu URL v úložišti, se kterou chcete místní soubor porovnat.

Můžete také použít průzkumníka úložiště a vybrat dva stromy k porovnání, třeba dvě značky, nebo větev/značku a kmen. Kontextové menu Vám je umožňuje porovnat s použitím Porovnat revize. Přečtěte si více v [4.10.3 – „Porovnávání adresářů“](#).

Rozdíl od předchozí revize

Pokud chcete vidět rozdíl mezi konkrétní revizí a Vaší pracovní kopie, použijte dialogové okno Záznamu Revizí, vyberte revizi, kterou chcete porovnat, pak z kontextového menu zvolte Porovnat s pracovní kopií.

Pokud chcete vidět rozdíl mezi poslední odevzdanou revizí a Vaší pracovní kopií, za předpokladu, že pracovní kopie nebyla změněna, stačí kliknout pravým tlačítkem myši na soubor. Pak vyberte TortoiseSVN → Porovnat s předchozí verzí. Tímto provedete porovnání mezi revizí před posledním datem odevzdání (který byl zaznamenán ve Vaší pracovní kopii) a pracovním originálem. Zde můžete vidět poslední změny v tomto souboru, které ho přivedly do stavu, který vidíte ve Vaší pracovní kopii. Nikdy nezobrazí změny novější než ty ve Vaší pracovní kopii.

Rozdíl mezi dvěma předchozími revizemi

Pokud chcete vidět rozdíly mezi dvěma revizemi, které jsou již odevzdané, použijte dialogové okno Záznam Revizí a vyberte dvě revize, které chcete porovnat (za použití obvyklých **Ctrl**-Modifikátor). Pak vyberte z kontextového menu Porovnat revize.

Pokud jste toto provedli ze záznamu revize pro adresář, zobrazí se dialogové okno Porovnat Revize, zobrazující seznam změněných souborů v tomto adresáři. Přečtěte si více na [4.10.3 – „Porovnávání adresářů“](#).

Všechny změny provedené v odevzdání

Pokud chcete vidět změny provedené ve všech souborech v určité revizi zobrazené najednou, můžete použít výstup sjednoceného rozdílu (záplata formátu GNU). Ukazuje pouze rozdíly s několika řádky kontextu. Je těžší číst než vizuální porovnání souborů, ale zobrazí všechny změny dohromady. V dialogovém okně Záznam Revizí vyberte revizi, kterou chcete a zvolte Zobrazit Rozdíly jako Jednotnou změnu v kontextovém menu.

Rozdíl mezi soubory

Pokud chcete vidět rozdíly mezi dvěma různými soubory, můžete to udělat přímo v Průzkumníku výběrem obou souborů (pomocí obvyklých **Ctrl**-Modifikátor). Pak z kontextové nabídky Průzkumníka zvolte TortoiseSVN → Porovnat.

Rozdíl mezi souborem/adresářem pracovní kopie a URL

Pokud chcete vidět rozdíly mezi souborem ve Vaší pracovní kopii a souborem v jakémkoli Subversion úložišti, můžete to provést přímo v Průzkumníkovi vybráním souboru a přidržením

klávesy **Shift** při kliknutí pravým tlačítkem myši pro získání kontextového menu. Vyberte **TortoiseSVN → Porovnat s URL**. Toto můžete udělat i pro adresář pracovní kopii. TortoiseMerge zobrazí tyto rozdíly stejným způsobem, jako zobrazuje soubor záplaty - seznam změněných souborů které si můžete prohlédnout jeden po druhém.

Rozdíl v informacích o obvinění

Pokud chcete vidět nejenom rozdíly ale také autory, revize a data, kdy byly změny provedeny, můžete spojit hlášení o porovnání a obvinění v rámci dialogového okna záznamu revizí. Přečtěte si [4.23.2 – „Obvinit Rozdíly“](#) pro další podrobnosti.

Rozdíl mezi adresáři

Vestavěné nástroje dodávané s TortoiseSVN nepodporují porovnávání rozdílů mezi hierarchií adresářů. Ale pokud máte externí nástroj, který podporuje tuto funkci, můžete ho místo něj použít. V [4.10.5 – „Externí Nástroje Porovnání/Sloučení“](#) Vám řekneme o některé nástroje, které jsme používali.

Pokud máte nastaven porovnávací nástroj třetí osoby, můžete použít **Shift** při výběru příkazu Porovnat pro použití jiného nástroje. Přečtěte si [4.30.5 – „Nastavení Externího Programu“](#) k zjištění ke konfiguraci dalších nástrojů porovnání.

4.10.2. Možnosti konců řádků a mezer

Někdy v životě projektu byste mohli změnit konce řádků z CRLF na LF, nebo můžete změnit odsazení sekce. Bohužel to bude znamenat velký počet změněných řádků, i když nedojde k žádné změně ve významu tohoto kódu. Možnosti zde Vám pomůžou zvládnout tyto změny, pokud jde o srovnání a použití rozdílů. Tato nastavení jsou v dialogových oknech **Sloučit** a **Obvinit**, stejně jako v nastavení TortoiseMerge.

Ignorovat konce řádků vyřadí změny, které jsou pouze z důvodu rozdílu stylu v konci řádků.

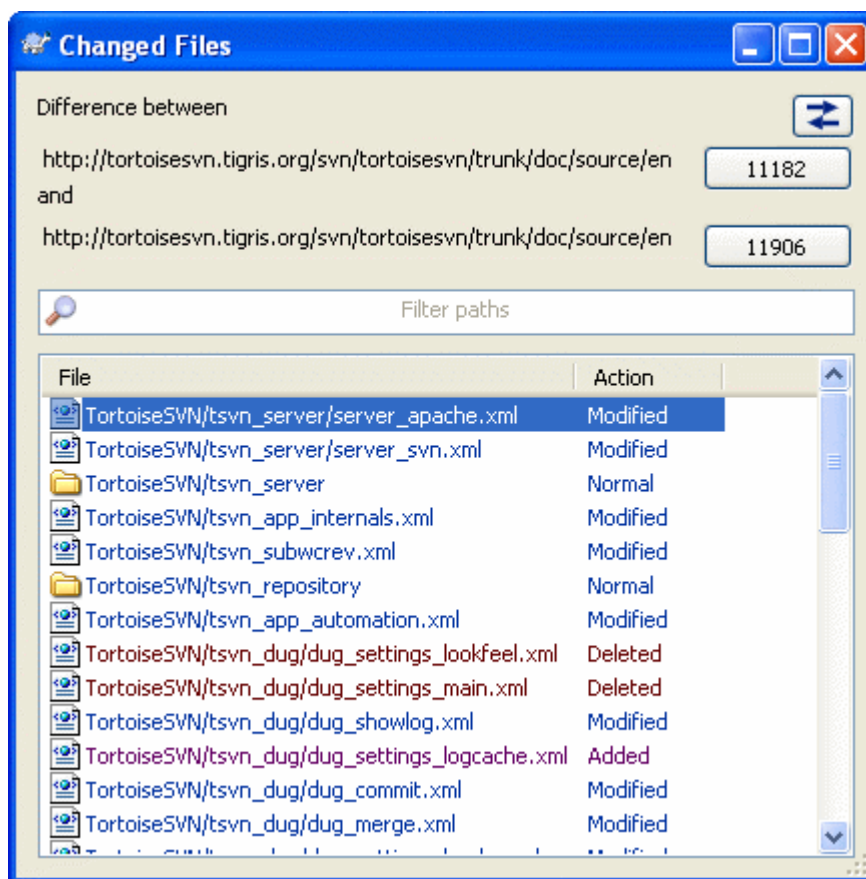
Porovnat mezery zahrnuje všechny změny v odsazení a mezer za sebou jako přidané/odstraněné řádky.

Ignore whitespace changes excludes changes which are due solely to a change in the amount or type of whitespace, eg. changing the indentation or changing tabs to spaces. Adding whitespace where there was none before, or removing a whitespace completely is still shown as a change.

Ignorovat všechny mezery vyřadí všechny změny způsobené mezerami.

Samořejmě, že jakýkoli řádek se změněným obsahem je vždy zahrnut v porovnání.

4.10.3. Porovnávání adresářů



Obrázek 4.24. Dialogové okno Porovnání Revizí

Když vyberete dva stromy v průzkumníku úložiště, nebo pokud zvolíte dvě revize adresáře v dialogovém okně záznamů můžete **Kontextové menu → Porovnat revize**.

Toto dialogové okno zobrazí seznam všech souborů, které se změnily, a umožňuje Vám je individuálně porovnat nebo obvinít pomocí kontextového menu.

Můžete exportovat *Změny Stromu*, což je užitečné, pokud potřebujete poslat někomu jinému Vaši strukturu stromu projektu, obsahující pouze soubory, které se změnily. Tato operace funguje pouze ve vybraných souborech, takže musíte zájmové soubory vybrat - obvykle to znamená, že všechny - a pak **Kontextové menu → Exportovat výběr do....** Budete vyzváni k zadání umístění pro uložení změn stromu.

Můžete také exportovat *Seznam* změněných souborů do textového souboru pomocí **Kontextové menu → Uložit seznam vybraných souborů do....**

Pokud chcete exportovat seznam souborů a činnosti (změněno, přidáno, smazáno), toto můžete také udělat pomocí **Kontextové menu → Výběr kopírovat do schránky**.

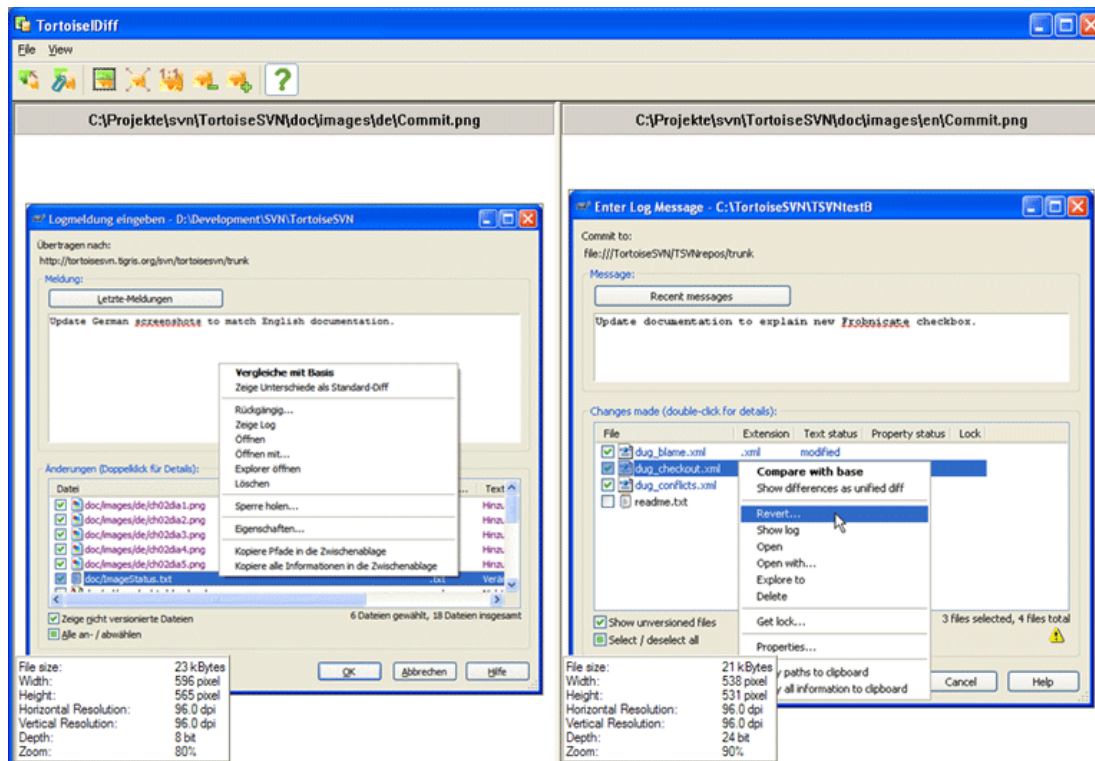
Tlačítko v horní části Vám umožní změnit směr srovnání. Můžete ukázat změnám, že se musí dostat z A do B, nebo chcete-li, z B do A.

Tlačítek s čísly revizí lze použít pro změnu na jiný rozsah revize. Když změníte rozsah, seznam položek, které se liší mezi dvěma revizemi, bude automaticky aktualizován.

Pokud je seznam souborů velmi dlouhý, můžete použít vyhledávací pole k snížení seznamu souborů obsahujících určitý text. Nezapomeňte, že se používá jednoduché textové vyhledávání, takže pokud chcete omezit seznam na zdrojové soubory C, měli byste zadat `.c` raději než `*.c`.

4.10.4. Porovnávání snímků pomocí TortoiseIDiff

Je k dispozici mnoho nástrojů k porovnání textových souborů, včetně našeho TortoiseMerge, ale my se často ocitáme v pozici, kdy chceme vidět, jak se také obrázkový soubor změnil. To je důvod, proč jsme vytvořili TortoiseIDiff.



Obrázek 4.25. Prohlížeč porovnání obrázku

TortoiseSVN → Porovnat některý z běžných formátů obrázkového souboru spustí TortoiseIDiff zobrazující grafické rozdíly. Ve výchozím nastavení jsou obrázky zobrazeny vedle sebe, ale můžete použít menu Zobrazení nebo panelu nástrojů a místo toho přejít na zobrazení pod sebou nebo pokud chcete, můžete obrázky překrýt a předstírat, že používáte lightbox.

Samozřejmě můžete také přibližovat a oddalovat a posouvat obraz. Můžete také posunout obraz jednoduše táhnutím. Pokud zvolíte volbu Spojit obrázky dohromady, pak ovládání posunutí (posuvníky, kolečko myši) jsou na obou obrazcích propojeny.

Okno informací o obrázku zobrazuje detaily o obrázkovém souboru, jako je velikost v pixelech, rozlišení a barevná hloubka. Pokud Vám tento rámeček překáží, použijte Zobrazit → Informace o obrázku pro jeho skrytí. Stejně informace můžete získat v popisku, pokud podržíte myš nad záhlavím názvu obrázku.

Když jsou překryty obrázky, je relativní intenzita obrazů (částečná průhlednost) kontrolována posuvníky na levé straně. Můžete kliknout na libovolné místo v posuvník pro nastavení průhlednosti přímo, nebo můžete táhnutím posuvníku změnit průhlednost interaktivně. **Ctrl+Shift**-Kolečko k změně průhlednosti.

Tlačítko nad posuvníkem přepíná mezi 0% a 100% průhledností, a pokud dvakrát kliknete na tlačítko, průhlednost se automaticky bude přepínat každou sekundu, dokud na tlačítko znovu nekliknete. To může být užitečné, když hledáte více malých změn.

Někdy chcete vidět rozdíly spíše než průhlednost. Možná máte dvě revize obrazových souborů vytištěné obvodové desky a chcete vidět, které spoje se změnily. Pokud zakážete prolnutí částečné

průhlednosti, rozdíl bude zobrazena jako *XOR* barevných hodnot pixelu. Nezměněné oblasti budou čistě bílé a změny budou vybarvené.

4.10.5. Externí Nástroje Porovnání/Sloučení

Pokud nástroje, které Vám poskytujeme, nedělají to, co potřebujete, zkuste některý z mnoha open-source nebo komerčních programů, které jsou k dispozici. Každý má své oblíbené a tento seznam není v žádném případě úplný, ale tady je několik, které byste mohli brát v úvahu:

WinMerge

WinMerge [<http://winmerge.sourceforge.net/>] je skvělý open-source nástroj pro porovnání, který pracuje i s adresáři.

Perforce Merge

Perforce je komerční systém pro kontrolu verzí, ale nástroj porovnání/sloučení si můžete stáhnout zdarma. Získejte více informací na *Perforce* [<http://www.perforce.com/perforce/products/merge.html>].

KDiff3

KDiff3 je bezplatný nástroj pro porovnání, který pracuje i s adresáři. Můžete si jej stáhnout *zde* [<http://kdiff3.sf.net/>].

ExamDiff

ExamDiff Standard je freeware. Je možné pracovat se soubory, ale i s adresáři. ExamDiff Pro je shareware a přidává řadu lahůdek včetně porovnání adresářů a editační funkce. V obou příchutích, verze 3.2 a vyšší zvládne unicode. Můžete si je stáhnout z *PrestoSoft* [<http://www.prestosoft.com/>].

Beyond Compare

Podobně jako ExamDiff Pro, toto je výborný shareware nástroj pro porovnání, který dokáže porovnat adresáře a umí unicode. Stáhněte si ji z *Scooter Software* [<http://www.scootersoftware.com/>].

Araxis Merge

Araxis Merge je užitečným komerčním nástrojem pro porovnání a sloučení souborů i adresářů. Používá trojitě porovnávání a můžete použít synchronizační spojení, pokud jste zaměnili pořadí funkcí. Stáhněte si ho z *Araxis* [<http://www.araxis.com/merge/index.html>].

SciTE

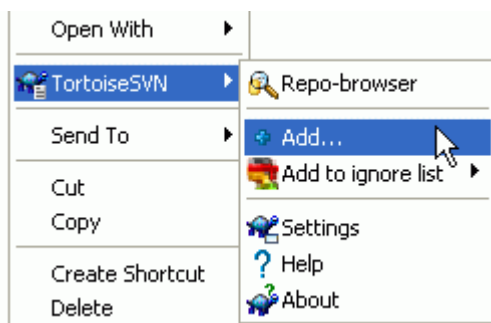
Tento textový editor zahrnuje barvení syntaxe pro sjednocené rozdíly, což je mnohem čitelnější. Stáhněte si ho z *Scintilla* [<http://www.scintilla.org/SciTEDownload.html>].

Notepad2

Notepad2 is designed as a replacement for the standard Windows Notepad program, and is based on the Scintilla open-source edit control. As well as being good for viewing unified diffs, it is much better than the Windows notepad for most jobs. Download it for free *here* [<http://www.flos-freeware.ch/notepad2.html>].

Přečtěte si **4.30.5 – „Nastavení Externího Programu“** pro Informace o tom, jak nastavit TortoiseSVN k používání těchto nástrojů.

4.11. Přidání nových souborů a adresářů



Obrázek 4.26. Kontextové menu průzkumníka pro soubory bez verze

Pokud jste vytvořili nové soubory a/nebo adresáře během Vašeho vývojového procesu, budete je muset také přidat do správy zdroje. Vyberte soubor(y) a/nebo adresáře a zvolte TortoiseSVN → Přidat.

Po přidání souborů a adresářů do správy zdroje se soubor zobrazí překrytím ikony přidáno, což znamená, že musíte nejprve odevzdat Vaši pracovní kopii, aby tyto soubory a adresáře byli k dispozici ostatním vývojářům. Přidání souboru/adresáře *nemá* vliv na úložiště!



Mnoho přidání

Můžete také použít příkaz Přidat na adresář, který má již verzi. V takovém případě dialogové okno Přidat zobrazí všechny soubory bez verze uvnitř adresáře s verzí. To Vám pomůže, pokud máte mnoho nových souborů a je třeba přidat všechny najednou.

Chcete-li přidat soubory mimo Vaši pracovní kopii, můžete použít obslužnou rutinu táhni a pusť:

1. vyberte soubory, které chcete přidat
2. right-drag them to the new location inside the working copy
3. pusťte pravé tlačítko myši
4. zvolte Kontextové menu → SVN Přidat soubory do této pracovní kopie. Soubory pak budou zkopírovány do pracovní kopie a přidány do správy verzí.

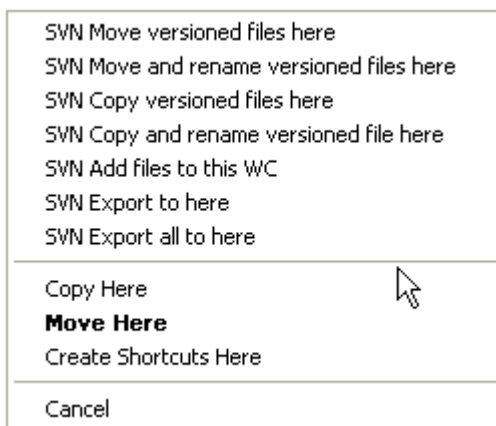
Můžete také přidat soubory uvnitř pracovní kopie jednoduše tažením levého tlačítka myši do dialogového okna odevzdání.

Přidáte-li soubor nebo adresář omylem, můžete přidání vzít zpět před odevzdáním použitím TortoiseSVN → Vrátit Přidání....

4.12. Kopírování/Přesun/Přejmenování Souborů a Adresářů

Často se stává, že již máte soubory, které potřebujete v jiném projektu v úložišti a vy je prostě chcete zkopírovat. Jednoduše by se mohli zkopírovat a přidat, jak je popsáno výše, ale to by Vám nedalo žádnou historii. A pokud následně opravíte chybu v původních souborech, můžete sloučit opravu automaticky pouze, pokud je nová kopie podobná originálu v Subversion.

The easiest way to copy files and folders from within a working copy is to use the right-drag menu. When you right-drag a file or folder from one working copy to another, or even within the same folder, a context menu appears when you release the mouse.



Obrázek 4.27. Menu táhnutí pravým tlačítkem pro adresář pod správou verzí

Now you can copy existing versioned content to a new location, possibly renaming it at the same time.

Můžete také kopírovat nebo přesouvat soubory va adresářích do pracovní kopie, nebo mezi dvěma pracovními kopiemi, pomocí známé metody vyjmout a vložit. Použijte standardní Windows Kopírovat nebo Vyjmout ke zkopírování jedné nebo více položek s verzí do schránky. Pokud schránka obsahuje takové položek s verzí, můžete pak použít TortoiseSVN → Vložit (Poznámka: ne standardní Windows Vložit) ke zkopírování nebo přesunu položek do nového umístění v pracovní kopii.

Můžete kopírovat soubory a adresáře z Vaší pracovní kopie na jiné místo v úložišti pomocí TortoiseSVN → Větev/Značka. Viz 4.19.1 – „Vytvoření Větvě nebo Značky“ pro více informací.

Můžete si najít starší verzi souboru nebo adresáře v dialogovém okně Záznamu a zkopírovat jej do nového umístění v úložišti přímo z dialogového okna Záznamu pomocí Kontextové menu → Vytvořit větev/značku z revize. Pro více informací si přečtete 4.9.3 – „Získání dodatečných informací“.

Můžete také použít průzkumníka úložiště k nalezení obsahu, který chcete a zkopírovat jej do Vaší pracovní kopie přímo z úložiště, nebo kopírovat mezi dvěma místy v rámci úložiště. V 4.24 – „Průzkumník Úložiště“ se dozvíte více.

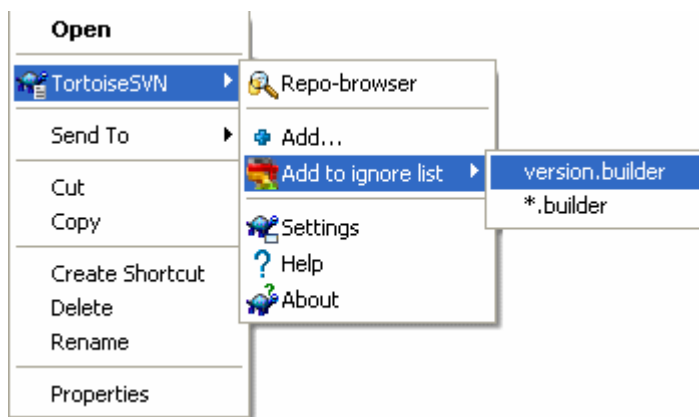


Nelze kopírovat mezi úložišťmi

Whilst you can copy files and folders *within* a repository, you *cannot* copy or move from one repository to another while preserving history using TortoiseSVN. Not even if the repositories live on the same server. All you can do is copy the content in its current state and add it as new content to the second repository.

Pokud si nejste jisti, zda dvě adresy URL na stejném serveru odkazují na stejné nebo různé úložiště, použijte průzkumníka úložiště a otevřete jednu URL, abyste zjistili, kde je kořen úložiště. Pokud vidíte obě místa v jednom okně průzkumníka úložiště, pak jsou ve stejném úložišti.

4.13. Ignorování Souborů a Adresářů



Obrázek 4.28. Kontextové menu průzkumníka pro soubory bez verze

Ve většině projektů budete mít soubory a adresáře, které by neměly být zahrnuty do správy verzí. To mohou být soubory vytvořené kompilátorem, *.obj, *.lst, či výstupní adresář pro uložení spustitelných souborů. Kdykoliv odevzdáte změny, TortoiseSVN zobrazuje Vaše soubory bez verze, které naplní seznam souborů v dialogovém okně odevzdání. Samozřejmě můžete vypnout tento displej, ale pak můžete zapomenout přidat nový zdrojový soubor.

Nejllepší způsob, jak se těmto problémům vyhnout, je přidat nepůvodní soubory do seznamu ignorovaných v projektu. Tak se nikdy objeví v dialogovém okně, ale skutečný zdrojový soubor bez verze bude stále označen.

Pokud kliknete pravým tlačítkem myši na jeden soubor bez verze a vyberete příkaz TortoiseSVN → Přidat do seznamu k ignorování z kontextového menu, dílčí menu se zobrazí, umožňující vybrat právě tento soubor, nebo všechny soubory stejného typu. Pokud vyberete více souborů, není žádné dílčí menu a můžete přidat jen zvolené soubory nebo adresáře.

Pokud chcete odebrat jednu nebo více položek ze seznamu ignorovaných, klikněte pravým tlačítkem myši na tyto položky a vyberte možnost TortoiseSVN → Odstranit ze seznamu ignorování. Můžete také vstoupit do vlastností svn:ignore adresáře přímo. To Vám umožní zadat obecnější vzorec pomocí globálního listu, který je popsán v následující části. Přečtěte si 4.17 – „Nastavení projektu“ pro více informací o nastavení vlastností přímo. Uvědomte si prosím, že každá předloha ignorování musí být uvedena na samostatném řádku. Oddělení mezerou nefunguje.



Globální seznam ignorovaných

Dalším způsobem, jak ignorovat soubory, je přidat je do *globálního seznamu ignorování*. Velký rozdíl je, že globální seznam ignorování je vlastnost klienta. To se vztahuje na *všechny* projekty Subversion, ale pouze na počítači klienta. Obecně je lepší použít vlastnost svn:ignore, kde je možné, protože může být aplikována na konkrétní oblastí projektu a funguje pro každého, kdo získá projekt. Přečtěte si 4.30.1 – „Obecná nastavení“ pro více informací.



Ignorování položek s verzí

Soubory s verzí a adresáře nikdy nemůžou být ignorovány - to je rys Subversion. Pokud přidáte verzi soubor omylem, přečtěte si B.8 – „Ignorovat soubory, které už mají verzi“ pro postup, jak mu verzi „odstranit“.

4.13.1. Shoda vzorů v seznamech k Ignorování

Vzory ignorování v Subversion využívají globbing souborů, technika, původně používaná v Unixu, určující soubory pomocí metaznaků jako zástupné znaky. Následující znaky mají speciální význam:

*

Jakýkoliv řetězec znaků, včetně prázdného řetězce (žádný znak).

?

Jeden libovolný znak.

[...]

Shoda s některým ze znaků uzavřených v hranatých závorkách. Pokud je v závorce pár znaků oddělen „-“ znamená libovolný znak shodující se abecedně mezi těmito dvěma znaky. Například `[AGm-p]` odpovídá všem A, G, m, n, o nebo p.

Pattern matching is case sensitive, which can cause problems on Windows. You can force case insensitivity the hard way by pairing characters, eg. to ignore `*.tmp` regardless of case, you could use a pattern like `*. [Tt] [Mm] [Pp]`.

Pokud chcete oficiální definici globbingu, najdete ji v IEEE specifikaci pro příkazový jazyk rozhraní *Zápis shody vzorů* [http://www.opengroup.org/onlinepubs/009695399/utilities/xcu_chap02.html#tag_02_13].

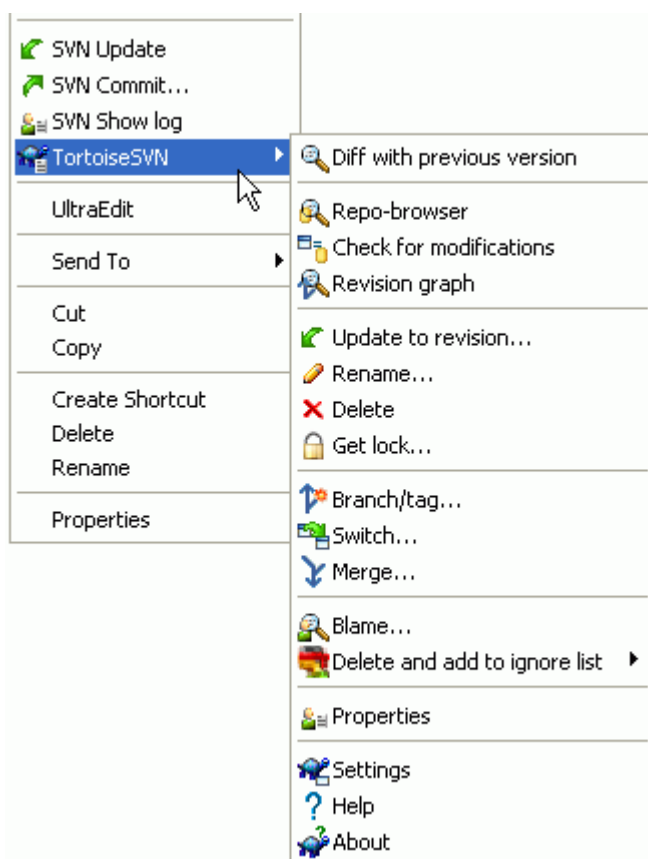


Žádné Cesty v Globálním Seznam k Ignorování

Do Vašeho vzoru byste neměli zahrnovat informace o cestě. Shoda vzoru je určena k použití na prosté názvy souborů a adresářů. Pokud chcete ignorovat všechny adresáře CVS, stačí přidat CVS do seznamu k ignorování. Není nutné specifikovat CVS `*/CVS` jako v předchozích verzích. Pokud chcete ignorovat všechny adresáře tmp, které se nacházejí v adresáři prog, ale ne v adresáři doc měli byste místo toho použít vlastnost `svn:ignore`. Neexistuje žádný spolehlivý způsob, jak tohoto dosáhnout pomocí globálních vzorů k ignorování.

4.14. Mazání, Přesunutí a Přejmenování

Na rozdíl od CVS, Subversion umožňuje přejmenování a přesun souborů a adresářů. Takže v menu TortoiseSVN existují volby pro smazání a přejmenování.



Obrázek 4.29. Kontextové menu průzkumníka pro soubory s verzí

4.14.1. Smazání souborů a adresářů

Použijte TortoiseSVN → Smazat k odstranění souborů nebo adresářů z Subversion.

Pokud zvolíte TortoiseSVN → Odstranit je soubor okamžitě odstraněn z Vaší pracovní kopie, také je označena v úložišti k odstranění při příštím odevzdání. Nadřazený adresář souboru ukazuje překryv „smazáno“. Dokud změnu neodevzdáte, můžete soubor získat zpět pomocí TortoiseSVN → Vrátit použitím na nadřazený adresář.

Když zvolíte TortoiseSVN → Odstranit adresář, zůstane v pracovní kopii, ale překryv se změní, aby naznačil, že je označen pro odstranění. Dokud změnu neodevzdáte, můžete ho získat zpět pomocí TortoiseSVN → Vrátit na samotný adresář. Tento rozdíl v chování mezi souborem a adresářem je součástí Subversion, ne TortoiseSVN.

Pokud chcete vymazat položku z úložiště, ale ponechat si ji místně jako soubor/adresář bez verze, použijte Rozšířené Kontextové menu → Smazat (ponechat místní). Musíte držet klávesu **Shift**, při klikání pravým tlačítkem myši na položku v panelu se seznamem v průzkumníku (pravý panel), aby toto rozšířené kontextové menu bylo vidět.

Pokud je *soubor* vymazán přes průzkumníka místo pomocí kontextového menu TortoiseSVN, dialogové okno odevzdání tyto soubory zobrazí a umožňuje je také odstranit ze správy verzí před odevzdáním. Nicméně pokud aktualizujete Vaší pracovní kopii, Subversion si všimne chybějícího souboru a nahradí jej nejnovější verzí z úložiště. Pokud potřebujete odstranit soubor spravovaný verzí, vždy použijte TortoiseSVN → Smazat, aby Subversion nemusel hádat, co opravdu chcete udělat.

Pokud je *adresář* smazán přes průzkumníka místo pomocí kontextového menu TortoiseSVN, Vaše pracovní kopie bude rozbitá a nebudete moci odevzdávat. Aktualizujete-li Vaši pracovní kopii, Subversion nahradí chybějící adresář nejnovější verzí z úložiště a pak ho můžete odstranit správným způsobem použitím TortoiseSVN → Smazat.



Získání smazaného souboru nebo adresáře zpět

If you have deleted a file or a folder and already committed that delete operation to the repository, then a normal TortoiseSVN → Revert can't bring it back anymore. But the file or folder is not lost at all. If you know the revision the file or folder got deleted (if you don't, use the log dialog to find out) open the repository browser and switch to that revision. Then select the file or folder you deleted, right-click and select Context Menu → Copy to... as the target for that copy operation select the path to your working copy.

4.14.2. Přesouvání souborů a adresářů

Pokud chcete udělat jednoduché přejmenování souboru nebo adresáře na místě, použijte Kontextové menu → Přejmenovat..., zadejte nový název položky a je hotovo.

If you want to move files around inside your working copy, perhaps to a different sub-folder, use the right-mouse drag-and-drop handler:

1. vyberte soubory nebo adresáře, které chcete přesunout
2. right-drag them to the new location inside the working copy
3. pusťte pravé tlačítko myši
4. v místní nabídce vyberte Kontextové menu → SVN Přesunout soubory s verzí sem



Odevzdat nadřazený adresář

Vzhledem k tomu, že přejmenování a přesouvání se provádí jako smazání následovano přidáním, musíte odevzdat nadřazený adresář přejmenovaného/přesunutého souboru tak, aby smazaná část přejmenování/přesunutí se objeví v dialogovém okně odevzdání. Pokud neodevzdáte odstraněnou část přejmenování/přesunutí, zůstane v úložišti, a když Vaši spolupracovníci budou aktualizovat starý soubor nebude odstraněn. tedy budou mít *obě* kopie, starou i novou.

Musíte odevzdat přejmenovaný adresář před změnou některých souborů uvnitř adresáře, jinak Vaše pracovní kopie může být velmi neuspořádaná.

Můžete také použít průzkumníka úložiště pro přesun předmětů po okolí. Přečtěte si 4.24 – „Průzkumník Úložiště“ a dozvíte se více.



SVN Nepřesouvat externí

Neměli byste používat TortoiseSVN příkazy Přesunout nebo Přejmenovat na adresář, kter byl vytvořen pomocí svn:externals. Tato akce by způsobila, že vnější položky jsou smazány z jejich nadřazeného úložiště. Tím pravděpodobně rozčílíte mnoho dalších lidí. Pokud potřebujete přesunout externí adresář, měli byste použít obyčejný přesun v rozhraní a poté nastavit vlastnost svn:externals v nadřazených adresářích zdroje a určení.

4.14.3. Změna velikosti písmen v souboru

Provádět změny velikosti písmen názvu souboru je složité se Subversion ve Windows, protože na krátkou dobu během přejmenování, oba názvy souboru musí existovat. Protože Windows má souborový systém citlivý na velikost písmen, použití obyčejného příkazu Přejmenovat nefunguje.

Naštěstí existují (nejméně) dva možné způsoby přejmenování souboru bez ztráty jeho záznamu historie. Je důležité ho přejmenovat v rámci Subversion. Přejmenování v průzkumníku naruší Vaši pracovní kopii!

Řešení A) (doporučeno)

1. Odevzdat změny ve Vaší pracovní kopii.
2. Přejmenovat soubor z VELKÉHO písma na velká PÍSMO přímo v úložišti pomocí průzkumníka úložiště.
3. Aktualizovat Vaši pracovní kopii.

Řešení B)

1. Přejmenujte z VELKÉHO písma na VELKÉ písmo_ pomocí příkazu přejmenovat v podmenu TortoiseSVN.
2. Odevzdejte změny.
3. Přejmenujte z VELKÉHO písma_ na velké PÍSMO.
4. Odevzdejte změny.

4.14.4. Řešení konfliktů v názvech souborů

Pokud úložiště již obsahuje dva soubory se stejným názvem, lišící se pouze velikostí písmen, (např. `Test.txt` a `test.txt`), nebudete moci aktualizovat nebo získat nadřazený adresář s Windows klientem. Zatímco Subversion podporuje názvy souborů s různou velikostí písmen, systém Windows ne.

Někdy se tak stane, že dva lidé odevzdají, z oddělených pracovních kopií, soubory, které náhodou mají stejný název, ale jinou velikost písmen. Také se může stát, že soubory jsou odevzdány ze systému se souborovým systémem, rozlišující velikost písmen, jako je Linux.

V tomto případě budete muset rozhodnout, který z nich v úložišti chcete zachovat a který smazat (nebo přejmenovat).



Zabránit souborům mít stejný název

Existuje hákový skript serveru k dispozici na adrese: <http://svn.collab.net/repos/svn/trunk/contrib/hook-scripts/>, který zabrání odevzdáním, které způsobí konflikt ve velikosti písmen.

4.14.5. Oprava Přejmenování Souborů

Někdy Vaše přátelské IDE přejmenuje pro Vás soubory jako součást cvičení refaktorování a samozřejmě nic Subversion neřekne. Pokusíte-li se odevzdat změny, Subversion uvidí staré jméno souboru jako chybějící a ten nový jako soubor bez verze. Stačilo by jen zaškrtnout nový název souboru, aby byl přidán, ale pak byste ztratili sledování historie, protože Subversion neví, že soubory spolu souvisejí.

Lepší způsob je oznámit Subversion, že tato změna je vlastně přejmenování. To můžete udělat v rámci dialogových oken Odevzdání a Zjistit změny. Stačí jen vybrat starý název (chybějící) i nové jméno (bez verze) a použít Kontextové menu → Opravit přesun pro spárování souborů jako přejmenování.

4.14.6. Smazání souborů bez verze

Obvykle nastavíte Váš seznam k ignorování tak, že všechny vytvořené soubory jsou v Subversion ignorovány. Ale co když chcete vymazat všechny ignorované položky, protože chcete provést čisté

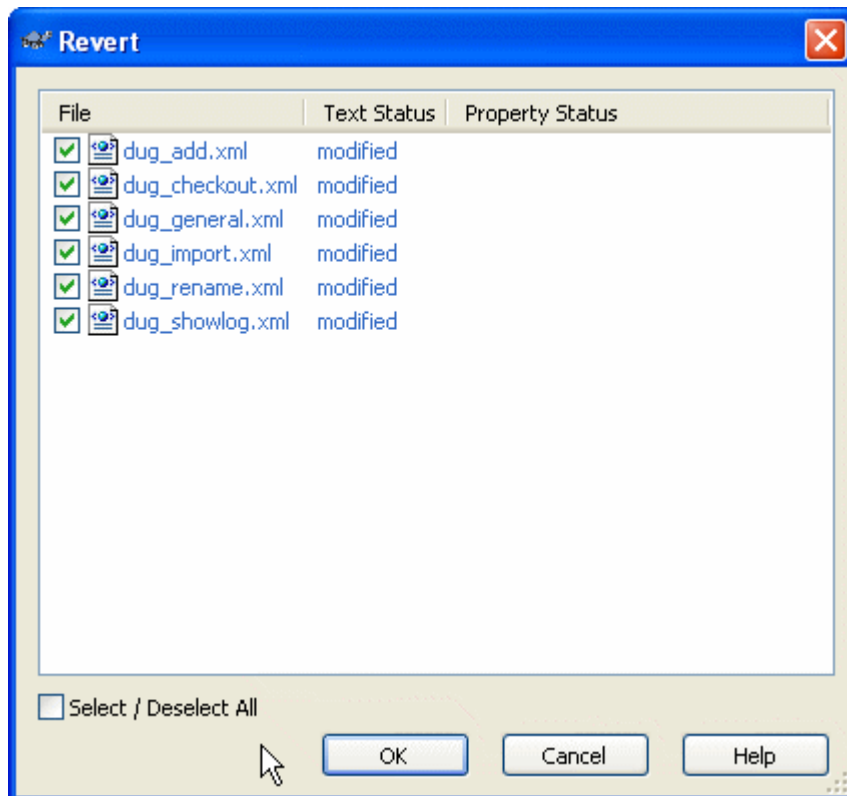
sestavení? Obvykle byste toto nastavili v makefile, ale pokud makefile ladíte, nebo měníte sestavovací systém, je užitečné mít způsob jak zamést podlahu.

TortoiseSVN poskytuje právě takovou možnost použitím **Rozšířená Kontextové menu** → **Smazat položky bez verze**.... Musíte držet **Shift** při kliknutí pravým tlačítkem na adresář v panel seznamu průzkumníka (pravý panel), aby bylo rozšířené kontextové menu vidět. Toto zobrazí dialogové okno, které vypíše všechny soubory bez verze kdekoli v pracovní kopii. Pak si můžete označit nebo odznačit položky, které mají být smazány.

Při mazání těchto položek je použit koš, takže pokud zde uděláte chybu a smažete soubor, který měl mít verzi, můžete ho stále obnovit.

4.15. Vrátit změny zpět

Pokud chcete zrušit všechny změny provedené v souboru od poslední aktualizace, je nutné vybrat soubor, klikněte pravým tlačítkem myši pro zobrazení kontextového menu a vyberte příkaz **TortoiseSVN** → **Vrátit**. Objeví se dialogové okno zobrazující soubory, které jste změnili a může být vráceny. Vyberte ty, které chcete vrátit a klikněte na OK.



Obrázek 4.30. Dialogové okno Vrátit

If you want to undo a deletion or a rename, you need to use Revert on the parent folder as the deleted item does not exist for you to right-click on.

Pokud chcete přidání položky vrátit zpět, v kontextovém menu se toto zobrazí jako **TortoiseSVN** → **Vrátit zpět přidání**.... Toto je také vrácení, ale jméno bylo změněno, aby to bylo více zřejmé.

Sloupce v tomto dialogovém okně lze upravit stejným způsobem jako sloupce v dialogovém okně **Zkontrolovat změny**. Přečtěte si [4.7.3 – „Místní a Vzdálený Stav“](#) pro další podrobnosti.



Vrácení Změn, které byly odevzdány, zpět

Vrátit Vráti zpět pouze místní změny. *Nevrací* zpět změny, které již byly odevzdány. Pokud chcete vrátit zpět všechny změny, které byly odevzdány v určité revizi, prohlédněte si [4.9](#) – „Dialogové Okno Záznamu Revizí“ pro další informace.



Vrácení je pomalé

Když vracíte změny můžete zjistit, že operace trvá mnohem déle, než očekáváte. Důvodem je, že upravená verze souboru je poslána do koše, takže můžete znovu změny načíst, pokud jste je omylem vrátili. Pokud ale je Váš Koš plný, systému Windows trvá dlouho najít místo pro uložení souboru. Řešení je jednoduché: buď vysypat koš, nebo odškrtnout políčko Při vrácení zpět použít odpadkový koš v nastavení TortoiseSVN.

4.16. Vyčistit

Pokud příkaz Subversion nemůže být úspěšně dokončen, třeba kvůli problémům serveru, může Vaše pracovní kopie zůstat v rozporuplném stavu. V takovém případě je třeba použít na adresář TortoiseSVN → Vyčistit. Je dobrý nápad toto provádět na nejvyšší úrovni v pracovní kopii.

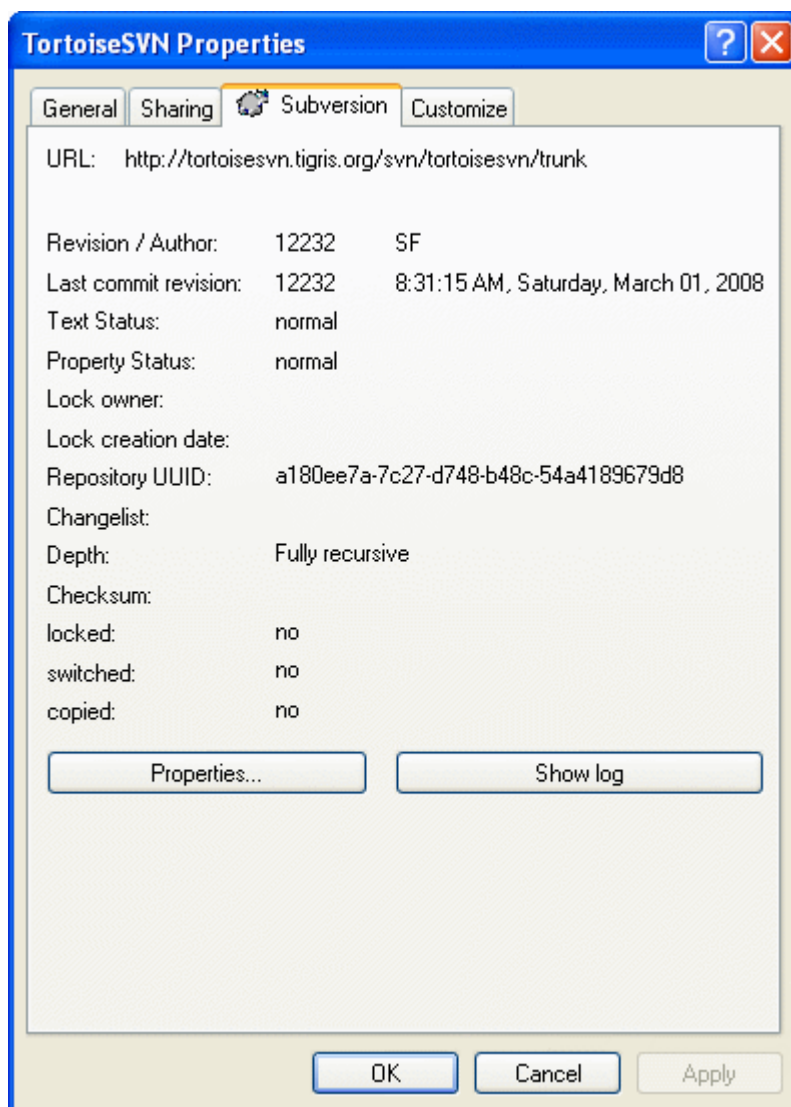
Vyčištění má další užitečný vedlejší efekt. Pokud se změnil datum souboru, ale jeho obsah ne, Subversion nemůže zjistit, zda se skutečně změnil, pokud, neporovná byte po bajtu s původní kopií. Pokud máte velké množství souborů v tomto stavu, získání stavu bude velmi pomalé, což způsobí, že mnoho dialogových oken bude pomalu reagovat. Spuštěním Vyčištění ve Vaší pracovní kopii budou tyto „rozbitá“ časová razítka opravena a kontrolu stavu obnovena na nejvyšší rychlost.



Use Commit Timestamps

Some earlier releases of Subversion were affected by a bug which caused timestamp mismatch when you check out with the **Use commit timestamps** option checked. Use the Cleanup command to speed up these working copies.

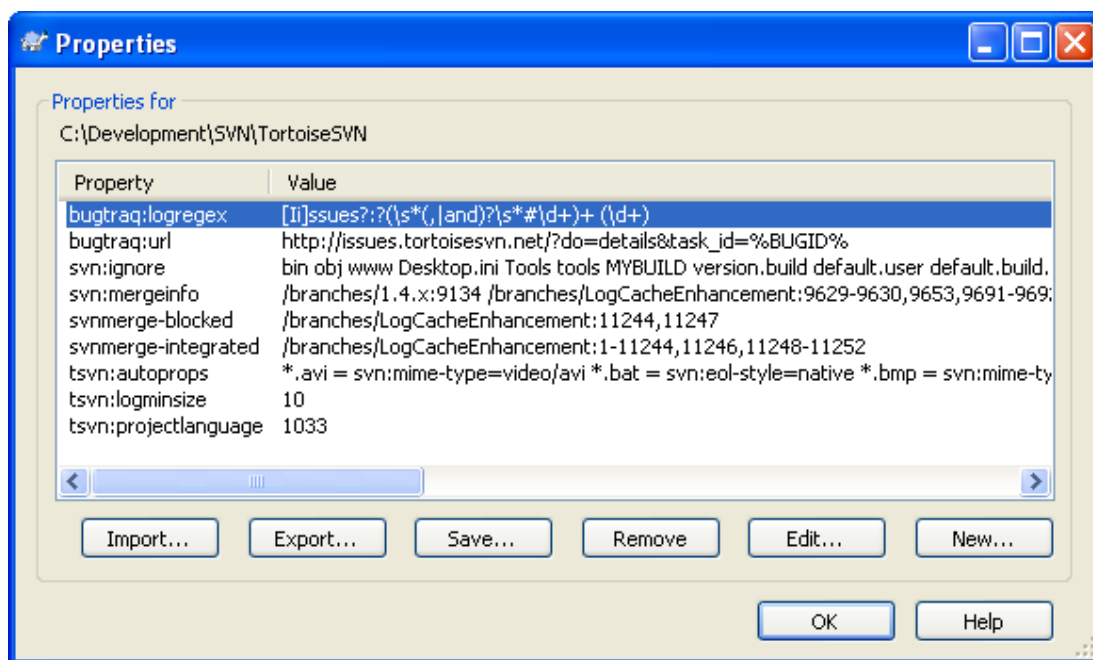
4.17. Nastavení projektu



Obrázek 4.31. Stránka vlastností v Průzkumníku, karta Subversion

Někdy budete chtít podrobnější informace o souboru/adresáři, než jen překryvem ikon. Můžete získat všechny informace, které Subversion nabízí v dialogovém okně vlastností v Průzkumníku. Stačí vybrat soubor nebo adresář a vyberte v kontextovém menu možnost **Windows Menu** → **Vlastnosti** (poznámka: toto je normální menu položky vlastnosti, který Průzkumník nabízí a ne ten v podmenu TortoiseSVN!). V dialogovém okně Vlastnosti přidal TortoiseSVN novou stránku vlastností pro soubory/adresáře pod kontrolou Subversion. Zde vidíte všechny důležité informace o vybraném souboru/adresáři.

4.17.1. Subversion Vlastnosti



Obrázek 4.32. Stránka Subversion vlastností

Můžete si přečíst a nastavit Subversion vlastnosti z dialogového okna vlastností v systému Windows, ale také z TortoiseSVN → vlastnosti a v seznámech stavu TortoiseSVN, z Kontextové menu → vlastnosti.

Můžete přidat vlastní vlastnosti, nebo nějaké vlastnosti se speciálním významem pro Subversion. Tyto začínají s svn:. svn:externals je taková vlastnost; podívejte se, jak ovládat externí v 4.18 – „Externí položky“.

4.17.1.1. svn:keywords

Subversion podporuje expanzi klíčových slov podobně jako CVS, které lze použít pro vložení informací o souboru a revizi v rámci samotného souboru. Klíčová slova, která jsou v současné době podporována:

\$Date\$

Datum posledního známého odevzdání. To je založeno na informacích získaných při aktualizaci Vaší pracovní kopie. *Nekontroluje* nejnovější změny v úložišti.

\$Revision\$

Revize posledního známého odevzdání.

\$Author\$

Autor, který provedl poslední známé odevzdání.

\$HeadURL\$

Úplná adresa URL tohoto souboru v úložišti.

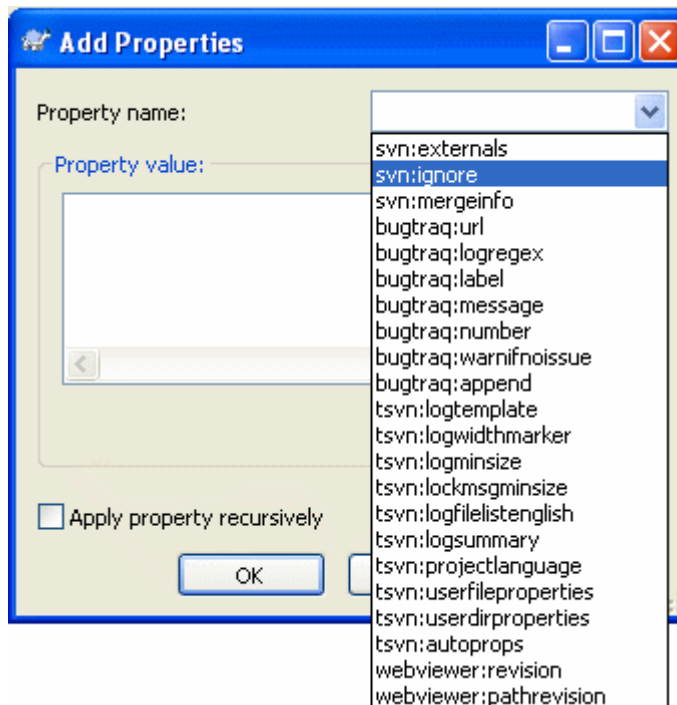
\$Id\$

Zhuštěná kombinace předchozích čtyř klíčových slov.

Chcete-li zjistit, jak můžete tyto klíčová slova využít, podívejte se na [sekcí svn:keywords](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] v knize Subversion, který dává kompletní popis těchto klíčových slov a jak je povolit a používat.

Pro více informací o Subversion vlastnostech se podívejte na [Zvláštní vlastnosti](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. Přidání a Úprava vlastností



Obrázek 4.33. Přidání vlastností

To add a new property, first click on **Add....** Select the required property name from the combo box, or type in a name of your own choice, then enter a value in the box below. Properties which take multiple values, such as an ignore list, can be entered on multiple lines. Click on **OK** to add that property to the list.

If you want to apply a property to many items at once, select the files/folders in explorer, then select **Context menu → properties**

Pokud chcete použít vlastnost na *každý* soubor a adresář v hierarchii pod aktuálním adresářem, zaškrtněte políčko **Rekurzivní**.

Some properties, for example `svn:needs-lock`, can only be applied to files, so the property name doesn't appear in the drop down list for folders. You can still apply such a property recursively to all files in a hierarchy, but you have to type in the property name yourself.

Pokud chcete upravit existující vlastnost, vyberte ji ze seznamu existujících vlastností, pak klikněte na **Upravit**

Pokud chcete odstranit existující vlastnost, vyberte ji ze seznamu existujících vlastností, pak klikněte na **Odstranit**.

Vlastnost `svn:externals` může být použita ke stáhnutí jiných projektů ze stejného úložiště, nebo úplně jiného úložiště. Pro více informací si přečtěte [4.18 – „Externí položky“](#).

4.17.1.3. Export a Import Vlastností

Často se ocitnete, že uplatňujete mnohokrát stejnou sadu vlastností, například `bugtraq:logregex`. Pro zjednodušení procesu kopírování vlastností z jednoho projektu do druhého, můžete použít funkci **Export/Import**.

Ze souboru nebo adresáře, kde jsou již nastavené vlastnosti, použijte **TortoiseSVN → vlastnosti**, zvolte vlastnosti, které chcete exportovat a klikněte na **Exportovat....** Budete vyzváni k zadání názvu souboru, kde budou názvy vlastností a hodnoty uloženy.

Z adresáře(ů), kde si přejete uplatnit tyto vlastnosti, použijte **TortoiseSVN** → **vlastnosti** a klikněte na **Importovat...** Budete vyzváni k zadání názvu souboru pro import, takže přejděte na místo, kde jste předtím uložili exportovaný soubor a vyberte ho. Vlastnosti budou nerekurzivně přidány do adresáře.

Pokud chcete přidat vlastnosti do stromu rekurzivně, postupujte podle výše uvedených kroků, pak v dialogovém okně vlastností, vyberte postupně vlastnosti, klikněte na **Upravit...**, zaškrtněte políčko **Aplikovat vlastnost rekurzivně** a klikněte na **OK**.

Formát importovaného souboru je binární a vlastnictvím TortoiseSVN. Jeho jediným účelem je přenos vlastností pomocí Importu a Exportu, takže není nutné tyto soubory upravovat.

4.17.1.4. Binární vlastnosti

TortoiseSVN zvládne binární hodnoty vlastností použitím souborů. Chcete-li číst binární hodnoty vlastností, zvolte **Uložit...** do souboru. Chcete-li nastavit binární hodnotu, použijte hex editor nebo jiný vhodný nástroj k vytvoření souboru s obsahem, který požadujete, pak **Načíst...** tento soubor.

Přestože binární vlastnosti nejsou často používány, v některých aplikacích mohou být užitečné. Například pokud ukládáte velké grafické soubory, nebo je-li aplikace pro načtení souboru je obrovská, možná budete chtít uložit miniaturu jako vlastnost, takže můžete rychle získat náhled.

4.17.1.5. Automatické nastavení vlastností

Můžete nastavit Subversion a TortoiseSVN, aby nastavovali vlastnosti souborů a adresářů automaticky, když jsou přidány do úložiště. Existují dva způsoby, jak toho dosáhnout.

Můžete upravit konfigurační soubor Subversion, abyste tuto funkci u Vašeho klienta povolili. Stránka **Obecné** dialogového okna nastavení TortoiseSVN má tlačítko upravit, které Vás tam vezme přímo. Konfigurační soubor je jednoduchý textový soubor, který řídí některé funkce Subversion. Musíte změnit dvě věci: za prvé v části nazvané **miscellany** vložte řádek `enable-auto-props = yes`. Za druhé je třeba upravit sekci níže, abyste určili, které vlastnosti chcete přidat a na které typy souborů. Tato metoda je standardní funkcí Subversion a funguje v jakémkoliv klientovi Subversion. Nicméně musí být určeny v každém klientovi individuálně - neexistuje žádný způsob, jak tato nastavení přenést z úložiště.

Alternativní metodou je nastavit vlastnost `tsvn:autoprops` v adresáři, jak je popsáno v další části. Tato metoda funguje pouze pro klienty TortoiseSVN, ale je možno šířit je na všechny pracovní kopie při aktualizaci.

Whichever method you choose, you should note that auto-props are only applied to files at the time they are added to the repository. Auto-props will never change the properties of files which are already versioned.

Pokud si chcete být naprosto jisti, že nové soubory mají nastaveny správné vlastnosti, měli byste nastavit v úložišti hák před-odevzdáním, který odmítne odevzdání, které nemá požadované vlastnosti nastaveny.



Vlastnosti odevzdání

Vlastnosti Subversion mají verzi. Po změně nebo přidání vlastnosti, musíte Vaše změny odevzdat.



Konflikty vlastností

Pokud existuje konflikt při odevzdání změn, protože jiný uživatel změnil stejnou vlastnost, Subversion vytvoří soubor `.prej`. Odstraňte tento soubor poté, co jste konflikt vyřešili.

4.17.2. TortoiseSVN Vlastnosti projektu

TortoiseSVN má několik vlastních speciálních vlastností, začínají `tsvn:`.

- `tsvn:logminsize` stanovuje minimální délku zprávy záznamu pro odevzdání. Zadáte-li kratší zprávu, než zde uvedenou, odevzdání je zablokováno. Tato funkce je velmi užitečná, aby Vám připomínala zadat řádnou popisnou zprávu pro každé odevzdání. Pokud tato vlastnost není nastavena, nebo je hodnota nula, prázdné zprávy záznamu jsou povoleny.

`tsvn:lockmsgminsize` stanovuje minimální délku zprávy zamknutí. Zadáte-li kratší zpráva zde uvedenou, je zámek zablokovan. Tato funkce je velmi užitečná, aby Vám připomněla zadat řádnou popisnou zprávu pro každý zámek, který dostanete. Pokud tato vlastnost není nastavena, nebo je hodnota nula, prázdné zprávy zámknutí jsou povoleny.

- `tsvn:logwidthmarker` Používá se pro projekty, které vyžadují, aby zprávy záznamu byly formátovány nějakou maximální šířkou (typicky 80 znaků) před zalomením řádku. Nastavením této vlastnosti na nenulovou hodnotu provede 2 věci v dialogovém okně záznamů: umístí značku pro označení maximální šířky a zakáže zalamování řádků na displeji, takže můžete vidět, zda je text, který jste zadali, příliš dlouhý. Poznámka: tato funkce bude správně fungovat pouze, když máte vybrané písmo s pevnou šířkou vybraných pro zprávy záznamu.
- `tsvn:logtemplate` Používá se pro projekty, které mají pravidla o formátování zpráv protokolu. Vlastnost má mnohořádkový textový řetězec, který bude vložen do zprávy odevzdání při spuštění odevzdání. Poté ji můžete upravit tak, aby zahrnovala požadované informace. Poznámka: Pokud také používáte `tsvn:logminsize`, nezapomeňte nastavit délku delší než šablona nebo ztratíte ochranný mechanismus.
- Subversion umožňuje nastavit „autovlastnosti“, které budou použity na nově přidané nebo importované soubory, na základě přípony souboru. Závisí, jestli má každý klient nastaveny vhodné autovlastnosti v jejich konfiguračním souboru Subversion. `tsvn:autoprops` lze nastavit pro adresáře a tyto budou sloučeny s místními autovlastnostmi uživatele při importu nebo přidávání souborů. Formát je stejný jako pro Subversion autovlastnosti, např. `*.sh = svn:eol-style = native;` `svn:executable` stanoví dvě vlastnosti na soubory s příponou `.sh`.

Dojde-li ke konfliktu mezi místními autovlastnostmi a `tsvn:autoprops`, nastavení projektu má přednost, protože jsou pro tento projekt specifické.

- V dialogovém okně Odevzdání máte možnost vložit seznam změněných souborů, včetně stavu každého souboru (přidaný, změněný atd.). `tsvn:logfilelistenglish` definuje, zda je stav souboru vložen v angličtině nebo v lokalizovaném jazyce. Pokud vlastnost není nastavena, je výchozí hodnota pravda.
- TortoiseSVN může použít moduly pro kontrolu pravopisu, které používají OpenOffice a Mozilla. Pokud je máte nainstalované, tato vlastnost určuje, kterou kontrolu pravopisu použít, tj. v jakém jazyce by měly zprávy záznamu ve Vašem projektu být. `tsvn:projectlanguage` nastaví jazykový modul, který by jádro kontroly pravopisu mělo použít při zadávání zprávy záznamu. Hodnoty pro Váš jazyk můžete najít na této stránce: [MSDN: Jazykové identifikátory](http://msdn.microsoft.com/en-us/library/dd318693.aspx) [http://msdn.microsoft.com/en-us/library/dd318693.aspx].

Tuto hodnotu můžete zadat v desítkové, nebo v šestnáctkové soustavě, pokud má předponu `0x`. Například angličtina (US) může být zapsána buď jako `0x0409` nebo `1033`.

- Vlastnost `tsvn:logsummary` se používá pro získání zprávy protokolu, která je pak uvedena v dialogovém okně záznamů jako souhrn zpráv záznamů.

Hodnoty vlastnosti `tsvn:logsummary` musí být nastavena na jeden řádek řetězce regulárního výrazu, který obsahuje jednu skupinu regulárního výrazu. Cokoliv se shoduje s touto skupinou, bude použito jako shrnutí.

Příklad: `\[Souhrn\]:\s+(.*)` zachytí všechno, co je po „[Souhrn]“ ve zprávách záznamů a použít je jako souhrn.

- Když chcete přidat nové vlastnosti, můžete je buď vybrat ze seznamu v rámečku, nebo můžete zadat jakékoli jméno vlastnosti, které se Vám líbí. Pokud Váš projekt používá nějaké vlastní vlastnosti a chcete, aby se tyto vlastnosti objevily v seznamu v rámečku (aby se předešlo překlepům, když zadáváte název vlastnosti), můžete vytvořit seznam Vašich vlastních vlastností pomocí `tsvn:userfileproperties` a `tsvn:userdirproperties`. Použijte tyto vlastnosti na adresář. Když pak upravujete vlastnost každé podřazené položky, budou Vaše uživatelské vlastnosti uvedeny v seznamu přednastavených názvů vlastností.

Některé vlastnosti `tsvn:` vyžadují hodnotu `true/false`. TortoiseSVN také rozumí `yes` jako synonymum pro `true` a `no` jako synonymum pro `false`.

TortoiseSVN je možné integrovat s některými nástroji pro sledování chyb. Využívá se vlastností projektu, které začínají `bugtraq:`. Přečtěte si [4.28 – „Integrace se Systémy pro Sledování Chyb/Problémů“](#) pro další informace.

Také jej můžete integrovat s některými on-line průzkumníky úložiště, pomocí vlastností projektu, které začínají `webviewer:`. Přečtěte si [4.29 – „Integrace s Internetovými Prohlížeči úložiště“](#) pro další informace.



Nastavit vlastností projektu na adresáře

These special project properties must be set on *folders* for the system to work. When you commit a file or folder the properties are read from that folder. If the properties are not found there, TortoiseSVN will search upwards through the folder tree to find them until it comes to an unversioned folder, or the tree root (eg. `C:\`) is found. If you can be sure that each user checks out only from e.g. `trunk/` and not some sub-folder, then it is sufficient to set the properties on `trunk/`. If you can't be sure, you should set the properties recursively on each sub-folder. A property setting deeper in the project hierarchy overrides settings on higher levels (closer to `trunk/`).

For project properties *only* you can use the **Recursive** checkbox to set the property to all sub-folders in the hierarchy, without also setting it on all files.

When you add new sub-folders using TortoiseSVN, any project properties present in the parent folder will automatically be added to the new child folder too.



Výstraha

Ačkoli vlastnosti projektu TortoiseSVN jsou velmi užitečné, fungují jen v TortoiseSVN a některé budou fungovat pouze v jeho novějších verzích. Pokud lidé pracující na projektu využívají různých klientů Subversion, případně mají starou verzi TortoiseSVN, možná byste mohli chtít použít háky úložiště pro prosazení zásad projektu. Vlastností projektu mohou pouze pomocí zásady zavést, nemohou je vymáhat.

4.18. Externí položky

Někdy je vhodné sestavit pracovní kopie, která je vyrobena z mnoha různých získání. Například můžete chtít, aby různé soubory nebo podadresáře přicházeli z různých míst v úložišti, nebo možná z úplně jiných úložišť. Pokud chcete, aby každý uživatel měl stejné rozložení, můžete nastavit vlastnost `svn:externals` pro stáhnutí uvedeného zdroje na místa, kde je ho zapotřebí.

4.18.1. Externí adresáře

Let's say you check out a working copy of `/project1` to `D:\dev\project1`. Select the folder `D:\dev\project1`, right click and choose **Windows Menu → Properties** from the context menu. The Properties Dialog comes up. Then go to the Subversion tab. There, you can set properties. Click **Add...**

Select the `svn:externals` property from the combobox and write in the edit box the repository URL in the format `url folder` or if you want to specify a particular revision, `-rREV url folder`. You can add multiple external projects, 1 per line. Suppose that you have set these properties on `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
```

Now click **Set** and commit your changes. When you (or any other user) update your working copy, Subversion will create a sub-folder `D:\dev\project1\sounds` and checkout the sounds project, another sub-folder `D:\dev\project1\quick_graphs` containing the graphics project, and finally a nested sub-folder `D:\dev\project1\skins\toolkit` containing revision 21 of the skin-maker project.

URLs must be properly escaped or they will not work, e.g. you must replace each space with `%20` as shown in the second example above.

Pokud chcete, aby místní cesta obsahovala mezery nebo jiné speciální znaky, můžete ji uzavřít do uvozovek, nebo použít `\` (Zpětné lomítko) jako unixový styl maskování v rozhraní před každým speciálním znakem. Samozřejmě to také znamená, že musíte použít `/` (Lomítko) jako oddělovač cesty. Všimněte si, že toto chování je nové v Subversion 1.6 a nebude u starších klientů fungovat.



Použit jednoznačná čísla revizí

Měli byste vážně uvažovat o používání jednoznačných čísel revizí pro všechny vaše externí definice, jak je popsáno výše. Pokud tak učiníte, znamená to, že můžete rozhodnout, kdy a který záběr externích informací přesně stáhnout. Kromě hlediska zdravého rozumu nebyt překvapen změnami v úložištích třetí strany, nad kterými nemusíte mít žádnou kontrolu, používání jednoznačných čísel revizí také znamená, že jak Vaše pracovní kopie postupuje zpět na předchozí revizi, Vaše externí nastavení se také vrátí do stavu, jak vypadali v té předchozí revizi, což znamená, že externí pracovní kopie bude aktualizována, aby odpovídala způsobu, jak *ony* vypadaly, když Vaše úložiště bylo v předchozí revizi. U softwarových projektů, toto může být rozdíl mezi úspěšným a neúspěšným sestavením staršího záběru Vašeho komplexní sady kódu.



Older svn:externals definitions

The format shown here was introduced in Subversion 1.5. You may also see the older format which has the same information in a different order. The new format is preferred as it supports several useful features described below, but it will not work on older clients. The differences are shown in the [Subversion Book](http://svnbook.red-bean.com/en/1.5/svn.advanced.externals.html) [http://svnbook.red-bean.com/en/1.5/svn.advanced.externals.html].

If the external project is in the same repository, any changes you make there there will be included in the commit list when you commit your main project.

Pokud je externí projekt v jiném úložišti, všechny změny v něm provedené budou při odevzdání hlavního projektu oznámeny, ale musíte tyto vnější změny odevzdat samostatně.

Pokud používáte absolutní adresu URL pro nastavení `svn:externals` a musíte Vaši pracovní kopii přemístit (tj. pokud se adresa URL úložiště změní), pak se Vaše externí nezmění a možná nebudou fungovat.

Aby se předešlo takovým problémům, klient Subversion verze 1.5 a vyšší podporuje relativní externí adresy URL. Jsou podporovány čtyři způsoby určení relativní adresy URL. V následujících příkladech,

předpokládáme, že máme dvě úložiště: jedno na `http://prikklad.cz/svn/uloz-1` a další na `http://prikklad.cz/svn/uloz-2`. Získání máme z `http://prikklad.cz/svn/uloz-1/projekt/kmen` ukládáme do `C:\Pracovní` a vlastnost `svn:externals` je nastavena na `kmen`.

Vzhledem k nadřazenému adresáři

Tyto adresy URL vždy začínat řetězcem `../` např.:

```
../../Nastroje/bla  spolecne/bla-nastroj
```

Toto vyjme `http://prikklad.cz/svn/uloz-1/nastroje/foo` do `C:\Práce\spolecne\bla-nastroj`.

Nezapomeňte, že URL je relativní k URL adresáře s vlastností `svn:externals`, ne do adresáře, kde vnější je zapsáno na disk.

Vzhledem ke kořeni úložiště

Tyto adresy URL vždy začínat řetězcem `^/` např.:

```
^/nastroje/bla  spolecne/bla-nastroj
```

Toto vyjme `http://prikklad.cz/svn/uloz-1/nastroje/bla` do `C:\Práce\spolecne\bla-nastroj`.

Můžete snadno odkazovat na jiná úložiště se stejnou `SVNParentPath` (Společný adresář obsahující několik úložišť). Například:

```
^/../Uloz-2/kladiva/drap  spolecne/drap-kladivo
```

Toto vyjme `http://prikklad.cz/svn/uloz-2/kladiva/drap` do `C:\Práce\spolecne\ drap-kladivo`.

Vzhledem ke schématu

URL začínající řetězcem `//` zkopírují pouze schématickou část URL. To je užitečné, když musí být přístup na stejné jméno hostitele přes různá schémata v závislosti na umístění v síti, např. klienti v intranetu používají `http://` zatímco externí klienti používají `svn+ssh://`. Například:

```
//Prikklad.cz/svn/uloz-1/nastroje/bla  spolecne/bla-nastroj
```

Toto vyjme `http://prikklad.cz/svn/uloz-1/nastroje/bla` nebo `svn+ssh://prikklad.cz/svn/uloz-1/nastroje/bla` podle toho, jaký způsob byl použit pro získání `C:\Pracovní`.

Vzhledem k jménu hostitele serveru

URL začínající řetězcem `/` kopíruje schéma a hostitelskou část URL, například:

```
/svn/uloz-1/nastroje/bla  spolecne/bla-nastroj
```

Toto vyjme `http://prikklad.cz/svn/uloz-1/nastroje/bla` do `C:\Práce\spolecne\bla-nastroj`. Ale pokud získáte Vaši pracovní kopii z jiného serveru na `svn+ssh://dalsi.mirror.cz/svn/uloz-1/projekt1/kmen` pak externí odkaz vyjme `svn+ssh://dalsi.mirror.cz/svn/uloz-1/nastroje/bla`.

You can also specify a peg revision after the URL if required, e.g. `http://sounds.red-bean.com/repos@19`.

Pokud potřebujete více informací o tom, jak TortoiseSVN s vlastnostmi zachází, přečtěte si [4.17 – „Nastavení projektu“](#).

Chcete-li zjistit o různých metodách přístupu společných podprojektů, přečtěte si [B.6 – „Zahrnout společný podprojekt“](#).

4.18.2. Externí soubory

Od Subversion verze 1.6, můžete přidat jeden soubor jako externí do Vaší pracovní kopie pomocí stejné syntaxe jako pro adresář. Nicméně, existují určitá omezení.

- Cesta k externímu souboru musí odkazovat na stávající adresář s verzí. Obecně je nejlepší umístit soubor přímo v adresáři, který má nastaveno `svn:externals`, ale v případě potřeby může být i v podadresáři s verzí. Naproti tomu externí adresáře automaticky vytvoří jakékoli mezilehlé adresáře bez verzí pokud je potřeba.
- URL pro externí soubor musí být ve stejném úložišti jako URL, do které bude externí soubor vložen, mezi-uložištní externí soubory nejsou podporovány.

Externí soubor se chová stejně jako jakýkoliv jiný soubor s verzí v mnoha ohledech, ale nemůže být přesunut nebo smazán pomocí běžných příkazů; místo toho musí být upravena vlastnost `svn:externals`.



File externals support incomplete in Subversion 1.6

In subversion 1.6 it is not possible to remove a file external from your working copy once you have added it, even if you delete the `svn:externals` property altogether. You have to checkout a fresh working copy to remove the file.

4.19. Větvení/Značkování

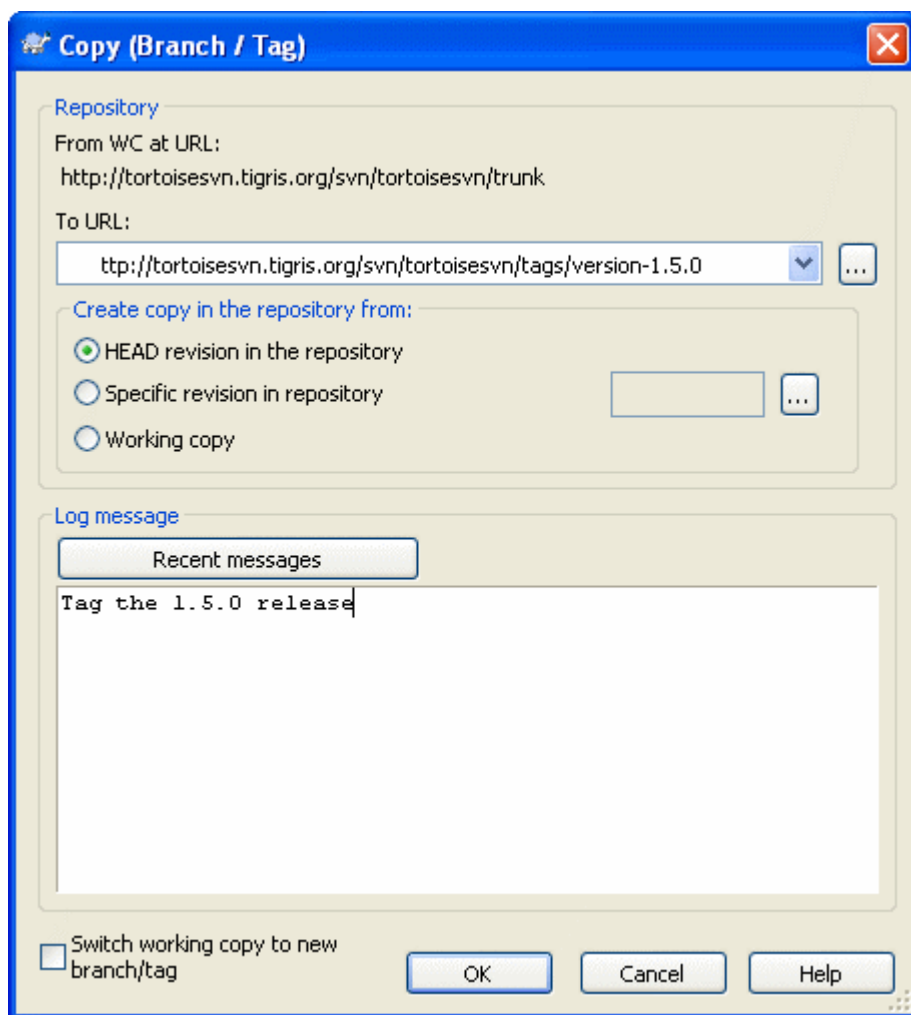
Jednou z funkcí systémů pro správu verzí je schopnost izolovat změny na samostatnou linku vývoje. Tato linka je známá jako *větev*. Větvě jsou často používány k vyzkoušení nových funkcí, aniž byste rušili hlavní vývojovou linii chybami kompilátoru a chybami programu. Jakmile je nová funkce dostatečně stabilní, pak vývojová větev je *sloučena* zpět do hlavní větve (kmen).

Dalším rysem systémů pro správu verzí je možnost označit konkrétní revizi (např. vydanou verzi), takže můžete kdykoliv obnovit některé sestavení či prostředí. Tento proces je znám jako *značkování*.

Subversion nemá žádné speciální příkazy pro větvení a značkování, místo toho používá takzvané „laciné kopie“. Laciné kopie jsou podobné pevným odkazům v Unixu, což znamená, že namísto toho, aby tvořili kompletní kopii v úložišti, je vytvořen interní odkaz, odkazující na konkrétní strom/revize. Výsledkem je, že větve a značky jsou velmi rychle vytvořeny a v úložišti nezabírají téměř žádný prostor navíc.

4.19.1. Vytvoření Větve nebo Značky

Pokud jste importovali projekt s doporučenou adresářovou strukturou, vytvoření verze větve nebo značky je velmi jednoduché:



Obrázek 4.34. Dialogové okno Větev/Značka

Vyberte adresář ve Vaší pracovní kopie, který chcete zkopírovat na větev nebo značku, pak vyberte příkaz TortoiseSVN → Větev/značka.

Výchozí cílová adresa URL pro novou větev bude zdrojová URL, na nichž je založena Vaše pracovní kopie. Budete muset tuto URL upravit na novou cestu Vaší větve/značky. Takže místo

```
http://svn.collab.net/uloz/JmenoProjektu/kmen
```

můžete nyní používat něco jako

```
http://svn.collab.net/uloz/JmenoProjektu/znacky/Vydani_1.10
```

Pokud si nemůžete vzpomenout na pojmenování, které jste použili naposledy, klikněte na tlačítko vpravo pro otevření průzkumníka úložiště, takže můžete zobrazit existující strukturu úložiště.

Nyní musíte vybrat zdroj kopie. Zde máte tři možnosti:

revize HEAD v úložišti

Nová větev je kopírována přímo v úložišti z revize HEAD. Z vaší pracovní kopie nemusí být převedeny žádné údaje a větev je vytvořena velmi rychle.

Jednoznačná revize v úložišti

Nová větev je kopírována přímo v úložišti, ale můžete si vybrat starší revizi. To je užitečné, pokud jste zapomněli udělat značku, když jste minulý týden vydali svůj projekt. Pokud si nemůžete vzpomenout na číslo revize, klikněte na tlačítko vpravo pro zobrazení záznamu revizi a odtamtud zvolte číslo revize. Opět žádná data nejsou přenášena z Vaší pracovní kopie a větev je vytvořena velmi rychle.

Pracovní kopie

Nová větev je identickou kopií Vaší místní pracovní kopie. Pokud jste aktualizovali některé soubory na starší revizi ve Vaší pracovní kopii, nebo pokud jste provedli místní změny, to je přesně to, co do kopie jde. Přirozeně tento druh složitěho značení může zahrnovat přenos dat z Vaší pracovní kopie zpět do úložiště, pokud tam už nejsou.

Pokud chcete, aby Vaše pracovní kopie byla automaticky překlopena na nově vytvořenou větev, zaškrtněte políčko **Překlopit pracovní kopie do jiné větve/značky**. Ale pokud to uděláte, nejprve se ujistěte, že Vaše pracovní kopie neobsahuje změny. Pokud ano, budou tyto změny sloučeny do větve pracovní kopie při překlopení.

Stiskněte OK pro odevzdání nové kopie do úložiště. Nezapomeňte přidat zprávu záznamu. Všimněte si, že kopie je vytvořena *uvnitř úložiště*.

Nezapomeňte, že pokud jste se rozhodly překlopit Vaši pracovní kopii do nově vytvořené větve, vytvoření větve nebo značky *neovlivní* Vaši pracovní kopii. I když vytvoříte větev z pracovní kopie, jsou tyto změny odevzdány do nové větve a ne do kmene, takže Vaše pracovní kopie může být stále označena jako změněná s ohledem na kmen.

4.19.2. Získat nebo Překlopit...

... Tot' je (ani ne) ta otázka. Zatímco získání stáhne vše od požadované větve v úložišti do Vašeho pracovního adresáře, TortoiseSVN → Překlopit... pouze převede změněná data do Vaší pracovní kopie. Dobré pro zatížení sítě, dobré pro Vaši trpělivost. :-)

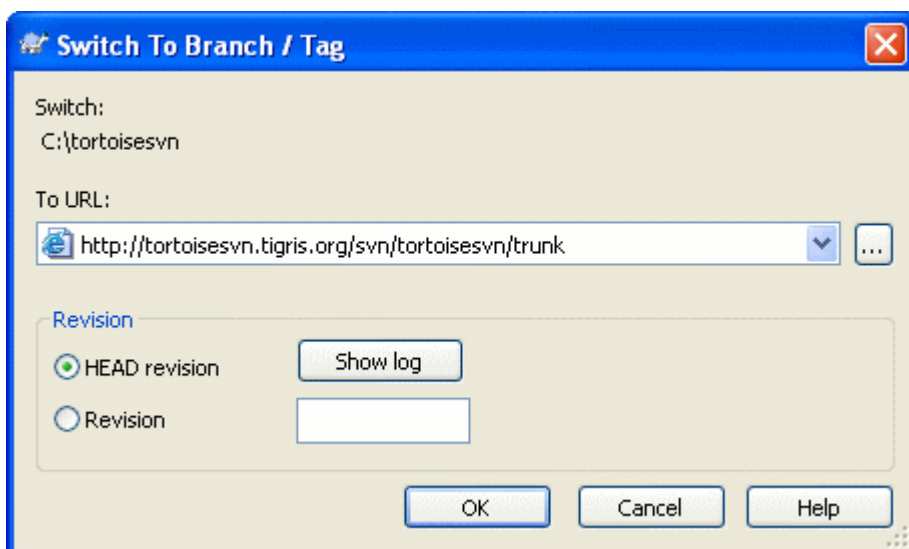
Abyste mohli pracovat s čerstvě vytvořenou větví nebo značkou máte několik způsobů, jak toho dosáhnout. Můžete:

- TortoiseSVN → Získat pro čerstvé získání do prázdné složky. Můžete získat do jakékoliv místa na místním disku a můžete vytvořit tolik pracovní kopie z úložiště, jak se Vám líbí.
- Překlopit aktuální pracovní kopii na nově vytvořenou kopii v úložišti. Opět vyberte adresář nejvyšší úrovně Vašeho projektu a použijte TortoiseSVN → Překlopit ... z kontextového menu.

V dalším dialogovém okně zadejte adresu URL větve, kterou jste právě vytvořili. Vyberte přepínač **Revize Head** a klikněte na OK. Vaše pracovní kopie bude překlopena do nové větve/značky.

Překlopení funguje podobně jako Aktualizace v tom, že nikdy nezahodí Vaše místní změny. Když Překlopíte, veškeré změny, které jste provedli ve Vaší pracovní kopii, které dosud nebyly odevzdány budou sloučeny. Pokud toto nechcete, pak musíte buď odevzdat změny před překlopením, nebo se vrátit Vaší pracovní kopii do již odevzdané revize (typicky HEAD).

- Pokud chcete pracovat na kmeni a větví, ale nechcete to na úkor nového získání, můžete použít Průzkumníka Windows pro vytvoření kopie získání Vašeho kmene v jiném adresáři, pak TortoiseSVN → Překlopit... tuto kopii do Vaší nové větve.



Obrázek 4.35. Dialogové okno Překlopení

Ačkoli Subversion sám nečiní žádný rozdíl mezi značkami a větvemi, způsob, jakým se obvykle používají se trochu liší.

- Tags are typically used to create a static snapshot of the project at a particular stage. As such they not normally used for development - that's what branches are for, which is the reason we recommended the `/trunk /branches /tags` repository structure in the first place. Working on a tag revision is *not a good idea*, but because your local files are not write protected there is nothing to stop you doing this by mistake. However, if you try to commit to a path in the repository which contains `/tags/`, TortoiseSVN will warn you.
- Je možné, že je třeba provést další změny ve vydání, které jste již označovali. Správný způsob, jak toto zvládnout je nejdříve vytvořit novou větev ze značky a větev odevzdat. Proveďte Vaše Změny této větve a pak vytvořte novou značku z této nové větve, např. `Verze_1.0.1`.
- Pokud změníte pracovní kopii vytvořenou z větve a odevzdáte ji, pak všechny změny jdou do nové větve a *ne* do kmene. Pouze změny jsou uloženy. Zbytek zůstává lacinou kopií.

4.20. Sloučení

Tam, kde jsou větve používány k udržení samostatných linií vývoje, v určité fázi budete chtít provedené změny sloučit v jedné větvi zpět do kmene, nebo naopak.

Je důležité pochopit, jak větvení a slučování v Subversion funguje předtím, než ho začnete používat, protože to může být docela složité. Důrazně doporučuje, abyste si přečetli kapitolu [Větvení a slučování](http://svnbook.red-bean.com/en/1.5/svn.branchmerge.html) [http://svnbook.red-bean.com/en/1.5/svn.branchmerge.html] v knize Subversion, která dává úplný popis a mnoho příkladů toho, jak se používá.

Další bod pro zapamatování je, že sloučení se *vždy* odehrává v pracovní kopii. Pokud chcete sloučit změny *do* větve, musíte získat pracovní kopii této větve a spustit Průvodce sloučením z této pracovní kopie pomocí TortoiseSVN → Sloučit....

Obecně je dobrý nápad provést sloučení do nezměněné pracovní kopie. Pokud jste již provedli další změny ve Vaší pracovní kopii, nejdříve je odevzdejte. Jestliže slučovací nevyjde tak, jak jste očekávali, můžete změny vrátit a příkaz **Vrátit** zahodí *všechny* změny včetně těch odevzdaných před sloučením.

Existují tři běžné případy použití slučování, které jsou provedeny mírně odlišnými způsoby, jak je popsáno níže. Na první straně Vás průvodce sloučením vyzve k výběru metody, kterou potřebujete.

Sloučit řadu revizí

Tato metoda se vztahuje na případ, když máte jednu nebo více revizí ve větvi (nebo v kmeni) a chcete tyto změny přenést na jinou větev.

Co požádáte, aby Subversion udělal, je toto: „Vypočítat nezbytné změny, jak se dostat [Z] revize 1 větve A [DO] revize 7 větve A a tyto změny provést v mé pracovní kopii (kmene nebo větve B).“

Znovu začlenit větev

Tato metoda řeší případ, kdy jste vytvořili větev funkcí, jak je popsáno v knize Subversion. Všechny změny kmene byly přenášeny týden po týdnu do větve funkcí a nyní, když je tato funkce kompletní ji chcete sloučit zpátky do kmene. Protože jste zachovali synchronizaci větve s kmenem, bude nejnovější verze větve a kmene naprosto identická s výjimkou změn ve Vaší větvi.

Toto je zvláštní případ sloučení stromu, který je popsán níže, a pouze vyžaduje URL pro sloučení z (obvykle) Vaší vývojové větve. Používá Subversion funkce sledování sloučení pro výpočet správného rozsahu revizí k použití a provede další kontroly, které zajistí, že větev je plně aktualizována změnami z kmene. To zaručuje, že náhodou nevrátíte zpět práci, kterou jiní odevzdali do kmene od Vaší poslední synchronizace změn.

Po sloučení, všechny vývoj ve větvi je kompletně sloučen zpět do hlavní vývojové linie. Větev je nyní zbytečná a může být smazána.

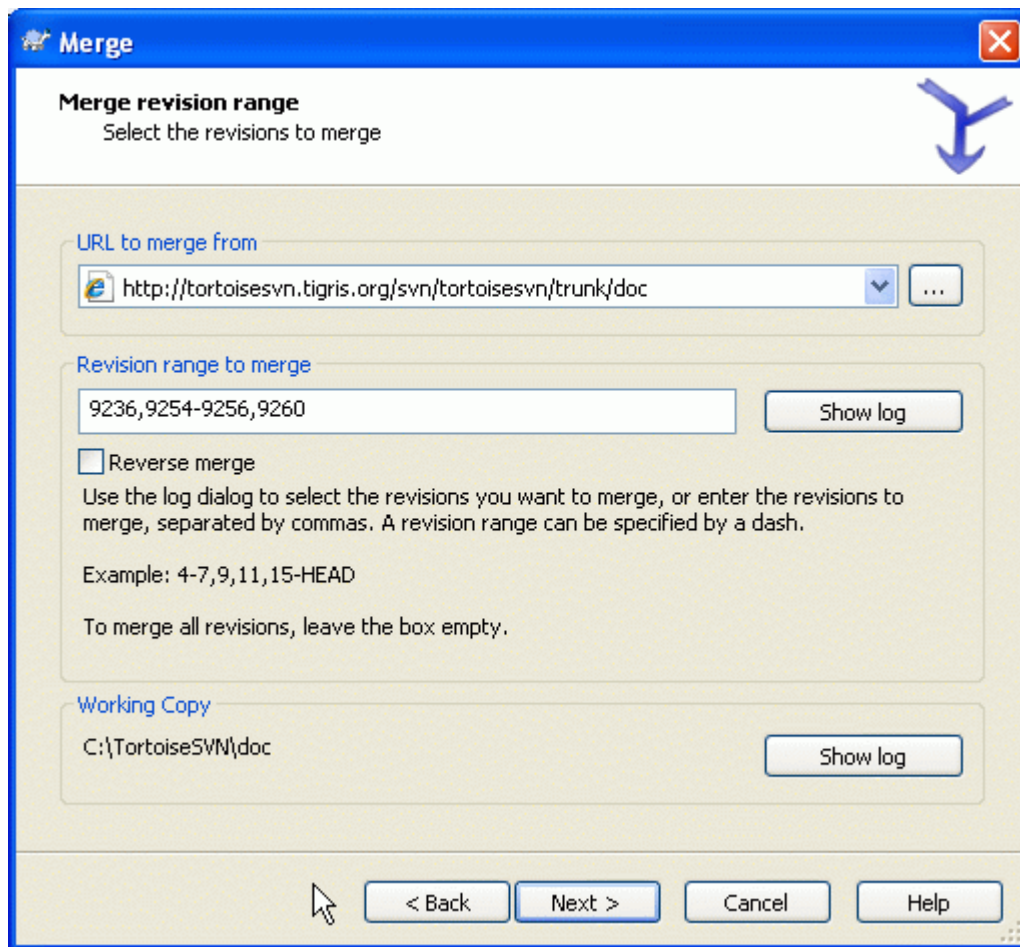
Jakmile jste provedli znovu začlenění sloučení, neměli byste ji nadále používat pro vývoj. Důvodem je to, že pokud se pokusíte později znovu synchronizovat Vaši stávající větev z kmene, sledování sloučení uvidí vaše znovuzačlenění jako změnu kmene, která dosud nebyly do větve sloučena a pokusí se sloučení větve do kmene sloučit zpět do větve! Řešením je jednoduše vytvořit novou větev z kmene, pro pokračování další fáze Vašeho vývoje.

Sloučení dvou různých stromů

Toto je obecnější případ znovu začleňovací metody. Co žádáte od Subversion, aby udělal, je: „Vypočítat změny nezbytné pro získání [Z] revize head kmene [DO] head revize větve a uplatnit tyto změny na moji pracovní kopii (kmene).“ Výsledkem je, že kmen nyní vypadá přesně jako větev.

Pokud Váš server/úložiště nepodporuje sledování sloučení pak je to jediný způsob, jak sloučit větev zpátky do kmene. Další případ použití nastane, když používáte dodavatelské větve a je třeba sloučit změny v důsledku nového vývoje dodavatele do kmenového kódu. Pro více informací si přečtěte kapitolu [Dodavatelské větve](http://svnbook.red-bean.com/en/1.5/svn.advanced.vendorbr.html) [http://svnbook.red-bean.com/en/1.5/svn.advanced.vendorbr.html] v knize Subversion.

4.20.1. Sloučení Rozsahu Revizí



Obrázek 4.36. Průvodce sloučením - Vyberte Rozsah Revizí

V poli Od: zadejte úplnou adresu URL adresáře větve nebo značky obsahující změny, které chcete přenést do Vaší pracovní kopie. Můžete také klepnout na ... pro procházení úložiště a vyhledání požadované větve. Pokud jste předtím slučovali z této větve, pak stačí použít rozbalovací seznam, který zobrazuje historii dříve použitých adres URL.

V poli Rozsah revizí ke sloučení zadejte seznam revizí, které chcete sloučit. To může být jedna revize, seznam konkrétních revizí oddělených čárkami nebo rozsah revizí oddělených pomlčkou, nebo jakákoliv jiná kombinace těchto možností.



Důležité

Existuje významný rozdíl ve způsobu, jakým je zadáván rozsah revizí v TortoiseSVN v porovnání s příkazovým řádkem klienta. Nejjednodušší způsob, jak si toto představit, je uvažovat o plotu se sloupky a plotovými výplněmi.

S příkazovým řádkem klienta zadáte změny ke sloučení pomocí dvou „sloupků plotu“ revizí, které stanoví body *před* a *po*.

S TortoiseSVN zadáváte seznam změn ke sloučení pomocí „plotových výplní“. Důvod pro to je jasný, když použijete dialogové okno záznamu k určení revizí ke sloučení, kde každá revize se objeví jako seznam změn.

Pokud slučujete revize v kusech, musíte podle metody uvedené v knize Subversion sloučit 100-200 teď a 200-300 příště. S TortoiseSVN byste slučovali 100-200 teď a 201-300 příště.

Tento rozdíl způsobil hodně nevraživosti v e-mailových konferencích. Jsme si vědomi, že existuje rozdíl v příkazovém řádku klienta, ale věříme, že pro většinu uživatelů GUI je snazší pochopit způsob, který jsme zavedli.

Nejjednodušší způsobem, jak zvolit rozsah revizí, který potřebujete, je kliknout na **Zobrazit záznamy**, protože toto zobrazí seznam nedávných změn s jejich komentáři záznamů. Pokud chcete sloučit změny z jedné revize, stačí tuto revizi vybrat. Pokud chcete sloučit změny z několika revizí, pak vyberte rozsah (za použití obvyklých **Shift**-Modifikátor). Klikněte na **OK** a seznam čísel revizí ke sloučení je vyplněn za Vás.

Pokud chcete sloučit změny zpět z Vaší pracovní kopie, vrátit změny, které jste již odevzdali, vyberte revize k vrácení a ujistěte se, že políčko **Reverzní sloučení** je zaškrtnuto.

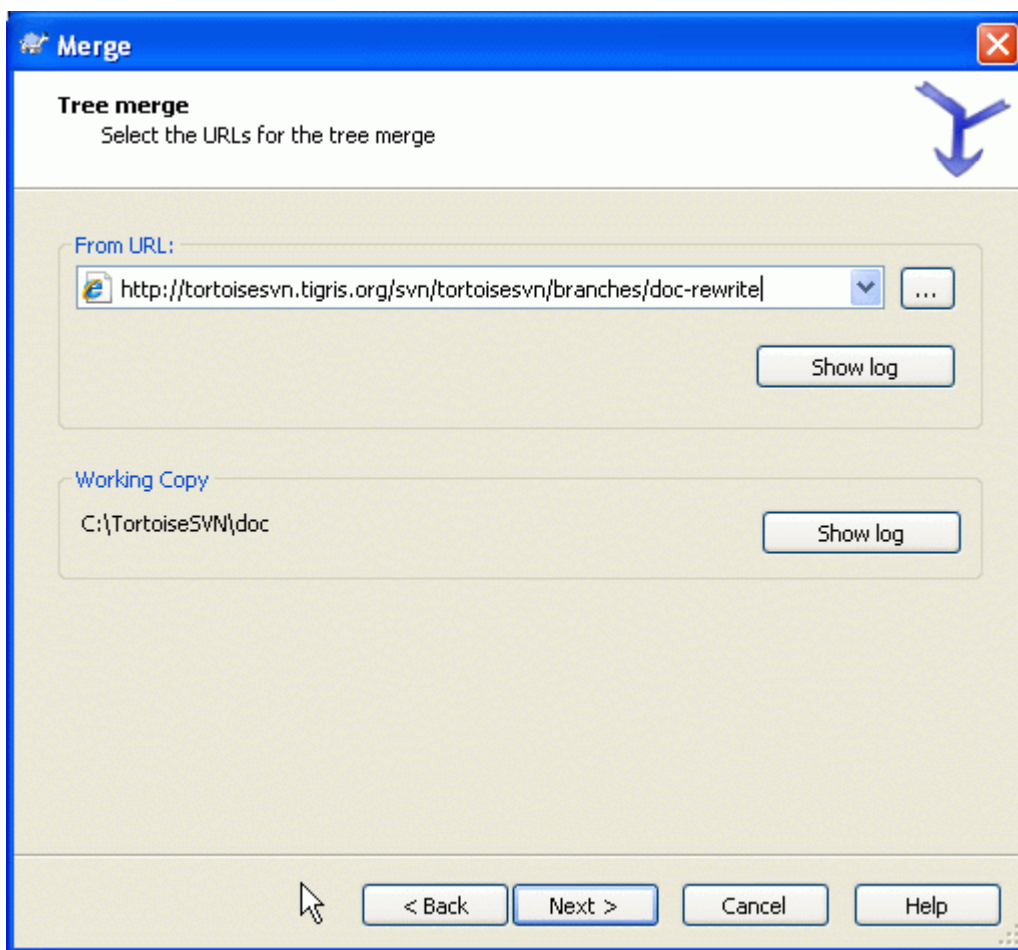
Pokud jste již sloučili některé změny z této větve, doufejme, že jste si poznamenali poslední sloučenou revizi ve zprávách záznamu, když jste odevzdali změny. V takovém případě můžete použít **Zobrazit záznam** pracovní kopie pro vyhledání této zprávy záznamu. Pamatujte si, že o revizích uvažujeme jako o sadě změn. Měli byste použít další revizi po bodu posledního sloučení jako výchozí bod pro toto sloučení. Například, pokud jste posledně sloučil revize 37 až 39, pak počátečním bodem pro toto sloučení by měla být revize 40.

Pokud používáte Subversion funkce sledování sloučení, nemusíte si pamatovat, která revize je již sloučená - Subversion to bude zaznamenávat za Vás. Ponecháte-li rozsahu revizí prázdný, všechny revize, které dosud nebyly sloučeny budou zahrnuty. Přečtěte si **4.20.6 – „Sledování Sloučení“** a dozvíte se více.

Pokud ostatní můžou odevzdávat změny, pak buďte opatrní při používání revize HEAD. Nemusí odkazovat na revizi, kterou myslíte, pokud někdo jiný provedl odevzdání po Vaší poslední aktualizaci.

Click Next and go to **4.20.4 – „Nastavení Sloučení“**

4.20.2. Znovu začlenit větev



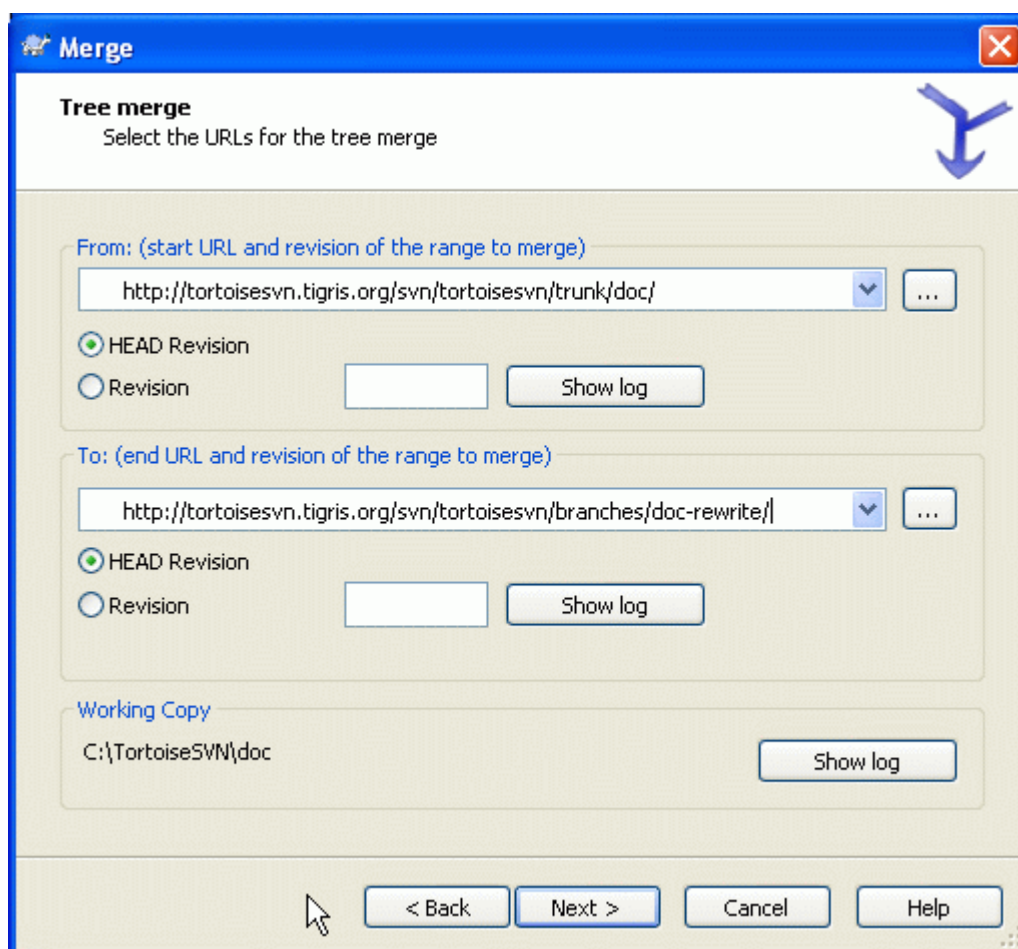
Obrázek 4.37. Průvodce Sloučením - Znovu začleňovací Sloučení

Chcete-li sloučit větev funkcí zpět do kmene, musíte spustit průvodce sloučením z pracovní kopie kmene.

V poli Z URL: zadejte úplnou adresu URL adresáře větve, kterou chcete sloučit zpět. Můžete také kliknout na ... k procházení úložiště.

Existují některé podmínky, které se vztahují na znovu začleňovací sloučení. Za prvé, server musí podporovat sledování sloučení. Pracovní kopie musí mít nekonečnou hloubku (žádné řídké získání) a nesmí mít žádné lokální úpravy, překlopené položky nebo položky, které byly aktualizovány na revizi jinou než HEAD. Všechny změny provedené v kmeni v průběhu vývoje větve musí být sloučeny do větve (nebo označeny jako sloučené). Rozsah revizí sloučení bude vypočten automaticky.

4.20.3. Sloučení Dvou Různých Stromů



Obrázek 4.38. Průvodce Sloučením - Sloučení Stromu

Pokud používáte tuto metodu ke sloučení větve funkcí zpět do kmene, budete muset spustit průvodce sloučením z pracovní kopie kmene.

V poli Od: zadejte úplnou adresu URL adresáře *kmene*. To může znít špatně, ale nezapomeňte, že kmen je počáteční bod, ke kterému chcete přidat změny větve. Můžete také kliknout na ... k procházení úložiště.

V poli Do: zadejte úplnou adresu URL adresáře větve funkcí.

V obou polích Od Revize a Na Revizi, zadejte číslo poslední revize, při které byly dva stromy synchronizovány. Pokud si jste jisti, že nikdo jiný neodevzdává, můžete použít revizi HEAD v obou případech. Pokud existuje možnost, že někdo jiný mohl odevzdat od synchronizace, použijte konkrétní číslo revize, aby se zabránilo ztrátě novějších odevzdání.

Můžete také použít Zobrazit záznamy pro vybrání revize.

4.20.4. Nastavení Sloučení

Tato stránka průvodce umožňuje zadat rozšířené možnosti před zahájením procesu sloučení. Většinou stačí použít výchozí nastavení.

Můžete určit hloubku použitou pro sloučení, tj. jak hluboko do Vaší pracovní kopie by sloučení mělo jít. Použité pojmy hloubky jsou popsány v 4.3.1 – „Hloubka Získání“. Výchozí hloubka je Pracovní kopie, která využívá stávající nastavení hloubky a téměř vždy je to, co chcete.

Most of the time you want merge to take account of the file's history, so that changes relative to a common ancestor are merged. Sometimes you may need to merge files which are perhaps related, but not in your repository. For example you may have imported versions 1 and 2 of a third party library into two separate directories. Although they are logically related, Subversion has no knowledge of this because

it only sees the tarballs you imported. If you attempt to merge the difference between these two trees you would see a complete removal followed by a complete add. To make Subversion use only path-based differences rather than history-based differences, check the **Ignore ancestry** box. Read more about this topic in the Subversion book, *Noticing or Ignoring Ancestry* [<http://svnbook.red-bean.com/en/1.5/svn.branchmerge.advanced.html#svn.branchmerge.advanced.ancestry>]

Můžete určit, jak jsou řešeny změny mezer a konce řádků. Tyto možnosti jsou popsány v **4.10.2 – „Možnosti konců řádků a mezer“**. Výchozí nastavení je zacházet se všemi rozdíly mezer a konce řádků jako skutečné změny, které mají být sloučeny.

Pokud používáte sledování sloučení a chcete označit revizi jako sloučenou, aniž by se skutečně slučovala, zaškrtněte políčko **Pouze zaznamenat sloučení**. Existují dva možné důvody, proč byste toto chtěli udělat. Je možné, že sloučení je příliš složité pro slučovací algoritmy, takže změny kódu uděláte ručně, pak označíte změny jako sloučené, takže algoritmus sledování sloučení je si toho vědom. Nebo budete chtít zabránit sloučení určité revize. Označit ji jako sloučenou se zabrání výskytu sloučení v klientech podporující sledování sloučení.

Now everything is set up, all you have to do is click on the **Merge** button. If you want to preview the results **Test Merge** performs the merge operation, but does *not* modify the working copy at all. It shows you a list of the files that will be changed by a real merge, and notes those areas where conflicts will occur.

Dialogové okno průběhu sloučení ukazuje každou fázi sloučení, s uvedeným rozsahem revizí. To může znamenat o jednu revizi více, než jste čekali. Například, pokud chcete sloučit revize 123, dialogové okno průběhu oznámí „Slučuji revize 122 až 123“. Je třeba si uvědomit, že sloučení je úzce spjato s porovnáním. Proces sloučení funguje tak, že tvoří seznam rozdílů mezi dvěma body v úložišti a tyto rozdíly vnese do pracovní kopie. Dialogové okno průběhu jen ukazuje počáteční a koncové body pro porovnání.

4.20.5. Zhodnocení výsledků sloučení

Sloučení je nyní dokončeno. Je dobrý nápad podívat se na sloučení a zkontrolovat, jestli dopadlo podle očekávání. Sloučení je obvykle poměrně složité. Konflikty často vznikají v případě, že větev se odchýlila příliš daleko od kmene.

V klientech a serverech Subversion před verzí 1.5, se žádné informace o slučování neukládaly a sloučené revize musely být sledovány ručně. Pokud jste testovali změny a chcete odevzdat tuto revizi, Vaše zpráva záznamu odevzdání by měla *vždy* obsahovat čísla revizí, které byly ve sloučení přeneseny. Pokud chcete později použít jiné sloučení budete potřebovat vědět, co jste již sloučili, protože nechcete přenést změnu více než jednou. Více informací o tomto najdete na *Nejlepší postupy pro slučování* [<http://svnbook.red-bean.com/en/1.4/svn.branchmerge.copychanges.html#svn.branchmerge.copychanges.bestprac>] v knize Subversion.

Pokud Váš server a všichni klienti mají Subversion 1.5 nebo vyšší, bude funkce pro sledování sloučení, zaznamenávat sloučené revize a vyhne se slučování revizí více než jednou. Tím se Váš život stane mnohem jednodušší, protože můžete vždycky jednoduše sloučit celou řadu revizí a víte, že pouze nové revize bude skutečně sloučeny.

Správa větve je důležitá. Pokud chcete, aby tato větev byla aktuální s kmenem, měli byste se ujistit, že slučujete často, aby se větev a kmen neodchýlili příliš daleko od sebe. Samozřejmě, měli byste se stále vyhnout opakovanému slučování změn, jak je vysvětleno výše.



Tip

Pokud jste právě sloučili větev funkcí zpět do kmene, kmen nyní obsahuje všechny nový kód funkce a větev je zbytečná. Nyní, jestli chcete, ji můžete smazat z úložiště.



Důležité

Subversion nesloučí soubor s adresářem a naopak - pouze adresáře do adresářů a soubory do souborů. Pokud kliknete na soubor a otevřete dialogové okno sloučení, pak musíte zadat

cestu k souboru v tomto dialogovém okně. Pokud vyberete adresář a spustíte dialogové okno, pak musíte zadat URL adresáře pro sloučení.

4.20.6. Sledování Sloučení

Subversion 1.5 představilo prostředky pro sledování sloučení. Při sloučení změn z jednoho stromu do druhého, čísla sloučených revizí jsou uložena a tyto informace mohou být použity pro několik různých účelů.

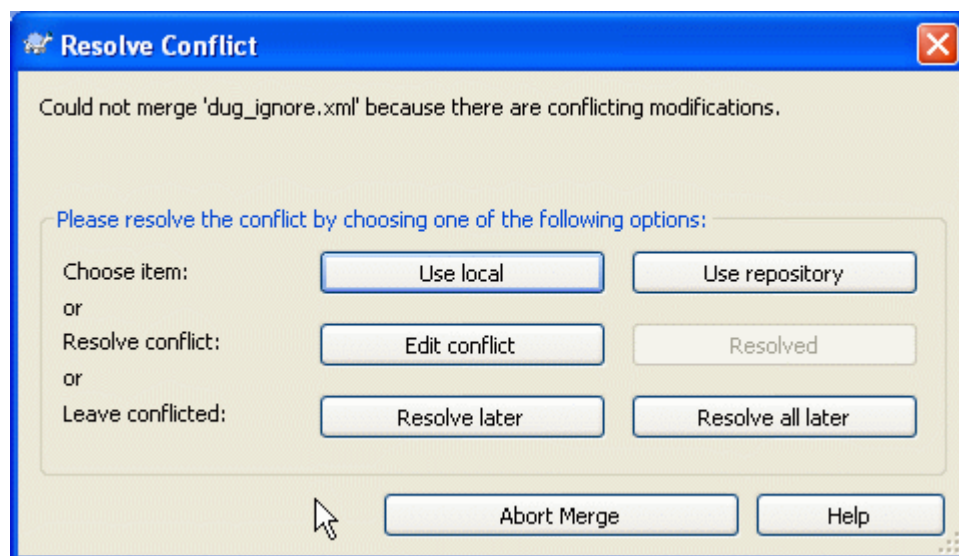
- Můžete se vyhnout nebezpečí sloučení stejné revize dvakrát (problém opakovaného sloučení). Jakmile je revize označena jako sloučená, budoucí sloučení, kde je tato revize v rozsahu bude přeskočena.
- Při sloučení větve zpět do kmene Vám dialogové okno záznamů může ukázat odevzdání do větve jako součást záznamu kmene, což dává lepší sledovatelnost změn.
- Když zobrazíte dialogové okno záznamů v rámci dialogového okna sloučení, jsou již sloučené revize zobrazeny v šedé barvě.
- Při zobrazování informací o obvinění v souboru, můžete zvolit zobrazení původního autora sloučených revizí, raději než osobu, která sloučení provedla.
- Můžete označit revize, které se *nemají slučovat* jejich začleněním do seznamu sloučených revizí, aniž by se sloučení skutečně provedlo.

Informace o sledování sloučení jsou uloženy ve vlastnosti `svn:mergeinfo` klientem, při provádění sloučení. Když je sloučení odevzdání, server uloží informace v databázi a když budete požadovat sloučení, záznam nebo informace o obvinění, server může reagovat odpovídajícím způsobem. Aby systém správně fungoval musíte zajistit, že server, úložiště a všechny klienty jsou aktualizovány. Starší klienty nebudou ukládat vlastnost `svn:mergeinfo` a starší servery nebudou poskytovat informace novým klientům.

Find out more about merge tracking from Subversion's [Merge tracking documentation](http://subversion.tigris.org/merge-tracking/index.html) [http://subversion.tigris.org/merge-tracking/index.html].

4.20.7. Řešení Konfliktů při Sloučení

Sloučení nemusí vždy jít hladce. Někdy dojde ke konfliktu, a pokud jste slučovali větší rozsah, většinou chcete konflikt vyřešit před zahájením sloučení další řady revizí. TortoiseSVN Vám pomůže skrz tento proces zobrazením dialogového okna *Zpětného Volání Sloučení při Konfliktu*.



Obrázek 4.39. Dialogové Okno Zpětného Volání Sloučení při Konfliktu

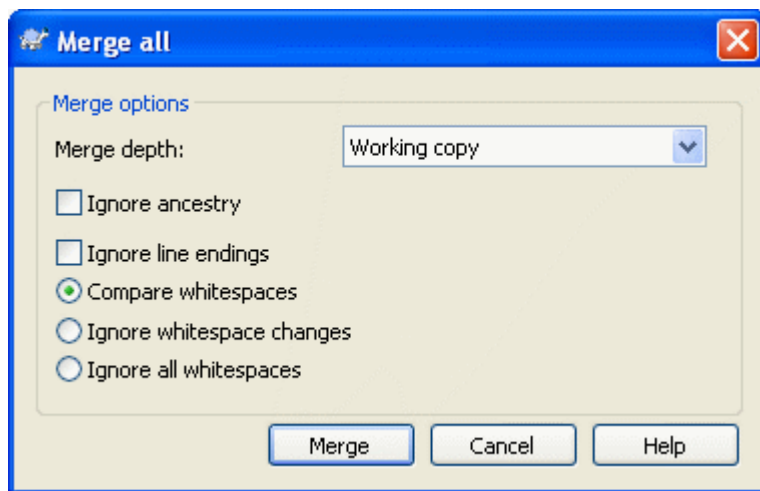
When a conflict occurs during the merge, you have three ways to handle it.

1. You may decide that your local changes are much more important, so you want to discard the version from the repository and keep your local version. Or you might discard your local changes in favour of the repository version. Either way, no attempt is made to merge the changes - you choose one or the other.
2. Normally you will want to look at the conflicts and resolve them. In that case, choose the **Edit Conflict** which will start up your merge tool. When you are satisfied with the result, click **Resolved**.
3. Poslední možností je rozhodnutí odložit a pokračovat se slučováním. Můžete si vybrat, co dělat s jedním konfliktním souborem nebo se všemi soubory po zbytek sloučení. Nicméně pokud existují v souboru další změny, nebude možné sloučení dokončit.

Pokud nechcete používat toto interaktivní zpětné volání, existuje zaškrťovací políčko v dialogovém okně průběhu sloučení, **Neinteraktivní slučování**. Pokud je toto nastaveno pro sloučení a sloučení by mělo konflikt, je soubor označen jako v konfliktu a sloučení pokračuje. Konflikty budete muset vyřešit, až celé sloučení skončí. Pokud toto není nastaveno, pak předtím, než je soubor označen jako konfliktní, máte šanci konflikt vyřešit *během* sloučení. Má to tu výhodu, že pokud je soubor slučován vícekrát (více revizí zahrnující změnu tohoto souboru), můžou následná sloučení uspět v závislosti na tom, které řádky jsou zasaženy. Ale samozřejmě si nemůžete odejít na kafe zatímco slučování běží ;)

4.20.8. Sloučení Dokončenou větev

Pokud chcete sloučit všechny změny z větve funkcí zpět do kmene, pak můžete použít TortoiseSVN → Znovu začlenění sloučení z rozšířeného kontextového menu (podržte klávesu **Shift**, při kliknutí pravým tlačítkem na soubor).



Obrázek 4.40. Dialogové okno znovu začlenění sloučení

Toto dialogové okno je velmi jednoduché. Jediné, co musíte udělat, je nastavit volby pro sloučení, jak je popsáno v [4.20.4 – „Nastavení Sloučení“](#). O zbytek se postará TortoiseSVN automaticky pomocí sledování sloučení.

4.20.9. Údržba Větvě Funkcí

Při vývoji nové funkce na samostatnou větev je dobrý nápad vymyslet zásadu pro znovuzачlenění, když je funkce hotova. Pokud probíhá i jiná práce v kmene ve stejnou dobu můžete zjistit, že rozdíly se v průběhu času stanou významnými a zpětné sloučení se stává noční můrou.

Pokud je tato funkce relativně jednoduchá a vývoj nebude trvat dlouho, pak můžete přijmout jednoduchý přístup, což je držet větev úplně oddělenou, dokud není funkce hotova, pak sloučit změny ve větvi zpět do kufří. V průvodci sloučením by toto bylo jednoduché kliknutí na **Sloučit rozsah revizí**, kde rozsah revizí je rozsah revizí větve.

Pokud tato funkce bude trvat déle a je třeba zohlednit změny v kmene, pak je třeba, aby větev byla synchronizována. To prostě znamená, že pravidelně slučujete změny kmene do větve, tak, aby větev obsahovala všechny změny kmene *plus* novou funkci. Proces synchronizace využívá **Sloučit rozsah revizí**. Když je funkce dokončena pak ji můžete sloučit zpět do kmene buď pomocí **Znovu začlenit větve** nebo **Sloučit dva různé stromy**.

4.21. Zamykání

Subversion většinou funguje nejlépe bez zamykání, použitím metody „Kopírovat-Změnit-Sloučit“ popsané v předchozí části **2.2.3 – „Řešení pomocí modelu Zkopírovat-Změnit-Sloučit“**. Nicméně existuje několik případů, kdy může být nutné zavést nějakou formu zásady zamykání.

- Používáte „neslučitelné“ soubory, například grafické. Pokud dva lidé změní stejný soubor, sloučení není možné, aniž by jeden z Vás neztratil svoji změnu.
- Vaše firma vždy v minulosti používala Zamykací systém pro správu revizí a vedení došlo k rozhodnutí, že „zamykání je nejlepší“.

Nejprve je nutné se ujistit, že Váš Subversion server je aktualizován alespoň na verzi 1.2. Dřívější verze vůbec nepodporují zamykání. Pokud vstupujete přes protokol `file://`, pak samozřejmě pouze Váš klient musí být aktualizován.

4.21.1. Jak Zamykání v Subversion Funguje

Ve výchozím nastavení nic není zamčené a každý, kdo má přístup k odevzdání může kdykoli odevzdat změny jakéhokoliv souboru. Ostatní budou pravidelně aktualizovat své pracovní kopie a změny v úložišti budou sloučeny s místními změnami.

Pokud *Zamknete* soubor, pak jenom Vy můžete tento soubor odevzdat. Odevzdání všech ostatních uživatelů bude blokováno, dokud Vy neuvolníte zámek. Zamknutý soubor nemůže být žádným způsobem v úložišti změněn, také nemůže být smazán nebo přejmenován nikým jiným, než majitelem zámku.

Nicméně, ostatní uživatelé nemusí vědět, že jste si vzali zámek. Pokud nebudou kontrolovat stav zámku pravidelně, budou poprvé vědět o zámku, když jejich odevzdání selže, což ve většině případů není příliš užitečné. Aby bylo snazší zámky spravovat, je zde nová vlastnost Subversion `svn:needs-lock`. Když je tato vlastnost nastavena (na jakoukoli hodnotu) na soubor, kdykoli je soubor získán, nebo aktualizován, je místní kopie pouze pro čtení *pokud* pracovní kopie neudrží zámek pro tento soubor. Toto se chová jako varování, že byste neměli upravovat soubor pokud jste nejdříve nezískali zámek. Soubory, které jsou s verzí a pouze pro čtení, jsou v TortoiseSVN označeny speciálním překryvem, který uvádí, že před úpravou potřebujete získat zámek.

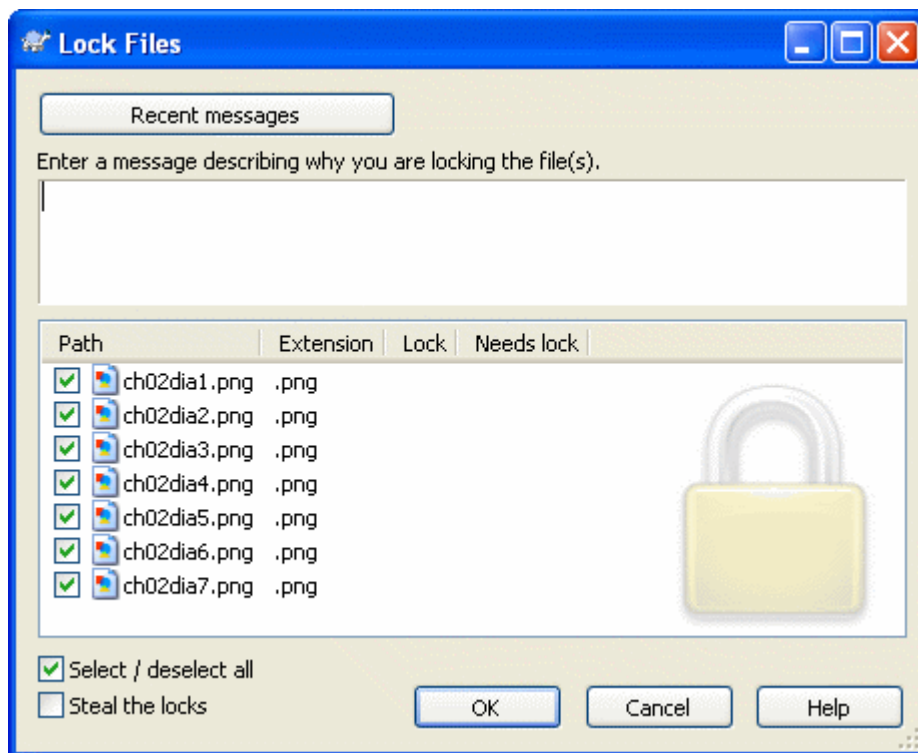
Zámky jsou zaznamenávány v umístění pracovní kopie, stejně jako majitelem. Pokud máte několik pracovních kopií (doma, v práci), pak můžete mít pouze zámek v *jedné* z těchto pracovních kopií.

Pokud jeden z vašich spolupracovníků získá zámek a pak odjede na dovolenou, aniž by ho uvolnil, co uděláte? Subversion poskytuje prostředky k vyrazení zámku. Uvolnění zámku v držení někoho jiného, je označováno jako *Rozbití* zámku a násilné získání zámku, který již někdo vlastní, je označován jako *Ukradnutí* zámku. Přirozeně to nejsou věci, které byste měli brát na lehkou váhu, chcete-li zůstat přáteli se svými spolupracovníky.

Zámky jsou zaznamenávány v úložišti, a znak zámku je vytvořen ve Vaší místní pracovní kopii. Pokud existuje rozpor, například pokud někdo jiný rozbil zámek, znak místního zámku se stává neplatným. Úložiště je vždy rozhodný odkaz.

4.21.2. Získání zámku

Vyberte soubor(y) v pracovní kopii, pro které chcete získat zámek, pak vyberte příkaz TortoiseSVN → Uzamknout....



Obrázek 4.41. Dialogové okno Uzamknutí

Zobrazí se dialogové okno, kde můžete zadat komentář, takže ostatní uvidí, proč jste zamkl soubor. Komentář je nepovinný a v současné době používán pouze se svnservu úložišti. Jestliže (a *pouze* jestliže), musíte ukrást zámek někoho jiného, zaškrtněte políčko **Ukrást zámek** a poté klikněte na OK.

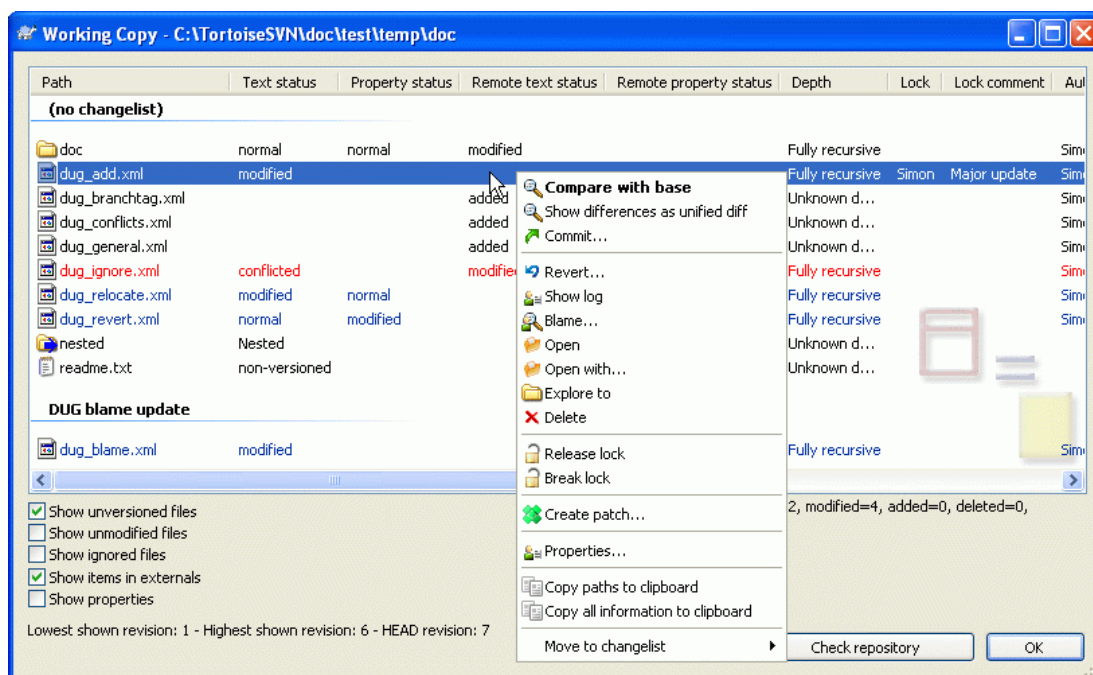
Pokud vyberete adresář a potom použijete TortoiseSVN → Uzamknout... dialogové okno uzamknutí se otevře s *každým* souborem v *každém* podadresáři vybraným pro zamknutí. Pokud opravdu chcete zamknout celou hierarchii, toto je způsob, jak to udělat, ale mohly byste se stát velmi neoblíbenými u svých spolupracovníků, pokud je do celého projektu nepustíte. Používejte opatrně ...

4.21.3. Uvolnění zámku

Abyste se ujistili, že nezapomenete uvolnit zámek, který už nepotřebujete, jsou uzamčené soubory zobrazeny v dialogovém okně odevzdání a ve výchozím nastavení jsou vybrány. Pokud budete v odevzdání pokračovat, zámky, které držíte na vybraných souborech, jsou odstraněny i když soubory nebyly změněny. Pokud na některém souboru nechcete uvolnit zámek, můžete zrušit jejich zaškrtnutí (pokud nejsou upraveny). Pokud chcete zachovat zámek na souboru, který jste upravili, musíte zaškrtnout políčko **Ponechat zámky před odevzdáním změn**.

K ručnímu uvolnění zámku, vyberte soubor(y) v pracovní kopii, kterým chcete uvolnit zámek, pak vyberte příkaz TortoiseSVN → Uvolnit zámek. Nic dalšího k zadání není a tak TortoiseSVN zkontaktuje úložiště a zámky uvolní. Můžete také použít tento příkaz na adresář, aby všechny zámky byly rekurzivně uvolněny.

4.21.4. Kontrola stavu zámku



Obrázek 4.42. Dialogové okno Zjistit Změny

Chcete-li vidět, jaké zámky Vy a ostatní držíte, můžete použít TortoiseSVN → Zjistit změny.... Místně držené znaky zámeků se objeví okamžitě. Chcete-li zkontrolovat zámky držené ostatními (a zjistit, zda některý z Vašich zámeků nebyl rozbit nebo ukraden), musíte kliknout na Zkontrolovat Úložiště.

Z kontextového menu tu můžete také získat a uvolnit zámky, stejně jako rozbit nebo ukrást zámky držené ostatními.



Vyhnete se Rozbíjení a Krádeži Zámeků

Pokud rozbijete nebo ukradnete zámek někoho jiného, aniž byste jim to řekli, mohli byste potenciálně způsobit ztrátu práce. Pokud pracujete s neslučitelnými typy souborů a ukradnete zámek někoho jiného, jakmile zámek uvolníte, můžete odevzdat své změny a přepsat Vaše. Subversion neztratí data, ale Vy jste ztratili ochranu proti týmové práci, kterou Vám uzamknutí dalo.

4.21.5. Udělat z Neuzamčených Souborů Soubory Pouze Pro Čtení

Jak bylo uvedeno výše, nejvíce efektivní způsob, jak používat zamykání je nastavit vlastnost `svn:needs-lock` na soubory. Viz 4.17 – „Nastavení projektu“ pro návod, jak nastavit vlastnosti. Soubory s touto vlastností budou vždy získávány a aktualizovány s nastaveným příznakem pouze pro čtení, pokud Vaše pracovní kopii nedrží zámek.



Pro připomenutí, TortoiseSVN používá speciální překryvy pro sdělení tohoto stavu.

Máte-li provozovat zásadu, kde každý soubor musí být uzamčen, pak můžete shledat snadnějším používat funkci auto-vlastnosti Subversion pro nastavení vlastností automaticky při každém přidání nových souborů. Přečtěte si 4.17.1.5 – „Automatické nastavení vlastnosti“ pro další informace.

4.21.6. Hákové Skripty Uzamknutí

Při vytváření nového úložiště se Subversion 1.2 nebo vyšší, jsou v úložišti vytvořeny čtyři šablony háků v adresáři `hooks`. Tyto skripty jsou volány před a po získání a před a po uvolnění zámku.

Je dobrý nápad instalovat na serveru hákový skript `po-zamknutí` a `po-odemknutí`, který pošle e-mail se jménem souboru, který byl uzamčen. S takovým skriptem na místě, můžou být všichni Vaši uživatelé oznámeni pokud někdo uzamkne/odemkne soubor. Můžete najít příklad hákového skriptu `hooks/post-lock.tmpl` v adresáři úložiště.

Můžete také použít háky k zakázání rozbití nebo kradení zámků, nebo třeba toto omezit na jmenovaného správce. Nebo možná budete chtít poslat e-mail majiteli, když je jeden z jejich zámků rozbit nebo ukraden.

Prečtěte si [3.3 – „Hook skripty na straně serveru“](#) pro více informací.

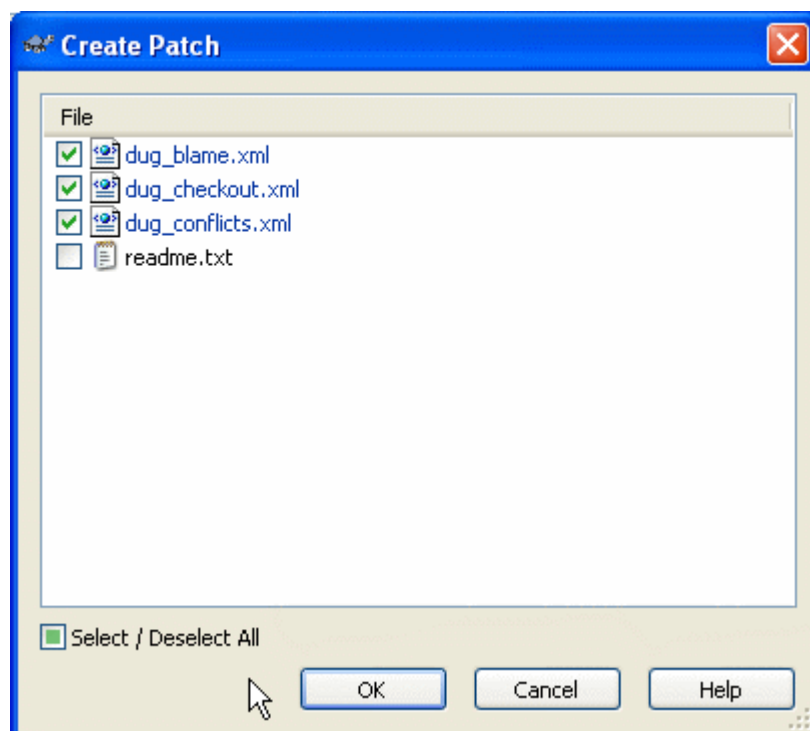
4.22. Vytváření a Používání Záplat

Pro open source projekty (jako je tento) má každý přístup ke čtení z úložiště a každý může k projektu přispět. Tak, jak jsou ty příspěvky kontrolovány? Pokud by kdokoli mohl změny odevzdat, projekt by byl trvale nestabilní a pravděpodobně trvale rozbitý. V této situaci je změna kontrolována posláním *záplatového* souboru vývojovému týmu, kteří mají přístup pro zápis. Mohou záplatu nejdříve prohlédnout a pak buď odeslat do úložiště, nebo odmítnout a vrátit zpět k autorovi.

Záplatové soubory jsou prostě soubory sjednoceného rozdílu popisující rozdíly mezi Vaší pracovní kopií a základní revizí.

4.22.1. Vytvoření Záplatového souboru

Nejdříve je potřeba, abyste změny *otestovali*. Pak namísto použití TortoiseSVN → Odevzdat... na nadřazený adresář, vyberte TortoiseSVN → Vytvořit Záplatu...



Obrázek 4.43. Dialogové okno Vytvořit Záplatu

nyní můžete vybrat soubory, které chcete do záplaty zahrnout, stejně jako byste udělali s úplným odevzdáním. Toto vyprodukuje jeden soubor, který obsahuje přehled všech změn, které jste provedli ve vybraných souborech od poslední aktualizace z úložiště.

Sloupce v tomto dialogovém okně lze upravit stejným způsobem jako sloupce v dialogovém okně Zkontrolovat změny. Přečtěte si 4.7.3 – „Místní a Vzdálený Stav“ pro další podrobnosti.

Můžete vytvářet oddělené záplaty obsahující změny v různé sadě souborů. Samozřejmě, pokud vytvoříte záplatový soubor, provede některé další změny ve *stejných* souborech a pak vytvoříte další záplatu, bude druhý záplatový soubor zahrnovat *obě* sady změn.

Jednoduše uložte soubor pomocí jména dle Vašeho výběru. Záplatové soubory mohou mít jakoukoli příponu, která se Vám líbí, ale podle úmluvy, by měl mít příponu `.patch` nebo `.diff`. Nyní jste připraveni odeslat Váš záplatový soubor.

Můžete také uložit záplatu do schránky místo do souboru. Možná to budete chtít tak udělat, abyste ji mohli vložit do e-mailu pro ostatní ke kontrole. Nebo pokud máte dvě pracovní kopie na jednom počítači a chcete přenést změny z jedné do druhé, záplata do schránky, je vhodný způsob, jak toto udělat.

4.22.2. Použití Záplatového souboru

Záplatové soubory jsou aplikovány na Vaší pracovní kopii. Toto mělo být provedeno ze stejné úrovně adresáře, která byla použita k vytvoření záplaty. Pokud si nejste jisti, která to je, stačí se podívat na první řádek záplatového souboru. Například, pokud první soubor, na kterém se pracovalo byl `dok/zdroj/česky/kapitolal.xml` a první řádek v záplatovém souboru je `Index: česky/kapitolal.xml` pak je třeba záplatu použít v adresáři `dok/zdroj/`. Nicméně pokud jste ve správné pracovní kopii, pokud vyberete špatnou úroveň adresáře, TortoiseSVN si toho všimne a navrhne správnou úroveň.

Aby bylo možno použít záplatový soubor na Vaší pracovní kopii, musíte mít do úložiště alespoň přístup pro čtení. Důvodem je, že program sloučení musí odkázat změny zpět na revizi, proti níž byly vyrobeny dálkovým vývojářem.

From the context menu for that folder, click on TortoiseSVN → Apply Patch... This will bring up a file open dialog allowing you to select the patch file to apply. By default only `.patch` or `.diff` files are shown, but you can opt for „All files“. If you previously saved a patch to the clipboard, you can use Open from clipboard... in the file open dialog.

Případně, pokud záplatový soubor má příponu `.patch` nebo `.diff`, můžete na něj přímo kliknout pravým tlačítkem myši a zvolit TortoiseSVN → Použít Záplatu.... V tomto případě budete vyzváni k zadání umístění pracovní kopie.

Tyto dvě metody nabízejí různé způsoby, jak dělat totéž. S první metodou zvolíte pracovní kopii a vyhledáte záplatový soubor. S druhým vyberete záplatový soubor a přejdete do pracovní kopie.

Jakmile jste si vybrali záplatový soubor a umístění pracovní kopie, je spuštěn TortoiseMerge pro sloučení změn ze záplatového souboru do Vaší pracovní kopie. Malé okno obsahuje seznam souborů, které byly změněny. Postupně dvakrát klikněte na každý z nich, zkontrolujte změny a sloučené soubory uložte.

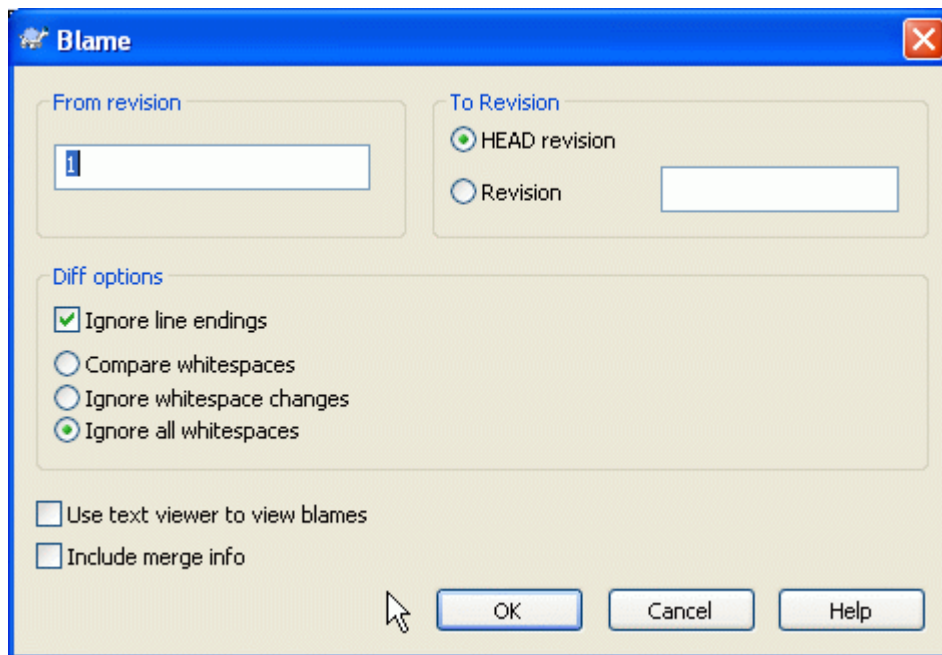
Záplata vzdáleného vývojáře je nyní použita na Vaší pracovní kopii, takže musíte odevzdat změny, aby všichni ostatní měli k nim z úložiště přístup.

4.23. Kdo Změnil Který Řádek?

Někdy je třeba vědět nejen to, které řádky se změnilly, ale také kdo přesně změnil určité řádky v souboru. V tu dobu příkaz TortoiseSVN → Obvinít..., někdy také označován jako *Okomentovat* přijde vhod.

Tento příkaz vypíše, pro každý řádek v souboru autora a revizi kdy byl řádek změněn.

4.23.1. Obviňovat ze souborů



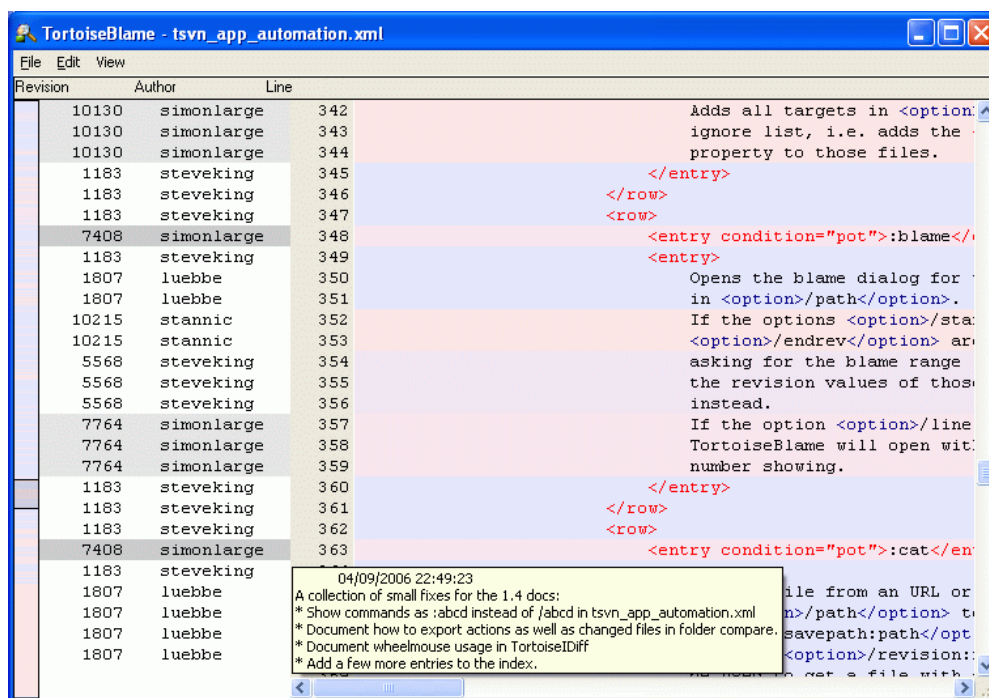
Obrázek 4.44. Dialogové okno Okomentování / Obvinění

Pokud nemáte zájem o změny z předchozích revizí, můžete nastavit revizi, z které by mělo obvinění začít. Nastavte toto na 1, chcete-li vinu za *každou* revizi.

By default the blame file is viewed using *TortoiseBlame*, which highlights the different revisions to make it easier to read. If you wish to print or edit the blame file, select **Use Text viewer to view blames**

Můžete určit způsob, jakým jsou změny mezery a konec řádků řešeny. Tyto možnosti jsou popsány v [4.10.2 – „Možnosti konců řádků a mezer“](#). Výchozí chování je zacházet se všemi rozdíly v mezerách a koncích řádků jako skutečné změny, ale chcete-li ignorovat změnu odsazení a najít původního autora, můžete si zde vybrat vhodnou volbu.

Po stisknutí OK TortoiseSVN začne schraňovat data pro vytvoření soubor obvinění. Pozor prosím: Toto může trvat několik minut, v závislosti na tom, jak moc se soubor změnil a samozřejmě vaše síťové připojení k úložišti. Jakmile je proces obvinění dokončen výsledek je zapsán do dočasného souboru a můžete ho prohlížet.



Obrázek 4.45. TortoiseBlame

TortoiseBlame, který je součástí TortoiseSVN, dělá soubor obvinění čitelnější. Když podržíte myš nad řádkem ve sloupci informací o obvinění, všechny řádky se stejnou revizí jsou zobrazeny s tmavším pozadím. Řádky z jiných revizí, které byly změněny od stejného autora, jsou zobrazeny se světlým pozadím. Zbarvení nemusí být tak jasné, pokud máte obrazovku nastavenou na režim zobrazení 256 barev.

Pokud kliknete levým tlačítkem na řádek, jsou všechny řádky se stejnou revizí zvýrazněny a řádky z jiných revizí stejného autora jsou zvýrazněny světlejší barvou. Toto zvýraznění je pevné, což vám umožní pohybovat myší bez ztráty zvýraznění. Klikněte opět na tuto změnu pro vypnutí zvýraznění.

Komentáře revizí (zprávy záznamu) jsou uvedeny v rámečku nápovědy kdykoliv myší najedete nad sloupec informací obvinění. Pokud chcete zkopírovat zprávu záznamu za revizi, použijte kontextové menu, které se objeví, když klepnete pravým tlačítkem na sloupec informací obvinění.

Můžete vyhledávat ve zprávě obvinění pomocí **Upravit → Najít....** To Vám umožní vyhledat čísla revizí, autorů a obsah souboru. Zprávy záznamů nejsou zahrnuty do hledání - pro jejich vyhledávání byste měli použít Dialogové okno záznamů.

Můžete také přeskočit na konkrétní číslo řádku pomocí **Upravit → Jdi na řádek....**

Pokud je myš nad sloupci informací o obvinění, je k dispozici kontextové, které pomáhá s porovnáváním revizí a zkoumáním historie, použitím čísla revize řádku pod kurzorem myši jako odkaz. **Kontextové menu → Obvinit předchozí revizi** vytvoří zprávu obvinění pro stejný soubor, ale pomocí předchozí revize jako horní hranice. To Vám dává zprávu obvinění pro stav souboru těsně před řádkem, na který se díváte, byl naposledy změněn. **Kontextové menu → Zobrazit změny** spustí prohlížeč rozdílu. Ukáže Vám, co se v odkazované revizi změnilo. **Kontextové menu → Zobrazit záznam** zobrazí dialogové okno záznamu revize začínající zmíněnou revizí.

Pokud potřebujete lepší vizuální indikátor kde jsou nejstarší a nejnovější změny, zvolte **Zobrazit → Zabarvit stáří řádků**. To použije barevný gradient pro zobrazení novějších linky červeně a starší řádky modře. Výchozí zbarvení je poměrně světlé, ale můžete jej změnit pomocí nastavení TortoiseBlame.

Používáte-li Sledování Sloučení, kde se řádky změnilo v důsledku sloučení z jiné cesty, TortoiseBlame zobrazí revize a autora poslední změny v původním souboru, spíše než revizi, kde ke sloučení došlo. Tyto

řádky jsou označeny zobrazením revize a autora v kurzívě. Pokud nechcete, aby se sloučené řádky takto zobrazovaly, zrušte zaškrtnutí políčka **Zahrnout informace o sloučení**.

Pokud chcete vidět cesty podílející se na sloučení, zvolte **Zobrazit → Sloučit cesty**.

Do Nastavení TortoiseBlame lze vstoupit pomocí **TortoiseSVN → Nastavení...** v kartě TortoiseBlame. Navštivte [4.30.9 – „Nastavení TortoiseBlame“](#).

4.23.2. Obvinit Rozdíly

Jedním z omezení zprávy Obvinění je, že pouze ukazuje soubor takový, jaký byl v určité revizi a ukazuje poslední osobu, která změnila každý řádek. Někdy chcete vědět, jaká byla provedena změna, stejně jako kdo ji provedl. Co tady potřebujete je kombinace porovnání a zprávy obvinění.

Dialogové okno záznamu revizí obsahuje několik voleb, které Vám toto umožní provést.

Obvinit Revize

V horním panelu vyberte 2 revize, poté zvolte **Kontextové menu → Obvinit revize**. Tím se stáhnou data obvinění za 2 revize, pak použijte prohlížeč rozdílů a porovnejte oba soubory obvinění.

Obvinit Změny

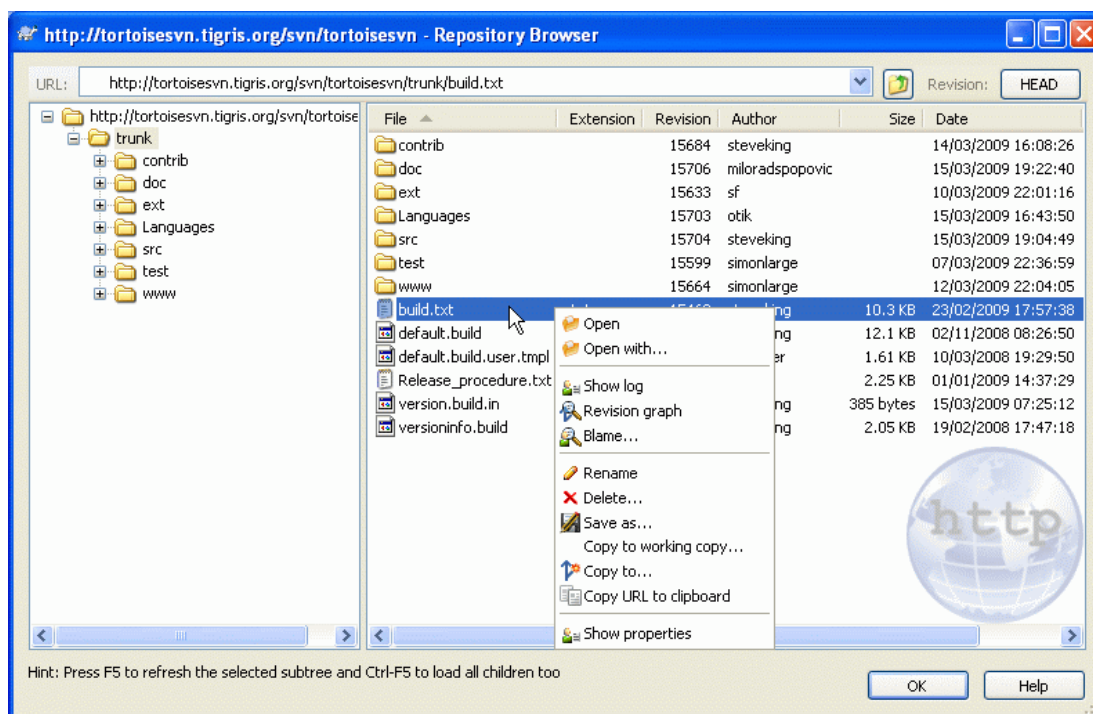
Vyberte jednu revizi v horním panelu, pak vyberte jeden soubor ve spodním panelu a zvolte **Kontextové menu → Obvinit změny**. Tím se stáhnou data obvinění pro vybranou a předchozí revizi, pak použijte prohlížeč rozdílů a porovnejte oba soubory obvinění.

Porovnat a Obvinit s Pracovním Originálem

Zobrazte záznam pro jeden soubor a v horním panelu vyberte jednu revizi, pak zvolte **Kontextové menu → Porovnat a obvinit s pracovním originálem**. Tím se stáhnou data vybrané revize a souboru v pracovním originálu, pak použijte prohlížeč rozdílů a porovnejte oba soubory obvinění.

4.24. Průzkumník Úložiště

Někdy je třeba pracovat přímo v úložišti, aniž byste měli pracovní kopii. To je úkol pro **Průzkumníka úložiště**. Stejně jako průzkumník a překryvy ikon Vám umožní zobrazit Vaši pracovní kopii, takže Průzkumník Úložiště Vám umožňuje zobrazit strukturu a stav úložiště.



Obrázek 4.46. Průzkumník Úložiště

S Průzkumníkem úložiště můžete provést příkazy jako kopírovat, přesunout, přejmenovat, ... přímo v úložišti.

Průzkumník úložiště vypadá velmi podobně jako průzkumník Windows, kromě toho, že ukazuje obsah úložiště při konkrétní revizi, nikoli soubory ve Vašem počítači. V levém panelu vidíte adresářovou strukturu a v pravém panelu obsah vybraného adresáře. V horní části Okna Průzkumníka Úložiště, můžete zadat URL úložiště a revize, které chcete procházet.

Tak jako průzkumník Windows, můžete kliknout na záhlaví sloupce v pravém panelu, pokud chcete nastavit jejich pořadí. A stejně jako v průzkumníkovi Windows je kontextové menu k dispozici v obou panelech.

Kontextové menu souboru Vám umožňuje:

- Otevře zvolený soubor a to buď ve výchozím prohlížeči pro tento typ souboru, nebo s programem, který vyberete.
- Uložit kopii souboru bez verze na Váš pevný disk.
- Zobrazit záznam revize pro daný soubor, nebo zobrazit graf všech revizí, abyste mohli vidět, odkud soubor pochází.
- Obvinít soubor, abyste zjistili, kdo změnil který řádek a kdy.
- Smazat nebo přejmenovat soubor.
- Vytvořit kopii souboru a to buď do jiné části úložiště, nebo do pracovní kopie s kořenem ve stejném úložišti.
- Zobrazit/Upravit vlastnosti souboru.

Kontextové menu pro adresář Vám umožňuje:

- Zobrazit záznam revize pro tento adresář nebo zobrazit graf všech revizí, takže můžete vidět, odkud adresář pochází.
- Export adresáře do místní kopie bez verze na Vašem pevném disku.
- Získání adresáře pro vytvoření místní pracovní kopie na Vašem pevném disku.
- Vytvořit nový adresář v úložišti.
- Přidat soubory nebo adresáře přímo do úložiště.
- Smazat nebo přejmenovat adresář.
- Make a copy of the folder, either to a different part of the repository, or to a working copy rooted in the same repository.
- Zobrazit/Upravit vlastnosti adresáře.
- Označte adresář pro srovnání. Označený adresář je zobrazen tučně.
- Porovnejte adresář s dříve označeným a to buď jako sjednocený rozdíl, nebo jako seznam změněných souborů, které pak mohou být vizuálně porovnány s výchozím nástrojem rozdílů. To může být užitečné zejména při porovnání dvou značek, nebo kmene a větve, abyste viděli, co se změnilo.

Zvolíte-li dva adresáře v pravém panelu, můžete zobrazit rozdíly buď jako sjednocený rozdíl, nebo jako seznam souborů, které lze vizuálně porovnat s výchozím nástrojem rozdílů.

Zvolíte-li v pravém podokně více adresářů, můžete získat všechny najednou do společného nadřazeného adresáře.

Pokud zvolíte 2 značky, které jsou zkopírovány ze stejného kořene (většinou /kmen/), můžete použít **Kontextové menu** → **Zobrazit záznam...** pro zobrazení seznamu revizí mezi dvěma body značek.

Můžete použít **F5** pro obnovení zobrazení jako normálně. Tím se obnoví vše, co je momentálně zobrazeno. Pokud chcete přednačíst nebo aktualizovat informace pro uzly, které ještě nebyly otevřeny použijte **Ctrl-F5**. Poté, k rozšíření libovolného uzlu dojde okamžitě bez zpoždění v síti, zatímco jsou informace načítány.

Průzkumníka úložiště můžete také použít pro táhnutí a puštění operace. Pokud přetáhnete adresář z průzkumníka Windows do průzkumníka úložiště, bude do úložiště importován. Nezapomeňte, že pokud přetáhnete více položek, budou importovány v samostatných odevzdáních.

Pokud chcete přesunout položku do úložiště, stačí ji přetáhnout levým tlačítkem myši na nové umístění. Pokud raději chcete vytvořit kopii než položkou pohybovat, tak použijte **Ctrl**-táhnutí levým tlačítkem myši místo toho. Při kopírování kurzor má znamínko „plus“ na něm, stejně jako v Průzkumníkovi Windows.

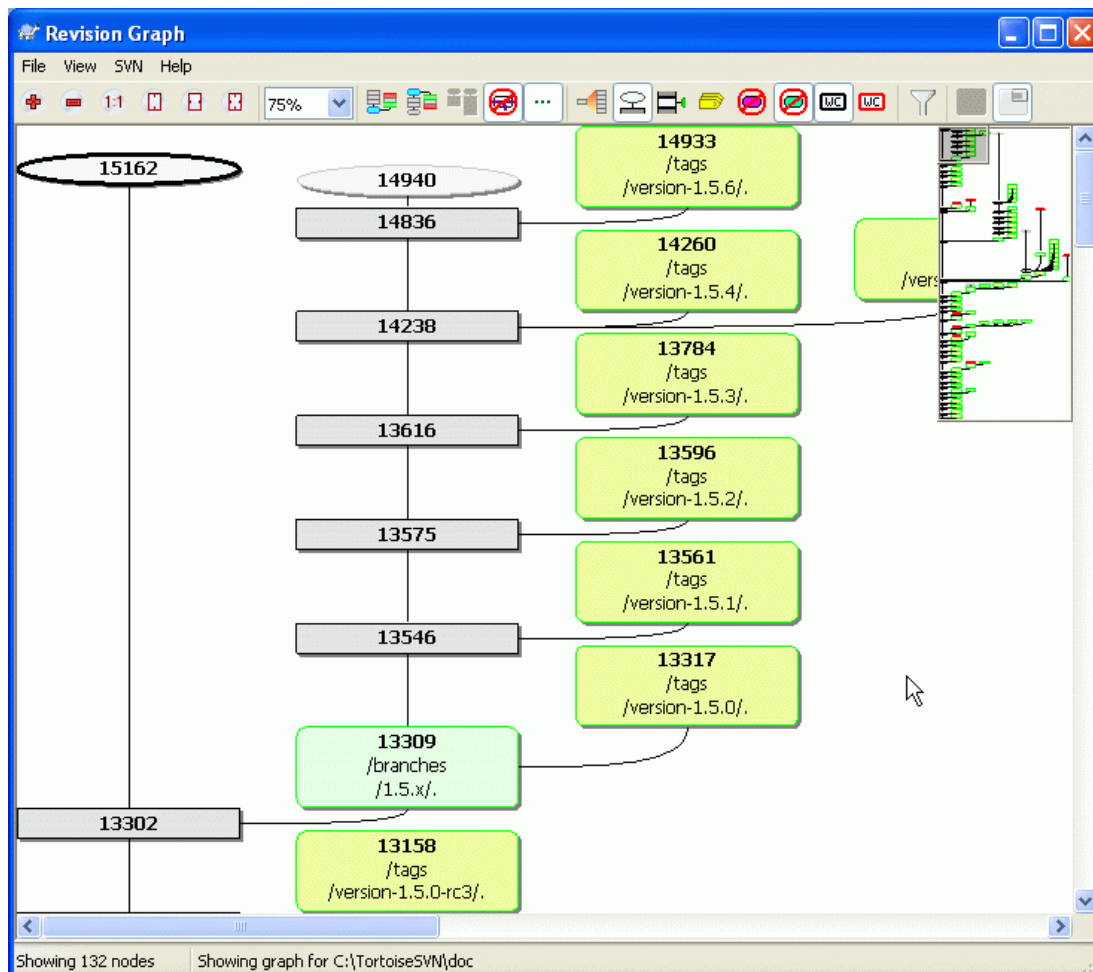
Pokud chcete kopírovat/přesunout soubor nebo adresář na jiné místo a také dát mu nové jméno najednou, můžete položku táhnout pravým tlačítkem myši nebo **Ctrl**-táhnout pravým tlačítkem myši místo použití táhnutí levým tlačítkem. V takovém případě je zobrazeno dialogové okno přejmenování, kde můžete zadat nový název souboru nebo adresáře.

Pokaždé, když provedete změny v úložišti pomocí jedné z těchto metod, zobrazí se Vám dialogové okno zadání zprávy záznamu. Pokud jste přetáhli něco omylem, je to také Vaše šance jak činnost zrušit.

Někdy, když se pokusíte otevřít cestu, obdržíte chybovou zprávu namísto detailů položky. K tomu může dojít, pokud jste zadali neplatnou adresu URL, nebo pokud nemáte oprávnění k přístupu, nebo pokud je na serveru nějaký jiný problém. Pokud potřebujete zkopírovat tuto zprávu a zahrnout ji do e-mailu, stačí

na ni kliknout pravým tlačítkem myši a zvolit Kontextové menu → Kopírovat chybovou zprávu do schránky, nebo jednoduše použít **Ctrl+C**.

4.25. Grafy Revizí



Obrázek 4.47. Graf Revize

Někdy je třeba vědět, kdy byly větve a značky vzaty z kmene a ideální způsob, jak zobrazit tento druh informací je ve formě grafu nebo stromové struktury. Pak potřebujete TortoiseSVN → Graf Revizí...

Tento příkaz analyzuje historii revizí a pokusí se vytvořit strom ukazující body, v nichž byly kopie odebrány a kdy byly větve/značky smazány.



Důležité

Za účelem vytvoření grafu, musí TortoiseSVN načíst všechny zprávy záznamu z kořene úložiště. Netřeba dodávat, že toto může trvat několik minut i s úložištěm o několik tisíc revizí, v závislosti na rychlosti serveru, síť atd. Pokud se pokusíte o něco takového v projektu *Apache*, který má v současnosti přes 500.000 revizí mohli byste nějakou dobu čekat.

Dobrou zprávou je, že pokud používáte uchovávání záznamu, musíte toto čekání protřpět pouze jednou. Poté, jsou data záznamů vedena na místní úrovni. Uchování záznamů se zapíná v nastavení TortoiseSVN.

4.25.1. Uzly Grafu Revizí

Každý uzel grafu revizí reprezentuje revizi v úložišti, kde se něco ve stromu změnilo, na který se díváte. Různé typy uzlů mohou být rozlišeny podle tvaru a barvy. Tvary jsou pevné, ale barvy lze nastavit pomocí TortoiseSVN → Nastavení

Přidané nebo zkopírované položky

Items which have been added, or created by copying another file/folder are shown using a rounded rectangle. The default colour is green. Tags and trunks are treated as a special case and use a different shade, depending on the TortoiseSVN → Settings

Smazané položky

Deleted items eg. a branch which is no longer required, are shown using an octagon (rectangle with corners cut off). The default colour is red.

Přejmenované položky

Přejmenované položky jsou také zobrazeny pomocí osmiúhelníku, ale výchozí barva je modrá.

Nejvyšší revize větve

Grafu je obvykle omezen na zobrazení bodů větví, ale často je také užitečné mít možnost vidět příslušné HEAD revize pro každou větev. Pokud zvolíte **Zobrazit HEAD revize**, bude každý uzel HEAD revize zobrazen jako elipsa. Nezapomeňte, že HEAD tady odkazuje na poslední revizi odevzdanou na této cestě, ne na HEAD revizi úložiště.

Revizi pracovní kopie

Pokud zobrazíte graf revizí z pracovní kopie, můžete se rozhodnout pro zobrazení základní revize v grafu pomocí **Zobrazit revize pracovní kopie**, která označuje základní uzel s tučným obrysem.

Změněná pracovní kopie

Pokud jste vyvolali graf revizí z pracovní kopie, můžete se rozhodnout pro zobrazení dalších uzlů reprezentující Vaše upravené pracovní kopie pomocí **Zobrazit změny pracovní kopie**. Ve výchozím nastavení se jedná o eliptický uzel s tučným červeným obrysem.

Normální položka

Všechny ostatní položky jsou zobrazeny pomocí prostého obdélníku.

Nezapomeňte, že ve výchozím nastavení graf zobrazuje pouze body, kde byly položky přidány, kopírovány nebo smazány. Ukázat všechny revize projektu vytvoří velký graf v netriviálních případech. Pokud opravdu chcete vidět *všechny* revize, kdy byly provedeny změny, existuje pro toto volba v menu **Zobrazit** a na panelu nástrojů.

Výchozí zobrazení (seskupení vypnuto) umísťuje uzly tak, že jejich svislá poloha je v přísném pořadí revizí, takže budete mít vizuální kritérium pro pořadí, ve kterém byly věci provedeny. Tam, kde jsou dva uzly ve stejném sloupci je pořadí velmi zřejmé. Když jsou dva uzly v sousedních sloupcích posun je mnohem menší, protože není nutné, bránit uzlům v překrývání a v důsledku toho, je pořadí trochu méně nápadné. Takové optimalizace jsou nutné k udržení složitých grafů v přiměřené velikosti. Vezměte prosím na vědomí, že toto uspořádání používá *okrajů* uzlu na *starší* straně jako odkaz, tj. spodní okraj uzlu, když je graf zobrazen s nejstarším uzlem vespod. Odkazující okraj je významný, protože tvary uzlů nejsou všechny ve stejné výšce.

4.25.2. Změna zobrazení

Vzhledem k tomu, že graf revizí je často docela složitý, existuje řada funkcí, které lze použít pro přizpůsobení zobrazení tak, jak chcete. Jsou k dispozici v menu **Zobrazit** menu a na panelu nástrojů.

Seskupit větve

Výchozí chování (seskupování vypnuto) má všechny řady řazené striktně revizí. Výsledkem je, že dlouho žijící větve s málo odevzdáními zabírají celý sloupec jen pro pár změn a graf je pak velmi široký.

Tento režim seskupí změny podle větví, tak, že neexistuje žádné globální řazení podle revizí: Po sobě jdoucí revize ve větvi budou uvedeny v (často) po sobě jdoucích řádcích. Podvětvě jsou ale uspořádány tak, aby pozdější větve budou zobrazeny ve stejném sloupci nad staršími větvemi, aby graf byl štíhlý. Výsledkem je, že daný řádek může obsahovat změny z různých revizí.

Nejstarší nahoře

Normálně graf ukazuje nejstarší revize dole a strom roste nahoru. Použijte-li tuto volbu graf místo toho poroste shora dolů.

Zarovnat stromy nahoru

Pokud je graf rozdělen do několika menších stromů, mohou se buď objevit v přirozeném seřazení podle revize, nebo zarovnaný v dolní části okna, v závislosti na tom, zda používáte možnost **Seskupit větve**. Použijte místo toho tuto volbu pro růst všech stromů ze shora dolů.

Snížit zkřížení linek

Pokud rozložení grafu vytvoří mnoho křížících se linií, použijte tuto volbu pro vyčištění. To může způsobit, že rozložení sloupců se objeví v méně logických místech, například v diagonální linii, spíše než ve sloupci a graf může vyžadovat větší prostor k zobrazení.

Rozdílné názvy cest

Dlouhé názvy cest můžou zabrat hodně prostoru a způsobit, že rámečky uzlů jsou velmi velké. Pomocí této možnosti zobrazíte pouze změněné části cest, nahradí společnou část tečkami. Např. pokud vytvoříte větev `/Větve/1.2.x/dok/html z /kmen/dok/ html` větev by mohla být zobrazena v zhuštěné formě jako `/Větve/1.2.x / . .` protože v posledních dvou úrovních, `dok` a `html` se nezměnila.

Zobrazit všechny revize

Toto provede přesně to, co čekáte, a zobrazí každou revizi, kde se něco (ve stromě, kterého graf zobrazujete). Při dlouhé historii toto může vytvořit opravdu obrovský graf.

Zobrazit HEAD revize

Tím je zajištěno, že poslední revize v každé větvi je vždy v grafu zobrazena.

Přesné zdroje kopie

Když je vytvořena větev/značka, výchozí chování je ukázat větev jako převzatou z posledního uzlu, kde byla provedena změna. Přesněji řečeno toto je nepřesné, protože větve jsou často vyrobeny ze současné vrchní revize spíše než z konkrétní revize. Takže je možné ukázat více správnou (ale méně užitečnou) revizi, která byla použita k vytvoření kopie. Všimněte si, že tato revize může být mladší, než revize HEAD zdrojové větve.

Složít značky

Když projekt má mnoho značek, zobrazovat každou značku jako samostatný uzel v grafu zabírá hodně prostoru a zahaluje zajímavější strukturu vývojové větve. Zároveň budete chtít mít snadný přístup k obsahu značky, abyste mohli porovnávat revize. Tato volba skryje uzly značek a zobrazí na jejich místě popis pro uzel, z kterého byly zkopírovány. Ikona značky na pravé straně zdrojového uzlu naznačuje, že byly vyrobeny značky.

Skrýt smazané cesty

Skryje cesty, které již nesou přítomny v revizi HEAD v úložiště, např. smazané větve.

Skrýt nezměněné větve

Skryje větve, kde nebyly žádné změny do souboru, nebo podadresáře nebyly odevzdány. To nemusí nutně znamenat, že větev nebyla použita, jen, že nebyly provedeny žádné změny v *této* její části.

Zobrazit revizi pracovní kopie

Označí revizi v grafu, která odpovídá aktualizaci revize položky, pro kterou byl graf vytvořen. Pokud jste právě aktualizovali, bude to HEAD, ale pokud ostatní odevzdali změny od poslední aktualizace Vaší pracovní kopie může být o několik revizí níže. Uzel je označen tím, že mu tučný obrys.

Zobrazit změny pracovní kopie

Pokud Vaše pracovní kopie obsahuje místní změny, tato možnost je vykreslí jako samostatné eliptické uzly, spojené zpátky k uzlu, kdy Vaše pracovní kopie byla naposledy aktualizována. Výchozí barva

obrysu je červená. Možná budete muset aktualizovat graf použitím **F5** pro zachycení posledních změn.

Filtr

Někdy graf revizí obsahuje více revizí, než chcete zobrazit. Tato volba otevře dialogové okno, které umožňuje omezit rozsah zobrazovaných revizí a schovat určité cesty podle názvu.

Pruhy stromu

V případě, že graf obsahuje několik stromů, je někdy užitečné střídat barvy na pozadí, aby bylo snadnější stromy rozlišit.

Zobrazit přehled

Zobrazí malý obrázek celého grafu, s aktuálně zobrazeným oknem jako obdélník, který můžete táhnout. To Vám umožní snadnější navigaci v grafu. Nezapomeňte, že pro velmi velké grafy se přehled může stát zbytečný kvůli extrémnímu zmenšení a proto v takových případech nebude zobrazen.

4.25.3. Použití Grafu

Aby byla navigace ve velkém grafu přehlednější, použijte okno přehledu. To ukazuje celý graf v malém okně, s aktuálně zobrazenou částí zvýrazněnou. Můžete táhnout zvýrazněnou část ke změně zobrazené oblasti.

Datum revize, autor a komentáře jsou uvedeny v rámečku nápovědy kdykoliv myší najedete nad rámečkem revize.

Zvolíte-li dvě revize (Použijte **Ctrl**-kliknutí levým tlačítkem myši), můžete použít kontextovou nabídku k zobrazení rozdílů mezi těmito revizemi. Můžete si vybrat k zobrazení rozdíly v bodech vytvoření větví, ale většinou budete chtít ukázat rozdíly v konečných bodech větve, tj. v revizi HEAD.

Můžete zobrazit rozdíly jako sjednocený rozdíl, který zobrazuje veškeré rozdíly v jediném souboru s minimálním kontextem. Pokud jste zvolili **Kontextové menu** → **Porovnat Revize** bude Vám předložen seznam změněných souborů. Dvakrát klikněte na název souboru pro získání obou revizí souboru a porovnejte je s využitím vizuální nástroje rozdílů.

Pokud kliknete pravým tlačítkem myši na revizi, můžete použít **Kontextové menu** → **Zobrazit záznam** pro zobrazení historie.

Můžete také sloučit změny ve vybraných revizích do jiné pracovní kopie. Dialogové okno výběru adresáře Vám umožňuje zvolit pracovní kopii, do které sloučit, ale poté už se nezobrazí žádné dialogové okno pro potvrzení, ani příležitost sloučení vyzkoušet. Je dobrý nápad sloučit do nezměněné pracovní kopie, takže můžete vrátit změny, pokud to nevyjde! Toto je užitečná funkce, pokud chcete sloučit vybrané revize z jedné větve do druhé.



Naučte se číst Graf Revizí

Noví uživatelé mohou být překvapeni tím, že graf revizí ukazuje něco, co neodpovídá mentálnímu model uživatele. Například, pokud revize mění více kopií nebo větví souboru nebo adresáře, pak bude pro tuto jedinou revizi zobrazeno více uzlů. Je dobrým zvykem začít s možnostmi nalevo na panelu nástrojů a přizpůsobit graf krok-za-krokem, dokud se nepřiblíží k Vašemu mentálnímu model.

Všechny možnosti filtrování se snaží, aby ztratili co nejméně informací jak je to možné. Například, to může způsobit, že některé uzly změní barvu. Kdykoliv je výsledek nečekaný, vraťte zpět poslední operaci filtrování a pokuste se si uvědomit, co je zvláštní té konkrétní revizi nebo větvi. Ve většině případů by byl původně očekávaný výsledek filtrové operace buď nepřesný, nebo zavádějící.

4.25.4. Obnovení zobrazení

Pokud chcete znovu zkontrolovat server pro novější informace, můžete jednoduše aktualizovat zobrazení pomocí **F5**. Pokud používáte úschovu záznamu (ve výchozím nastavení zapnuta), toto zkontroluje úložiště pro novější odevzdání a načte pouze ty nové. Pokud je úschova záznamu v režimu offline, pokusí se také vrátit online.

Pokud používáte úschovu záznamu a myslíte si, že obsah zprávy, nebo autor se mohl změnit, měli byste použít dialogové okno záznamu k obnovení zpráv, které potřebujete. Vzhledem k tomu, že graf revizí pracuje z kořene úložiště, museli bychom zrušit celou úschovu záznamu a znovunaplnění by mohlo trvat *velmi* dlouhou dobu.

4.25.5. Čištění stromů

Ve velkém stromu může být obtížné se orientovat a někdy budete chtít skrýt části, nebo rozbít do lesa s menšími stromy. Pokud najedete ukazatelem myši nad místo, kde čára uzle vstupuje, nebo vystupuje z uzlu, uvidíte jednu nebo více výstupných tlačítek, která Vám toto umožní udělat.



Klikněte na tlačítko minus ke sbalení připojeného podstromu.



Klikněte na tlačítko plus pro rozbalení sbaleného stromu. Když byl strom sbalen, toto tlačítko zůstane viditelné pro aby naznačilo skrytý podstrom.



Klikněte na znaménko křížku k oddělení připojeného podstromu a zobrazit ho jako samostatný strom na grafu.

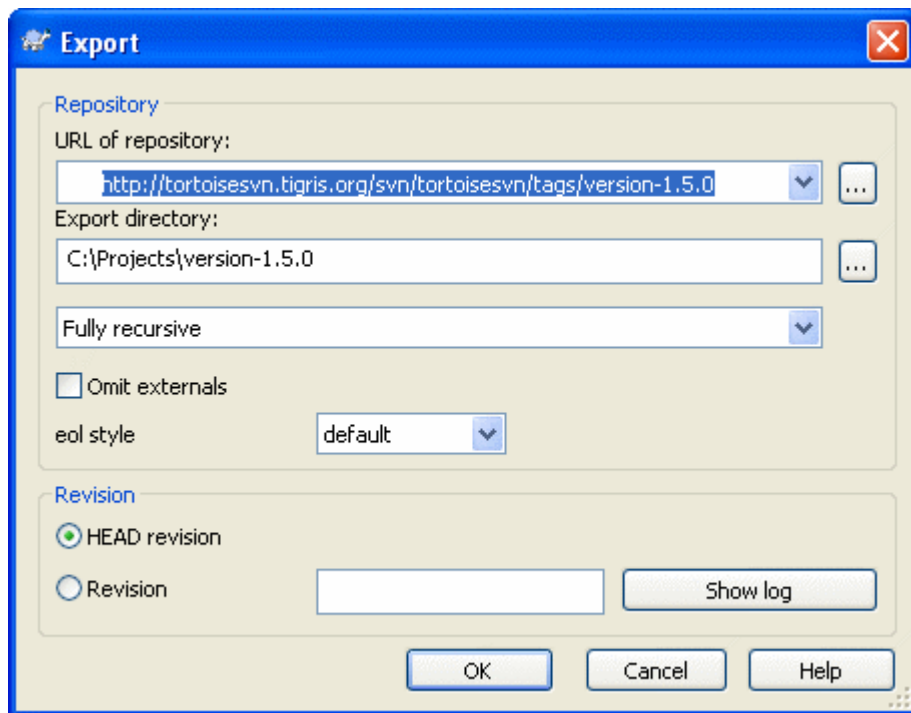


Klikněte na tlačítko kruhu k připojení rozděleného stromu. Když je strom oddělen, toto tlačítko zůstane viditelné, aby naznačilo, že existuje oddělený podstrom.

Klikněte na pozadí grafu pro hlavní kontextové menu, které nabízí možnosti **Rozbalit vše** a **Spojit vše**. Pokud žádná větev nebyla sbalena, nebo rozdělena, kontextové menu se nezobrazí.

4.26. Export Pracovní Kopie Subversion

Někdy možná budete chtít kopii Vašeho pracovního stromu, bez všech adresářů `.svn`, např. k vytvoření archivu tar Vašeho zdroje, nebo pro export na webový server. Namísto vytvoření kopie a pak smazání všech adresářů `.svn` ručně, TortoiseSVN nabízí příkaz **TortoiseSVN → Exportovat...** . Export z URL a export z pracovní kopie jsou prováděny trochu jinak.



Obrázek 4.48. Dialogové okno Export z URL

Spustíte-li tento příkaz na adresář bez verze, TortoiseSVN bude předpokládat, že vybraný adresář je cíl a otevře dialogové okno pro zadání URL a revize k exportu. Toto dialogové okno má možnosti pro export adresáře pouze horní úrovně, vynechat externí odkazy a potlačit styly konce řádků pro soubory, které mají nastavenou vlastnost `svn:eol-style`.

Samozřejmě můžete také exportovat přímo z úložiště. Použijte Průzkumníka Úložiště pro navigaci k příslušnému podstromu v úložišti, pak použijte Kontextové menu → Exportovat. Zobrazí se dialogové okno Exportovat z URL, které je popsáno výše.

Spustíte-li tento příkaz z Vaší pracovní kopie budete požádáni o místo pro uložení *čisté* pracovní kopie, bez adresářů `.svn`. Ve výchozím nastavení jsou pouze exportovány soubory s verzí, ale můžete použít zaškrtnutí políčko **Exportovat také soubory bez verze**, aby zahrnul jakékoliv další soubory bez verze, které existují ve Vaší pracovní kopii a ne v úložišti. Externí odkazy používající `svn:externals` můžou být v případě potřeby vynechány.

Dalším způsobem, jak exportovat z pracovní kopie je táhnout pravým tlačítkem myši adresář pracovní kopie do jiného umístění a zvolit Kontextové menu → SVN Exportovat sem nebo Kontextové menu → SVN Exportovat vše sem. Druhá možnost také zahrne i soubory bez verze.

When exporting from a working copy, if the target folder already contains a folder of the same name as the one you are exporting, you will be given the option to overwrite the existing content, or to create a new folder with an automatically generated name, eg. Target (1).



Export jednotlivých souborů

Dialogové okno exportu neumožňuje export jednotlivých souborů, i když Subversion to dokáže.

Chcete-li exportovat jednotlivé soubory s TortoiseSVN, musíte použít průzkumníka úložiště (4.24 – „Průzkumník Úložiště“). Stačí přetáhnout soubor(y), které chcete exportovat z průzkumníka úložiště tam, kde je chcete mít v průzkumníku Windows, nebo pomocí kontextové nabídky průzkumníka úložiště pro export souborů.



Export Změn Stromu

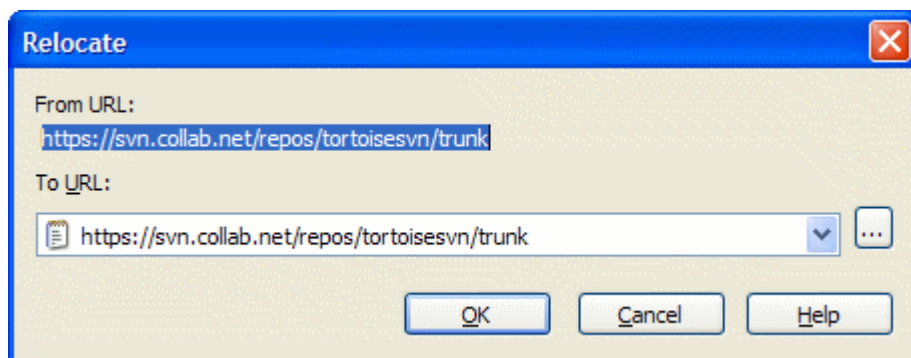
Pokud chcete exportovat kopii struktury stromu projektu obsahující pouze soubory, které se změnilly v konkrétní revizi, nebo mezi jakýmkoli dvěma revizemi, použijte funkci porovnání revizí popsanou v 4.10.3 – „Porovnávání adresářů“.

4.26.1. Smazání pracovní kopie ze správy verzí

Někdy máte pracovní kopii, kterou chcete převést zpět na normální adresář, bez adresářů `.svn`. Co opravdu potřebujete, je příkaz exportovat na místě, který odstraní pouze kontrolní adresáře, raději než vytvářet nový čistý adresářový strom.

Odpověď je překvapivě jednoduchá - exportovat adresář do sebe! TortoiseSVN detekuje tento zvláštní případ a zeptá se, zda chcete vytvořit pracovní kopii bez verze. Pokud odpovíte *ano*, budou kontrolní adresáře odstraněny a budete mít jednoduchý, adresářový strom bez verze.

4.27. Přemístit pracovní kopii



Obrázek 4.49. Dialogové okno přemístění

If your repository has for some reason changed it's location (IP/URL). Maybe you're even stuck and can't commit and you don't want to checkout your working copy again from the new location and to move all your changed data back into the new working copy, TortoiseSVN → Relocate is the command you are looking for. It basically does very little: it scans all `entries` files in the `.svn` folder and changes the URL of the entries to the new value.

Můžete být překvapeni zjištěním, že TortoiseSVN se spojí s úložištěm v rámci této operace. Vše, co dělá, je provedení několika jednoduchých kontrol, aby se ujistil, že nová adresa URL skutečně odkazuje na stejné úložiště jako stávající pracovní kopie.



Varování

Jedná se o velmi řídkou používanou operaci. Příkaz přemístit je použit pouze pokud se URL kořene úložiště změnila. Možné důvody jsou:

- IP adresa serveru se změnila.
- Protokol se změnil (např. `http://` na `https://`).
- Kořenová cesta úložiště v nastavení serveru se změnila.

Jinak řečeno, budete muset přemístit, pokud Vaše pracovní kopie odkazuje na stejné místo ve stejném úložišti, ale úložiště se přesunulo.

Nepoužívá se, když:

- Chcete-li přesunout do jiného úložiště Subversion. V takovém případě byste měli provést čisté získání z nového umístění úložiště.
- Chcete-li překlopit na jinou větev, nebo adresář ve stejném úložišti. Pro toto použijte TortoiseSVN → Překlopit.... Přečtěte si 4.19.2 – „Získat nebo Překlopit...“ pro více informací.

Použijete-li přemístit v obou výše uvedených případech, *naruší to Vaši pracovní kopii* a dostanete mnoho nevysvětlitelných chybových zpráv při aktualizaci, odevzdání, atd. Jakmile k tomuto dojde, opravou je úplně nové získání.

4.28. Integrace se Systémy pro Sledování Chyb/Problémů

V oblastech vývoje softwaru je velmi běžné, aby se změny vztahovaly k určitému problému nebo ID chyby. Uživatelé systémů pro sledování chyb (sledování problémů) mohou přiřadit změny provedené v Subversion ke konkrétnímu ID v jejich systému pro sledování problémů. Většina systémů pro sledování problémů proto poskytne hákový skript před odevzdáním, který rozebere zprávu záznamu pro nalezení ID chyby, s níž je odevzdání spojené. Je to poněkud náchylné k chybám, neboť závisí na uživateli, aby napsal vhodnou zprávu záznamu tak, že hákový skript ji může správně zpracovat.

TortoiseSVN může pomoci uživateli dvěma způsoby:

1. Když uživatel zadá zprávu záznamu, dobře formulovaná věta zahrnující číslo problému spojeným s odevzdáním může být přidáno automaticky. Tím se snižuje riziko, že uživatel zadá číslo problému způsobem, který nástroje sledování chyb nemůžou správně rozebrat.

Nebo TortoiseSVN může zvýraznit část zadané zprávy záznamu, které je pak rozpoznám systémem pro sledování problémů. Tímto způsobem uživatel ví, že zpráva záznamu může být správně rozebrána.

2. Když uživatel prochází zprávy záznamu, TortoiseSVN vytvoří odkaz z každé ID chyby ve zprávě záznamu, který zobrazí prohlížeč s uvedeným problémem.

4.28.1. Přidání Čísel Problémů do zpráv záznamu

V TortoiseSVN můžete integrovat nástroj pro sledování chyb podle vašeho výběru. Pro toto musíte nastavit některé vlastnosti, které začínají `bugtraq:`. Musí být nastaveny na adresáře: (4.17 – „Nastavení projektu“)

Existují dva způsoby, jak integrovat TortoiseSVN se systémem pro sledování problémů. Jedním z nich je založen na jednoduchých řetězcích, druhý je založen na *regulárních výrazech*. Vlastnosti používané oběma přístupy jsou:

`bugtraq:url`

Set this property to the URL of your bug tracking tool. It must be properly URI encoded and it has to contain `%BUGID%`. `%BUGID%` is replaced with the Issue number you entered. This allows TortoiseSVN to display a link in the log dialog, so when you are looking at the revision log you can jump directly to your bug tracking tool. You do not have to provide this property, but then TortoiseSVN shows only the issue number and not the link to it. e.g the TortoiseSVN project is using `http://issues.tortoisesvn.net/?do=details&id=%BUGID%`

Můžete také použít relativní URL místo absolutních. To je užitečné, když Váš systém pro sledování problémů je na stejné doméně/serveru jako zdrojové úložiště. V případě, že se jméno domény změní, nemusíte upravovat vlastnost `bugtraq:url`. Existují dva způsoby, jak zadat relativní URL:

Pokud začíná řetězcem `^/` předpokládá se, že je vzhledem ke kořeni úložiště. Například, `^/../?do=details&id=%BUGID%` bude bráno jako `http://`

`tortoisesvn.net/?do=details&id=%BUGID%` Pokud se Vaše úložiště nachází na `http://tortoisesvn.net/svn/trunk/`.

U URL začínající řetězcem `/` se předpokládá, že je vzhledem ke jménu hostitelského serveru. Například `?do=details&id=%BUGID%` je bráno jako `http://tortoisesvn.net/?do=details&id=%BUGID%` Pokud se Vaše úložiště nachází kdekoli na `http://tortoisesvn.net`.

`bugtraq:warnifnoissue`

Nastavte toto na `true` pokud chcete, aby Vás TortoiseSVN varoval kvůli prázdnému textovému poli čísla problému. Platné hodnoty jsou `true/false`. *Pokud není nastaveno, předpokládá se `false`.*

4.28.1.1. Číslo Problému v Textovém Poli

Jednoduše, TortoiseSVN ukazuje uživateli samostatné vstupní pole, kde může být zadáno ID chyby. Pak je samostatný řádek připojen na začátek/konec zprávy záznamu, kterou uživatel zadal.

`bugtraq:message`

Tato vlastnost aktivuje systém sledování chyb v režimu *Vstupního pole*. Pokud je tato vlastnost nastavena, potom Vás TortoiseSVN vyzve k zadání čísla problému, při odevzdání změn. Je použita na přidání řádku ke konci zprávy záznamu. Musí obsahovat `%BUGID%`, který je při odevzdání nahrazen číslem problému. To zaručuje, že Váš záznam odevzdání obsahuje odkaz na číslo problému, které je vždy v jednotném formátu a může být rozebráno Vaším nástrojem pro sledování chyb pro přiřazení čísla problému s určitým odevzdáním. Jako příklad můžete použít `Issue: %BUGID%`, ale to závisí na Vašem nástroji.

`bugtraq:append`

Tato vlastnost určuje, zda je ID chyby připojeno (`true`) na konec zprávy záznamu nebo je vloženo (`false`) na začátek zprávy. Platné hodnoty jsou `true/false`. *Pokud není nastavena, předpokládá se `true`, aby se neporušili stávající projekty.*

`bugtraq:label`

Tento text je zobrazen TortoiseSVN v dialogovém okně odevzdání k označení rámečku úprav, kde zadáte číslo problému. Pokud není nastavena, je zobrazeno `Bug-ID / Issue-Nr: .` Mějte na paměti, že okno nezmění svoji velikost pro tento štítek, takže udržujte velikost štítku pod 20-25 znaků.

`bugtraq:number`

Pokud je nastavena na `true`, pouze čísla jsou povolena v textovém poli čísla problému. Výjimkou je čárka, takže můžete oddělit čárkou několik čísel. Platné hodnoty jsou `true/false`. *Pokud není nastavena, předpokládá se `true`.*

4.28.1.2. Čísla Problému Pomoci Regulárních Výrazů

V přístupu s *regulárními výrazy*, TortoiseSVN neukazuje samostatné vstupní pole, ale označuje části zprávy záznamu, které uživatel zadá, které je pak rozpoznána systémem pro sledování problémů. Toto se provádí, zatímco uživatel píše zprávu záznamu. To také znamená, že ID chyby může být kdekoli uvnitř zprávy záznamu! Tato metoda je mnohem flexibilnější a používá se v rámci projektu TortoiseSVN.

`bugtraq:logregex`

Tato vlastnost aktivuje systém pro sledování chyb v režimu *Regulárních výrazů*. Obsahuje buď jeden regulární výraz, nebo dva regulární výrazy oddělené novým řádkem.

If two expressions are set, then the first expression is used as a pre-filter to find expressions which contain bug IDs. The second expression then extracts the bare bug IDs from the result of the first regex. This allows you to use a list of bug IDs and natural language expressions if you wish. e.g. you might fix several bugs and include a string something like this: „This change resolves issues #23, #24 and #25“

If you want to catch bug IDs as used in the expression above inside a log message, you could use the following regex strings, which are the ones used by the TortoiseSVN project: `[Ii]ssues?:? (\s*(, |and)?\s*#\d+)+ and (\d+)`

První výraz vybere „problémy #23, #24 a #25“ z okolní zprávy záznamu. Druhý vytáhne prostá desítková čísla z výstupu prvního výrazu, takže vrátí „23“, „24“ a „25“ pro použití jako ID chyby.

Rozebereme si trochu první regulární výraz, musí začínat se slovem „problem“, případně velkými písmeny. Může být následován „y“ (více než jeden problém) a i dvojtečkou. Toto je následované jednou nebo více skupinami, z nichž každá má nulovou nebo více odsazenou mezeru, dobrovolnou čárku nebo volitelně „a“ a další volitelné místo. Nakonec je povinné „#“ a povinné číslo z desítkové soustavy.

Pokud je nastaven pouze jeden výraz, pak se musí prosté ID chyby shodovat ve skupinách řetězce regulárního výrazu. Příklad: `[Ip]roblem(?:s)?#?(\\d+)` Tato metoda je vyžadována několika systémy pro sledování problémů, např. Trac, ale je těžší sestavit regulární výraz. Doporučujeme tuto metodu používat pouze pokud dokumentace Vašeho systému pro sledování problémů neřekne.

Pokud nejste obeznámeni s regulárními výrazy, podívejte se na úvod v http://cs.wikipedia.org/wiki/Regulární_výraz, a on-line dokumentace s návody na <http://www.regularnivyrazy.info> [<http://www.regularnivyrazy.info/>].

Pokud obě vlastnosti `bugtraq:message` a `bugtraq:logregex` jsou nastaveny, `logregex` má přednost.



Tip

I v případě, že nemáte systém pro sledování problémů s háčkovým skriptem před odevzdáním zpracovávajícího zprávy záznamu, stále můžete toto použít pro přetvoření problémů uvedených ve zprávách záznamu do odkazů!

A i když odkazy nepotřebujete, čísla problémů se zobrazují jako samostatné sloupce v dialogovém okně záznamu, takže je snazší najít změny, které se vztahují k určitému problému.

Některé vlastnosti `tsvn`: vyžadují hodnotu `true/false`. TortoiseSVN také rozumí `yes` jako synonymum pro `true` a `no` jako synonymum pro `false`.



Nastavení Vlastností Adresáře

These properties must be set on folders for the system to work. When you commit a file or folder the properties are read from that folder. If the properties are not found there, TortoiseSVN will search upwards through the folder tree to find them until it comes to an unversioned folder, or the tree root (eg. `C:\`) is found. If you can be sure that each user checks out only from e.g. `trunk/` and not some sub-folder, then it's enough if you set the properties on `trunk/`. If you can't be sure, you should set the properties recursively on each sub-folder. A property setting deeper in the project hierarchy overrides settings on higher levels (closer to `trunk/`).

For `tsvn`: properties *only* you can use the **Recursive** checkbox to set the property to all sub-folders in the hierarchy, without also setting it on all files.



Žádné Informace z systému pro sledování problémů v Průzkumníku Úložiště

Because the issue tracker integration depends upon accessing subversion properties, you will only see the results when using a checked out working copy. Fetching properties remotely is a slow operation, so you will not see this feature in action from the repo browser.

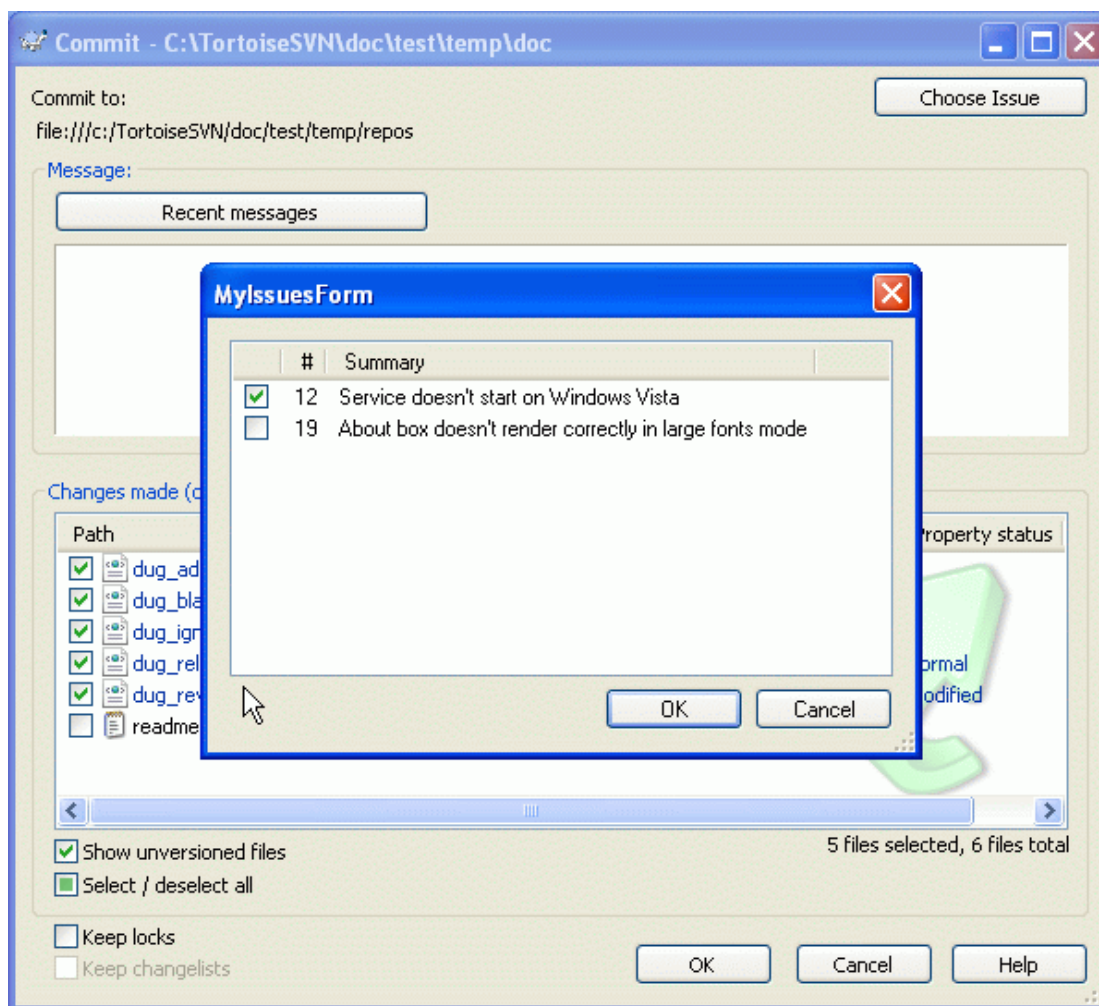
This issue tracker integration is not restricted to TortoiseSVN; it can be used with any Subversion client. For more information, read the full *Issue Tracker Integration Specification* [<http://tortoisesvn.googlecode.com/svn/trunk/doc/issuetrackers.txt>] in the TortoiseSVN source repository. (3 – „TortoiseSVN je zadarmo!“ explains how to access the repository).

4.28.2. Získávání informací od Systému pro Sledování Problémů

Předchozí část se zabývala přidáním informace do zpráv záznamů. Ale co když potřebujete získat informace ze systému pro sledování problémů? Dialogové okno odevzdání má rozhraní COM, které umožňuje integraci s externím programem, který umí mluvit s Vaším systémem pro sledování problémů. Typicky se budete chtít systému dotázat na seznam otevřených problémů přiřazených Vám, takže si můžete vybrat problémy, které se v tomto odevzdání řeší.

Any such interface is of course highly specific to your issue tracker system, so we cannot provide this part, and describing how to create such a program is beyond the scope of this manual. The interface definition and sample plugins in C# and C++/ATL can be obtained from the `contrib` folder in the *TortoiseSVN repository* [<http://tortoisesvn.googlecode.com/svn/trunk/contrib/issue-tracker-plugins>]. (3 – „TortoiseSVN je zadarmo!“ explains how to access the repository). A summary of the API is also given in 6 – „Rozhraní *IBugtraqProvider*“. Another (working) example plugin in C# is *Gurtle* [<http://code.google.com/p/gurtle/>] which implements the required COM interface to interact with the *Google Code* [<http://code.google.com/hosting/>] issue tracker.

Pro ilustraci předpokládejme, že Váš správce systému Vám poskytl zásuvný modul pro systém pro sledování problémů, který jste nainstalovali a že jste nastavili některé Vaše pracovní kopie, aby zásuvný modul používali v dialogovém okně nastavení TortoiseSVN. Při otevření dialogového okna odevzdání z pracovní kopie, které byl zásuvný modul přidělen, zobrazí se nové tlačítko v horní části dialogového okna.



Obrázek 4.50. Ukázka dialogového okna dotazu na systém pro sledování problémů

V tomto příkladu si můžete vybrat jeden nebo více otevřených problémů. Zásuvný modul pak může vytvořit speciálně formátovaný text, který se přidává ke zprávě záznamu.

4.29. Integrace s Internetovými Prohlížeči úložiště

Existuje několik on-line prohlížečů úložišť k dispozici pro použití se Subversion jako například [ViewVC](http://www.viewvc.org/) [http://www.viewvc.org/] a [WebSVN](http://websvn.tigris.org/) [http://websvn.tigris.org/]. TortoiseSVN poskytuje prostředky pro spojení s těmito prohlížeči.

Můžete integrovat prohlížeč úložiště dle vlastního výběru v TortoiseSVN. Pro toto musíte nastavit některé vlastnosti, které určují vazby. Musí být nastaveny na adresáře: (4.17 – „Nastavení projektu“)

webviewer:revision

Set this property to the URL of your repo viewer to view all changes in a specific revision. It must be properly URI encoded and it has to contain %REVISION%. %REVISION% is replaced with the revision number in question. This allows TortoiseSVN to display a context menu entry in the log dialog Context Menu → View revision in webviewer

webviewer:pathrevision

Set this property to the URL of your repo viewer to view changes to a specific file in a specific revision. It must be properly URI encoded and it has to contain %REVISION% and %PATH%. %PATH%

% is replaced with the path relative to the repository root. This allows TortoiseSVN to display a context menu entry in the log dialog **Context Menu** → **View revision and path in webviewer**. For example, if you right-click in the log dialog bottom pane on a file entry `/trunk/src/file` then the `%PATH%` in the URL will be replaced with `/trunk/src/file`.

Můžete také použít relativní URL místo absolutní. To je užitečné v případě, že Váš internetový prohlížeč je na stejné doméně/serveru jako zdrojové úložiště. V případě, že se jméno domény změní, nemusíte upravovat vlastnosti `webviewer:revision` a `webviewer:pathrevision`. Formát je stejný jako u vlastnosti `bugtraq:url`. Viz 4.28 – „Integrace se Systémy pro Sledování Chyb/Problémů“.



Nastavení Vlastností Adresáře

These properties must be set on folders for the system to work. When you commit a file or folder the properties are read from that folder. If the properties are not found there, TortoiseSVN will search upwards through the folder tree to find them until it comes to an unversioned folder, or the tree root (eg. `C:\`) is found. If you can be sure that each user checks out only from e.g. `trunk/` and not some sub-folder, then it's enough if you set the properties on `trunk/`. If you can't be sure, you should set the properties recursively on each sub-folder. A property setting deeper in the project hierarchy overrides settings on higher levels (closer to `trunk/`).

For `tsvn:` properties *only* you can use the **Recursive** checkbox to set the property to all sub-folders in the hierarchy, without also setting it on all files.



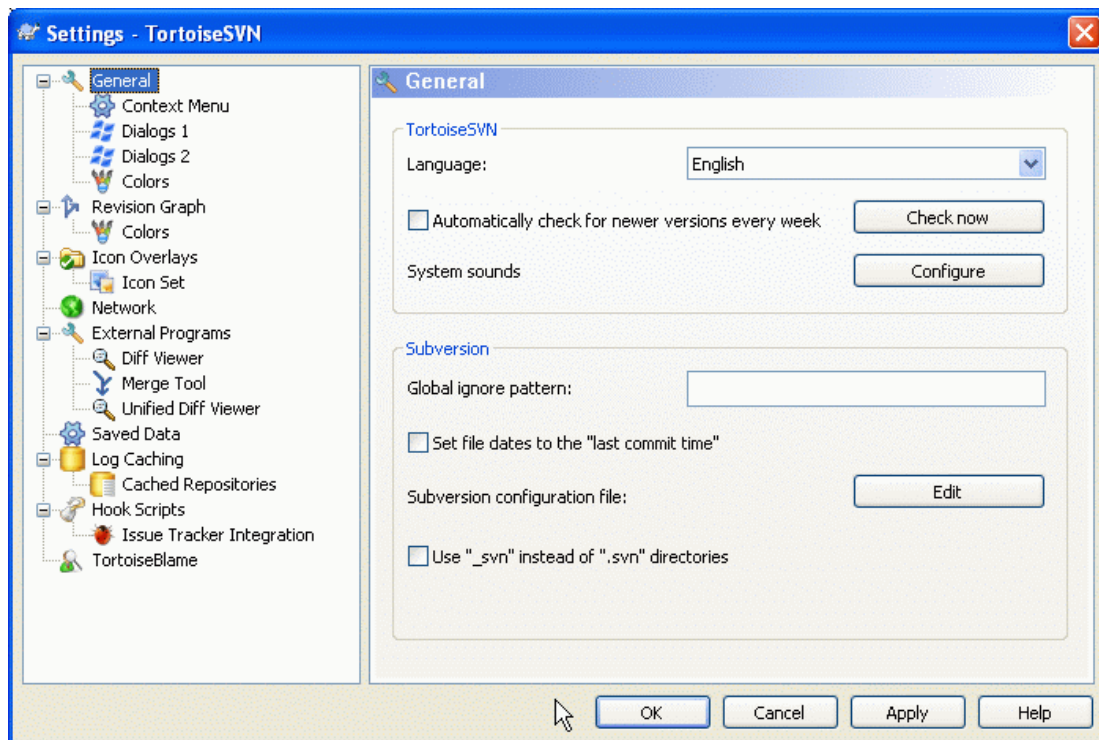
No Repo Viewer Links from Repository Browser

Because the repo viewer integration depends upon accessing subversion properties, you will only see the results when using a checked out working copy. Fetching properties remotely is a slow operation, so you will not see this feature in action from the repo browser.

4.30. Nastavení TortoiseSVN

Chcete-li zjistit, na co různá nastavení jsou, nechte na chvíli kurzor myši na rámečku pro úpravy/zaškrtávacím políčkem ... a vyskočí užitečný popis.

4.30.1. Obecná nastavení



Obrázek 4.51. Dialogové okno nastavení, stránka Obecné

Toto dialogové okno umožňuje zadat preferovaný jazyk a specifická nastavení Subversion.

Jazyk

Vybere jazyk uživatelského rozhraní. Co jiného jste čekali?

Automatically check for newer versions every week

If checked, TortoiseSVN will contact its download site once a week to see if there is a newer version of the program available. Use **Check now** if you want an answer right away. The new version will not be downloaded; you simply receive an information dialog telling you that the new version is available.

Systémové zvuky

TortoiseSVN má tři vlastní zvuky, které jsou nainstalovány ve výchozím nastavení.

- Chyba
- Upozornění
- Varování

Můžete si vybrat jiné zvuky (nebo tyto zvuky vypnout úplně) pomocí ovládacího panelu Windows. Konfigurace je zkratka pro Kontrolní Panel.

Vzorek globálního ignorování

Vzorky globálního ignorování jsou používány k prevenci souborů bez verze objevovat se např. v dialogovém okně odevzdání. Soubory shodující se se vzory jsou také ignorovány importem. Ignorujte soubory či adresáře zadáním jména nebo přípony. Vzory jsou odděleny mezerami, např. `bin obj *.bak *.~?? *.Jar *. [Tt]mp`. Tyto vzory by neměly obsahovat žádné oddělovače cest. Všimněte si také, že neexistuje žádný způsob, jak rozlišovat mezi soubory a adresáři. Přečtěte si [4.13.1 – „Shoda vzorů v seznamech k Ignorování“](#) pro více informací o syntaxi shody vzorů.

Nezapomeňte, že vzory ignorování, které zde zadáte, budou mít vliv také na další Subversion klienty běžící na Vašem PC, včetně příkazové řádky klienta.



Výstraha

Pokud použijete konfigurační soubor pro nastavení vzorů globálního ignorování, přepíše nastavení, které provedete zde. Do konfiguračního souboru Subversion můžete vstoupit pomocí **Upravit** jak je popsáno níže.

Tento vzorec ignorovat bude mít vliv na všechny Vaše projekty. Nemá verzi, takže nebude mít vliv na ostatní uživatele. Naproti tomu můžete také použít vlastnost `svn:ignore` k vyloučení souborů nebo adresářů ze správy verzí. Přečtěte si [4.13 – „Ignorování Souborů a Adresářů“](#) pro více informací.

Nastavit data souboru na „poslední čas odevzdání“

Tato volba říká TortoiseSVN, aby nastavil data souborů na poslední čas odevzdání, při provádění odevzdání nebo aktualizace. Jinak TortoiseSVN použije aktuální datum. Pokud vyvíjíte software je obecně nejlepší používat aktuální datum, protože sestavovací systémy se běžně dívají na časová razítka, aby se rozhodli, které soubory potřebujete ke kompilaci. Pokud používáte „poslední čas odevzdání“ a vrátíte se ke starší revizi souboru, nemusí se Váš projekt kompilovat tak, jak jste očekávali.

Konfigurační soubor Subversion

Použijte **Upravit** pro úpravu konfiguračního souboru Subversion přímo. Některá nastavení nelze měnit přímo v TortoiseSVN a je třeba je nastavit zde. Pro více informací o Subversion souboru konfigurační prohlédněte si [Oblast Konfigurace běhu programu](http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html) [http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html]. Část o [Automatické Nastavení Vlastností](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] je obzvlášť zajímavá a to se nastavuje zde. Všimněte si, že Subversion umí číst informace o konfiguraci z několika míst a musíte vědět, který z nich má přednost. V části [Konfigurace a registr systému Windows](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] se dozvíte více.

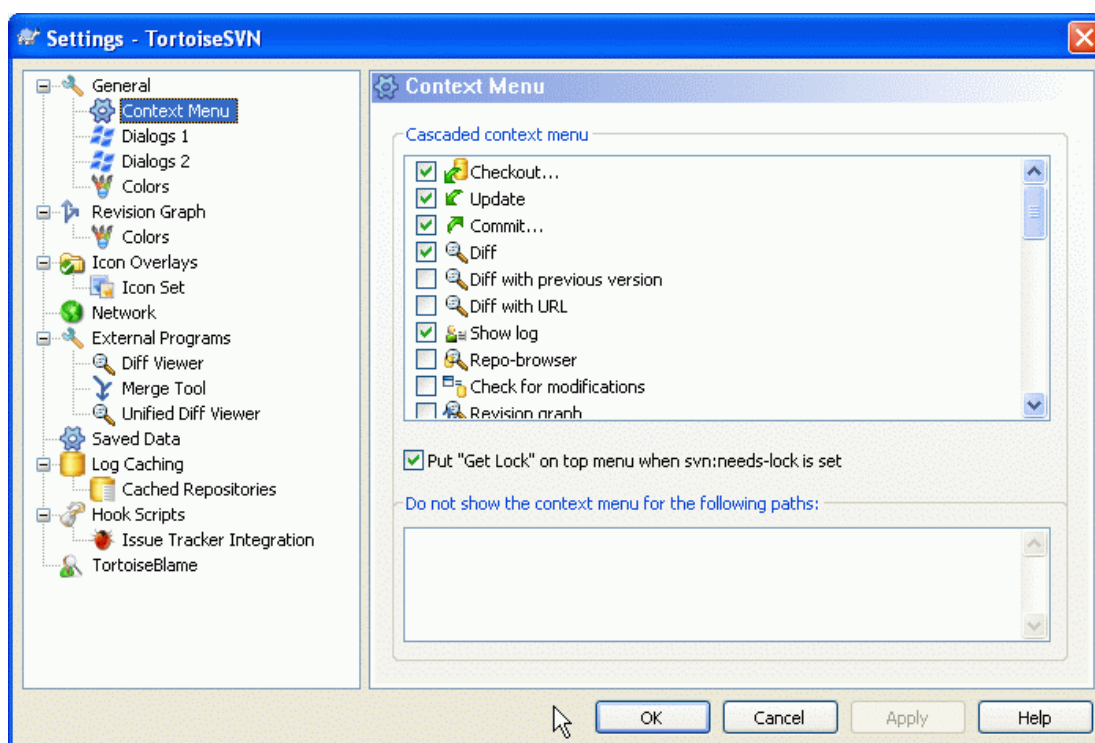
Použití adresářů `_svn` místo `.svn`

VS.NET při použití s internetovými projekty, nezvládá adresáře `.svn`, které Subversion používá k ukládání svých vnitřních informací. To není chyba v Subversion. Chyba je ve VS.NET a FrontPage Rozšíření, které používá. Přečtěte si [4.30.11 – „Pracovní adresáře Subversion“](#), abyste se dozvěděli více o tomto problému.

Pokud chcete změnit chování Subversion a TortoiseSVN, můžete použít toto políčko pro nastavení proměnné prostředí, která toto ovládá.

Měli byste si uvědomit, že změnou této volby nebudou automaticky převedeny stávající pracovní kopie, aby používali nový správcovský adresář. Toto budete muset udělat sami pomocí skriptu (viz náš FAQ) nebo jednoduše získat novou pracovní kopii.

4.30.1.1. Nastavení Kontextového Menu



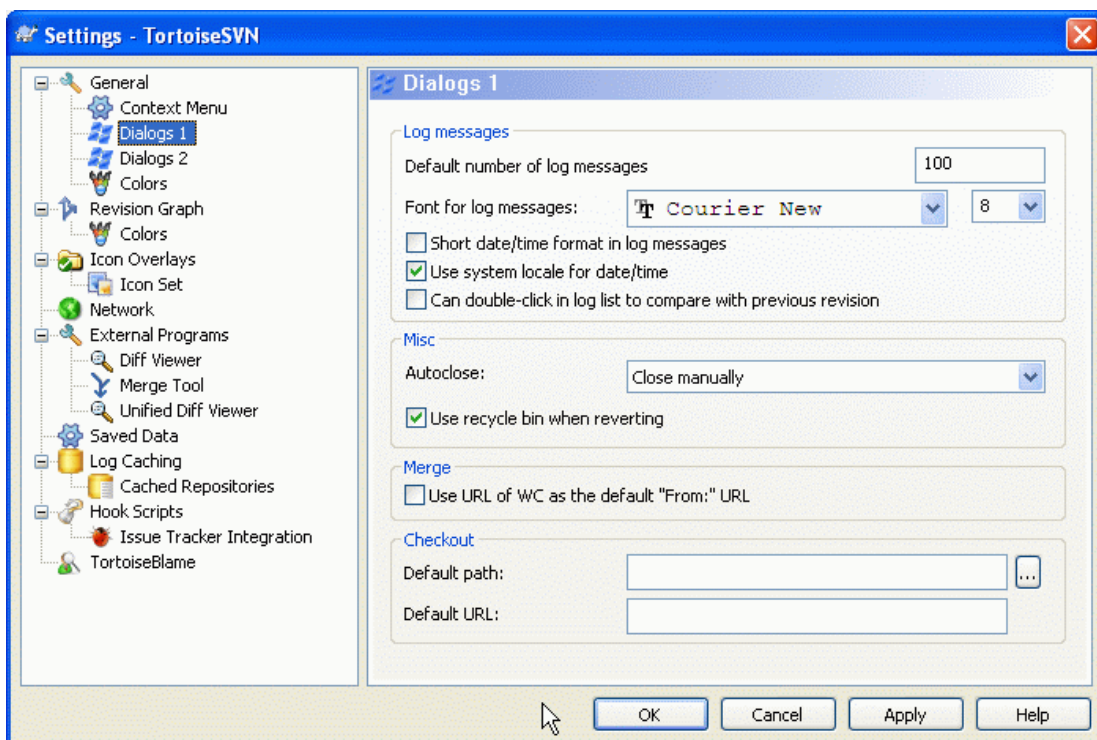
Obrázek 4.52. Dialogové okno Nastavení, Stránka Kontextového menu

Tato stránka Vám umožňuje určit, které z položek kontextového menu TortoiseSVN se objeví v hlavním kontextovém menu a které se objeví v menu TortoiseSVN. Ve výchozím nastavení je většina položek odškrtnuta a objeví se v podmenu.

K dispozici je speciální případ pro **Získat zámek**. Můžete ho samozřejmě povýšit na nejvyšší úroveň s využitím výše uvedeného seznamu, ale protože většina souborů nepotřebuje zamknout, toto jen udělá větší nepořádek. Nicméně, soubor s vlastností `svn:needs-lock` potřebuje tuto činnost pokaždé, když je upraven, takže v tom případě je velmi užitečné mít ho na nejvyšší úrovni. Zaškrtnutím políčka zde znamená, že když je vybrán soubor, který má nastavenou vlastnost `svn:needs-lock`, **Získat zámek** se objeví vždy na nejvyšší úrovni.

Pokud existují některé cesty na Vašem počítači, kde prostě nechcete, aby se kontextové menu TortoiseSVN vůbec neobjevovalo, můžete je uvést v kolonce v dolní části.

4.30.1.2. Dialogové Okno Nastavení TortoiseSVN 1



Obrázek 4.53. Dialogové Okno Nastavení, Dialogová okna 1 Stránka

Toto dialogové okno Vám umožňuje nastavit některé z dialogových oken TortoiseSVN tak, jak se Vám líbí.

Výchozí počet zpráv záznamu

Omezuje počet zpráv záznamu, které TortoiseSVN načte při prvním výběru TortoiseSVN → Zobrazit záznam Užitečné pro pomalé připojení k serveru. Můžete vždy použít Zobrazit vše nebo Další 100 pro získání více zpráv.

Typ písma pro zprávy záznamu

Vybere typ písma a velikost, které použít pro zobrazení samotné zprávy záznamu ve prostředním panelu dialogového okna záznamu revizí a při psaní zprávy záznamu v dialogovém okně Odevzdání.

Krátký formát data a času ve zprávách záznamu

Pokud standardní dlouhé zpráv používají až příliš mnoho místa na obrazovce použijte krátký formát.

Can double-click in log list to compare with previous revision

If you frequently find yourself comparing revisions in the top pane of the log dialog, you can use this option to allow that action on double-click. It is not enabled by default because fetching the diff is often a long process, and many people prefer to avoid the wait after an accidental double-click, which is why this option is not enabled by default.

Progress Dialog

TortoiseSVN může automaticky zavřít všechna dialogová okna průběhu, když je akce dokončena bez chyb. Toto nastavení Vám umožňuje vybrat podmínky pro uzavření dialogových oken. Výchozí nastavení (doporučeno) je Zavřít ručně, které umožňuje přezkoumat všechny zprávy a zkontrolovat, co se stalo. Můžete se však rozhodnout, že chcete ignorovat některé typy zpráv a dialogové okno zavřít automaticky, pokud nejsou k dispozici žádné zásadní změny.

Automaticky zavřít, pokud nejsou žádná sloučení, přidání nebo smazání znamená, že dialogové okno průběhu se uzavře, pokud proběhly jednoduché aktualizace, ale pokud změny z

úložiště byly sloučeny s Vašimi, nebo pokud některé soubory byly přidány nebo odstraněny, zůstane dialogové okno otevřené. Zůstane také otevřené, pokud se vyskytly nějaké konflikty nebo chyby v průběhu operace.

Auto-close if no merges, adds or deletes for local operations means that the progress dialog will close as for Auto-close if no merges, adds or deletes but only for local operations like adding files or reverting changes. For remote operations the dialog will stay open.

Automaticky zavřít, pokud nejsou žádné konflikty uvolní více kritéria a zavře dialogové okno i když došlo ke sloučení, přidání nebo odstranění. Nicméně pokud se nevyskytly žádné konflikty nebo chyby, zůstane dialogové okno otevřené.

Automaticky zavřít, pokud nejsou žádné chyby vždy zavře dialogové okno i když se vyskytly konflikty. Jedinou podmínkou, která dialogové okno nechá otevřené, je chybový stav, který nastane, když Subversion není schopno dokončit úkol. Například, pokud aktualizace selže, protože server je nedostupný, nebo se odevzdání nezdařilo, protože je pracovní kopie zastaralá.

Při vrácení zpět použít koš

Když vrátíte místní úpravy, jsou změny zahozeny. TortoiseSVN Vám dává záchrannou síť navíc zasláním upraveného souboru do koše před vrácením původní kopie. Pokud chcete koš přeskočit, tuto volbu odškrtněte.

Použít URL pracovní kopie jako výchozí „Z:“ URL

V dialogovém okně sloučení výchozí chování je, aby URL Z: byla mezi sloučeními uložena do paměti. Nicméně, někteří lidé chtějí slučovat z mnoha různých míst v jejich hierarchii a je pro ně snadnější začít s URL aktuální pracovní kopie. Pak může být upravena, aby odkazovala na obdobnou cestu na jinou větev.

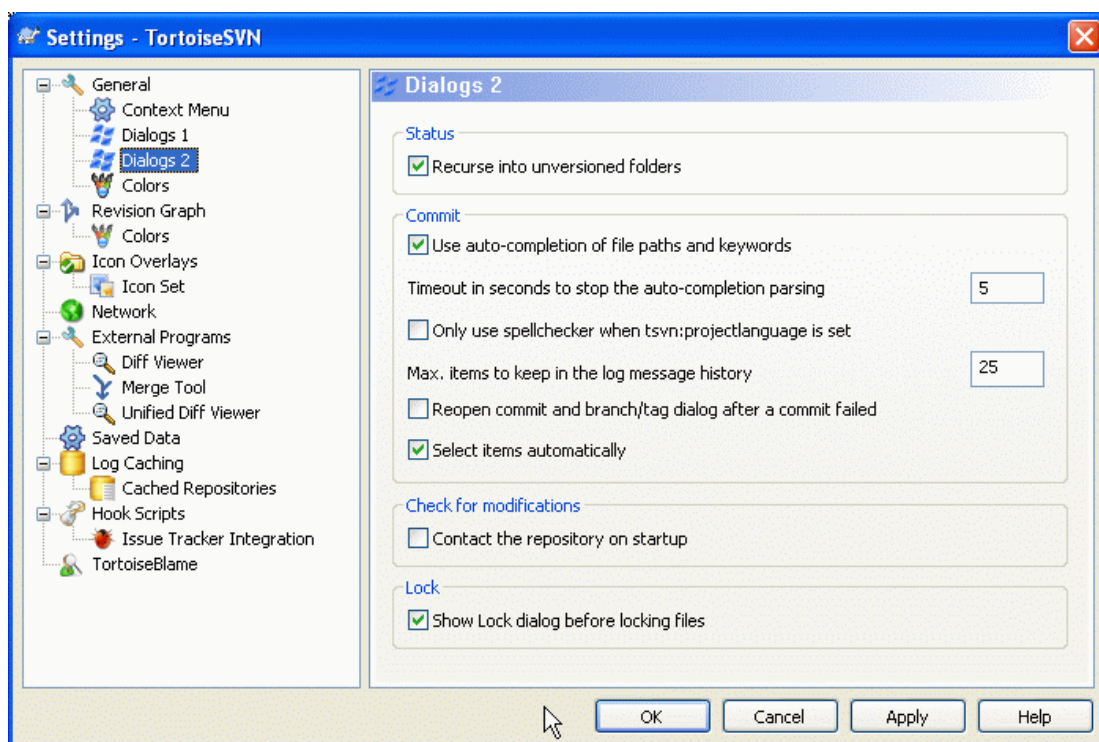
Výchozí cesta získání

Můžete určit výchozí cestu pro získání. Pokud máte všechny Vaše získání na jednom místě, je užitečné mít jednotku a adresář předvyplněný, aby pouze stačilo přidat na konec nový název adresáře.

Výchozí URL získání

Můžete také určit výchozí URL pro získání. Pokud často získáváte podprojekty z velmi velkých projektů, může být užitečné mít URL předvyplněnou, takže stačí na konec přidat jméno podprojektu.

4.30.1.3. Dialogové Okno Nastavení TortoiseSVN 2



Obrázek 4.54. Dialogové okno Nastavení, Dialogová okna Stránka 2

Rekurzivně do adresářů bez verze

Pokud je toto políčko zaškrtnuto (výchozí stav), pak kdykoli je stav adresáře bez verze uveden v dialogovém okně Přidat, Odevzdat nebo Zjistit změny, každý podřazený soubor a adresář jsou rovněž uvedeny. Zrušíte-li zaškrtnutí tohoto políčka, je zobrazen pouze nadřazený soubor/adresář bez verze. Odškrtnutí redukuje nepořádek v těchto dialogových oknech. V takovém případě, pokud vyberete adresář bez verze pro Přidání, je přidán rekurzivně.

Použit automatické doplňování cest souborů a klíčových slov

Dialogové okno odevzdání obsahuje zařízení k rozebrání seznamu souborů, které jsou odevzdávány. Zadáte-li první 3 písmena položky v seznamu, zobrazí se okno automatického dokončení a můžete stisknout Enter k dokončení názvu souboru. Zaškrtněte políčko, pokud chcete tuto funkci povolit.

Časový limit v sekundách pro zastavení zpracování automatického dokončení

Analýzátor automatického dokončování, může být velmi pomalý, pokud existuje hodně velkých souborů ke kontrole. Tento limit zabrání, aby dialogové okno odevzdání bylo příliš dlouho nedostupné. Pokud Vám chybí důležité informace v automatickém dokončení, můžete časový limit prodloužit.

Kontrolu pravopisu používat pouze když je nastavena `tsvn:projectlanguage`

Pokud nechcete používat kontrolu pravopisu pro všechny odevzdání, zaškrtněte toto políčko. Kontrola pravopisu bude i nadále povolena, kde to vlastnosti projektu vyžadují.

Max. počet položek, které se mají držet v historii zpráv záznamu

Když píšete zprávu záznamu v dialogovém okně odevzdání, TortoiseSVN ji uloží pro případné další použití později. Ve výchozím nastavení bude uloženo posledních 25 zpráv záznamu pro každé úložiště, ale tento počet můžete nastavit zde. Pokud máte mnoho různých úložišť, můžete ho snížit, aby se zabránilo zaplnění Vašeho registru.

Nezapomeňte, že toto nastavení se vztahuje pouze na zprávy, které napíšete na tomto počítači. Nemá to nic společného s úschovou záznamu.

Znovu otevřít po neúspěchu dialogové okno pro odevzdání větví/značek

When a commit fails for some reason (working copy needs updating, pre-commit hook rejects commit, network error, etc), you can select this option to keep the commit dialog open ready to try again. However, you should be aware that this can lead to problems. If the failure means you need to update your working copy, and that update leads to conflicts you must resolve those first.

Položky vybrat automaticky

Normální chování v dialogovém okně odevzdání je pro všechny změněné položky (s verzí), aby byly vybrány k odevzdání automaticky. Pokud dáváte přednost, aby ze začátku nic nebylo vybráno a vybírat položky k odevzdání ručně, zrušte zaškrtnutí tohoto políčka.

Kontaktovat úložiště při spuštění

Ve výchozím nastavení dialogové okno Zjistit změny zkontroluje pracovní kopii a úložiště kontaktuje pouze po kliknutí na tlačítko **Zkontrolovat úložiště**. Chcete-li, aby úložiště bylo zkontrolováno vždy, můžete použít toto nastavení, aby se tato činnost děla automaticky.

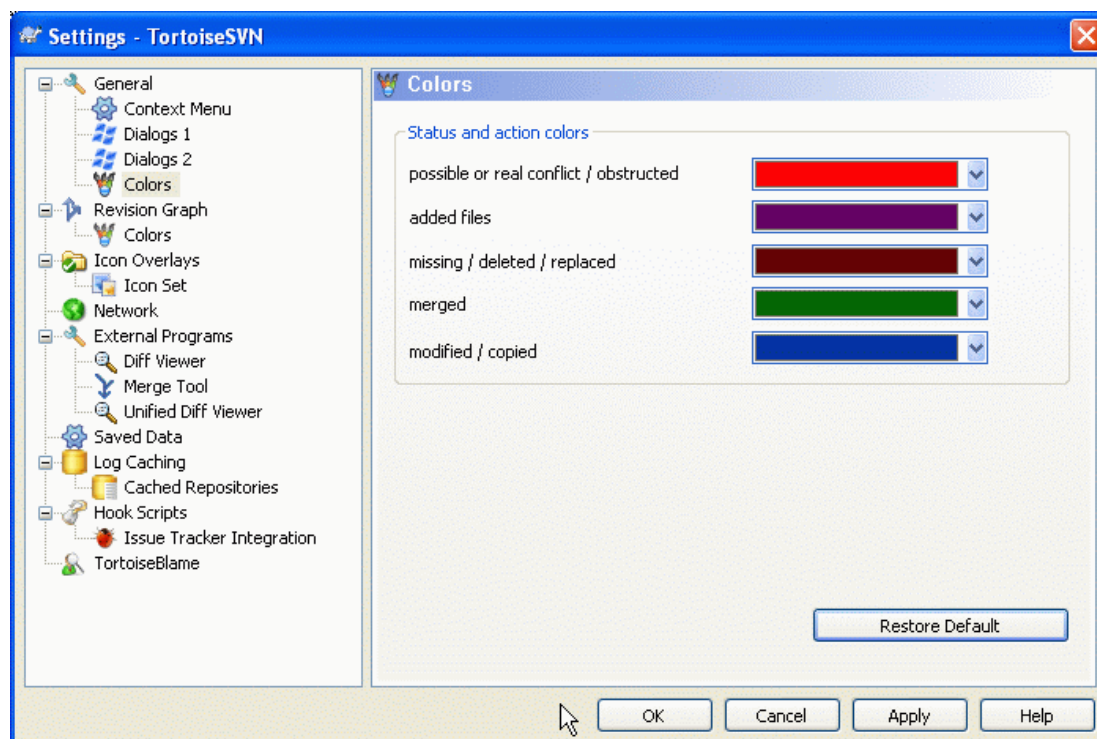
Zobrazit dialogové okno Uzamčení před zamknutím souborů

Když vyberete jeden nebo více souborů a pak použijte TortoiseSVN → **Zámek** pro získání zámku na tyto soubory, v některých projektech je obvyklé psát textovou zprávu vysvětlující, proč jste zamkli soubory. Pokud nechcete používat zprávy o zamčení, můžete zrušit zaškrtnutí tohoto políčka k přeskočení dialogového okna a soubory uzamknout okamžitě.

Pokud použijete příkaz **zámek** na adresář, dialogové okno zamknout je vždy zobrazeno. To Vám také dává možnost vybrat soubory k uzamknutí.

Pokud Váš projekt používá vlastnost `tsvn:lockmsgminsize`, zobrazí se dialogové okno zamknutí, bez ohledu na toto nastavení, protože projekt *vyžaduje* zprávy o zamknutí.

4.30.1.4. Nastavení Barev TortoiseSVN



Obrázek 4.55. Dialogové okno Nastavení, Strana Barvy

Toto dialogové okno Vám umožňuje nastavit barvy textu použité v dialogových oknech TortoiseSVN tak, jak se Vám líbí.

možný nebo reálný konflikt / zablokovaný

Došlo ke konfliktu při aktualizaci, nebo k ní může dojít během sloučení. Aktualizace je blokována existujícím souborem/adresářem bez verze se stejným názvem jako s verzí.

Tato barva se používá také pro chybové zprávy v dialogových oknech průběhu.

Přidané soubory

Položky přidané do úložiště.

Chybějící / smazané / nahrazené

Položky smazané z úložiště, chybějící, nebo smazané z pracovní kopie a nahrazené jiným souborem se stejným názvem.

Sloučeno

Změny z úložiště byly úspěšně sloučeny s pracovní kopií bez vytváření jakýchkoli konfliktů.

Změněno / kopírováno

Přidat s historií, nebo cestami zkopírovanými z úložiště. Také použity v dialogovém okně záznamu pro záznamy, které zahrnují kopírované položky.

Smazaný uzel

Položka, která byla smazána z úložiště.

Přidaný uzel

Položka, která byla přidána do úložiště operací přidání, kopírování nebo přesunutí.

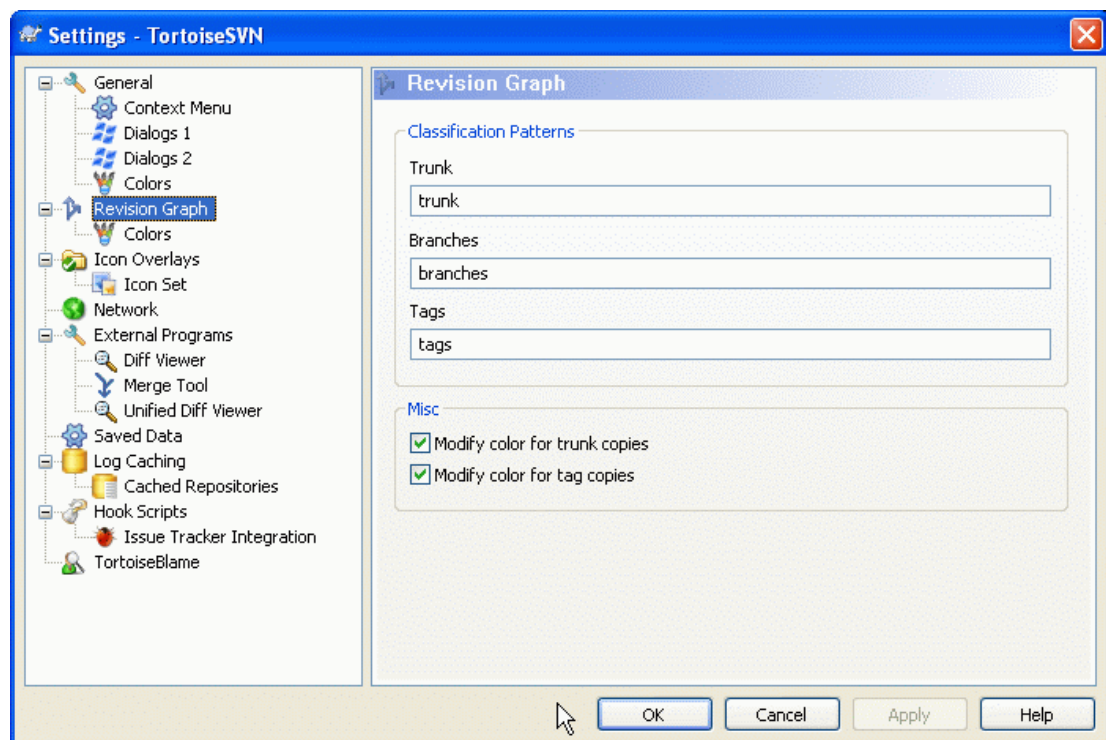
Přejmenovaný uzel

Položka, která byla v úložišti přejmenována.

Nahrazený uzel

Původní položka byla smazána a nová položka se stejným názvem ji nahradila.

4.30.2. Nastavení Grafu Revizí



Obrázek 4.56. Dialogové okno nastavení, Stránka Graf Revizí

Klasifikační Vzory

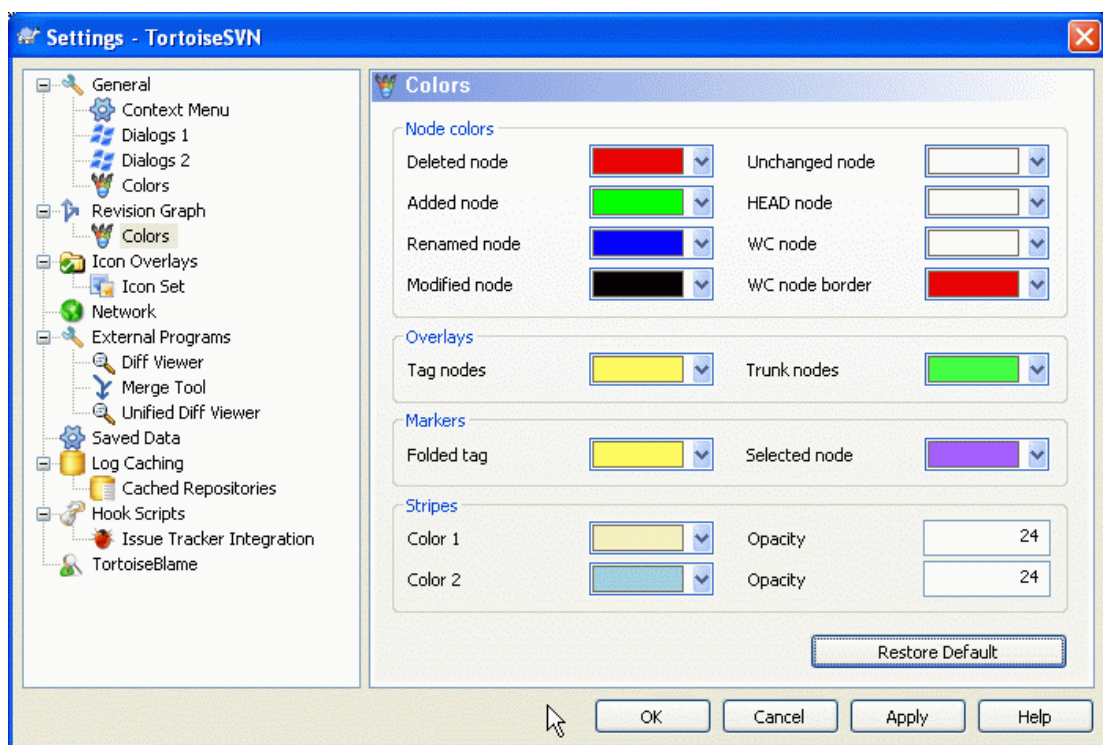
Graf revizí se pokusí ukázat jasnější představu o Vaší struktuře úložiště tím, že rozlišuje mez kmenem, větvemi značkami. Protože žádná taková klasifikace není do Subversion zabudována, je tato informace vyjmuta ze jmen cest. Výchozí nastavení předpokládá, že používáte klasické anglické názvy, jak je navrženo v dokumentaci Subversion, ale samozřejmě Vaše využití se mohou lišit.

Určete vzory používané k rozpoznání těchto cest ve třech rámečcích. Vzory budou porovnávány bez citlivosti na velikost písmen, ale musíte je zadat malými písmeny. Zástupné znaky * a ? budou fungovat jako obvykle a můžete použít ; k oddělení více vzorů. Nezadávejte žádné extra mezery, protože budou zahrnuta do odpovídajícího vzoru.

Změnit barvy

Barvy jsou používány v rámci grafu revizí označující typ uzlu, tj. zda je uzel přidáný, smazaný, přejmenovaný. Abychom Vám pomohli vybrat klasifikace uzlů, můžete povolit graf revizí míchat barvy pro sdělení údaje o typu a klasifikaci uzlu. Pokud je políčko zaškrtnuto, je míchání používáno. Pokud toto políčko není zaškrtnuto, je barva použita pouze k označení uzlu typu. Pomocí dialogového okna výběru barvy přiřadíte konkrétní barvy k použití.

4.30.2.1. Barvy Grafu Revizí



Obrázek 4.57. Dialogové okno nastavení, Stránka Barvy Grafu Revizí

Tato stránka Vám umožňuje nastavit používané barvy. Nezapomeňte, že barva zde uvedená je plná barva. Většina uzlů je vybarvena pomocí směsi barvy typu uzlu, barvy pozadí a nepovinnou barvou klasifikace.

Smazaný Uzel

Položky, které byly ve stejné revizi smazány a nebyly zkopírovány nikam jinam.

Přidaný Uzel

Nově přidané položky, nebo zkopírované (přidané s historií).

Přejmenovaný Uzel

Položky odstraněné z jednoho místa a přidané do jiného ve stejné revizi.

Změněný Uzel

Jednoduchá změna bez jakéhokoliv přidání nebo smazání.

Nezměněný uzel

Může být použita k zobrazení revizi použité jako zdroj kopie i když žádná změna (položky, pro kterou byl graf vytvořen) se v této revizi nekonala.

HEAD uzel

Současná revize HEAD v úložišti.

Uzel pracovní kopie

Pokud jste se rozhodli zobrazit další uzel pro Vaši změněnou pracovní kopii, připojenou ke své revizi posledního odevzdání na grafu, použije se tato barva.

Okraj uzlu pracovní kopie

Pokud jste se rozhodli zobrazit, zda je pracovní kopie změněná, použije se tento barevný okraj na uzel pracovní kopie když jsou změny nalezeny.

Uzly značek

Uzly klasifikované jako značky mohou být smíchány s touto barvou.

Uzly kmene

Uzly klasifikované jako kmen mohou být smíchány s touto barvou.

Vyznačení Složených Značek

Používáte-li složení značek pro úsporu místa, značky jsou vyznačeny ve zdroji kopírování pomocí bloku této barvy.

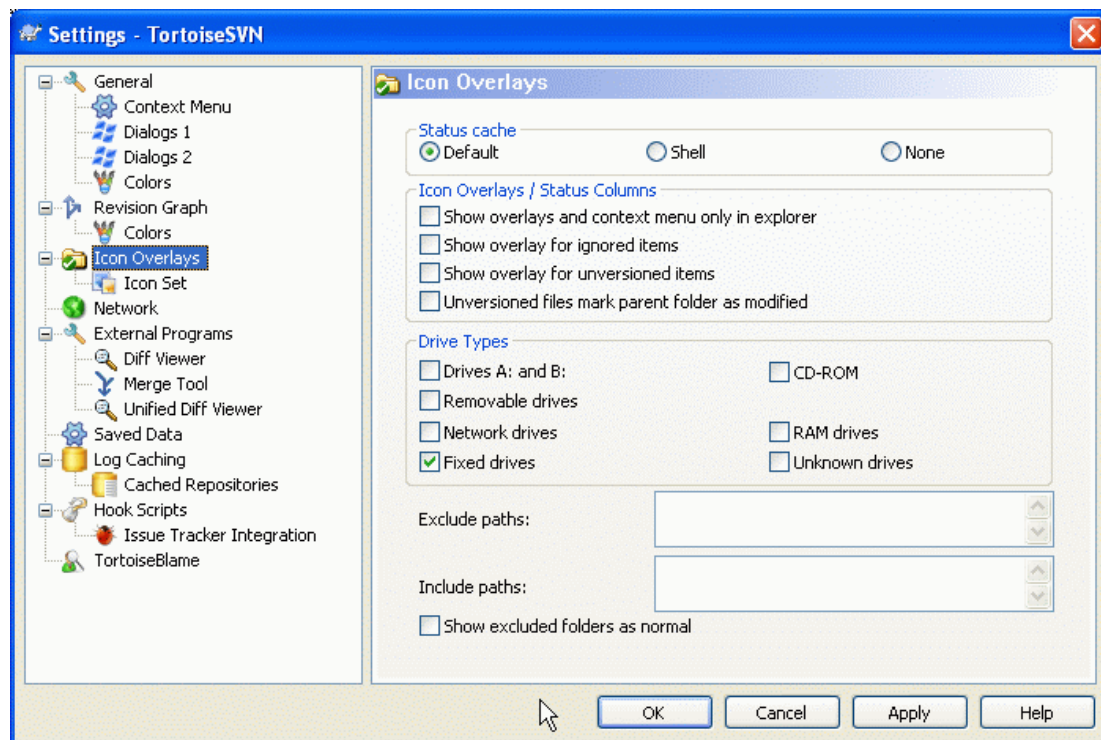
Vyznačení Vybraného uzlu

Když kliknete levým tlačítkem myši na uzel, je výběr vyznačen blokem této barvy.

Pruhy

Tyto barvy jsou používány, když je graf rozdělen do podstromů a na pozadí je zbarven střídavými pruhy pro rozlišení jednotlivých stromů.

4.30.3. Nastavení Překryvu Ikon



Obrázek 4.58. Dialogové okno nastavení, Stránka Překryv Ikon

Tato stránka Vám umožňuje vybrat položky, na kterých se zobrazí ikona překryvu TortoiseSVN.

Ve výchozím nastavení se překryv ikon a kontextové menu zobrazí ve všech dialogových oknech otevřít/uložit, stejně jako v Průzkumníku Windows. Pokud chcete, aby se objevily *pouze* v Průzkumníku Windows, zaškrtněte políčko **Zobrazovat překryvy a kontextové menu pouze v průzkumníku Windows**.

Ignored items and Unversioned items are not usually given an overlay. If you want to show an overlay in these cases, just check the boxes.

Můžete také zvolit označit adresáře jako změněné, pokud obsahují položky bez verze. To by mohlo být užitečné pro připomenutí, že jste vytvořili nové soubory, které ještě nemají verzi. Tato volba je dostupná pouze při použití *výchozího* uchovávání stavu (viz níže).

Vzhledem k tomu, že trvá poměrně dlouho načíst stav pracovní kopie, TortoiseSVN používá úschovu pro uložení stavu, takže průzkumník není příliš zpomalován, když zobrazuje překryvy. Můžete si vybrat, který typ úschovy TortoiseSVN má použít podle Vašeho systému a velikosti pracovní kopie zde:

Výchozí

Uchovává všechny informace o stavu v samostatném procesu (TSVNCache.exe). Tento proces sleduje všechny disky pro změny a načte stav, budou-li soubory v pracovní kopii změněny. Proces běží s co nejmenší prioritou, takže ostatní programy nebudou kvůli tomu mít nedostatek zdrojů. To také znamená, že informace o stavu není v *reálném čase*, ale může trvat několik sekund, než se překryvy změní.

Výhody: překryvy ukazují stav rekurzivně, tj. pokud soubor hluboko uvnitř pracovní kopie je upraven, všechny adresáře až po kořen pracovní kopie bude také zobrazovat překryv změny. A protože tento proces může poslat oznámení do rozhraní, překryvy v levém stromovém zobrazení se obvykle také mění.

Nevýhody: proces probíhá neustále, i když nepracujete na Vašich projektech. Také používá asi 10-50 megabajtů RAM v závislosti na počtu a velikosti Vašich pracovních kopií.

Rozhraní

Uchovávání se provádí přímo uvnitř DLL rozšíření rozhraní, ale pouze pro aktuálně zobrazený adresář. Pokaždé, když přejdete do jiného adresáře, informace o stavu jsou načteny znovu.

Výhody: potřebuje pouze velmi málo paměti (cca 1 MB RAM) a může zobrazit stav v *reálném čase*.

Nevýhody: Vzhledem k tomu, že pouze jeden adresář se uchovává, překryvy neukazují stav rekurzivně. Pro velké pracovní kopie může trvat déle zobrazit adresáře v průzkumníku, než s výchozím uchováváním. Také sloupec mime-type není k dispozici.

Žádný

S tímto nastavením, TortoiseSVN vůbec nenačítá stav v Průzkumníku. Díky tomu soubory nemají překryv a adresáře mají pouze 'normální' překryv, pokud mají verzi. Žádné jiné překryvy nejsou zobrazeny a ani žádné další sloupce nejsou k dispozici.

Výhody: nepoužívá absolutně žádnou další paměť a vůbec nezpomaluje Průzkumníka při procházení.

Nevýhody: Informace o stavu souborů a adresářů nejsou V Průzkumníku zobrazeny. Chcete-li zjistit, zda pracovní kopie jsou změněné, budete muset použít dialogové okno „Zjistit změny“.

Další skupina Vám umožňuje vybrat, které typy úložných médií by mělo zobrazovat překryvy. Ve výchozím nastavení jsou vybrány pouze pevné disky. Můžete dokonce vypnout všechny překryvy ikon, ale jaká by to pak byla legrace?

Síťové jednotky mohou být velmi pomalé, takže ve výchozím nastavení se ikony nezobrazují pro pracovní kopie nacházející se na síťových discích.

USB Flash disky jsou zvláštní případ v tom, že typ jednotky je určen samotným přístrojem. Některé vypadají jako pevné disky a některé jako vyměnitelné jednotky.

The **Exclude Paths** are used to tell TortoiseSVN those paths for which it should *not* show icon overlays and status columns. This is useful if you have some very big working copies containing only libraries which you won't change at all and therefore don't need the overlays. For example:

`f:\development\SVN\Subversion` will disable the overlays *only* on that specific folder. You still can see the overlays on all files and folder inside that folder.

`f:\development\SVN\Subversion*` will disable the overlays on *all* files and folders whose path starts with `f:\development\SVN\Subversion`. That means you won't see overlays for any files and folders below that path.

Totéž platí pro **Zahrnout Cesty**. Kromě toho, že pro tyto cesty jsou překryvy zobrazeny i když jsou překryvy zakázány pro tento konkrétní typ jednotky, nebo vyloučením cesty jak je uvedeno výše.

Users sometimes ask how these three settings interact, and the definitive answer is:

```
if (path is in include list)
    show overlays
if (path is allowed drive type) AND (path is not in exclude list)
    show overlays
```

The include list *always* makes the overlays show. Otherwise, overlays are shown for all marked drive types *unless* the path is excluded.

TSVNCache.exe také používá tyto cesty k omezení jeho skenování. Pokud chcete, aby hledal pouze ve zvláštních adresářích, zakažte všechny druhy jednotek a zahrňte pouze adresáře, které konkrétně chcete kontrolovat.



Vyloučit Jednotky SUBST

Je často vhodné použít jednotku SUBST k přístupu do Vašich pracovních kopií, např. pomocí příkazu

```
subst T: C:\TortoiseSVN\kmen\dok
```

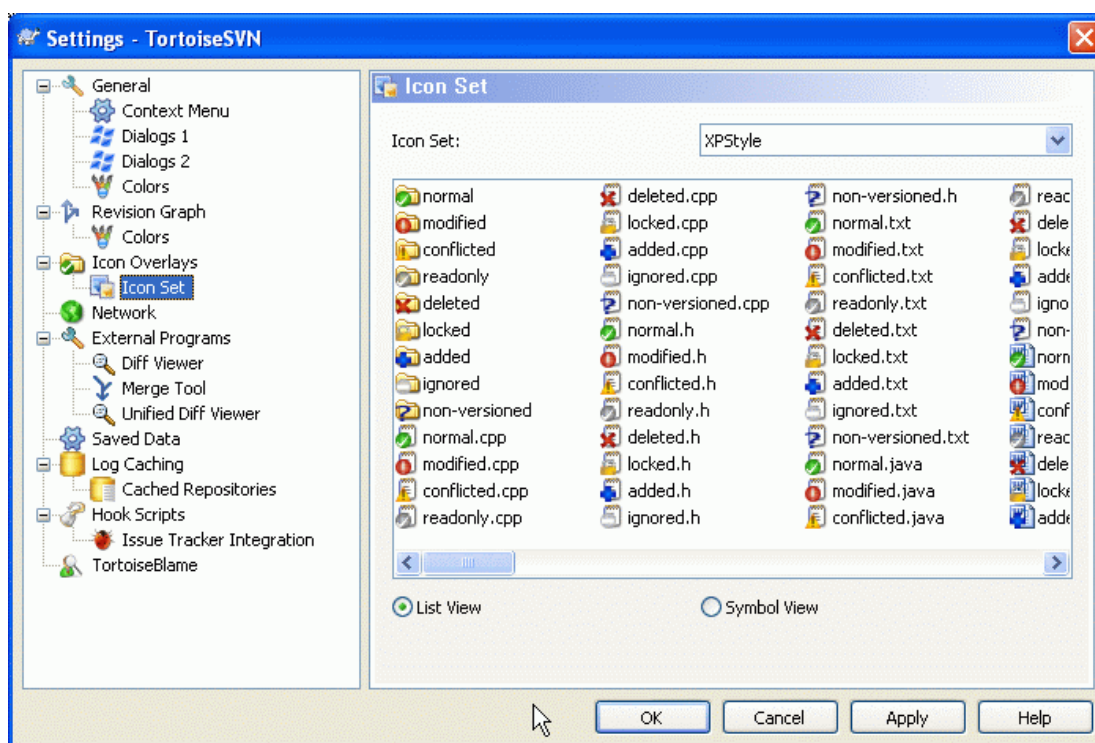
nicméně to může způsobit, že se překrývání neaktualizuje, protože TSVNCache dostane pouze jedno oznámení, kdy se soubor změní a to je obecně v původní cestě. To znamená, že Vaše překryvy na subst cestě nemůžou být nikdy aktualizovány.

Snadný způsob, jak toto obejít je vyloučit původní cestu ze zobrazení překryvů, takže překryvy se místo toho zobrazí na subst cestě.

Někdy vyloučíte oblasti, které obsahují pracovní kopie, což ušetří TSVNCache od snímání a monitorování změn, ale stále chcete vizuální indikaci, že tyto adresáře mají verze. Políčko **Zobrazit vyloučené adresáře jako 'normální'** toto umožňuje provést. S touto volbou se adresáře s verzí v každé vyloučené lokalitě (typ jednotky odškrtnut, nebo výslovně vyloučen) zobrazí jako normální a aktuální, se zeleným zaškrtnutím. Toto Vám připomene, že se díváte na pracovní kopii, i když překryv adresáře nemusí být správný. Soubory nemají žádné překryvy. Nezapomeňte, že kontextová menu stále fungují, i když překryvy nejsou zobrazeny.

Jako zvláštní výjimka, jednotky A: a B: nejsou pro volbu **Zobrazit vyloučené adresáře jako 'normální'** nikdy považovány. Důvodem je, že systém Windows je nucen hledat na jednotce, což může mít za následek zpoždění několika sekund při startu Průzkumníka, i když Váš počítač má disketovou mechaniku.

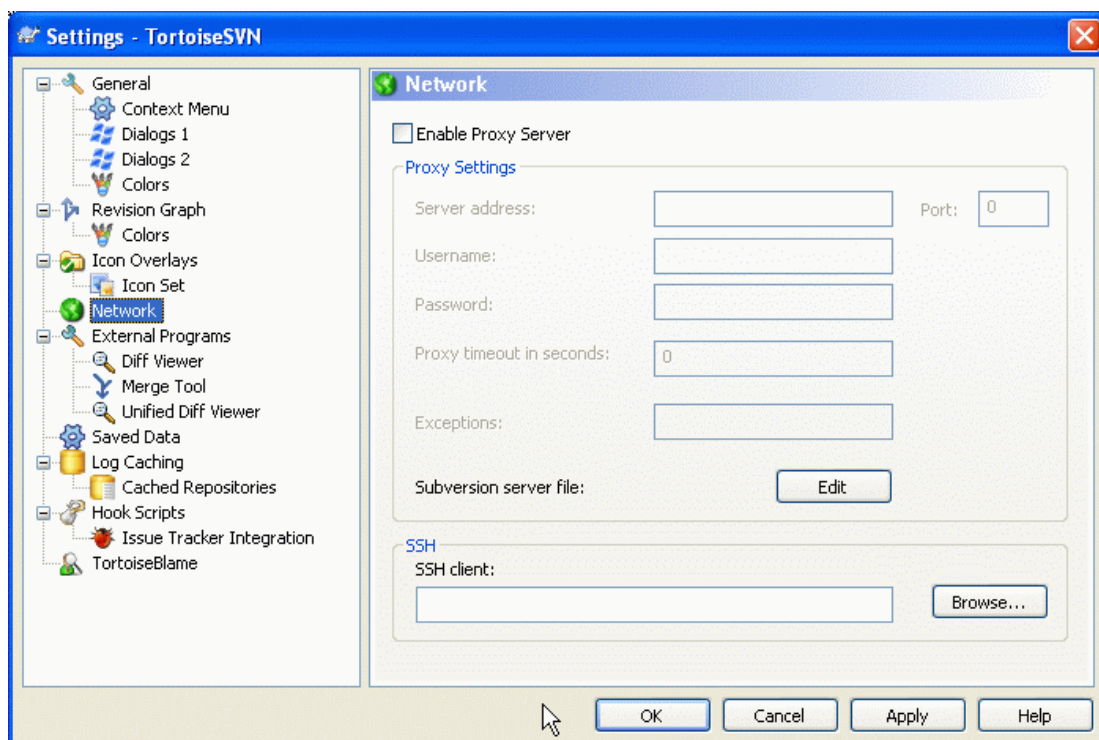
4.30.3.1. Výběr Sady Ikon



Obrázek 4.59. Dialogové okno nastavení, Stránka Sady Ikon

Můžete změnit sady ikon překrytí na tu, která se Vám nejvíce líbí. Nezapomeňte, že když změníte nastavení překryvu, budete muset restartovat počítač, aby se změny projevíly.

4.30.4. Nastavení sítě



Obrázek 4.60. Dialogové okno nastavení, Stránka Sít'

Zde si můžete nastavit proxy server, pokud ho potřebujete, abyste se dostali přes firemní firewall.

Pokud potřebujete mít nastavení proxy serveru pro každé úložiště zvlášť, budete muset použít Subversion soubor `servers` k této konfiguraci. Použitím **Upravit** se tam dostanete přímo. Prohlédněte si [Oblast konfigurace běhu programu](http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html) [http://svnbook.red-bean.com/en/1.5/svn.advanced.confarea.html] pro podrobnosti o tom, jak tento soubor použít.

Můžete také určit, který program by měl TortoiseSVN použít k vytvoření zabezpečené připojení svn+ssh do úložiště. Doporučujeme použít TortoisePlink.exe. Jedná se o verzi oblíbeného programu Plink a je součástí TortoiseSVN, ale je sestavena jako aplikace bez okna, takže se neobjeví okno DOS pokaždé, když se přihlásíte.

Musíte zadat celou cestu ke spustitelnému souboru. Pro TortoisePlink.exe toto je standardní TortoiseSVN bin. Použijte tlačítko **Procházet** pro pomoc při hledání umístění. Upozorňujeme, že pokud cesta obsahuje mezery, musíte ji uzavřít do uvozovek, např.

```
"C:\Program Files\TortoiseSVN\bin\TortoisePlink.exe"
```

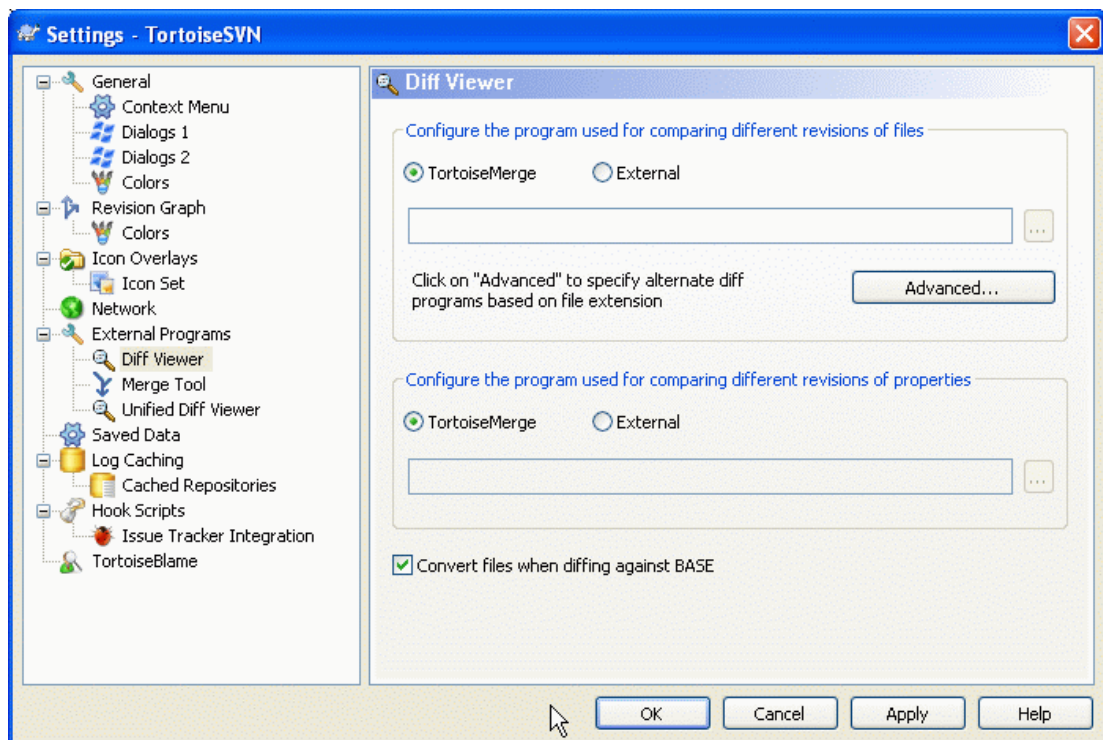
Jeden vedlejší účinek toho, že nemá okno je to, že není nikde místo, kde se můžou zobrazit chybové zprávy, takže pokud se ověření nezdaří, jednoduše dostanete zprávu, něco jako „Unable to write to standard output“. Z tohoto důvodu doporučujeme, abyste nejprve spojení nastavili pomocí standardního Plink. Když všechno funguje, můžete také použít TortoisePlink s přesně stejnými parametry.

TortoisePlink does not have any documentation of its own because it is just a minor variant of Plink. Find out about command line parameters from the [PuTTY website](http://www.chiark.greenend.org.uk/~sgtatham/putty/) [http://www.chiark.greenend.org.uk/~sgtatham/putty/]

Abyste se vyhnuli opakovanému dotazu, můžete také zvážit použití nástroje pro ukládání hesel, jako je Pageant. Ten je také k dispozici ke stažení z webových stránek PuTTY.

Nakonec, nastavení SSH na serveru a v klientech je složitý proces, který je nad rámec tohoto souboru nápovědy. Nicméně, můžete najít návod v nejčastějších otázkách TortoiseSVN v položce [Subversion/TortoiseSVN SSH Jak na to](http://tortoisesvn.net/ssh_howto) [http://tortoisesvn.net/ssh_howto].

4.30.5. Nastavení Externího Programu



Obrázek 4.61. Dialogové okno Nastavení, Stránka Prohlížeč Rozdílů

Zde můžete určit vlastní programy rozdílů/ sloučení, které by TortoiseSVN měl použít. Ve výchozím nastavení je použit TortoiseMerge, který je instalován společně s TortoiseSVN.

Přečtěte si [4.10.5 – „Externí Nástroje Porovnání/Sloučení“](#) pro seznam některých vnějších programů rozdílů/sloučení, které používají lidé s TortoiseSVN.

4.30.5.1. Prohlížeč Rozdílů

Externí program rozdílů lze použít pro srovnání různých revizí souborů. Externí program bude muset získat názvy souborů z příkazového řádku, spolu s jinými parametry příkazového řádku. TortoiseSVN používá substituce parametrů s předponou %. Když se setká s jedním z nich, nahradí jej příslušnou hodnotou. Pořadí parametrů bude záležet na programu Rozdílů, který používáte.

%base

Původní soubor bez změny

%bname

Název okna pro základní soubor

%mine

Váš vlastní soubor, s Vašimi změnami

%yname

Název okna pro Váš soubor

The window titles are not pure filenames. TortoiseSVN treats that as a name to display and creates the names accordingly. So e.g. if you're doing a diff from a file in revision 123 with a file in your working copy, the names will be `filename : revision 123` and `filename : working copy`

For example, with ExamDiff Pro:

```
C:\Path-To\ExamDiff.exe %base %mine --left_display_name:%bname
--right_display_name:%yname
```

or with KDiff3:

```
C:\Path-To\kdiff3.exe %base %mine --L1 %bname --L2 %yname
```

or with WinMerge:

```
C:\Path-To\WinMerge.exe -e -ub -dl %bname -dr %yname %base %mine
```

or with Araxis:

```
C:\Path-To\compare.exe /max /wait /title1:%bname /title2:%yname
%base %mine
```

Pokud používáte vlastnost `svn:keywords` k rozšíření klíčových slov a zejména *revize* v souboru, pak může existovat rozdíl mezi soubory, které jsou čistě z důvodu aktuální hodnoty klíčového slova. Také, pokud používáte `svn:eol-style = native` originální soubor bude mít jenom LF konce řádků, zatímco Váš soubor bude mít CR-LF konce řádků. TortoiseSVN obvykle tyto rozdíly skryje automaticky tím, že nejprve rozebere originální soubor pro rozšíření klíčových slov a konce řádků před operací porovnání. Nicméně, toto může s velkými soubory trvat dlouhou dobu. Pokud je zaškrtnuta volba **Konvertovat soubory při porovnávání** oproti originálu TortoiseSVN vynechá předzpracovávání souborů.

Můžete také zadat jiný nástroj rozdílů pro použití pro vlastnosti Subversion. Protože vlastnosti mají tendenci být krátké jednoduché textové řetězce, můžete použít jednodušší kompaktnější prohlížeč.

Pokud máte nastaven alternativní nástroj rozdílů, můžete vstoupit do TortoiseMerge a nástroje třetí strany z kontextové nabídky. **Kontextové menu** → **Porovnat** použije základní nástroj rozdílů a **Shift+Kontextové menu** → **Porovnat** použije sekundární nástroj rozdílů.

4.30.5.2. Nástroj Sloučení

Externí program sloučení sloužící k řešení konfliktních souborů. Zastupující parametry se používají stejným způsobem jako s programem rozdílů.

`%base`
původní soubor bez Vašich změn nebo změn ostatních

`%bname`
Název okna pro základní soubor

`%mine`
Váš vlastní soubor, se změnami

`%yname`
Název okna pro Váš soubor

`%theirs`
soubor tak jak je v úložišti

`%tname`
Název okna pro soubor v úložišti

`%merged`
konfliktní soubor, výsledek operace sloučení

%mname

Název okna pro nový soubor

Například, s Perforce Merge:

```
C:\Cesta-k\P4Merge.exe %base %theirs %mine %merged
```

nebo s KDiff3:

```
C:\Cesta-k\kdiff3.exe %base %mine %theirs -o %merged
--L1 %bname --L2 %yname --L3 %tname
```

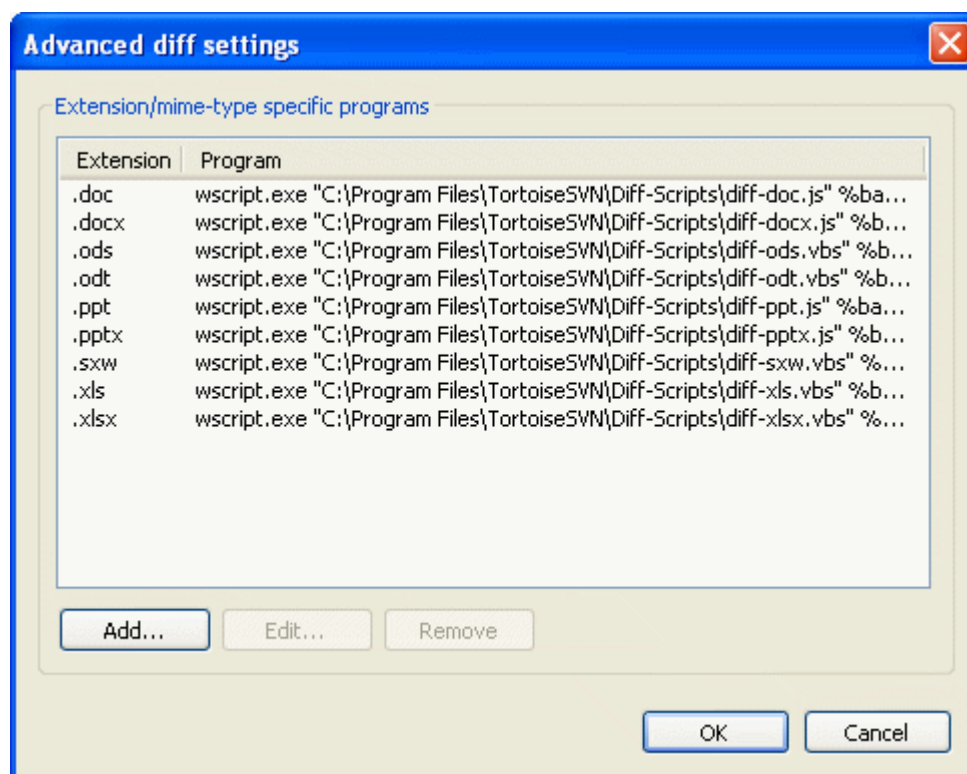
nebo s Araxis:

```
C:\Cesta-ke\compare.exe /max /wait /3 /title1:%tname /title2:%bname
/title3:%yname %theirs %base %mine %merged /a2
```

nebo s WinMerge (verze 2.8 a vyšší):

```
C:\Cesta-k\WinMerge.exe %merged
```

4.30.5.3. Pokročilá Nastavení Rozdílů/Sloučení



Obrázek 4.62. Dialogové okno Nastavení, Pokročilé Dialogové Okno Rozdílů/Sloučení

V pokročilém nastavení můžete určit jiný program pro rozdílů a sloučení pro každou příponu souboru. Například byste mohli přiřadit Photoshop jako program „Rozdílů“ pro soubory .jpg :-). Můžete také přiřadit vlastnost svn:mime-type s programem rozdílů nebo sloučení.

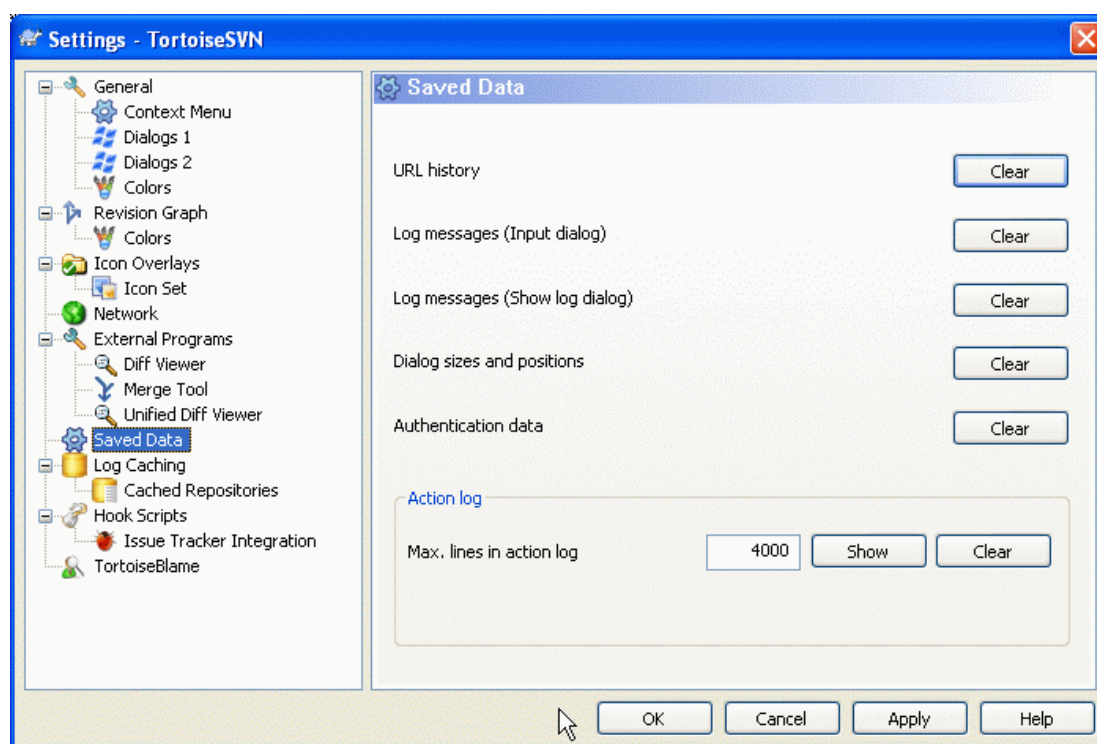
Chcete-li přidružit pomocí přípony souboru, je nutné zadat příponu. Použijte `.bmp` k popsání bitmapových souborů Windows. Chcete-li přidružit pomocí vlastnosti `svn:mime-type`, zadejte MIME typ, včetně lomítka, například `text/xml`.

4.30.5.4. Prohlížeč Sjednoceného Rozdílu

Program pro prohlížení souborů sjednoceného rozdílu (záplatové soubory). Žádné parametry nejsou povinné. Výchozí možnost je zkontrolovat přidružení pro soubory `.diff` a pak pro `.txt`. Pokud nemáte prohlížeč pro soubory `.diff`, budete s největší pravděpodobností použít Poznámkový blok.

Původní program Poznámkový blok Windows se nechová dobře v souborech, které nemají standardní CR-LF zakončení řádků. Protože většina souborů sjednoceného rozdílu mají pouze LF zakončení řádků, nejsou dobře v poznámkovém bloku zobrazeny. Nicméně, můžete si zdarma stáhnout náhradní Poznámkový blok [Notepad2](http://www.flos-freeware.ch/notepad2.html) [http://www.flos-freeware.ch/notepad2.html] který nejen zobrazuje správně konce řádků, ale i barevně kóduje přidávané a smazané řádky.

4.30.6. Nastavení Uložených dat



Obrázek 4.63. Dialogové okno nastavení, Stránka Uložená Data

Pro vaše pohodlí, TortoiseSVN ukládá mnohá nastavení, které používáte, a pamatuje si, kde jste byli v poslední době. Pokud chcete tuto úschovu dat vymazat, můžete tak učinit zde.

Historie URL

Kdykoli získáte pracovní kopii, sloučíte změny nebo použijete průzkumníka úložiště, TortoiseSVN uchovává záznamy o naposledy použitých adresách URL a nabízí je Vám v poli se seznamem. Někdy je tento seznam přeplněn zastaralými URL, takže je užitečné vyprazdňovat ji pravidelně.

Pokud chcete odebrat jednu položku z jednoho z polí se seznamem, můžete toto udělat na místě. Stačí kliknout na šipku k rozvinutí seznamu pole, najedte myší nad položku, kterou chcete odstranit a stiskněte **Shift+Del**.

Zprávy záznamu (vstupní dialogové okno)

TortoiseSVN uchovává nedávné zprávy záznamu odevzdání, které jste zadali. Tyto jsou uloženy pro každé úložiště zvlášť, takže pokud máte přístup k mnoha úložištím tento seznam se může poměrně hodně rozrůst.

Zprávy záznamu (Zobrazit dialogové okno záznamu)

TortoiseSVN uchovává zprávy záznamu načtené dialogovým oknem záznamu, pro ušetření času při příštím zobrazení záznamu. Pokud někdo jiný upraví zprávu záznamu a Vy už máte tuto zprávu uchovanou, nevidíte změnu, dokud nevymažete úschovu. Uchovávání zpráv záznamu se povoluje v kartě Uchovávání záznamu.

Velikost a umístění dialogových oken

Mnoho dialogových oken si pamatuje svoji velikost a umístění, které jste naposledy použili.

Ověřovací údaje

Při ověření se serverem Subversion, jsou uživatelské jméno a heslo uložené místně, takže je nemusíte pořád zadávat. Možná budete chtít tyto data z bezpečnostních důvodů odstranit, nebo protože chcete mít přístup do úložiště pod jiným uživatelským jménem ... ví Marek, že používáte jeho PC?

Pokud chcete vymazat ověřovací údaje pro jeden konkrétní server, přečtěte si [4.1.5 – „Ověření“](#) návod, jak najít uschovaná data.

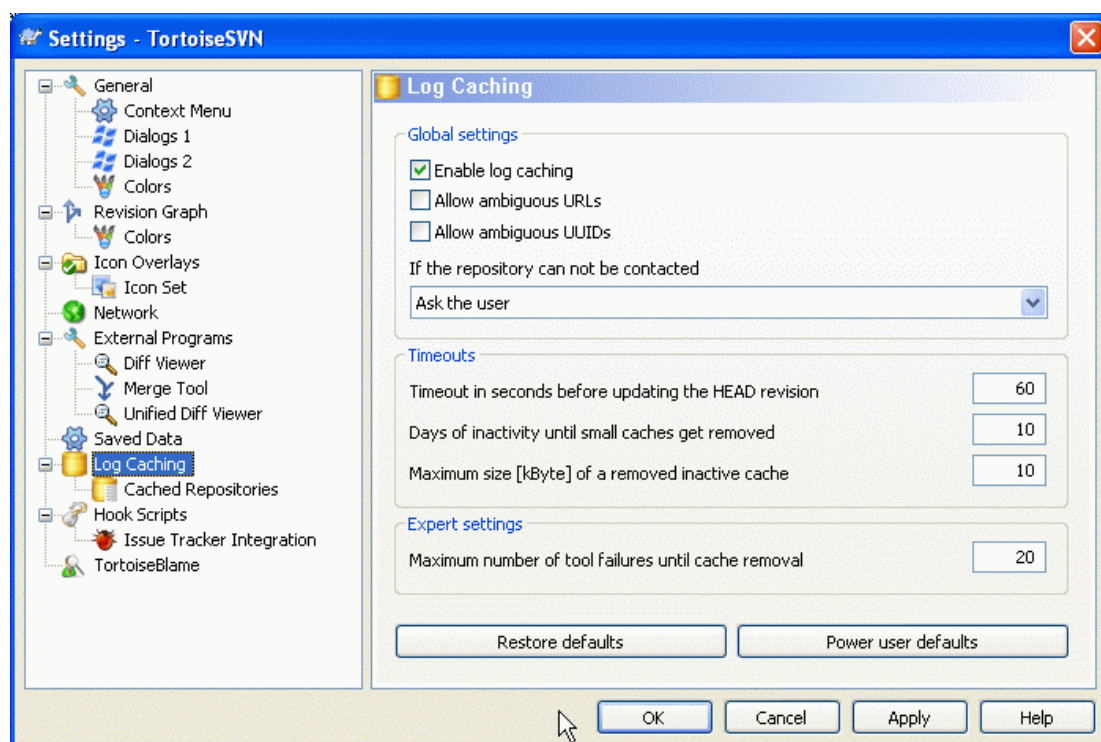
Záznam činnosti

TortoiseSVN uchovává záznam o všem, co bylo napsané v jeho dialogovém okně průběhu. To může být užitečné, pokud například chcete zkontrolovat, co se stalo v nedávném příkazu aktualizace.

Soubor záznamů je omezen v délce, a když se stane příliš velkým, nejstarší obsah je zahozen. Ve výchozím nastavení je uchováno 4000 řádků, ale toto číslo můžete upravit.

Odtud si můžete prohlédnout obsah souboru záznamů a také vymazat.

4.30.7. Uchovávání záznamu



Obrázek 4.64. Dialogové okno Nastavení, Stránka Úschova Záznamu

Toto dialogové okno Vám umožňuje nastavit funkci úschovy záznamu TortoiseSVN, což udrží místní kopii zpráv záznamu a změněné cesty, aby se zabránilo časově náročnému stahování ze serveru. Pomocí

úschovy záznamu můžete dramaticky urychlit dialogové okno záznamu a graf revizí. Další užitečnou vlastností je, že zprávy záznamu jsou stále přístupné i offline.

Povolit uchování záznamu

Povoluje uchování záznamu, kdykoli jsou data záznamu požadována. Pokud je zaškrtnuto, budou data načtena z úschovy je-li k dispozici a jakékoliv zprávy, které nejsou uloženy, budou stáhnuty ze serveru a přidány do úschovy.

Je-li uchování zakázáno, budou data vždy získávána přímo ze serveru a nejsou místně ukládána.

Povolit nejednoznačné URL

Občas se budete muset připojit k serveru, který používá stejnou adresu URL pro všechny úložiště. Starší verze svnbridge by toto provedly. Pokud potřebujete přístup k těmto úložištím, budete muset tuto možnost zaškrtnout. Pokud nechcete, nechte odškrtnuté ke zlepšení výkonnosti.

Povolit nejednoznačný UUID

Některé hostingové služby dávají všem svým úložištím stejné UUID. Toto můžete dokonce udělat sami pomocí kopírování adresáře úložiště k vytvoření nového. Ze všelijakých důvodů je tohle špatný nápad - UUID by měl být *jedinečný*. Nicméně, úschova záznamů stále bude v této situaci fungovat, pokud toto políčko zaškrtnete. Pokud toto nepotřebujete, nechte ho odškrtnuté ke zlepšení výkonnosti.

Když není možno kontaktovat úložiště

Pokud pracujete v režimu offline, nebo je-li server úložiště nedostupný, úschova záznamu může být stále používána k dodání zpráv záznamu již uložených. Samozřejmě úschova nemusí být aktuální, proto existují možnosti, které Vám umožní vybrat, zda tato funkci by měla být použita.

Když jsou data záznamu brána z úschovy, aniž by byl zkontaktován server, dialogové okno používající tyto zprávy zobrazí stav offline v jeho záhlaví.

Časový limit před aktualizací HEAD revize

Když spustíte dialogové okno záznamu, budete normálně chtít kontaktovat server pro kontrolu veškerých nových zpráv záznamu. Pokud je časový limit nastaven zde nenulový, pak server bude kontaktován pouze při překročení časového limitu od posledního kontaktu. To může snížit komunikaci mezi serverem a klientem, pokud často otevíráte dialogové okno záznamu a server je pomalý, ale uvedené údaje nemusí být zcela aktuální. Pokud chcete tuto funkci použít, doporučujeme použít hodnotu 300 (5 minut) jako kompromis.

Počet dnů neaktivity do vymazání malých úschoven

Procházíte-li spousty úložišť budete hromadit velké množství uchovaných záznamů. Pokud je aktivně nepoužíváte, úschova záznamů příliš neporoste, takže je TortoiseSVN ve výchozím nastavení vyčistí po uplynutí nastaveného času. Pomocí této položky kontrolujete čištění úschovy.

Maximální velikost odstraněné neaktivní úschovny

Větší úschovy jsou horší na znovunabytí a tak TortoiseSVN čistí pouze malé úschovny. Touto hodnotou doladíte práh.

Maximální počet chyb nástroje před odebrání úschovny

Občas se něco pokazí s ukládáním a způsobí havárii. Pokud se tak stane, úschova je obvykle automaticky odstraněna, aby se zabránilo opakování tohoto problému. Pokud používáte méně stabilní noční build můžete zvolit si úschovu i tak nechat.

4.30.7.1. Uchovaná úložiště

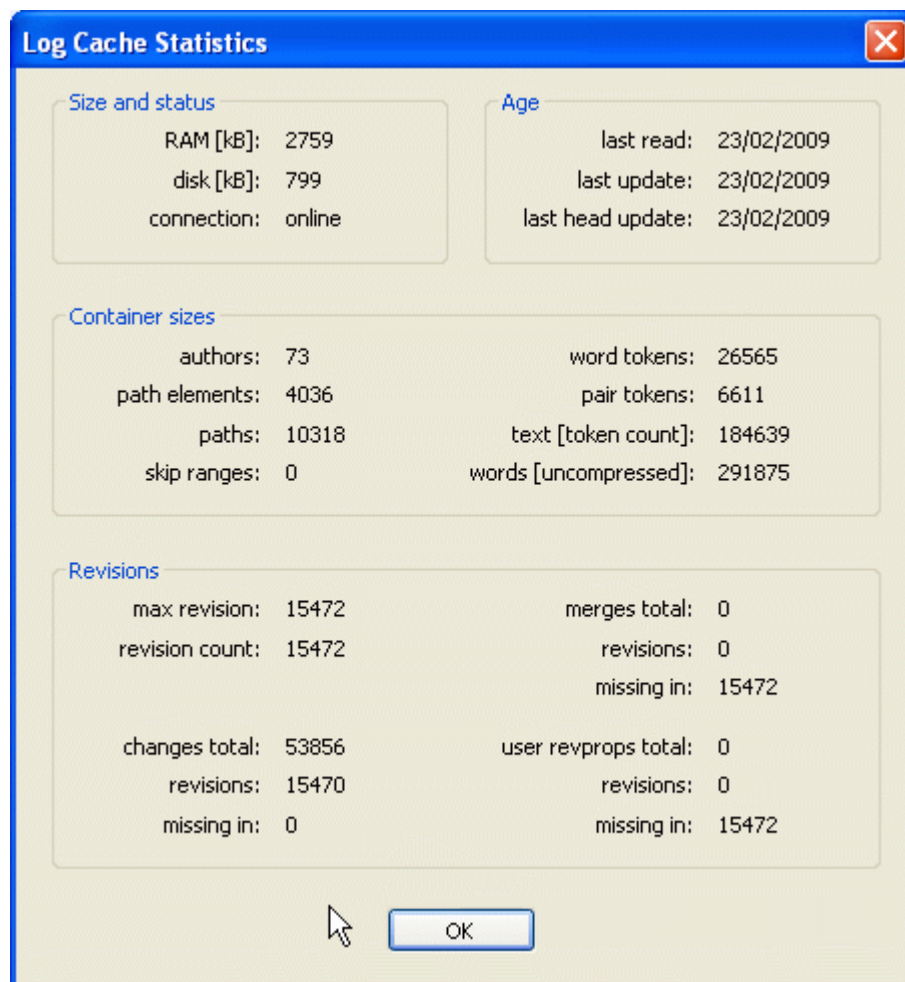
Na této stránce si můžete prohlédnout seznam úložišť, které jsou uloženy místně a prostor využitý pro úschovu. Pokud vyberete jedno z úložišť, pak můžete použít tlačítka pod ním.

Klikněte na **Aktualizovat** pro úplnou obnovu úschovy a vyplnit všechny díry. Pro velké úložiště toto může být velmi časově náročné, ale užitečné, pokud se chystáte přejít do režimu offline a chcete nejlepší dostupnou úschovu.

Klikněte na tlačítko **Exportovat** pro export celé úschovy jako sadu CSV souborů. To by mohlo být užitečné, pokud chcete zpracovat záznam dat pomocí externího programu, ačkoli to je hlavně užitečné pro vývojáře.

Klikněte na **Smazat** ke smazání všech uložených dat pro vybrané úložiště. Toto nezakáže úschovu úložiště, takže příště, když požádat o záznam dat, bude vytvořena nová úschova.

4.30.7.2. Statistika Úschovy Záznamu



Obrázek 4.65. Dialogové okno Nastavení, Statistika Úschovy Záznamu

Klikněte na tlačítko **Podrobnosti** pro zobrazení podrobné statistiky pro konkrétní úschovy. Mnoho z polí zde uvedené jsou hlavně v zájmu vývojářů TortoiseSVN, takže nejsou všechny podrobně popsány.

RAM

Množství paměti potřebné k provozu této úschovy.

Disk

Množství místa na disku používané úschovou. Data jsou komprimována, takže využití disku je obecně poměrně skromné.

Spojení

Ukazuje, zda úložiště bylo k dispozici v poslední době, kdy byla úschova použita.

Poslední aktualizace

V poslední době, kdy byl obsah úschovy změněn.

Poslední HEAD aktualizace

Kdy naposledy jsme požádali o revizi HEAD ze serveru.

Autoři

Počet různých autorů se zprávami zaznamenaný v úschově.

Cesty

Počet uvedených cest, jako byste vidět pomocí `svn log -v`.

Vynechat rozsahy

Počet rozsahů revizí, které jsme nenačetli, prostě proto, že o ně nebylo požádáno. Jedná se o měřítko počtu děr v úschově.

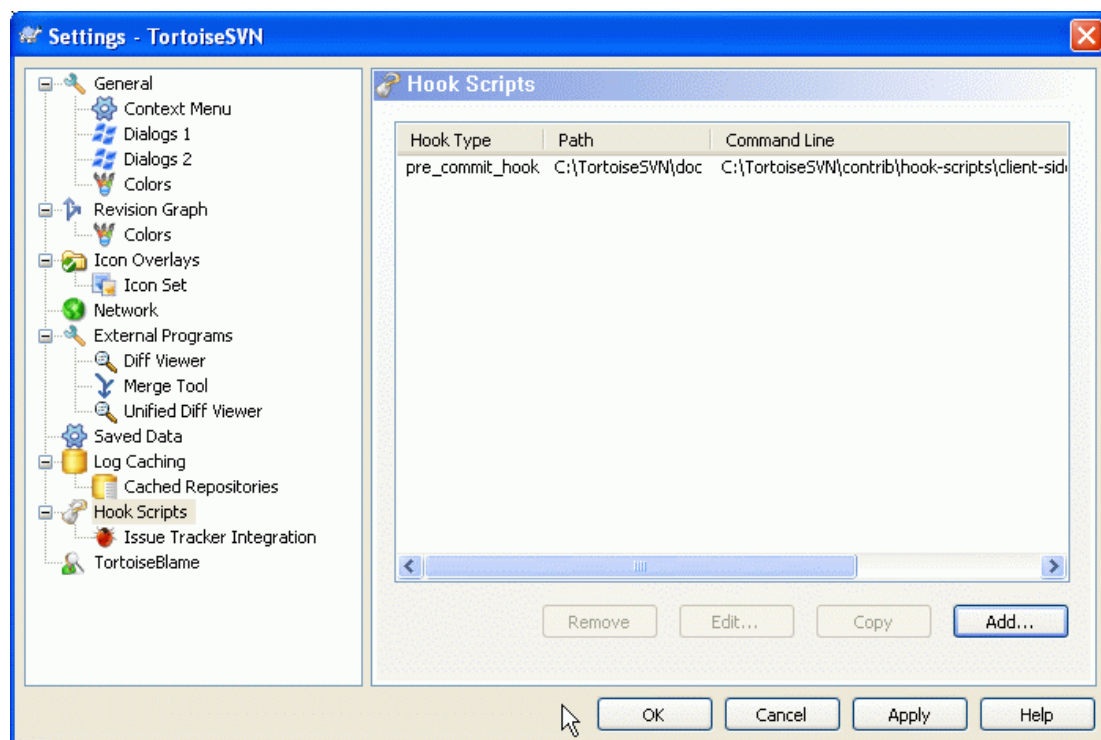
Nejvyšší revize

Nejvyšší číslo revize uložené v úschově.

Počet revizí

Počet revizí uložených v úschově. To je další měřítko úplnosti úschovy.

4.30.8. Hákové Skripty na Straně Klienta

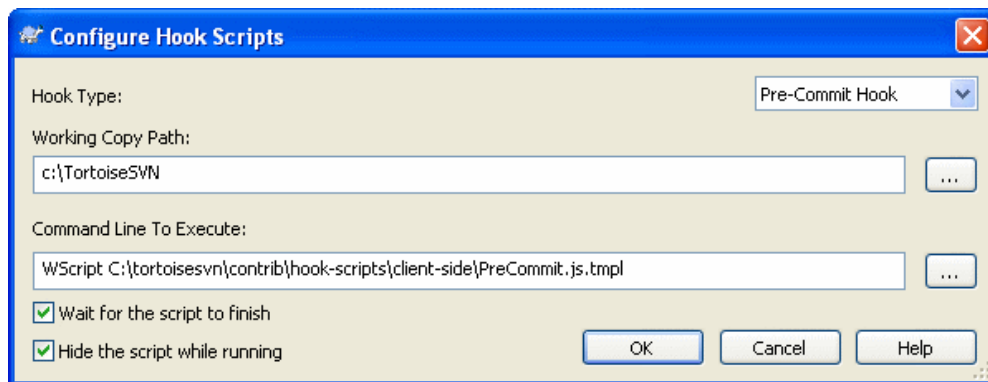


Obrázek 4.66. Dialogové okno nastavení, Stránka Hákové Skripty

Toto dialogové okno Vám umožňuje nastavit hákové skripty, které budou automaticky provedeny, při některých činnostech Subversion. Na rozdíl od skriptů probíraných v 3.3 – „Hook skripty na straně serveru“, jsou tyto skripty spouštěny místně na straně klienta.

Jedno použití pro takové háky může být volat program, jako je `SubWCRev.exe` pro aktualizaci čísel verzí po odevzdání a možná spustit znovu sestavení.

Z různých bezpečnostních a uskutečňujících důvodů, jsou hákové skripty určované místně na počítači, spíše než jako vlastnosti projektu. Můžete určit, co se stane, bez ohledu na to, co někdo jiný odevzdává do úložiště. Samozřejmě si můžete vždy vybrat volat skript, který je sám pod správou verzí.



Obrázek 4.67. Dialogové okno Nastavení, Konfigurace Hákových Skriptů

Chcete-li přidat nový hákový skript, stačí kliknout na **Přidat** a vyplnit údaje.

V současné době existuje šest typů hákového skriptu k dispozici

Zahájení odevzdání

Volán před zobrazením dialogového okna odevzdání. Toto byste mohli chtít použít, pokud hák upravuje soubor s verzí a ovlivňuje seznam souborů, které musí být odevzdány a/nebo zprávu odevzdání. Nicméně byste měli vzít na vědomí, že hák je volán v rané fázi, úplný seznam objektů vybraných pro odevzdání není k dispozici.

Před odevzdáním

Volán po kliknutí na **OK** v dialogovém okně a než začne vlastní odevzdání. Tento hák má seznam toho, co bude přesně odevzdáno.

Po odevzdání

Volán po ukončení odevzdání (ať už bylo úspěšné či nikoli).

Zahájení aktualizace

Volán před zobrazením dialogového okna Aktualizovat na revizi.

Před aktualizací

Called before the actual Subversion update begins.

Po aktualizaci

Called after the update finishes (whether successful or not).

Hák je určen pro konkrétní cestu pracovní kopie. Stačí pouze zadat cestu na nejvyšší úrovni; pokud budete provádět operace v podadresáři, bude TortoiseSVN automaticky hledat vzhůru pro odpovídající cestu.

Next you must specify the command line to execute, starting with the path to the hook script or executable. This could be a batch file, an executable file or any other file which has a valid windows file association, eg. a perl script.

Příkazový řádek obsahuje několik parametrů, které jsou vyplněny TortoiseSVN. Předané parametry závisí na háku, který je volán. Každý hák má své vlastní parametry, které jsou předávány v následujícím pořadí:

Zahájení odevzdání

PATHMESSAGEFILECWD

Před odevzdáním

PATHDEPTHMESSAGEFILECWD

Po odevzdání

PATHDEPTHMESSAGEFILEREVISIONERRORCWD

Zahájení aktualizace

PATHCWD

Před aktualizací

PATHDEPTHREVISIONCWD

Po aktualizaci

PATHDEPTHREVISIONERRORCWD

Význam každého z těchto parametrů je popsán zde:

PATH

Cesta k dočasnému souboru, který obsahuje všechny cesty, pro které byla operace spuštěna. Každá cesta je v dočasném souboru na samostatném řádku.

DEPTH

Hloubka, s níž je odevzdání/aktualizace provedena.

Možné hodnoty jsou:

-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

Cesta k souboru, obsahující zprávu záznamu o odevzdání. Soubor obsahuje text s UTF-8 kódováním. Po úspěšném spuštění háku zahájení odevzdání, je zpráva záznamu přečtena zpět, což dává háku možnost ji změnit.

REVISION

Revize úložiště, na kterou by se mělo aktualizovat, nebo po dokončení odevzdání.

ERROR

Cesta k souboru obsahující chybovou zprávu. Pokud nedošlo k chybě, bude soubor prázdný.

CWD

Aktuální pracovní adresář, v níž je skript spuštěn. Toto je nastaveno na společný kořenový adresář všech dotčených cest.

Nezapomeňte, že i když jsme tyto parametry pojmenovali pro naše pohodlí, nemusíte odkazovat na tyto názvy v nastavení háku. Všechny parametry uvedené pro daný hák jsou vždy předány, ať chcete, nebo ne ;-)

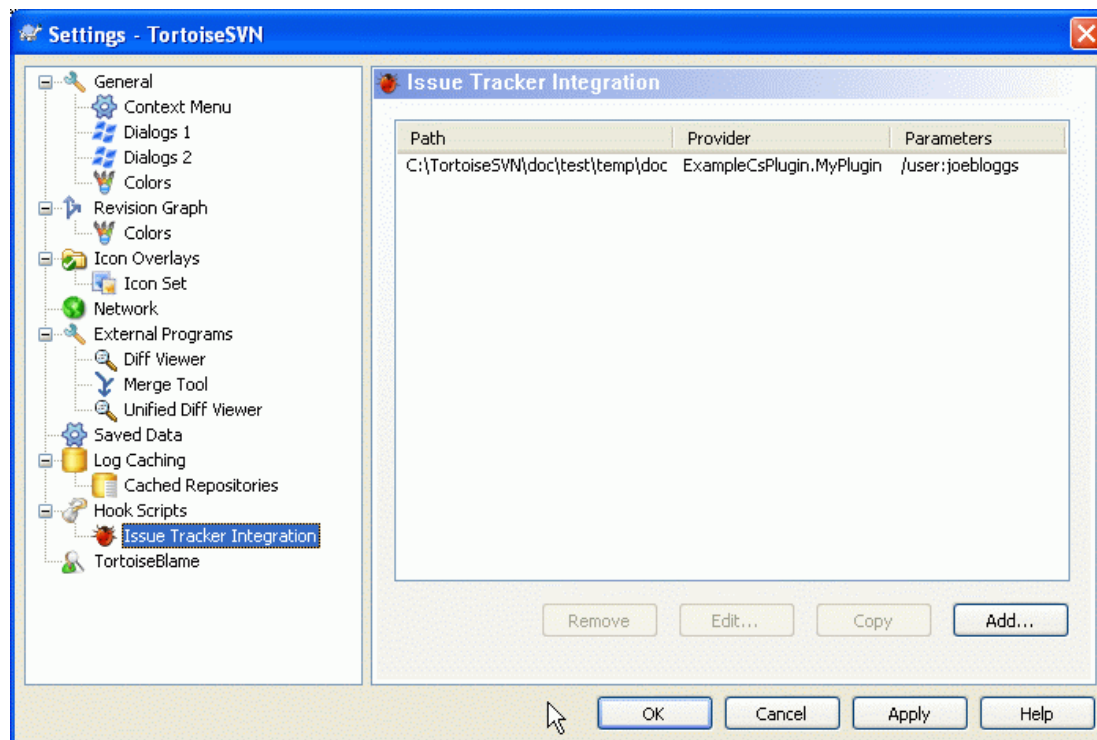
Pokud chcete, aby byla operace Subversion pozdržena, dokud hák nebyl dokončen zaškrtněte **Počkat, až skript skončí**.

Za normálních okolností budete chtít schovat ošklivá okna DOS, když je skript spuštěn, takže **Skrýt skript při běhu** je ve výchozím nastavení zaškrtnuto.

Sample client hook scripts can be found in the `contrib` folder in the *TortoiseSVN repository* [<http://tortoisesvn.googlecode.com/svn/trunk/contrib/hook-scripts>]. (3 – „TortoiseSVN je zadarmo!“ explains how to access the repository).

4.30.8.1. Integrace se Systémem pro Sledování Problémů

TortoiseSVN může použít zásuvný modul COM pro dotazy systémům pro sledování problémů v dialogovém okně odevzdání. Použití těchto zásuvných modulů je popsáno v 4.28.2 – „Získávání informací od Systému pro Sledování Problémů“. Pokud Vám Váš správce systému zásuvný modul poskytl, který jste už nainstalovali a zaregistrovali, toto je místo, kde určit, jak bude zaveden do Vaší pracovní kopie.



Obrázek 4.68. Dialogové okno nastavení, Stránka Integrace se Systémem pro Sledování Problémů

Klikněte na Přidat... pro použití zásuvného modulu s konkrétní pracovní kopií. Zde můžete zadat cestu pracovní kopie, zvolit, který zásuvný modul použít z rozbalovacího seznamu všech registrovaných zásuvných modulů pro systém pro sledování problémů, a všechny parametry k předání. Parametry budou specifické pro zásuvný modul, ale může také obsahovat Vaše uživatelské jméno v systému pro sledování problémů, aby zásuvný modul mohl načíst problémy, které jsou Vám přiřazeny.

If you want all users to use the same COM plugin for your project, you can specify the plugin also with the properties `bugtraq:provideruuid` and `bugtraq:providerparams`.

`bugtraq:provideruuid`

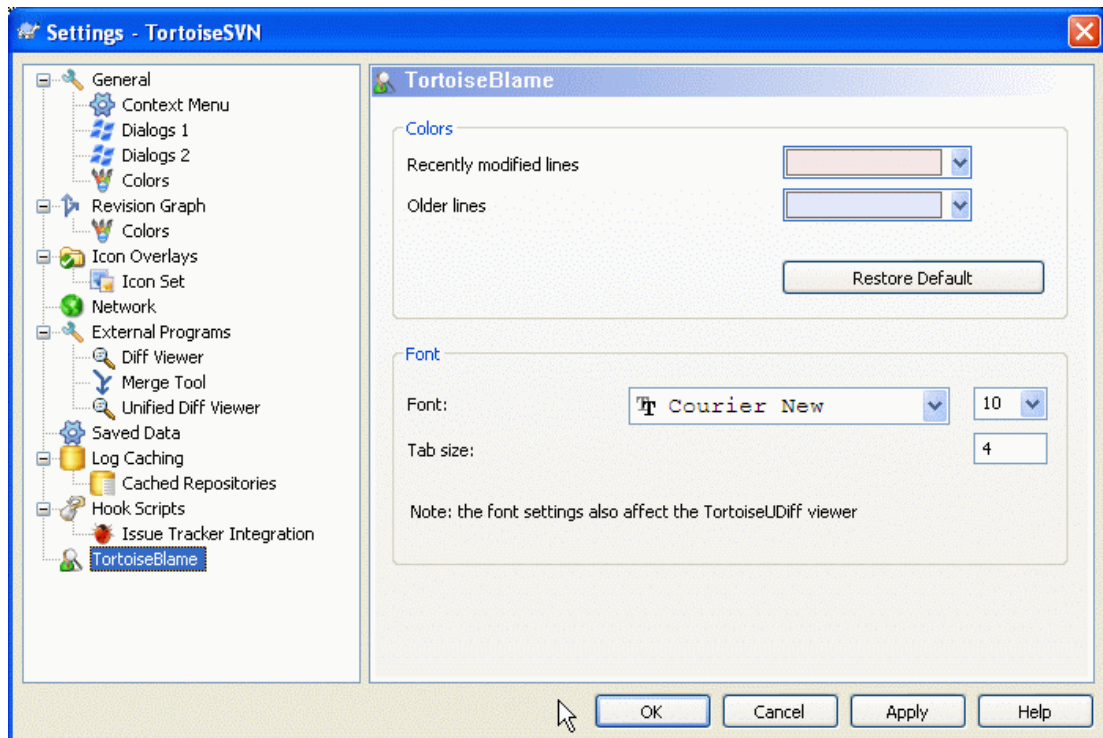
This property specifies the COM UUID of the `IBugtraqProvider`, for example `{91974081-2DC7-4FB1-B3BE-0DE1C8D6CE4E}`. (this example is the UUID of the *Gurtle bugtraq provider* [<http://code.google.com/p/gurtle/>], which is a provider for the *Google Code* [<http://code.google.com/hosting/>] issue tracker).

`bugtraq:providerparams`

Tato vlastnost určuje parametry předávané `IBugtraqProvider`.

Please check the documentation of your `IBugtraqProvider` plugin to find out what to specify in these two properties.

4.30.9. Nastavení TortoiseBlame



Obrázek 4.69. Dialogové okno nastavení, Stránka TortoiseBlame

Nastavení používané TortoiseBlame jsou řízeny z hlavní kontextové nabídky, ne přímo ze samotného TortoiseBlame.

Barvy

TortoiseBlame může použít barvu pozadí pro označení věku řádků v souboru. Nastavte koncové body zadáním barev pro nejmladší a nejstarší revize a TortoiseBlame použije lineární interpolace mezi těmito dvěma barvami podle revize úložiště uvedená pro každý řádek.

Typ písma

Můžete si vybrat písmo použité pro zobrazení textu a velikost bodu, která se má použít. To platí jak pro obsah souboru, tak i pro informace o autorovi a revizi uvedené v levém panelu.

Tabulátory

Určuje, kolik prostoru využít pro rozpětí, když je znak tabulátoru nalezen v obsahu souboru.

4.30.10. Registry Settings

A few infrequently used settings are available only by editing the registry directly. It goes without saying that you should only edit registry values if you know what you are doing.

Konfigurace

You can specify a different location for the Subversion configuration file using registry location HKCU\Software\TortoiseSVN\ConfigDir. This will affect all TortoiseSVN operations.

Cache tray icon

To add a cache tray icon for the TSVNCache program, create a DWORD key with a value of 1 at HKCU\Software\TortoiseSVN\CacheTrayIcon. This is really only useful for developers as it allows you to terminate the program gracefully.

Debug

To show the command line parameters passed from the shell extension to TortoiseProc.exe create a DWORD key with a value of 1 at HKCU\Software\TortoiseSVN\Debug.

Context Menu Icons

This can be useful if you use something other than the windows explorer or if you get problems with the context menu displaying correctly. create a DWORD key with a value of 0 at HKCU\Software\TortoiseSVN\ShowContextMenuIcons if you don't want TortoiseSVN to not show icons for the shell context menu items. Set this value to 1 to show the icons again.

Block Overlay Status

If you don't want the explorer to update the status overlays while another TortoiseSVN command is running (e.g. Update, Commit, ...) then create a DWORD key with a value of 1 at HKCU\Software\TortoiseSVN\BlockStatus.

Update Check URL

HKCU\Software\TortoiseSVN\UpdateCheckURL contains the URL from which TortoiseSVN tries to download a text file to find out if there are updates available. You can also set this under HKLM instead of HKCU if you want, but HKCU overwrites the setting in HKLM. This might be useful for company admins who don't want their users to update TortoiseSVN until they approve it.

Filenames without extensions in auto-completion list

The auto-completion list shown in the commit message editor displays the names of files listed for commit. To also include these names with extensions removed, create a DWORD key with a value of 1 at HKCU\Software\TortoiseSVN\AutocompleteRemovesExtensions.

Explorer columns everywhere

The extra columns the TortoiseSVN adds to the details view in Windows Explorer are normally only active in a working copy. If you want those to be accessible everywhere, not just in working copies, create a DWORD key with a value of 1 at HKCU\Software\TortoiseSVN\ColumnsEveryWhere.

Merge log separator

When you merge revisions from another branch, and merge tracking information is available, the log messages from the revisions you merge will be collected to make up a commit log message. A pre-defined string is used to separate the individual log messages of the merged revisions. If you prefer, you can create a SZ key at HKCU\Software\TortoiseSVN\MergeLogSeparator containing a separator string of your choice.

Always blame changes with TortoiseMerge

TortoiseSVN allows you to assign external diff viewer. Most such viewers, however, are not suited for change blaming (4.23.2 – „Obvinit Rozdíly“), so you might wish to fall back to TortoiseMerge in this case. To do so, create a DWORD key with a value of 1 at HKCU\Software\TortoiseSVN\DiffBlamesWithTortoiseMerge.

Current revision highlighting for folders in log dialog

The log dialog highlights the current working copy revision when the log is shown for a file. To do the same thing for a folder requires a working copy crawl, which is the default action, but it can be a slow operation for large working copies. If you want to change the operation of this feature you must create a DWORD registry key at HKCU\Software\TortoiseSVN\RecursiveLogRev. A value of 0 disables the feature (no highlighting for folders), a value of 1 (default) will fetch the status recursively (find the highest revision in the working copy tree), and a value of 2 will check the revision of the selected folder itself, but will not check any child items.

Make checkout fail if an item of the same name exists

By default, if you checkout a working copy over an existing unversioned folder structure, as you might do after import, then any existing which differ from the repository content will be left unchanged and marked as modified. When you come to commit, it is your local copy which will then be sent back to the repository. Some people would prefer the checkout to fail if the existing content differs, so that if two people add the same file the second person's version does

not overwrite the original version by mistake. If you want to force checkouts to fail in this instance you must create a DWORD registry key with value 0 at HKCU\Software\TortoiseSVN\AllowUnversionedObstruction.

4.30.11. Pracovní adresáře Subversion

Když je použit VS.NET 2003 pro webové projekty, nepracuje s adresáři `.svn`, které Subversion používá k ukládání svých vnitřních informací. To není chyba v Subversion. Chyba je v VS.NET 2003 a Rozšíření FrontPage, které používá.

Note that the bug is fixed in VS2005 and later versions.

Od verze 1.3.0 Subversion a TortoiseSVN, můžete nastavit proměnnou prostředí `SVN_ASP_DOT_NET_HACK`. Pokud je tato proměnná nastavena, pak Subversion bude používat adresáře `_svn` místo `.svn`. Musíte restartovat rozhraní, aby se tato proměnná prostředí projevila. Normálně to znamená, restartovat počítač. Aby to bylo jednodušší, nyní toto můžete udělat ze stránky obecného nastavení použitím jednoduchého políčka - viz [4.30.1 – „Obecná nastavení“](#).

Pro více informací a další způsoby, jak se v první řadě tomuto problému vyhnout se podívejte na článek o této problematice v našich [Nejčastějších Otázkách](#) [<http://tortoisesvn.net/aspdotnethack>].

4.31. Poslední krok

Přispějte!

I když TortoiseSVN a TortoiseMerge jsou zdarma, můžete podpořit vývojáře zasláním záplat a hrát aktivní roli ve vývoji. Můžete také pomoci nás povzbudit během nekonečných hodin, které trávíme před našimi počítači.

While working on TortoiseSVN we love to listen to music. And since we spend many hours on the project we need a *lot* of music. Therefore we have set up some wish-lists with our favourite music CDs and DVDs: <http://tortoisesvn.tigris.org/donate.html> Please also have a look at the list of people who contributed to the project by sending in patches or translations.

Kapitola 5. Program SubWCRev

SubWCRev je konzolový program ve Windows, který lze použít ke čtení stavu pracovní kopie Subversion a případně provést nahrazení klíčového slova v šablonovém souboru. Toto se často používá jako součást procesu sestavení jako prostředek obsahující informace o pracovní kopii do objektu, který sestavujete. Typicky to může být použito k obsazení čísla revize do rámečku „O programu“.

5.1. Příkazový řádek SubWCRev

SubWCRev reads the Subversion status of all files in a working copy, excluding externals by default. It records the highest commit revision number found, and the commit timestamp of that revision. It also records whether there are local modifications in the working copy, or mixed update revisions. The revision number, update revision range and modification status are displayed on stdout.

SubWCRev.exe je volán z příkazové řádky nebo skriptu a je ovládán pomocí parametrů příkazového řádku.

```
SubWCRev WorkingCopyPath [SrcVersionFile DstVersionFile] [-nmdfe]
```

WorkingCopyPath je cesta k pracovní kopii ke kontrole. SubWCRev lze použít pouze na pracovní kopie, ne přímo na úložiště. Cesta může být absolutní nebo relativní k aktuálnímu pracovní adresář.

Pokud chcete, aby SubWCRev provedl nahrazení klíčového slova, takže pole jako revize úložiště a URL jsou uloženy do textového souboru, musíte zadat šablonový soubor SrcVersionFile a výstupní soubor DstVersionFile který obsahuje nahrazovanou verzi šablony.

There are a number of optional switches which affect the way SubWCRev works. If you use more than one, they must be specified as a single group, eg. -nm, not -n -m.

Překlopit	Popis
-n	Pokud je tento přepínač zadán, SubWCRev skončí s ERRORLEVEL 7 v případě, že pracovní kopie obsahuje místní změny. Toto může být použito k zabránění sestavení pokud jsou přítomny neodevzdané změny.
-m	Pokud je tento přepínač zadán, SubWCRev skončí s ERRORLEVEL 8 v případě, že pracovní kopie obsahuje smíšené revize. Toto může být použito k zabránění sestavení s částečně aktualizovanou pracovní kopií.
-d	Pokud je tento přepínač zadán, SubWCRev skončí s ERRORLEVEL 9 v případě, že cílový soubor již existuje.
-f	Pokud je tento přepínač zadán, zahrne SubWCRev poslední změněnou revizi adresáře. Výchozí chování je používat pouze soubory při získávání čísel revizí.
-e	Pokud je tento přepínač zadán, bude SubWCRev zkoumat adresáře, které jsou součástí svn:externals, ale pouze v případě, že jsou ze stejného úložiště. Výchozí chování je externí ignorovat.
-x	Je-li tento přepínač zadán, SubWCRev vypíše čísla revizí v šestnáctkové soustavě.
-X	Je-li tento přepínač zadán, SubWCRev vypíše čísla revizí v šestnáctkové soustavě a přidá '0X' na začátek

Tabulka 5.1. Seznam dostupných přepínačů v příkazovém řádku

5.2. Nahrazení Klíčového Slova

Pokud jsou zadány zdrojové a cílové soubory, SubWCRev zkopíruje zdroj do cíle, nahrazujíc klíčová slova takto:

Klíčové slovo	Popis
\$WCREV\$	Nahrazeno nejvyšší odevzdanou revizí pracovní kopie.
\$WCDATES\$	Replaced with the commit date/time of the highest commit revision. By default, international format is used: <code>rrrr-mm-dd hh:mm:ss</code> . Alternatively, you can specify a custom format which will be used with <code>strftime()</code> , for example: <code>\$WCDATE=%a %b %d %I:%M:%S %p %Z</code> . For a list of available formatting characters, look at the odkaz on-line [http://www.cppreference.com/stddate/strftime.html].
\$WCNOW\$	Nahrazeno aktuálním systémovým datem/časem. Toto může být použito k označení času sestavení. Formátování času lze také použít, jak je popsáno v <code>\$WCDATES\$</code> .
\$WCRANGES\$	Replaced with the update revision range in the working copy. If the working copy is in a consistent state, this will be a single revision. If the working copy contains mixed revisions, either due to being out of date, or due to a deliberate update-to-revision, then the range will be shown in the form <code>100:200</code>
\$WCMIXED\$	<code>\$WCMIXED?TText:FText\$</code> je nahrazen <code>TText</code> pokud existují smíšené aktualizace revizí, nebo <code>FText</code> pokud ne.
\$WCMODS\$	<code>\$WCMODS?TText:FText\$</code> je nahrazeno <code>TText</code> v případě, že existují místní úpravy, nebo <code>FText</code> pokud ne.
\$WCURL\$	Nahrazeno URL adresou úložiště cesty pracovní kopie předaná SubWCRev.
\$WCINSVN\$	<code>\$WCINSVN?TText:FText\$</code> je nahrazeno <code>TText</code> pokud má položka verzi, nebo <code>FText</code> pokud ne.
\$WCNEEDSLOCK\$	<code>\$WCNEEDSLOCK?TText:FText\$</code> je nahrazeno <code>TText</code> v případě, že položka má nastavenou vlastnost <code>svn:needs-lock</code> , nebo <code>FText</code> pokud ne.
\$WCISLOCKED\$	<code>\$WCISLOCKED?TText:FText\$</code> je nahrazeno <code>TText</code> v případě, že záznam je uzamčen, nebo <code>FText</code> pokud ne.
\$WCLOCKDATE\$	Nahrazeno datem zamknutí. Formátování času lze použít, jak je popsáno v <code>\$WCDATES\$</code> .
\$WCLOCKOWNER\$	Nahrazeno jménem majitele zámku.
\$WCLOCKCOMMENT\$	Nahrazeno komentářem k zámku.

Tabulka 5.2. Seznam dostupných přepínačů v příkazovém řádku



Tip

Některá z těchto klíčových slov platí pro jednotlivé soubory, nikoli na celou pracovní kopii, takže má smysl je použít když SubWCRev je zavolán pro skenování jednoho souboru. To se týká `$WCINSVN$`, `$WCNEEDSLOCK$`, `$WCISLOCKED$`, `$WCLOCKDATE$`, `$WCLOCKOWNER$` a `$WCLOCKCOMMENT$`.

5.3. Příklad klíčového slova

Příklad uvedený níže ukazuje, jak jsou klíčová slova v šablonovém souboru nahrazena ve výstupním souboru.

```
// Testovací soubor: testsoubor.tmpl
```



```
char *Revize = "$WCREV$";
char *Změna = "$WCMODS?Změněná:Nezměněná$";
char *Datum = "$WCDATE$";
char *Rozsah = "$WCRANGE$";
char *Smíšená = "$WCMIXED?Smíšené revize pracovní kopie:Nesmíšené$";
char *URL = "$WCURL$";

#if $WCMODS?1:0$
#error Zdroj se změnil
#endif

// Konec souboru
```

Po spuštění SubWCRev.exe cesta\k\pracovníkopii testsoubor.tmp1 testspubor.txt, výstupní soubor testsoubor.txt by vypadal takto:

```
// Testovací soubor pro SubWCRev: testsoubor.txt

char *Revize = "3701";
char *Změna = "Změněno";
char *Datum = "2005/06/15 11:15:12";
char *Rozsah = "3699:3701";
char *Smíšená = "Mixed revision WC";
char *URL = "http://domena.projektu.cz/svn/kmen/zdr";

#if 1
#error Zdroj se změnil
#endif

// Konec souboru
```



Tip

Takovýto soubor bude zahrnut do sestavení, takže byste očekávat, že bude mít verzi. Ujistěte se, že šablonový soubor má verzi a ne vytvořený soubor, jinak pokaždé, když znovu vytvoříte soubor s verzí budete muset změny odevzdat, což znamená, že souboru s verzí je třeba aktualizovat.

5.4. Rozhraní COM

Pokud potřebujete přístup k informacím revizí Subversion z jiných programů, můžete použít COM rozhraní SubWCRev. Objekt vytvoření je SubWCRev.object a tyto metody jsou podporovány:

Metoda	Popis
.GetWCInfo	Tato metoda prochází pracovní kopii a shromažďuje revize. Přirozeně je třeba toto zavolat předtím, než budete moci přistupovat k informacím pomocí zbývajících metod. První parametr je cesta. Druhý parametr by měl být true, pokud chcete zahrnout revize adresářů. Obdoba přepínače -f příkazové řádky. Třetí parametr by měl být true, pokud chcete zahrnout svn:externals. Obdoba přepínače -e příkazové řádky.
.Revision	The highest commit revision in the working copy. Equivalent to \$WCREV\$
.Date	The commit date/time of the highest commit revision. Equivalent to \$WCDATE\$

Metoda	Popis
.Author	Autor nejvyšší odevzdané revize, to znamená, poslední člověk, který odevzdal změny do pracovní kopie.
.MinRev	Minimální aktualizace revize, jak je uvedeno v \$WCRANGE\$
.MaxRev	Maximální aktualizace revize, jak je uvedeno v \$WCRANGE\$
.HasModifications	True, pokud existují místní úpravy
.Url	Replaced with the repository URL of the working copy path used in GetWCInfo. Equivalent to \$WCURL\$
.IsSvnItem	True, pokud má položka verzi.
.NeedsLocking	True, pokud má položka nastavenou vlastnost svn:needs-lock.
.IsLocked	True, pokud je položka uzamčena.
.LockCreationDate	Řetězec reprezentující datum, kdy byl zámek vytvořen, nebo prázdný řetězec, pokud položka není uzamčena.
.LockOwner	Řetězec představující majitele zámku, nebo prázdný řetězec, pokud položka není uzamčena.
.LockComment	Zpráva zadaná při vytvoření zámku.

Tabulka 5.3. Podporované metody COM/automatizace

Následující příklad ukazuje, jak by rozhraní mohlo být použito.

```
// testCOM.js - javascript file
// test script for the 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);

```

Kapitola 6. Rozhraní IBugtraqProvider

Chcete-li získat těsnější integraci se systémy pro sledování problémů, než prosté používání vlastností bugtraq:, TortoiseSVN může využít zásuvné moduly COM. S takovými zásuvnými moduly je možné načíst informace přímo ze systému, komunikovat s uživatelem a poskytovat informace zpátky TortoiseSVN o otevřených problémech, ověření zpráv záznamu zadané uživatelem a dokonce i spustit činnosti po úspěšném odevzdání např. zavřít problém.

We can't provide information and tutorials on how you have to implement a COM object in your preferred programming language, but we have example plugins in C++/ATL and C# in our repository in the contrib/issue-tracker-plugins folder. In that folder you can also find the required include files you need to build your plugin. (3 – „TortoiseSVN je zadarmo!“ explains how to access the repository).

6.1. Rozhraní IBugtraqProvider

TortoiseSVN 1.5 can use plugins which implement the IBugtraqProvider interface. The interface provides a few methods which plugins can use to interact with the issue tracker.

```
HRESULT ValidateParameters (
    // Nadřazené okno pro jakékoliv uživatelské rozhraní, které
    // je třeba zobrazit během ověřování.
    [in] HWND hParentWnd,

    // Řetězec parametru, který je třeba ověřit.
    [in] BSTR parameters,

    // Je tento řetězec platný?
    [out, retval] VARIANT_BOOL *valid
);
```

This method is called from the settings dialog where the user can add and configure the plugin. The `parameters` string can be used by a plugin to get additional required information, e.g., the URL to the issue tracker, login information, etc. The plugin should verify the `parameters` string and show an error dialog if the string is not valid. The `hParentWnd` parameter should be used for any dialog the plugin shows as the parent window. The plugin must return `TRUE` if the validation of the `parameters` string is successful. If the plugin returns `FALSE`, the settings dialog won't allow the user to add the plugin to a working copy path.

```
HRESULT GetLinkText (
    // Nadřazené okno pro jakékoli (chybové) uživatelské rozhraní, které je třeba z
    [in] HWND hParentWnd,

    // Řetězec parametru, jen pro případ, že musíte mluvit se
    // svou internetovou službou (např.) zjistit, jaký je správný text.
    [in] BSTR parameters,

    // Který text chcete zobrazit?
    // Použít aktuální jazykové vlákno.
    [out, retval] BSTR *linkText
);
```

The plugin can provide a string here which is used in the TortoiseSVN commit dialog for the button which invokes the plugin, e.g., "Choose issue" or "Select ticket". Make sure the string is not too long,

otherwise it might not fit into the button. If the method returns an error (e.g., `E_NOTIMPL`), a default text is used for the button.

```
HRESULT GetCommitMessage (  
    // Nadřazené okno pro uživatelské rozhraní poskytovatele.  
    [in] HWND hParentWnd,  
  
    // Parametry pro Vašeho poskytovatele.  
    [in] BSTR parameters,  
    [in] BSTR commonRoot,  
    [in] SAFEARRAY(BSTR) pathList,  
  
    // Předtím zadaný text ve zprávě odevzdání.  
    // Váš poskytovatel by měl zahrnout tuto zprávu ve zprávě nové,  
    // tam, kde je to vhodné.  
    [in] BSTR originalMessage,  
  
    // Nový text pro zprávu odevzdání.  
    // Ta nahradí zprávu původní.  
    [out, retval] BSTR *newMessage  
);
```

This is the main method of the plugin. This method is called from the TortoiseSVN commit dialog when the user clicks on the plugin button. The `parameters` string is the string the user has to enter in the settings dialog when he configures the plugin. Usually a plugin would use this to find the URL of the issue tracker and/or login information or more. The `commonRoot` string contains the parent path of all items selected to bring up the commit dialog. Note that this is *not* the root path of all items which the user has selected in the commit dialog. The `pathList` parameter contains an array of paths (as strings) which the user has selected for the commit. The `originalMessage` parameter contains the text entered in the log message box in the commit dialog. If the user has not yet entered any text, this string will be empty. The `newMessage` return string is copied into the log message edit box in the commit dialog, replacing whatever is already there. If a plugin does not modify the `originalMessage` string, it must return the same string again here, otherwise any text the user has entered will be lost.

6.2. Rozhraní IBugtraqProvider2

V TortoiseSVN 1.6 bylo přidáno nové rozhraní, které nabízí více funkcí pro zásuvné moduly. Toto rozhraní IBugtraqProvider2 je odvozeno z IBugtraqProvider.

```
HRESULT GetCommitMessage2 (  
    // Nadřazené okno pro uživatelské rozhraní Vašeho poskytovatele.  
    [in] HWND hParentWnd,  
  
    // Parametry pro Vašeho poskytovatele.  
    [in] BSTR parameters,  
    // Společná URL odevzdání  
    [in] BSTR commonURL,  
    [in] BSTR commonRoot,  
    [in] SAFEARRAY(BSTR) pathList,  
  
    // Text, který je už přítomen ve zprávě odevzdání.  
    // Váš poskytovatel by měl zahrnout tento text v nové zprávě,  
    // kde je to vhodné.  
    [in] BSTR originalMessage,  
  
    // Můžete přiřadit vlastní vlastnosti revizi k odevzdání  
    // nastavením následujících dvou parametrů.
```

```
// Poznámka: Oba safearrays musí být stejné délky.  
//          Pro každé jméno vlastnosti musí být i její hodnota  
  
// Obsah pole bugID (když je zobrazeno)  
[in] BSTR bugID,  
  
// Změněný obsah pole bugID  
[out] BSTR * bugIDOut,  
  
// Seznam jmen vlastností revize.  
[out] SAFEARRAY(BSTR) * revPropNames,  
  
// Seznam hodnot vlastností revize.  
[out] SAFEARRAY(BSTR) * revPropValues,  
  
// Nový text pro zprávu odevzdání.  
// Toto nahradí původní text  
[out, retval] BSTR * newMessage  
);
```

This method is called from the TortoiseSVN commit dialog when the user clicks on the plugin button. This method is called instead of `GetCommitMessage()`. Please refer to the documentation for `GetCommitMessage` for the parameters that are also used there. The parameter `commonURL` is the parent URL of all items selected to bring up the commit dialog. This is basically the URL of the `commonRoot` path. The parameter `bugID` contains the content of the bug-ID field (if it is shown, configured with the property `bugtraq:message`). The return parameter `bugIDOut` is used to fill the bug-ID field when the method returns. The `revPropNames` and `revPropValues` return parameters can contain name/value pairs for revision properties that the commit should set. A plugin must make sure that both arrays have the same size on return! Each property name in `revPropNames` must also have a corresponding value in `revPropValues`. If no revision properties are to be set, the plugin must return empty arrays.

```
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  
);
```

This method is called right before the commit dialog is closed and the commit begins. A plugin can use this method to validate the selected files/folders for the commit and/or the commit message entered by the user. The parameters are the same as for `GetCommitMessage2()`, with the difference that `commonURL` is now the common URL of all *checked* items, and `commonRoot` the root path of all checked items. The return parameter `errorMessage` must either contain an error message which TortoiseSVN shows to the user or be empty for the commit to start. If an error message is returned, TortoiseSVN shows the error string in a dialog and keeps the commit dialog open so the user can correct whatever is wrong. A plugin should therefore return an error string which informs the user *what* is wrong and how to correct it.

```
HRESULT OnCommitFinished (  
    // Nadřazené okno pro jakékoli (chybové) uživatelské rozhraní, které je třeba z  
    [in] HWND hParentWnd,  
  
    // Společný kořen všech cest, které byly odevzdány.  
    [in] BSTR commonRoot,
```

```
// Všechny cesty, které byly odevzdány.  
[in] SAFEARRAY(BSTR) pathList,  
  
// Už zadaný text ve zprávě záznamu.  
[in] BSTR logMessage,  
  
// Revize odevzdání.  
[in] ULONG revision,  
  
// Chyba, která se má ukázat uživateli, pokud tato funkce  
// vrátí něco jiného než S_OK  
[out, retval] BSTR * error  
);
```

This method is called after a successful commit. A plugin can use this method to e.g., close the selected issue or add information about the commit to the issue. The parameters are the same as for `GetCommitMessage2`.

```
HRESULT HasOptions(  
    // Zda poskytovatel nabízí možnosti  
    [out, retval] VARIANT_BOOL *ret  
);
```

This method is called from the settings dialog where the user can configure the plugins. If a plugin provides its own configuration dialog with `ShowOptionsDialog`, it must return `TRUE` here, otherwise it must return `FALSE`.

```
HRESULT ShowOptionsDialog(  
    // Nadřazené okno dialogového okna možností  
    [in] HWND hParentWnd,  
  
    // Parametry pro Vašeho poskytovatele.  
    [in] BSTR parameters,  
  
    // Řetězec parametrů  
    [out, retval] BSTR * newparameters  
);
```

This method is called from the settings dialog when the user clicks on the "Options" button that is shown if `HasOptions` returns `TRUE`. A plugin can show an options dialog to make it easier for the user to configure the plugin. The `parameters` string contains the plugin parameters string that is already set/entered. The `newparameters` return parameter must contain the parameters string which the plugin constructed from the info it gathered in its options dialog. That `parameters` string is passed to all other `IBugtraqProvider` and `IBugtraqProvider2` methods.

Příloha A. Často kladené otázky (FAQ)

Because TortoiseSVN is being developed all the time it is sometimes hard to keep the documentation completely up to date. We maintain an [online FAQ](http://tortoisesvn.tigris.org/faq.html) [http://tortoisesvn.tigris.org/faq.html] which contains a selection of the questions we are asked the most on the TortoiseSVN mailing lists <dev@tortoisesvn.tigris.org> and <users@tortoisesvn.tigris.org>.

We also maintain a project [Issue Tracker](http://issues.tortoisesvn.net) [http://issues.tortoisesvn.net] which tells you about some of the things we have on our To-Do list, and bugs which have already been fixed. If you think you have found a bug, or want to request a new feature, check here first to see if someone else got there before you.

If you have a question which is not answered anywhere else, the best place to ask it is on one of the mailing lists. <users@tortoisesvn.tigris.org> is the one to use if you have questions about using TortoiseSVN. If you want to help out with the development of TortoiseSVN, then you should take part in discussions on <dev@tortoisesvn.tigris.org>.

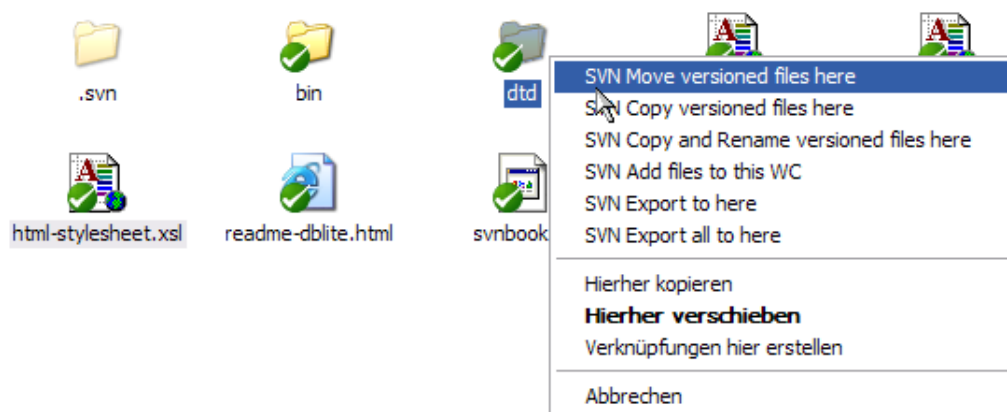
Příloha B. Jak...

Tato příloha obsahuje řešení problémů/otázek, které můžete mít při používání TortoiseSVN.

B.1. Přesunout/zkopírovat větší množství souborů najednou

Přesouvání/Kopírování jednotlivých souborů lze provést pomocí TortoiseSVN → Přejmenovat.... Ale pokud chcete přesunout/kopírovat velké množství souborů, toto je prostě příliš pomalé a příliš pracné.

The recommended way is by right-dragging the files to the new location. Simply right-click on the files you want to move/copy without releasing the mouse button. Then drag the files to the new location and release the mouse button. A context menu will appear where you can either choose Context Menu → SVN Copy versioned files here. or Context Menu → SVN Move versioned files here.



B.2. Přinutit uživatele zadat zprávu záznamu

Existují dva způsoby, jak zabránit uživatelům odevzdávat s prázdnou zprávou záznamu. Jeden z nich je specifický pro TortoiseSVN, další funguje ve všech klientech Subversion, ale vyžaduje přímý přístup k serveru.

B.2.1. Hákový skript na serveru

Pokud máte přímý přístup k serveru úložiště, můžete nainstalovat skript před odevzdáním, který odmítne všechny odevzdání s prázdnou, nebo příliš krátkou zprávou záznamu.

V adresáři úložiště na serveru, je podadresář `hooks`, který obsahuje některé příklady hákových skriptů, které můžete použít. Soubor `pre-commit.tmpl` obsahuje ukázkový skript, který odmítne odevzdání, pokud není dodána žádná zpráva záznamu, nebo je zpráva příliš krátká. Soubor obsahuje také poznámky o tom, jak nainstalovat/používat tento skript. Stačí sledovat instrukce v tomto souboru.

Tato metoda je doporučený způsob, pokud Vaši uživatelé používají i jiné klienty Subversion než TortoiseSVN. Nevýhodou je, že odevzdání je odmítnuto serverem a díky tomu uživatelé dostanou chybové hlášení. Klient nemůže před odevzdáním vědět, že bude odmítnut. Pokud chcete, aby TortoiseSVN měl tlačítko OK vypnuté, dokud zpráva záznamu není dostatečně dlouhá, pak použijte metodu popsanou níže.

B.2.2. Vlastnosti projektu

TortoiseSVN využívá vlastnosti pro řízení některých jeho funkcí. Jedna z těchto vlastností je `tsvn:logminsize`.

Pokud ji nastavíte na adresář, pak TortoiseSVN vypne tlačítko OK ve všech dialogových oknech odevzdání, dokud uživatel nezadá zprávu záznamu s minimální délkou uvedenou ve vlastnosti.

For detailed information on those project properties, please refer to [4.17 – „Nastavení projektu“](#)

B.3. Aktualizovat vybrané soubory z úložiště

Normálně aktualizujete pracovní kopii pomocí TortoiseSVN → Aktualizovat . Ale pokud chcete jen vyzvednout nějaké nové soubory, které kolega přidal bez sloučení jakýkoliv změn v jiných souborech ve stejnou dobu, budete potřebovat jiný přístup.

Použijte TortoiseSVN → Zjistit změny a klikněte na Zkontrolovat úložiště, abyste viděli, co se v úložišti změnilo. Vyberte soubory, které chcete aktualizovat místně, pak pomocí kontextového menu aktualizujte pouze tyto soubory.

B.4. Vrátit zpět revize v úložišti

B.4.1. Pomocí dialogového okna záznamu revizí

Nejjednodušší způsob, jak vrátit změny z jedné nebo řady revizí je použít dialogové okno záznamu revizí. To je také způsob, pokud chcete zahodit nedávné změny a udělat z dřívější revize nový HEAD.

1. Vyberte soubor nebo adresář, v kterém chcete vrátit změny. Pokud chcete vrátit všechny změny, měl by to být adresář nejvyšší úrovně.
2. Vyberte TortoiseSVN → Zobrazit záznam k zobrazení seznamu revizí. Možná budete muset použít Zobrazit Vše nebo Dalších 100 pro zobrazení revize(i), která Vás zajímá.
3. Vyberte revize, kterou chcete vrátit. Pokud chcete vrátit řadu revizí, zvolte první a držte klávesu **Shift** při výběru té poslední. Všimněte si, že pro více revizí, řada musí být neporušená, bez mezer. Klikněte pravým tlačítkem myši na vybranou revizi(e), pak zvolte Kontextové menu → Vrátit změny z této revize.
4. Nebo pokud chcete, aby dřívější revizi se stala nová HEAD revize, klikněte pravým tlačítkem myši na vybranou revizi, pak zvolte Kontextové menu → Vrátit se k této revizi. Tím se zbavíte *všech* změn po vybrané revizi.

Vrátili jste změny v pracovní kopii. Podívejte se na výsledky, pak změny odevzdejte.

B.4.2. Pomocí dialogového okna sloučení

Chcete-li vrátit větší rozsah revizí, můžete použít dialogové okno Sloučení. Předchozí metoda používá sloučení v zákulisí, tato metoda ho používá explicitně.

1. Ve Vaší pracovní kopie zvolte TortoiseSVN → Sloučit.
2. V poli Z: zadejte úplnou adresu URL adresáře větve nebo značky obsahující změny, které chcete vrátit do Vaší pracovní kopie. To by se mělo zobrazit jako výchozí URL.
3. V poli Od revize zadejte číslo revize, které v současnosti máte. Pokud si jste jisti, že nikdo jiný neprovedl změny, můžete použít revizi HEAD.
4. ujistěte se, že volba Použít "Z:" URL je zaškrtnuta.
5. V poli Na Revizi zadejte číslo revize, do které se chcete vrátit, a sice na tu *před* první revizi, která má být vrácena.

6. Klikněte na OK k dokončení sloučení.

Vrátili jste změny v pracovní kopii. Podívejte se na výsledky, pak změny odevzdejte.

B.4.3. Pomocí `svndumpfilter`

Vzhledem k tomu, že TortoiseSVN nikdy neztrácí data, Vaše „vrácené“ revize stále existují jako mezilehlé revize v úložišti. Pouze revize HEAD byla změněna do předchozího stavu. Pokud chcete, aby revize zcela zmizely z Vašeho úložiště, jako kdyby vůbec neexistovaly, budete muset použít více extrémní opatření. Pokud není opravdu dobrý důvod k tomu toto udělat, *není* to doporučeno. Jedním z možných důvodů je, že někdo odevzdal důvěrný dokument do veřejného úložiště.

Jediným způsobem, jak odstranit data z úložiště je použití nástroje `svnadmin` příkazového řádku Subversion. Popis funkce naleznete v [Údržbě Úložiště](http://svnbook.red-bean.com/en/1.5/svn.reposadmin.maint.html) [http://svnbook.red-bean.com/en/1.5/svn.reposadmin.maint.html].

B.5. Porovnat dvě revize souboru nebo adresáře

Pokud chcete porovnat dvě revize v historii položky, například revize 100 a 200 téhož souboru, stačí použít TortoiseSVN → Zobrazit záznam pro seznam historie revizí pro tento soubor. Vyberte si dvě revize, které chcete porovnat pak použijte Kontextové menu → Porovnat Revize.

Pokud chcete porovnat stejné položky ve dvou různých stromech, například kmen a větev, můžete použít průzkumníka úložiště pro otevření obou stromů, vyberte soubor v obou místech, pak použijte Kontextové menu → Porovnat Revize.

Pokud chcete porovnat dva stromy, abyste viděli, co se změnilo, například kmen a označené vydání, můžete použít TortoiseSVN → Graf Revizí Vyberte dva uzly k porovnání, pak použijte Kontextové menu → Porovnat HEAD revize. Tím se zobrazí seznam změněných souborů a pak můžete vybrat jednotlivé soubory k zobrazení změn v detailu. Můžete také exportovat stromovou strukturu obsahující všechny změněné soubory, nebo prostě jen seznam všech změněných souborů. Přečtěte si [4.10.3 – „Porovnávání adresářů“](#) pro více informací. Můžete také použít Kontextové menu → Sjednocení rozdíl z HEAD revizí, abyste viděli přehled všech rozdílů, s minimálním kontextem.

B.6. Zahrnout společný podprojekt

Někdy budete chtít zahrnout další projekty v rámci Vaší pracovní kopie, možná některé kódové knihovny. Nechcete ale kopii tohoto kódu ve Vašem úložišti, protože pak byste ztratily spojení s původním (a udržovaným) kódem. Nebo možná máte několik projektů, které sdílejí základní kód. Existují minimálně 3 způsoby jak toto řešit.

B.6.1. Pomocí `svn:externals`

Nastavte vlastnost `svn:externals` na adresář v projektu. Tato vlastnost se skládá z jednoho nebo více řádků, každý řádek má název podadresáře, který chcete použít jako adresář získání společného kódu a URL úložiště, které tam chcete získat. Pro podrobnosti si prohlédněte [4.18 – „Externí položky“](#).

Odevzdejte nový adresář. Nyní, když aktualizujete, Subversion stáhne kopii tohoto projektu z jeho úložiště do Vaší pracovní kopie. Podadresáře jsou automaticky vytvořeny v případě potřeby. Pokaždé, když budete aktualizovat svoji hlavní pracovní kopii, obdržíte také nejnovější verze všech externích projektů.

If the external project is in the same repository, any changes you make there there will be included in the commit list when you commit your main project.

Pokud je externí projekt v jiném úložišti, všechny změny v něm provedené budou při odevzdání hlavního projektu oznámeny, ale musíte tyto vnější změny odevzdat samostatně.

Ze tří uvedených metod, tato je jediná, který nepotřebuje žádné nastavení na straně klienta. Jakmile jsou externí uvedeny ve vlastnostech adresáře, všichni klienty při aktualizaci dostanou zaplněný adresář.

B.6.2. Pomocí vnořené pracovní kopie

Create a new folder within your project to contain the common code, but do not add it to Subversion

Vyberte TortoiseSVN → Získat na nový adresář a získejte do něj kopii společného kódu. Nyní máte samostatnou pracovní kopii vnořenou ve své hlavní pracovní kopii.

Tyto dvě pracovní kopie jsou nezávislé. Když odevzdáváte změny do nadřazené pracovní kopie, jsou ve vnořené ignorovány. Podobně při aktualizaci nadřazené pracovní kopie, není vnořená aktualizována.

B.6.3. Pomocí relativního umístění

If you use the same common core code in several projects, and you do not want to keep multiple working copies of it for every project that uses it, you can just check it out to a separate location which is related to all the other projects which use it. For example:

```
C:\Projects\Proj1
C:\Projects\Proj2
C:\Projects\Proj3
C:\Projects\Common
```

and refer to the common code using a relative path, eg. `..\..\Common\DSPCore`.

If your projects are scattered in unrelated locations you can use a variant of this, which is to put the common code in one location and use drive letter substitution to map that location to something you can hard code in your projects, eg. Checkout the common code to `D:\Documents\Framework` or `C:\Documents and Settings\{login}\My Documents\framework` then use

```
SUBST X: "D:\Documents\framework"
```

to create the drive mapping used in your source code. Your code can then use absolute locations.

```
#include "X:\superio\superio.h"
```

Tato metoda bude fungovat pouze v prostředí PC a budete muset zdokumentovat požadované mapování disku, takže váš tým bude vědět, kde jsou tyto záhadné soubory. Tato metoda je výhradně pro použití v uzavřených vývojových prostředích a nedoporučuje se pro všeobecné použití.

B.7. Vytvořit zkratku úložiště

Pokud často potřebujete otevřít průzkumníka úložiště v konkrétním umístění, můžete vytvořit zástupce na ploše pomocí rozhraní automatizace TortoiseProc. Stačí vytvořit nového zástupce a stanovit cíl na:

```
TortoiseProc.exe/command:repobrowser/path: "url/do/uloziste"
```

Samozřejmě je nutné zadat URL skutečného úložiště.

B.8. Ignorovat soubory, které už mají verzi

Pokud jste omylem přidali některé soubory, které měly být ignorovány, jak je dostanete ze správy verzí, aniž by se ztratili? Možná že máte svůj vlastní konfigurační soubor IDE, který není součástí projektu, ale který Vám zabral spoustu času nastavit přesně tak, jak se Vám líbí.

If you have not yet committed the add, then all you have to do is use TortoiseSVN → Revert... to undo the add. You should then add the file(s) to the ignore list so they don't get added again later by mistake.

If the files are already in the repository, you have to do a little more work.

1. Hold the **Shift** key to get the extended context menu and use TortoiseSVN → Delete (keep local) to mark the file/folder for deletion from the repository without losing the local copy.
2. TortoiseSVN → Commit the parent folder.
3. Add the file/folder to the ignore list so you don't get into the same trouble again.

B.9. Odstranit verzi pracovní kopie

Pokud máte pracovní kopii, kterou chcete převést zpět na běžný adresářový strom, bez adresářů `.svn`, můžete jednoduše exportovat do sebe. Přečtěte si [4.26.1 – „Smazání pracovní kopie ze správy verzí“](#), abyste zjistili, jak na to.

B.10. Odebrat pracovní kopii

If you have a working copy which you no longer need, how do you get rid of it cleanly? Easy - just delete it in Windows Explorer! Working copies are private local entities, and they are self-contained.

Příloha C. Užitečné tipy pro administrátory

Tato příloha obsahuje řešení problémů/otázek, které můžete mít, pokud jste zodpovědní za nasazení TortoiseSVN do více klientských počítačů.

C.1. Nasazení TortoiseSVN pomocí zásad skupiny

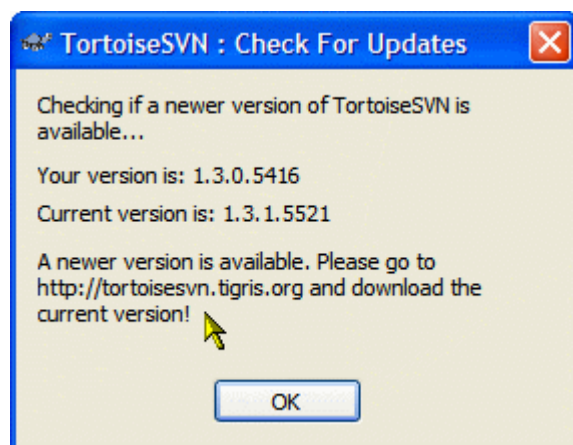
Instalační soubor TortoiseSVN je ve formátu MSI, což znamená, že byste neměli mít žádné problémy s přidáním tohoto MSI souboru do zásad skupiny Vašeho správce domény.

Dobrý návod, jak toto provést, lze nalézt v článku 314934 znalostní báze společnosti Microsoft: <http://support.microsoft.com/?kbid=314934>.

Versions 1.3.0 and later of TortoiseSVN must be installed under *Computer Configuration* and not under *User Configuration*. This is because those versions need the new CRT and MFC DLLs, which can only be deployed *per computer* and not *per user*. If you really must install TortoiseSVN on a per user basis, then you must first install the MFC and CRT package version 8 from Microsoft on each computer you want to install TortoiseSVN as per user.

C.2. Přesměrovat kontrolu nové verze

TortoiseSVN checks if there's a new version available every few days. If there is a newer version available, a dialog shows up informing the user about that.



Obrázek C.1. The upgrade dialog

If you're responsible for a lot of users in your domain, you might want your users to use only versions you have approved and not have them install always the latest version. You probably don't want that upgrade dialog to show up so your users don't go and upgrade immediately.

Versions 1.4.0 and later of TortoiseSVN allow you to redirect that upgrade check to your intranet server. You can set the registry key HKCU\Software\TortoiseSVN\UpdateCheckURL (string value) to an URL pointing to a text file in your intranet. That text file must have the following format:

```
1.4.1.6000
A new version of TortoiseSVN is available for you to download!
http://192.168.2.1/downloads/TortoiseSVN-1.4.1.6000-svn-1.4.0.msi
```

The first line in that file is the version string. You must make sure that it matches the exact version string of the TortoiseSVN installation package. The second line is a custom text, shown in the upgrade dialog. You can write there whatever you want. Just note that the space in the upgrade dialog is limited. Too long messages will get truncated! The third line is the URL to the new installation package. This URL is opened when the user clicks on the custom message label in the upgrade dialog. You can also just point the user to a web page instead of the MSI file directly. The URL is opened with the default web browser, so if you specify a web page, that page is opened and shown to the user. If you specify the MSI package, the browser will ask the user to save the MSI file locally.

C.3. Nastavení proměnné prostředí SVN_ASP_DOT_NET_HACK

As of version 1.4.0 and later, the TortoiseSVN installer doesn't provide the user with the option to set the SVN_ASP_DOT_NET_HACK environment variable anymore, since that caused many problems and confusions with users which always install *everything* no matter if they know what it is for.

Ale tato možnost je skryta pouze uživatelům. Stále je možné donutit instalátor TortoiseSVN nastavit tuto proměnnou nastavením vlastnosti ASPDOTNETHACK na hodnotu true. Například je možné spustit instalátor takto:

```
msiexec /i TortoiseSVN-1.4.0.msi ASPDOTNETHACK=TRUE
```

C.4. Zakázání položek kontextového menu

Ve verzi 1.5.0 a vyšší, TortoiseSVN umožňuje zakázat (vlastně skrýt) položky kontextového menu. Protože toto je funkce, která by neměla být použita jen tak, ale pouze v případě, že je pádný důvod, neexistuje pro toto žádné grafické rozhraní a je třeba to provést přímo v registru. Toto může být použito k zakázání určitých příkazů pro uživatele, kteří by je neměli používat. Ale prosím vezměte na vědomí, že pouze položky kontextového menu v *průzkumníku* jsou skryty a příkazy jsou stále k dispozici jiným způsobem, např. z příkazové řádky nebo i v jiných dialogových oknech TortoiseSVN!

Klíče registru, který drží informace, které kontextové menu zobrazit, jsou HKEY_CURRENT_USER\Software\TortoiseSVN\ContextMenuEntriesMaskLow a HKEY_CURRENT_USER\Software\TortoiseSVN\ContextMenuEntriesMaskHigh.

Každá z těchto položek registru má hodnotu DWORD, přičemž každý bit odpovídá konkrétní položce menu. Nastavený bit znamená, že odpovídající položka menu je vypnuta.

Hodnota	Položka menu
0x0000000000000001	Získat
0x0000000000000002	Aktualizovat
0x0000000000000004	Odevzdat
0x0000000000000008	Přidat
0x0000000000000010	Vrátit
0x0000000000000020	Vyčistit
0x0000000000000040	Vyřešit
0x0000000000000080	Překlopit
0x0000000000000100	Import
0x0000000000000200	Exportovat
0x0000000000000400	Vytvořit Úložiště zde
0x0000000000000800	Větev/značka

Hodnota	Položka menu
0x0000000000001000	Sloučit
0x0000000000002000	Smazat
0x0000000000004000	Přejmenovat
0x0000000000008000	Aktualizovat na Revizi
0x0000000000010000	Porovnat
0x0000000000020000	Zobrazit Záznam
0x0000000000040000	Upravit konflikty
0x0000000000080000	Přemístit
0x0000000000100000	Zjistit změny
0x0000000000200000	Ignorovat
0x0000000000400000	Průzkumník Úložiště
0x0000000000800000	Obvinít
0x0000000001000000	Vytvořit Záplatu
0x0000000002000000	Použít Záplatu
0x0000000004000000	Graf Revizí
0x0000000008000000	Zámek
0x0000000010000000	Odstranit zámek
0x0000000020000000	Vlastnosti
0x0000000040000000	Porovnat s URL
0x0000000080000000	Smazat položky bez verze
0x2000000000000000	Nastavení
0x4000000000000000	Nápověda
0x8000000000000000	O aplikaci

Tabulka C.1. Položky menu a jejich hodnoty

Příklad: Pro vypnutí položek menu „Přemístit“, „Smazat položky bez verze“ a „Nastavení“, přidejte hodnoty přiřazené položkám, jako jsou tyto:

```

0x0000000000008000
+ 0x0000000080000000
+ 0x2000000000000000
= 0x2000000080080000

```

Nižší hodnota DWORD (0x80080000) pak musí být uložena v HKEY_CURRENT_USER\Software\TortoiseSVN\ContextMenuEntriesMaskLow, vyšší hodnota DWORD (0x20000000) v HKEY_CURRENT_USER\Software\TortoiseSVN\ContextMenuEntriesMaskHigh.

K znovu zobrazení těchto položek menu, stačí smazat tyto dva klíče registru.

Příloha D. Automatizace TortoiseSVN

Protože všechny příkazy TortoiseSVN jsou řízeny přes parametry příkazového řádku, můžete je automatizovat pomocí dávkových skriptů nebo spustit konkrétní příkazy a dialogová okna z jiných programů (např. Váš oblíbený textový editor).



Důležité

Nezapomeňte, že TortoiseSVN je klient grafického uživatelského rozhraní a tento průvodce automatizací Vám ukáže, jak zobrazit dialogová okna TortoiseSVN pro sběr vstupů uživatele. Pokud chcete napsat skript, který nevyžaduje žádné vstupy, měli byste místo toho použít oficiální příkazový řádek klienta Subversion.

D.1. Příkazy TortoiseSVN

Program grafického uživatelského rozhraní TortoiseSVN se nazývá `TortoiseProc.exe`. Všechny příkazy jsou uvedeny s parametrem `/command:abcd` kde `abcd` je povinný název příkazu. Většina těchto příkazů potřebujete alespoň jeden argument cesty, který je předáván s `/path:"nějaká\cesta"`. V následující tabulce se příkaz vztahuje k parametru `/command:abcd` a cesta se vztahuje k parametru `/path:"nějaká\cesta"`.

Protože některým z těchto příkazů můžete předat seznam cílových cest (např. odevzdání několika konkrétních souborů) parametr `/path` může mít i několik cest, oddělených znakem `*`.

TortoiseSVN používá dočasné soubory k předání více argumentů mezi rozšířením rozhraní a hlavním programem. Od TortoiseSVN 1.5.0 a později, je parametr `/notempfile` je zastaralý a není třeba už ho přidávat.

Dialogové okno průběhu, který se používá pro odevzdání, aktualizace a mnoho dalších příkazů obvykle zůstane otevřené i po dokončení příkazu, dokud uživatel nestiskne tlačítko OK. Toto může být změněno zaškrtnutím příslušné volby v dialogovém okně nastavení. Ale použitím tohoto nastavení se bude dialogové okno průběhu bez ohledu na to, pokud spustíte příkaz z dávkového souboru nebo z kontextového menu TortoiseSVN.

Chcete-li zadat jiné umístění konfiguračního souboru, použijte parametr `/configdir:"cesta\k\adresáři astavení"`. Tím se přepíše výchozí cesta, včetně všech nastavení registru.

To close the progress dialog at the end of a command automatically without using the permanent setting you can pass the `/closeonend` parameter.

- `/closeonend:0` dialogové okno se automaticky nezavře
- `/closeonend:1` automaticky zavřít, pokud nejsou žádné chyby
- `/closeonend:2` automaticky zavřít, pokud nejsou žádné chyby a konflikty
- `/closeonend:3` automaticky zavřít, pokud nejsou žádné chyby, konflikty a sloučení
- `/closeonend:4` auto close if no errors, conflicts and merges for local operations

Níže uvedená tabulka uvádí všechny příkazy, ke kterým lze přistupovat pomocí příkazového řádku `TortoiseProc.exe`. Jak je uvedeno výše, tyto by měly být použity ve formě `/command:abcd`. V tabulce je předpona `/command` vynechána pro úsporu místa.

Příkaz	Popis
<code>:about</code>	Zobrazuje informace o programu. Také je zobrazen, není-li zadán příkaz.

Příkaz	Popis
:log	Opens the log dialog. The <code>/path</code> specifies the file or folder for which the log should be shown. Three additional options can be set: <code>/startrev:xxx</code> , <code>/endrev:xxx</code> and <code>/strict</code>
:checkout	Opens the checkout dialog. The <code>/path</code> specifies the target directory and the <code>/url</code> specifies the URL to checkout from.
:import	Opens the import dialog. The <code>/path</code> specifies the directory with the data to import.
:update	Updates the working copy in <code>/path</code> to HEAD. If the option <code>/rev</code> is given then a dialog is shown to ask the user to which revision the update should go. To avoid the dialog specify a revision number <code>/rev:1234</code> . Other options are <code>/nonrecursive</code> and <code>/ignoreexternals</code> .
:commit	Otevře dialogové okno odevzdání. <code>/path</code> určuje cílový adresář nebo seznam souborů k odevzdání. Můžete také zadat přepínač <code>/logmsg</code> pro předání předem nastavené zprávy záznamu v dialogovém okně. Nebo pokud nechcete předat zprávu záznamu na příkazovém řádku, použijte <code>/logmsgfile:cesta</code> , kde <code>cesta</code> odkazuje na soubor, obsahující zprávu záznamu. Pro předvyplnění rámečku s ID chyby (v případě, že jste nastavili integraci se systémem pro sledování chyb správně), můžete k tomuto použít <code>/bugid:"id chyby zde"</code> .
:add	Přidá soubory v <code>/path</code> pod správu verzí.
:revert	Vrátí místní změny pracovní kopie. <code>/path</code> říká, které položky chcete vrátit.
:cleanup	Cleans up interrupted or aborted operations and unlocks the working copy in <code>/path</code> .
:resolve	Označí konfliktní soubor zadaný v <code>/path</code> jako vyřešený. Pokud je předán <code>/noquestion</code> , pak řešení je provedeno, aniž by byl uživatel dotázán, jestli je to opravdu třeba udělat.
:repocreate	Vytvoří úložiště v <code>/path</code>
:switch	Otevře dialogové okno překlopení. <code>/path</code> určuje cílový adresář.
:export	Exports the working copy in <code>/path</code> to another directory. If the <code>/path</code> points to an unversioned directory, a dialog will ask for an URL to export to the directory in <code>/path</code> .
:merge	Opens the merge dialog. The <code>/path</code> specifies the target directory. For merging a revision range, the following options are available: <code>/fromurl:URL</code> , <code>/revrange:řetězec</code> . For merging two repository trees, the following options are available: <code>/fromurl:URL</code> , <code>/tourl:URL</code> , <code>/fromrev:xxx</code> and <code>/torev:xxx</code> . These pre-fill the relevant fields in the merge dialog.
:mergeall	Otevře dialogové okno sloučit vše. <code>/path</code> určuje cílový adresář.
:copy	Zobrazí dialogové okno větev/značka. <code>/path</code> je pracovní kopie z, které má být větev/tag vytvořena. A <code>/url</code> je cílová URL. Můžete také zadat přepínač <code>/logmsg</code> k předání předem nastavené zprávy záznamu do dialogového okna větve/značky. Nebo pokud nechcete předat zprávu záznamu na příkazovém řádku, použijte <code>/logmsgfile:cesta</code> , kde <code>cesta</code> odkazuje na soubor, obsahující zprávu záznamu.
:settings	Otevře dialogové okno nastavení.
:remove	Odstraní soubor(y) v <code>/path</code> ze správy verzí.
:rename	Přejmenuje soubor v <code>/path</code> . O nový název souboru bude požádáno v dialogovém okně. Chcete-li se vyhnout otázce o přejmenování podobných souborů v jednom kroku, předejte <code>/noquestion</code> .

Příkaz	Popis
:diff	Starts the external diff program specified in the TortoiseSVN settings. The /path specifies the first file. If the option /path2 is set, then the diff program is started with those two files. If /path2 is omitted, then the diff is done between the file in /path and its BASE. To explicitly set the revision numbers use /startrev:xxx and /endrev:xxx. If /blame is set and /path2 is not set, then the diff is done by first blaming the files with the given revisions.
:showcompare	<p>V závislosti na URL a revizích k porovnání, buď ukazuje sjednocený rozdíl (v případě, že volba sjednocený je nastavena), dialogové okno se seznamem souborů, které se změnily, nebo pokud adresa URL odkazuje na soubory, spustí prohlížeč rozdílů pro tyto dva soubory.</p> <p>Možnosti url1, url2, revize1 a revize2 musí být zadány. Možnosti udržetrevizi, ignorovatpůvod, obvinít a sjednocený jsou volitelné.</p>
:conflicteditor	Spustí editor konfliktů, který je uveden v nastavení TortoiseSVN se správnými soubory pro konfliktní soubor v /path.
:relocate	Otevře se dialogové okno přemístit. /path Určuje cestu pracovní kopie k přemístění.
:help	Otevře soubor nápovědy.
:repostatus	Opens the check-for-modifications dialog. The /path specifies the working copy directory.
:repobrowser	Starts the repository browser dialog, pointing to the URL of the working copy given in /path or /path points directly to an URL. An additional option /rev:xxx can be used to specify the revision which the repository browser should show. If the /rev:xxx is omitted, it defaults to HEAD. If /path points to an URL, the /projectpropertiespath:path/to/wc specifies the path from where to read and use the project properties.
:ignore	Přidá všechny cíle v /path do seznamu k ignorování, tj. přidá vlastnost svn:ignore k těmto souborům.
:blame	<p>Otevře dialogové okno obvinění souboru uvedeném v /path.</p> <p>Pokud jsou volby /startrev a /endrev nastaveny, pak se dialogové okno s dotazem o rozsahu obvinění nezobrazí, ale místo toho jsou použity hodnoty revizí těchto voleb.</p> <p>Pokud možnost /line:nnn je nastavena, TortoiseBlame bude otevřen zobrazující zadané číslo řádku.</p> <p>Možnosti /ignoreeol, /ignorespaces a /ignoreallspaces jsou také podporovány.</p>
:cat	Uloží soubor z adresy URL nebo z cesty pracovní kopie uvedené v /path na místo uvedené v /savepath:path. Revize je uvedena v /revision:xxx. Toto může být použito k získání souboru s konkrétní revizí.
:createpatch	Vytvoří záplatový soubor pro cestu zadanou v /path.
:revisiongraph	Zobrazí graf revizí pro cesty uvedené v /path.
:lock	Zamkne soubor nebo všechny soubory v adresáři uvedené v /path. Zobrazí se dialogové okno, takže uživatel může zadat komentář k zámku.
:unlock	Odemkne soubor nebo všechny soubory v adresáři uvedené v /path.

Příkaz	Popis
:rebuildiconcache	Znovu obnoví windows rozhraní uložených ikon. Použijte pouze v případě, že ikony Windows jsou poškozené. Vedlejším efektem tohoto (kterému se nelze vyhnout) je, že ikony na ploše budou zpřeházeny. Chcete-li potlačit rámeček se zprávou, předejte /noquestion.
:properties	Zobrazí dialogové okno vlastností pro cesty uvedené v /path .

Tabulka D.1. Seznam dostupných příkazů a možností

Příklady (které by měly být uvedeny na jednom řádku):

```
TortoiseProc.exe /command:commit
                  /path:"c:\svn_wc\file1.txt*c:\svn_wc\file2.txt"
                  /logmsg:"zkouška zprávy záznamu" /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. Příkazy TortoiseIDiff

Nástroj rozdílů obrázků má několik možností v příkazovém řádku, které můžete použít ke kontrole spuštění nástroje. Tento program se nazývá TortoiseIDiff.exe.

Níže uvedená tabulka uvádí všechny možnosti, které mohou být předány nástroji rozdílů obrázků na příkazovém řádku.

Možnost	Popis
:left	Cesta k souboru zobrazenému vlevo.
:lefttitle	Řetězec názvu. Tento řetězec se používá v názvu zobrazení obrázku místo úplné cesty k souboru s obrázkem.
:right	Cesta k souboru zobrazenému vpravo.
:righttitle	Řetězec názvu. Tento řetězec se používá v názvu zobrazení obrázku místo úplné cesty k souboru s obrázkem.
:overlay	Pokud je zadán, nástroj rozdílů obrázků přepne do režimu překrytí (částečná průhlednost).
:fit	Pokud je zadán, nástroj rozdílů obrázků nastaví shodnou velikost obou obrázků
:showinfo	Zobrazí informační rámeček obrázku.

Tabulka D.2. Seznam dostupných možností

Příklad (který by měl být uveden na jednom řádku):

```
TortoiseIDiff.exe /left:"c:\obrázky\obr1.jpg" /lefttitle:"obrazek 1"
                  /right:"c:\obrázky\obr2.jpg" /righttitle:"obrazek 2"
                  /fit /overlay
```

Příloha E. Odkazy Rozhraní Příkazové Řádky

Někdy tato příručka odkazuje na hlavní dokumentaci Subversion, který popisuje Subversion v rozhraní příkazové řádky (CLI). Abychom Vám pomohli pochopit, co TortoiseSVN dělá v zákulisí, sestavili jsme seznam zobrazující odpovídající CLI příkazy pro každou z operací grafického rozhraní TortoiseSVN.

Poznámka

I když zde jsou CLI ekvivalenty toho, co TortoiseSVN dělá, nezapomeňte, že TortoiseSVN *nevolá* CLI, ale používá přímo knihovnu Subversion.

Pokud si myslíte, že jste našli chybu v TortoiseSVN, můžeme Vás požádat, abyste se ji pokusili zopakovat pomocí CLI, abychom mohli rozlišit problémy TortoiseSVN od problémů Subversion. Tento odkaz Vám řekne, který příkaz zkusit.

E.1. Úmluvy a Základní pravidla

In the descriptions which follow, the URL for a repository location is shown simply as URL, and an example might be `http://tortoisesvn.googlecode.com/svn/trunk`. The working copy path is shown simply as PATH, and an example might be `C:\TortoiseSVN\trunk`.



Důležité

Protože TortoiseSVN je Rozšíření Rozhraní Windows, není schopen používat pojem současného pracovního adresáře. Všechny cesty pracovní kopie musí být uvedeny s použitím absolutní cesty, ne relativní.

Některé položky jsou nepovinné a ty jsou často ovládány zaškrťovacími políčky nebo přepínači v TortoiseSVN. Tyto možnosti jsou uvedeny v [hranatých závorkách] v definicích příkazového řádku.

E.2. Příkazy TortoiseSVN

E.2.1. Získat

```
svn checkout [-N] [--ignore-externals] [-r rev] URL PATH
```

If Only checkout the top folder is checked, use the `-N` switch.

Pokud je Vynechat externí zaškrtnuto, použijte přepínač `--Ignore-externals`.

Pokud získáváte konkrétní revizi, zadejte ji po URL pomocí přepínače `-r`.

E.2.2. Aktualizovat

```
svn info URL_Pracovní_Kopie
svn update [-r rev] CESTA
```

Aktualizace více položek není v současné době v Subversion atomová operace. Takže TortoiseSVN nejprve nalezne HEAD revizi úložiště a pak aktualizuje všechny položky na konkrétní číslo revize, aby se zabránilo vytvoření smíšených revizi pracovní kopie.

Pokud je pouze jedna položka vybrána pro aktualizaci nebo vybrané položky nejsou všechny ze stejného úložiště, TortoiseSVN aktualizuje pouze na HEAD.

Žádné parametry příkazového řádku nejsou zde použity. Aktualizovat na revizi také provádí příkaz aktualizovat, ale nabízí více možností.

E.2.3. Aktualizovat na Revizi

```
svn info URL_of_WC
svn update [-r rev] [-N] [--ignore-externals] PATH
```

If Only update the top folder is checked, use the `-N` switch.

Pokud je Vynechat externí zaškrtnuto, použijte přepínač `--Ignore-externals`.

E.2.4. Odevzdat

V TortoiseSVN, dialogové okno odevzdání používá několik příkazů Subversion. V první fázi je kontrola stavu, který určuje položky ve Vaší pracovní kopii, které mohou být potenciálně odevzdány. Můžete si prohlédnout seznam, porovnat soubory s originálem a vybrat položky, které chcete zahrnout do odevzdání.

```
svn status -v CESTA
```

Pokud je zaškrtnuto **Ukázat soubory bez verze**, TortoiseSVN také zobrazí všechny soubory a adresáře bez verze v hierarchii pracovní kopie s přihlédnutím k pravidlům ignorování. Tato konkrétní vlastnost nemá žádnou přímou obdobu v Subversion, protože příkaz `svn status` nevstupuje do adresářů bez verze.

Chcete-li zkontrolovat všechny soubory a adresáře bez verze, tyto položky nejprve budou přidány do Vaší pracovní kopie.

```
svn add CESTA...
```

Když kliknete na OK, spustí se odevzdání Subversion. Pokud jste nechali všechny políčka pro výběr souborů ve výchozím stavu, TortoiseSVN použije jediné rekurzivní odevzdání pracovní kopie. Pokud zrušíte výběr některých souborů, pak nerekurzivní odevzdání (`-N`) musí být použito a každá cesta musí být zadána individuálně na příkazovém řádku odevzdání.

```
svn commit -m "LogMessage" [-N] [--no-unlock] PATH...
```

Zpráva záznamu zde představuje obsah editačního pole zprávy záznamu. Může být prázdný.

Pokud **Ponechat zámky** je zaškrtnuto, použijte přepínač `--no-unlock`.

E.2.5. Porovnat

```
svn diff CESTA
```

Pokud používáte Porovnat z hlavní kontextové nabídky, porovnáváte upravený soubor proti jeho původní revizi. Výstup z příkazu CLI výše toto dělá také a produkuje výstup ve formátu sjednoceného rozdílu. Nicméně, není to stejné, jako pomocí TortoiseSVN. TortoiseSVN používá TortoiseMerge (nebo program rozdílů dle Vašeho výběru) k zobrazení rozdílů vizuálně mezi full-textovými soubory, takže neexistuje žádný přímý ekvivalent CLI.

Můžete také porovnat jakékoli 2 obrázky pomocí TortoiseSVN, ať už jsou nebo nejsou pod správou verzí. TortoiseSVN jen předá dva soubory do zvoleného programu rozdílů a nechá ho zjistit, kde se vyskytují rozdíly.

E.2.6. Zobrazit Záznam

```
svn log -v -r 0:N --limit 100 [--stop-on-copy] CESTA  
nebo  
svn log -v -r M:N [--stop-on-copy] CESTA
```

Ve výchozím nastavení TortoiseSVN se snaží stáhnout 100 zpráv záznamu pomocí metody `--limit`. Pokud mu nastavení říká, že má použít staré API, pak je použita druhá forma, k stáhnutí zpráv záznamu za 100 revizí uložiště.

Pokud je zaškrtnuto **Zastavit na kopírování/přejmenování**, použijte přepínač `--stop-on-copy`.

E.2.7. Zjistit změny

```
svn status -v CESTA  
nebo  
svn status -u -v CESTA
```

Počáteční kontrola stavu se dívá pouze do Vaší pracovní kopie. Pokud kliknete na **Zkontrolovat úložiště** pak úložiště je také zkontrolováno, aby se zjistilo, které soubory by se změnily aktualizací, což vyžaduje přepínač `-u`.

Pokud je zaškrtnuto **Ukázat soubory bez verze**, TortoiseSVN také zobrazí všechny soubory a adresáře bez verze v hierarchii pracovní kopie s přihlédnutím k pravidlům ignorování. Tato konkrétní vlastnost nemá žádnou přímou obdobu v Subversion, protože příkaz `svn status` nevstupuje do adresářů bez verze.

E.2.8. Graf Revizí

Graf revizí je funkce TortoiseSVN. Neexistuje žádná obdoba v příkazovém řádku klienta.

Co TortoiseSVN dělá je

```
svn info URL_Pracovní_Kopie  
svn log -v URL
```

where URL je *kořen* úložiště a pak vrácená data zpracuje.

E.2.9. Průzkumník Úložiště

```
svn info URL_Pracovní_Kopie  
svn list [-r rev] -v URL
```

Můžete použít `svn info` k určení kořene úložiště, což je nejvyšší úroveň zobrazená v průzkumníku úložiště. Nemůžete jít nahoru nad tuto úroveň. Také tento příkaz vrátí všechny informace o zamknutí uvedené v průzkumníku úložiště.

Volání `svn list` vypíše obsah adresáře, dané URL a revize.

E.2.10. Upravit konflikty

Tento příkaz nemá obdobu CLI. Vyvolá TortoiseMerge nebo externí nástroj trojitého porovnání/sloučení, podívá se na soubory zapojené do konfliktu a vytřídí, které řádky použít.

E.2.11. Vyřešeno

```
svn resolved CESTA
```

E.2.12. Přejmenovat

```
svn rename SOUČAS_CESTA NOVÁ_CESTA
```

E.2.13. Smazat

```
svn delete CESTA
```

E.2.14. Vrátit

```
svn status -v CESTA
```

V první fázi je kontrola stavu, který určuje položky v pracovní kopii, které mohou být potenciálně vráceny. Můžete si prohlédnout tento seznam, porovnat soubory s originálem a vybrat položky, které chcete zahrnout do vrácení.

Když kliknete na OK, spustí se vrácení Subversion. Pokud jste nechali všechna políčka pro výběr souboru v jejich výchozím stavu, TortoiseSVN použije jediné rekurzivní (-R) vrácení v pracovní kopii. Pokud zrušíte výběr některých souborů, pak každá cesta musí být na příkazové řádce vrácení uvedena samostatně.

```
svn revert [-R] CESTA...
```

E.2.15. Vyčistit

```
svn cleanup CESTA
```

E.2.16. Uzamknout

```
svn status -v CESTA
```

V první fázi je kontrola stavu, která určí soubory v pracovní kopii, které můžou být potenciálně uzamčeny. Můžete si vybrat položky, které chcete zamknout.

```
svn lock -m "Zprávazamknutí" [--force] CESTA...
```

Zprávazamknutí zde představuje obsah editačního pole zprávy o uzamčení. Může být prázdné.

Pokud je Ukrást zámky zaškrtnuto, použijte přepínač --force.

E.2.17. Uvolnit zámek

```
svn unlock CESTA
```

E.2.18. Větev/značka


```
svn copy -m "LogMessage" URL URL
    nebo
svn copy -m "LogMessage" URL@rev URL@rev
    nebo
svn copy -m "LogMessage" CESTA URL
```

Dialogové okno Větev/Značka zkopíruje úložiště. K dispozici jsou 3 přepínače možností:

- revize HEAD v úložišti
- Určitá revize v úložišti
- Pracovní kopie

, kterým odpovídají 3 varianty příkazového řádku výše.

Zpráva záznamu zde představuje obsah editačního pole zprávy záznamu. Může být prázdný.

E.2.19. Překlopit

```
svn info URL_Pracovní_Kopie
svn switch [-r rev] URL CESTA
```

E.2.20. Sloučit

```
svn merge [--dry-run] --force Z_URL@revN Do_URL@revM CESTA
```

Testovat Sloučení provede stejné sloučení s přepínačem `--dry-run`.

```
svn diff Z_URL@revN Do_URL@revM
```

Sjednocený rozdíl zobrazí operace porovnání, která bude použita k provedení sloučení.

E.2.21. Exportovat

```
svn export [-r rev] [--ignore-externals] URL Export_CESTA
```

Tento formulář se používá při přístupu z adresáře bez verze a adresář je použit jako cíl.

Exportování pracovní kopie na jiné místo se provádí bez použití knihovny Subversion, takže není odpovídající obdoba příkazové řádky.

Co TortoiseSVN dělá, je zkopírování všech souborů do nového umístění, zatímco zobrazuje průběh operace. Soubory/adresáře bez verze lze také volitelně exportovat.

V obou případech, pokud je zaškrtnuto **Vynechat externí**, použijte přepínač `--ignore-externals`.

E.2.22. Přemístit

```
svn switch --relocate Z_URL Do_URL
```

E.2.23. Vytvořit Úložiště Zde

```
svnadmin create --fs-type fsfs CESTA
```

E.2.24. Přidat

```
svn add CESTA...
```

Pokud vyberete adresář, TortoiseSVN nejprve prohledá rekurzivně položky, které mohou být přidány.

E.2.25. Import

```
svn import -m LogMessage CESTA URL
```

Zpráva záznamu zde představuje obsah editačního pole zprávy záznamu. Může být prázdný.

E.2.26. Obvinit

```
svn blame -r N:M -v CESTA  
svn log -r N:M CESTA
```

Pokud používáte TortoiseBlame k zobrazení informací o obvinění, je soubor záznamu také požadován k zobrazení zpráv záznamu v popisku. Zobrazíte-li obvinění jako textový soubor, tato informace není zapotřebí.

E.2.27. Přidat do seznamu k ignorování

```
svn propget svn:ignore CESTA > dočsoubor  
{upravit nové ignorovanou položku do dočasněho souboru}  
svn propset svn:ignore -F dočsoubor CESTA
```

protože vlastnost `svn:ignore` je často mnohořádková hodnota, je zde měněna prostřednictvím textového souboru, spíše než přímo v příkazovém řádku.

E.2.28. Vytvořit Záplatu

```
svn diff CESTA > záplatovýsoubor
```

TortoiseSVN vytvoří záplatový soubor ve formátu sjednoceného rozdílu porovnáním pracovní kopie s jeho základní verzí.

E.2.29. Použít Záplatu

Aplikování záplat je ošemetná věc, pokud není záplata a pracovní kopie ve stejné revizi. Naštěstí pro Vás, můžete použít TortoiseMerge, který nemá v Subversion žádnou přímou obdobu.

Příloha F. Detaily zavedení

Tato příloha obsahuje podrobnější diskusi o zavádění některých funkcí TortoiseSVN.

F.1. Překryv ikon

Každý soubor a adresář má Subversion hodnotu stavu, jak je uvádí knihovna Subversion. V příkazovém řádku klienta, jsou zastoupeny jediným písmenem kódu, ale v TortoiseSVN jsou zobrazeny graficky pomocí překryvu ikon. Protože počet překryvů je velmi omezen, každé překrytí představuje jednu z několika hodnot stavu.



Konfliktní překrytí je používáno k označení konfliktního stavu, kde aktualizace nebo přepínač vyústí v konflikt mezi místními změnami a změny stáhnutými z úložiště. Je také používáno k označení stavu zablokováno, který může nastat, když operaci nebylo možné dokončit.



Změněný překryv představuje změněný stav, kde jste udělali místní změny, sloučený stav, kde změny z úložiště byly sloučeny s místními změnami a nahrazený stav, kde byl soubor smazán a nahrazen jiným se stejným názvem.



Překrytí *Smazané* představuje smazaný stav, kde je položka naplánována k smazání, nebo stav *chybí*, kde položka není přítomna. Přirozeně položka, která chybí, nemůže mít překrytí, ale nadřazený adresář může být označen, pokud jedna z jeho podřízených položek chybí.



Přidaný překryv je jednoduše používán k označení stavu *přidán*, kdy položka byla přidána do správy verzí.



Překryv *V Subversion* je používán k označení předmětu, který je v normálním stavu, nebo položka s verzí, jejíž stav není dosud znám. Vzhledem k tomu, že TortoiseSVN má uchovávací proces v pozadí ke shromažďování stavu, může trvat několik sekund, než je překryv aktualizován.



The *Needs Lock* overlay is used to indicate when a file has the `svn:needs-lock` property set. For working copies which were created using Subversion 1.4.0 and later, the `svn:needs-lock` status is cached locally by Subversion and this is used to determine when to show this overlay. For working copies which are in pre-1.4.x format, TortoiseSVN shows this overlay when the file has read-only status. Note that Subversion automatically upgrades working copies when you update them, although the caching of the `svn:needs-lock` property may not happen until the file itself is updated.



Uzamčený překryv se používá, když místní pracovní kopie drží zámek pro tento soubor.



Překryv *Ignorovaný* je používán k označení předmětu, který je ve stavu ignorován, ať už kvůli globálnímu vzoru ignorování, nebo vlastnosti `svn:ignore` v nadřazeném adresáři. Tento překryv je nepovinný.



Překrytí *Bez verze* je používán pro sdělení, že předmět je ve stavu `bez verze`. To znamená, že položka v adresáři s verzí, ale není samotná pod správou verzí. Tento překryv je nepovinný.

Pokud položka má Subversion stav *žádný* (Položka není v pracovní kopii), pak není zobrazeno žádné překrytí. Pokud jste se rozhodli zakázat překryvy *Ignorován* a *Bez verze* pak pro tyto soubory také nebudou uvedeny žádné překryvy.

Položka může mít pouze jednu hodnotu Subversion stavu. Například by soubor mohl mít místní úpravy a ve stejnou dobu by mohla být označena pro odstranění. Subversion vrátí jednu hodnotu stavu - v tomto případě *smazáno*. Tyto priority jsou definovány v rámci samotného Subversion.

Když TortoiseSVN zobrazuje stav rekurzivně (výchozí nastavení), jednotlivé adresáře zobrazují překryv odrážející jejich vlastní postavení a stav všech jejich podřazených položek. Pro zobrazení jednoho *shrnujícího* překrytí, používáme pořadí priorit uvedený výše k určení, který překryv použít, kde *Konfliktní* překryv má nejvyšší prioritu.

In fact, you may find that not all of these icons are used on your system. This is because the number of overlays allowed by Windows is limited to 15. Windows uses 4 of those, and the remaining 11 can be used by other applications. If there are not enough overlay slots available, TortoiseSVN tries to be a „Good Citizen (TM)“ and limits its use of overlays to give other apps a chance.

- *Normální*, *Změněný* a *Konfliktní* jsou vždy načteny a viditelné.
- *Smazán* je načten pokud je to možné, ale vrátí se do *Změněný*, pokud není dostatek volného místa.
- *Pouze pro čtení* je načten, pokud je to možné, ale vrátí se do *Normální*, pokud není dostatek volného místa.
- *Locked* is only loaded if there are fewer than 13 overlays already loaded. It falls back to *Normal* if there are not enough slots.
- *Added* is only loaded if there are fewer than 14 overlays already loaded. It falls back to *Modified* if there are not enough slots.

Příloha G. Zabezpečení svnserve pomocí SSH

Tato část poskytuje průvodce krok za krokem nastavením Subversion a TortoiseSVN na používání protokolu `svn+ssh`. Pokud již používáte ověřené SSH spojení pro přihlášení na Váš server, pak jste už hotovi a podrobnosti můžete nalézt v knize Subversion. Pokud nepoužíváte SSH, ale rádi byste tak učinili k ochraně instalace Subversion, tento průvodce poskytuje jednoduchou metodu, která nevyžaduje vytvoření samostatného uživatelského SSG účtu na serveru pro každého uživatele Subversion.

V tomto zavedení vytvoříme jednotný uživatelský SSH účet pro všechny uživatele Subversion s použitím různých ověřovacích klíčů k rozlišení skutečných uživatelů Subversion.

V této příloze předpokládáme, že již máte nainstalovány nástroje Subversion a že jste vytvořili úložiště, jak je uvedeno jinde v této příručce. Nezapomeňte, že byste *neměli* spouštět svnserve jako službu nebo daemona při použití SSH.

Much of the information here comes from a tutorial provided by Marc Logemann, which can be found at www.logemann.org [<http://www.logemann.org/2007/03/13/subversion-tortoisesvn-ssh-howto/>] Additional information on setting up a Windows server was provided by Thorsten Müller. Thanks guys!

G.1. Nastavení Serveru Linux

Musíte mít SSH na serveru zapnuté a zde předpokládáme, že budete používat OpenSSH. Na většině distribucí toto již bude nainstalováno. Chcete-li to zjistit, zadejte:

```
ps xa | grep sshd
```

a podívejte se na ssh operace.

Jeden bod k poznámce je, že pokud budete sestavovat Subversion ze zdroje a nepředáte `./configure` žádný argument, Subversion vytvoří adresář `bin` v `/usr/local` a umístí tam své spustitelné soubory. Pokud chcete použít režim tunelování se SSH, musíte být vědomi toho, že uživatel přihlašující se přes SSH potřebuje spustit program `svnserve` a některé další spustitelné soubory. Z tohoto důvodu buď umístíte `/usr/local/bin` do proměnné `PATH` nebo vytvoříte symbolické odkazy z Vašich soborů do adresáře `/usr/sbin`, nebo jiného adresáře, který je běžně v `PATH`.

Chcete-li zkontrolovat, že je vše v pořádku, přihlašte se jako cílový uživatel přes SSH a napište:

```
which svnserve
```

Tento příkaz by Vám měl sdělit, jestli je `svnserve` dosažitelný.

Vytvořte nového uživatele, kterého budeme používat pro přístup do SVN úložiště:

```
useradd -m svnuzivatel
```

Ujistěte se, aby tento uživatel měl úplná přístupová práva do úložiště.

G.2. Nastavení Windows Serveru

Install Cygwin SSH daemon as described here: <http://pigtail.net/LRP/printsrv/cygwin-sshd.html>

Vytvořte nový uživatelský účet systému Windows `svnuzivatel` kterého budeme používat pro přístup do úložiště. Ujistěte se, aby tento uživatel měl úplná přístupová práva do úložiště.

Pokud ještě žádný soubor vytvořen nebyl, pak ho vytvořte z konzole Cygwin pomocí:

```
mkpasswd -l > /etc/passwd
```

G.3. Nástroje klienta SSH pro použití s TortoiseSVN

Stáhněte si nástroje, které potřebujeme pro použití SSH v klientu Windows z této stránky: <http://www.chiark.greenend.org.uk/~sgtatham/putty/> Stačí jít do sekce Download a stáhnout Tmel, Plink, Pageant a Puttygen.

G.4. Vytváření Certifikátů OpenSSH

Dalším krokem je vytvoření páru klíčů pro ověření. Existují dva možné způsoby, jak vytvořit klíče. Prvním je vytvoření klíče s PuTTYgen na straně klienta, nahrát veřejný klíč na server a použít soukromý klíč s PuTTY. Druhý je vytvořit dvojici klíčů s OpenSSH nástrojem ssh-keygen, stáhnout soukromý klíč pro Vašeho klienta a převést soukromý klíč do PuTTY stylu.

G.4.1. Vytvoření Klíčů pomocí ssh-keygen

Přihlaste se na server jako root nebo svnuzivatel a zadejte příkaz:

```
ssh-keygen -b 1024 -t dsa -N heslo -f klicsoubor
```

nahrad'te heslo skutečným (které znáte pouze Vy) a soubor s klíčem. Právě jsme vytvořili SSH2 DSA klíč s 1024 bitovou frází. Pokud zadáte

```
ls -l klicsoubor *
```

uvidíte dva soubory, klicsoubor a klicsoubor.pub. Jak asi tušíte, soubor .pub je veřejný klíč, druhý je soukromý.

Append the public key to those in the .ssh folder within the svnuser home directory:

```
cat keyfile.pub >> /home/svnuser/.ssh/authorized_keys
```

Aby bylo možné použít soukromý klíč, který jsme vytvořili, musíme jej převést do formátu putty. Je to proto, že formát souboru soukromého klíče není specifikován v těle standardu. Po stažení souboru se soukromým klíčem Vašeho PC klienta, spusťte PuTTYgen a použijte **Conversions** → **Import key**. Přejděte na Váš soubor klicsoubor který jste dostali od serveru s frází, kterou jste použili při vytváření klíče. Nakonec klikněte na **Save private key** a uložte soubor jako klicsoubor.PPK.

G.4.2. Vytvoření klíčů pomocí PuTTYgen

Použijte PuTTYgen pro vytvoření páru veřejného/soukromého klíče a uložte je. Zkopírujte veřejný klíč na server a připojit ho k těm v adresáři .ssh v rámci domovského adresáře svnuzivatel:

```
cat klicsoubor.pub >> /home/svnuzivatel/
.ssh/authorized_keys
```

G.5. Test pomocí PuTTY

Pro zkoušku spojení použijeme PuTTY. Spusťte program a na kartě **Session** nastavte hostname na jméno nebo IP adresu serveru, protokol na SSH a uložte relaci jako SvnSpojeni nebo jakékoliv jiné jméno, kterému dáváte přednost. Na kartě **SSH** nastavte preferovanou verzi SSH protokolu na 2 a v **Auth** nastavte

úplnou cestu k souboru `. PPK` se soukromým klíčem, který jste dříve převedli. Vraťte se do karty **Session** a stiskněte tlačítko **Save**. Nyní uvidíte `SvnSpojení` v seznamu uložených relací.

Klikněte na **Open** a měli byste vidět výzvu k přihlášení ve stylu telnet. Použijte `svnuzivatel` jako uživatelské jméno a pokud je vše v pořádku, měli byste se připojit přímo bez vyzvání k zadání hesla.

Možná budete muset na serveru upravit `/etc/ssh/sshd_config`. Upravte řádky takto a poté restartujte službu SSH.

```
PubkeyAuthentication yes
PasswordAuthentication no
PermitEmptyPasswords no
ChallengeResponseAuthentication no
```

G.6. Testování SSH s TortoiseSVN

Zatím jsme jen zkoušeli, zda se můžete přihlásit pomocí SSH. Nyní se musíme ujistit, že spojení SSH může skutečně spustit `svnserve`. Na serveru upravte `/home/svnuzivatel/.ssh/authorized_keys` takto, abyste povolili mnoha autorům Subversion použít stejný systémový účet, `svnuzivatel`. Nezapomeňte, že každý Subversion autor používá stejné přihlašovací údaje, ale jiný ověřovací klíč, proto musíte přidat jeden řádek pro každého autora.

Poznámka: tohle vše je na jednom velmi dlouhém řádku.

```
command="svnserve -t -r <Kořenovácestaúložiště> --tunnel-user=<autor>",
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,
no-pty ssh-rsa <Veřejnýklíč> <Komentář>
```

Je zde několik hodnot, které je potřeba nastavit podle Vašeho systému.

`<Kořenovácestaúložiště>` by měl být nahrazen cestou k adresáři, který obsahuje Vaše úložiště. To zamezuje nutnost zadávat úplné cesty serveru v URL. Nezapomeňte, že musíte používat lomítka i na Windows server, např. `c:/svn/kořenuloz`. V příkladech níže předpokládáme, že máte adresář úložiště v kořenovii úložiště názvem `uloz`.

`<autor>` by měl být nahrazen `svn` autorem, kterého chcete mít uložen při odevzdání. To také umožňuje `svnserve` používat vlastní přístupová práva v rámci `svnserve.conf`.

`<Veřejnýklíč>` by měl být nahrazen veřejným klíčem, který jste dříve vytvořili.

`<Komentář>` může být jakýkoli komentář, který se Vám líbí, ale je užitečné mapovat `svn` jméno autora ke skutečnému jménu osoby.

Klikněte pravým tlačítkem na libovolný adresář v Průzkumníku Windows a vyberte **TortoiseSVN** → **Průzkumník Úložiště**. Budete vyzváni k zadání URL, takže ji zadejte v této podobě:

```
svn+ssh://svnuzivatel@SvnSpojení/uloz
```

Co tato URL znamená? Název schématu je `svn+ssh`, který říká TortoiseSVN, jak zpracovat požadavky na server. Po dvojitém lomítku, zadáte uživatele pro připojení k serveru, v našem případě `svnuzivatel`. Po `@` dodáváme náš název relace PuTTY. Tento název relace obsahuje všechny detaily, jako kde najít soukromý klíč a IP serveru nebo DNS. Nakonec musíme zadat cestu k úložišti, vzhledem ke kořeni úložiště na serveru, jak je uvedeno v souboru `authorized_keys`.

Klikněte na **OK** a měli byste být schopni procházet obsah úložiště. Pokud ano, tak Vám nyní běží SSH tunel ve spojení s TortoiseSVN.

Všimněte si, že ve výchozím nastavení TortoiseSVN používá vlastní verzi Plink pro připojení. Tím se zabrání vyskakování okna konzole pro každý pokus o ověření, ale také to znamená, že se nikde nezobrazí chybová zpráva. Pokud se zobrazí chyba „Unable to write to standard output“, můžete zkusit uvést Plink jako klienta v nastavení sítě TortoiseSVN. To Vám umožní vidět skutečné chybové zprávy vytvořené v Plink.

G.7. Konfigurační varianty SSH

Jeden způsob, jak zjednodušit URL v TortoiseSVN je nastavit uživatele uvnitř relace PuTTY. Za tímto účelem je nutné načíst již zavedenou relaci `SvnSpojeni` v PuTTY a v kartě **Connection** nastavit **Auto login user** na uživatelské jméno, např. `svnuzivatel`. Uložte relaci PuTTY jako předtím a zkuste následující URL uvnitř TortoiseSVN:

```
svn+ssh://SvnSpojeni/uloz
```

Tentokrát jsme poskytli pouze relaci PuTTY `SvnSpojeni` klientu SSH, který TortoiseSVN používá (TortoisePlink.exe). Tento klient zkontroluje relaci pro všechny potřebné údaje.

V době psaní PuTTY nekontroluje všechny uložené konfigurace, takže pokud máte více konfigurací se stejným názvem serveru, vybere ji při první shodě. Také pokud upravíte výchozí konfiguraci a uložíte ji, uživatelské jméno pro automatické přihlášení *není* uloženo.

Mnozí lidé rádi používají Pageant pro ukládání všech jejich klíčů. Protože relace PuTTY je schopna klíče uložit, Pageant vždycky nepotřebujete. Ale představte si, že chcete uložit klíče pro několik různých serverů, v tomto případě budete muset upravovat relace PuTTY znovu a znovu, v závislosti na serveru, s kterým se snažíte spojit. V této situaci Pageant dává smysl, protože když PuTTY, Plink, TortoisePlink nebo jakékoli jiný nástroj založený na PuTTY se snaží připojit k SSH serveru, kontroluje všechny soukromé klíče, které Pageant drží pro zahájení spojení.

Pro tento úkol, stačí jednoduše spustit Pageant a přidat soukromý klíč. Měl by to být stejný soukromý klíč, který jste nastavili v relaci PuTTY výše. Pokud používáte Pageant pro ukládání soukromých klíčů, můžete odstranit odkaz na soubor se soukromým klíčem uložen v relaci PuTTY. Můžete přidat více klíčů pro jiné servery i samozřejmě pro jiné uživatele.

If you don't want to repeat this procedure after every reboot of your client, you should place Pageant in the auto-start group of your Windows installation. You can append the keys with complete paths as command line arguments to Pageant.exe

Poslední způsob, jak se připojit k SSH serveru je jednoduše pomocí tohoto URL v TortoiseSVN:

```
svn+ssh://svnuuziivatel@100.101.102.103/uloz  
svn+ssh://svnuzivatel@madomena.cz/uloz
```

Jak vidíte, nepoužíváme uložené relace PuTTY, ale IP adresu (nebo jméno domény) jako cíl spojení. Dodáváme i uživatele, ale můžete se ptát, jak bude soubor se soukromým klíčem nalezen. Vzhledem k tomu, že TortoisePlink.exe je jen upravená verze standardního nástroje Plink z balíku PuTTY, TortoiseSVN také vyzkouší všechny klíče uložené v Pageant.

If you use this last method, be sure you do not have a default username set in PuTTY. We have had reports of a bug in PuTTY causing connections to close in this case. To remove the default user, simply clear `HKEY_CURRENT_USER\Software\SimonTatham\Putty\Sessions\Default%20Settings\HostName`

Slovník pojmů

Aktualizovat	Tento příkaz Subversion stáhne poslední změny z úložiště do Vaší pracovní kopie, sloučí změny provedené ostatními s místními změnami v pracovní kopii.
BDB	Berkeley Databáze. Prověřený databázový backend pro úložiště, které nelze použít ve sdílené síti. Výchozí pro úložiště před verzí 1.2.
Exportovat	Tento příkaz vytvoří kopii adresáře s verzí, stejně jako pracovní kopie, ale bez místních adresářů <code>.svn</code> .
FSFS	Vlastní backend souborového systému Subversion pro úložiště. Může být použit v sdílené síti. Výchozí pro úložiště od verze 1.2 a novější.
GPO	Group policy object
Historie	Zobrazit historii revizí souboru nebo adresář. Také znám jako „záznam“.
Import	Subversion příkaz pro import celé hierarchie adresářů do úložiště v jedné revizi.
Konflikt	Když se změny z úložiště sloučí s místními změnami, někdy tyto změny nastanou na stejných řádcích. V tomto případě Subversion nemůže automaticky rozhodnout, kterou verzi použít a soubor je tak řečeno v konfliktu. Pro vyřešení konfliktu musíte upravit soubor ručně, než budete moci spáchat další změny.
Kopie	V Subversion můžete vytvořit kopii jednoho souboru nebo celého stromu. Ty jsou realizovány jako „laciné kopie“ které působí trochu jako odkaz na původní kopii v tom, že nezabírají téměř žádný prostor. Zhotovení kopie zachová historii položky v kopii, takže můžete sledovat změny provedené dříve, než byla kopie vytvořena.
Obvinit	Tento příkaz je pouze pro textové soubory, okomentuje každý řádek pro zobrazení revize úložiště, v němž byl naposledy změněn a autora, který tuto změnu provedl. Naše GUI implementace se nazývá TortoiseBlame a také ukazuje datum, čas a zprávu odevzdání, když podržíte myš nad číslem revize.
Odevzdat	Tento příkaz Subversion slouží k předání změn v místní pracovní kopii zpět do úložiště, vytvářející novou revizi úložiště.
Porovnat	Zkratka pro „Zobrazit Rozdíly“. Velmi užitečné, pokud chcete přesně zjistit, jaké změny byly provedeny.
Pracovní kopie	Toto je Vaše místní „pískoviště“, Oblast, kde budete pracovat na souborech s verzemi a obvykle sídlí na Vašem místním pevném disku. Můžete vytvořit pracovní kopii tím, že ji „Získáte“ z úložiště a přivádíte své změny zpět do úložiště pomocí „Odevzdat“.
Překlopit	Stejně jako „Aktualizovat na revizi“ změní časové okno pracovní kopie, aby se podívalo na jiné místo v historii, tak „Překlopit“ změní prostorové okno pracovní kopie tak, aby ukazovalo na jinou část úložiště. To je zvláště užitečné při práci na kmeni a větvi, kde se liší jen několik souborů. Můžete mezi nimi svoji pracovní kopii překlopit a jen změněné soubory budou převedeny.

Přemístit	<p>Pokud se Vaše úložiště přesune, snad proto, že jste ho přestěhovali do jiného adresáře na Vašem serveru, nebo název domény serveru se změnil, je třeba „přemístit“ Vaši pracovní kopie tak, aby její URL úložiště odkazovalo na nové umístění.</p> <p>Poznámka: měli byste používat tento příkaz pouze, pokud Vaše pracovní kopie odkazuje na stejné místo ve stejném úložišti, ale samotné úložiště se přesunulo. Za všech ostatních okolností, pravděpodobně budete muset místo toho použít příkaz „Překlopit“.</p>
Přidat	<p>Příkaz Subversion, který slouží k přidání souboru nebo adresáře do Vaší pracovní kopie. Nové položky jsou do úložiště přidány při odevzdání.</p>
Revize	<p>Pokaždé, když odevzdáte sadu změn, vytvoříte v úložišti jednu novou „revize“. Každá revize představuje stav stromu úložiště v určitém okamžiku v jeho historii. Pokud se chcete vrátit zpět v čase můžete úložiště prozkoumat, jaké bylo v revizi N.</p> <p>V jiném smyslu, revize odkazuje na sadu změn, které byly provedeny, když byla tato revize vytvořena.</p>
revize HEAD	<p>Poslední revize souboru nebo adresáře v <i>úložišti</i>.</p>
Sloučit	<p>Proces, při kterém změny z úložiště jsou přidány do Vaší pracovní kopie bez přerušení jakýchkoli změn, které jste již místně udělali. Někdy tyto změny nemohou být srovnány automaticky a pracovní kopie je tzv. v konfliktu.</p> <p>Sloučení se děje automaticky při aktualizaci Vaší pracovní kopie. Můžete také sloučit konkrétní změny z jiné větve pomocí TortoiseSVN v příkazu Sloučit.</p>
Smazat	<p>Když odstraníte položku s verzí (a změnit odevzdáte), položka již v úložišti po odevzdané revizi neexistuje. Ale samozřejmě, že ještě existuje v předchozích revizích úložiště, takže můžete stále k ní mít přístup. Pokud je to nutné, můžete zkopírovat odstraněnou položku a „vzkřísit“ ji úplně i s historií.</p>
SVN	<p>Často používaná zkratka pro Subversion.</p> <p>Název vlastního protokolu Subversion používaného serverem úložiště „svnserve“.</p>
Vlastnost	<p>Kromě Vašich adresářů a souborů Subversion umožňuje přidat i metadata s verzí - známá jako „vlastnosti“ každému z Vašich adresářů a souborů s verzí. Každá vlastnost má název a hodnotu, podobně jako klíč v registru. Subversion má některé speciální vlastnosti, které používá vnitřně, jako je <code>svn:EOL-style</code>. TortoiseSVN má taky některé, jako například <code>tsvn:logminsize</code>. Můžete přidat vlastní vlastnosti s libovolným názvem a hodnotou, kterou zvolíte.</p>
Vlastnosti revize (revvlast)	<p>Stejně jako soubory mohou mít vlastnosti, může je mít i každá revize v úložišti. Některé speciální vlastnosti jsou přidány automaticky při vytvoření revize a sice: <code>svn:Datum</code> <code>svn:autor</code> <code>svn:záznam</code> které představují datum a čas odevzdání, respektive autora a zprávu záznamu. Tyto vlastnosti lze upravit, ale nemají verzi, takže každá změna je trvalá a nelze ji vrátit zpět.</p>
Vrátit	<p>Subversion udržuje místní „nedotčenou“ kopii každého souboru, jaký byl, když jste naposledy aktualizovali Vaši pracovní kopii.</p>

	<p>Pokud jste provedli změny a rozhodnete se, že je chcete vrátit, můžete použít příkaz „vrátit“, abyste se vrátili k původní kopii.</p>
Vyčistit	<p>Citujeme z knihy Subversion: „Rekurzivně vyčistí pracovní kopii, odstraní zámky a pokračuje v nedokončených operacích. Pokud někdy máte chybu <i>pracovní kopie uzamčena</i>, spusťte tento příkaz k odstranění starých zámků a dostaňte Vaši pracovní kopie zpět do použitelného stavu.“ Nezapomeňte, že v této souvislosti <i>zámek</i> odkazuje na zamknutí v místním souborovém systému, ne zamknutí úložiště.</p>
Vyřešit	<p>Pokud jsou soubory v pracovní kopii ponechány po sloučení ve stavu konfliktu, musí být tyto konflikty vyřešeny člověkem pomocí editoru (nebo možná TortoiseMerge). Tento proces je označován jako „Řešení konfliktů“. Až toto skončí, můžete označit konfliktní soubory jako vyřešené, což jim umožní být odevzdány.</p>
Větev	<p>Termín často používaný v systémech pro správu verzí pro popis co se stane, když se vývoj na určitém místě rozdělí a pokračuje po 2 samostatných cestách. Můžete vytvořit větev mimo hlavní linii vývoje, abyste mohli vyvíjet novou funkci, aniž byste hlavní linii učinili nestabilní. Nebo můžete rozvětvit stabilní verzi, na které uděláte jen opravy chyb, zatímco nový vývoj probíhá v nestabilním kmenu. V Subversion je větev zavedena jako „laciná kopie“.</p>
Základní revize	<p>Aktuální základní revize souboru nebo adresáře ve Vaší <i>pracovní kopii</i>. Jedná se o revizi, v které soubor nebo adresář byl, při posledním spuštění získání, aktualizace nebo odevzdání. BASE revize se obvykle rovná revizi HEAD.</p>
Zámek	<p>Když si vezmete zámek na položku s verzí, označíte ji v úložišti jako neodevzdatelnou, s výjimkou pracovní kopie, kde byl zámek získán.</p>
Záplata	<p>Pokud pracovní kopie má změny pouze v textových souborech, je možné použít Subversion příkaz <i>porovnat</i> pro vytvoření jednoho souboru shrnující tyto změny ve formátu sjednoceného rozdílu. Soubor tohoto typu je často označován jako „Záplata“ a může být poslán e-mailem někomu jinému (nebo do poštovního adresáře) a použit na jinou pracovní kopii. Někdo bez oprávnění k odevzdání může provést změny a předložit záplatový soubor, aby ho oprávněný vývojář mohl použít. Nebo pokud si nejste jisti provedenou změnou, může poslat záplatu ostatním k přezkoumání.</p>
Záznam	<p>Zobrazit historii revizí souboru nebo adresář. Také znám jako „Historie“.</p>
Získat	<p>Příkaz Subversion, který vytvoří místní pracovní kopii v prázdném adresáři stažením verzí souborů z úložiště.</p>
Úložiště	<p>Úložiště je centrální místo, kde jsou data uložena a udržována. Úložiště může být místo, kde se nachází více databází nebo souborů pro distribuci po síti, nebo úložiště může být také místo, které je přímo přístupné uživatelům, aniž by museli cestovat po celé síti.</p>

Rejstřík

Symboly

'pracovní kopie' bez verze, 125

A

adresář .svn, 161
adresář _svn, 161
aktualizace, 51, 173
Apache, 25
ASP projekty, 178
auto-vlastnosti, 94
automatizace, 180, 183

C

Často kladené otázky, 171
chválit, 115
činnosti na straně serveru, 118
číslo verze v souborech, 162
CLI, 184
COM, 162, 167

D

dočasné soubory, 40
Doména Windows, 30

E

export, 125
export změn, 78
externí, 96, 174
externí úložiště, 96

F

filtr, 72

G

globální ignorování, 134
globování, 84
GPO, 177
graf, 121
graf revize, 121

H

historie, 63
hook skripty, 19, 155
hook skripty na straně serveru, 19
hooky, 19

I

IBugtraqProvider, 167
ignorovat, 83
ikony, 57
import, 40
import na místě, 42

Index projektů, 29
instalovat, 3
internetová stránka, 19

J

jazykové balíčky, 3
jednotky SUBST, 145

K

klíčová slova, 92
klientské háky, 155
Kniha Subversion, 5
konflikt, 9, 53
konflikt stromu, 53
konflikty sloučení, 109
kontextové menu, 36
kontrola nové verze, 177
kontrola pravopisu, 3
kontrola verzí, 1
kopie, 99, 118
kopírování souborů, 82

M

maximalizovat, 40
mimořádné soubory, 42
Mnohonásobné ověření, 31
mod_authz_svn, 26, 29
msi, 177

N

nasazení, 177
nastavení, 133
nástroje porovnání, 81
nástroje slučování, 81
NTLM, 30

O

obvinit, 115
oddělit od úložiště, 175
odeslat změny, 45
odevzdání, 45
odkaz, 19
odkaz pro získání, 19
odstranit, 86
odstranit verze, 175
odstranit verzi, 127, 175
okomentovat, 115
ověření, 39
ovladač táhnutí, 38
označit vydání, 99

P

položky kontextového menu, 178
porovnání adresářů, 174
porovnání obrázku, 80
porovnání souborů, 174
porovnat, 76, 76, 114

porovnat revize, 78
 porovnávání, 60
 pouze pro čtení, 111
 Pověření, 29
 pracovní kopie, 10
 prázdná zpráva, 172
 přejmenování souborů, 82
 přejmenovat, 87, 118, 172
 překlady, 3
 překlopit, 101
 překryvy, 57, 190
 přemístit, 127
 přesun, 172
 přesunout, 87
 přesunování souborů, 82
 přesunutý server, 127
 přidání souborů do úložiště, 40
 přidat, 81
 přidat verzi novým souborům, 81
 příkazový řádek, 180, 183
 příkazový řádek klienta, 184
 přinést změny, 51
 priorita překrytí, 190
 Přístup, 16
 prohlížeč serveru, 118
 prohlížeč úložiště, 132
 prohlížení na internetu, 132
 projekty dodavatele, 174
 properties, 90
 proxy server, 146
 průzkumník, 1
 průzkumník úložiště, 118

R

registry, 159
 reorganizovat, 172
 revize, 12, 121
 right-click, 36
 Rohraní Windows, 1
 rozšířit klíčová slova, 92

S

SASL, 23
 Sdílený síťový disk, 16
 Server přesunut, 127
 seznam změn, 61
 shoda vzorů, 84
 sjednocený rozdíl, 114
 sledování chyb, 128
 sledování sloučení, 109
 sledování záznamu o sloučení, 70
 sloučit, 102

- dva stromy, 106
- rozsah revizí, 103
- znovu začlenit, 105

 Sloupce Průzkumníka, 58
 slovník, 3

smazat, 86
 soubory/adresáře bez verze, 83
 společné projekty, 174
 správce domény, 30, 177
 SSL, 32
 SSPI, 30
 statistika, 72
 stav, 57, 59
 stav pracovní kopie, 57
 Subversion vlastnosti, 91
 SubWCRev, 162
 SubWCRev Rozhraní COM, 164
 SVNParentPath, 28, 29
 SVNPath, 28
 svnserve, 20, 21
 SVN_ASP_DOT_NET_HACK, 178
 systém pro sledování chyb, 128
 systém pro sledování problémů, 128, 167
 systémprosledováníchyb, 128

T

táhnout-a-pustit, 38
 táhnutí pravým tlačítkem, 38
 TortoiseIDiff, 80
 TortoiseSVN odkaz, 19

U

úložiště, 5, 40
 UNC cesty, 16
 upravit záznam/autora, 71
 URL úložiště změněna, 127
 URL změněna, 127
 úschova záznamu, 152

V

verze, 177
 větev, 82, 99
 ViewVC, 132
 vlastností projektu, 94
 vlastnosti revize, 71
 Vlastnosti TortoiseSVN, 94
 vlastrev, 71
 vrátit, 89, 173
 vrátit zpět, 173
 vrátit zpět odevzdání, 173
 vrátit zpět změnu, 173
 VS2003, 178
 vyčistit, 90
 vyjmutí verze, 162
 vyřešit, 53
 vytvoření pracovní kopie, 43
 Vytvořit

- Příkazový řádek klienta, 15
- TortoiseSVN, 15

 vytvořit úložiště, 15
 Vzorek pro vynechání, 134

W

WebDAV, 25

WebSVN, 132

Z

zakázání funkce, 178

záloha, 18

zamykání, 111

záplata, 114

zapsat, 45

zásady skupiny, 177, 178

zásuvný modul, 167

záznam, 63

získání, 43

zkontrolovat novou verzi, 177

zkratka, 175

změna velikosti písmen, 87

změny, 59, 174

značka, 82, 99

znovu začleňovací sloučení, 110

zobrazit změny, 57

zpět, 89

zpráva odevzdání, 172

zpráva záznamu, 172

zprávy odevzdání, 63

zprávy záznamu, 63

zvuky, 133