

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Úvod do verzovacích systémů

(informativní)

Jan Faigl

Katedra počítačů
Fakulta elektrotechnická
České vysoké učení technické v Praze

Přednáška 11

A0B36PR2 – Programování 2

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 1 / 63

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Základní pojmy

- Správce verzí - má na starosti automatické číslování verzí Sada nástrojů pro přístup k verzovaným souborům
- Repozitář (Repository) - místo, kde jsou uloženy verzované soubory
- Lokální kopie (Working copy) - lokální kopie repozitáře, nebo konkrétní verze souborů z repozitáře
 - Uživatel pracuje s kopiemi verzovaných souborů, které modifikuje

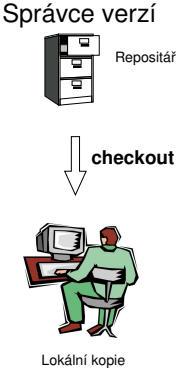
U konkrétních systémů verzování můžeme dále rozlišit lokální a pracovní kopii Například subversion udržuje v pracovní kopii ještě adresář .svn, ve kterém je lokální kopie konkrétní verze repozitáře, na které uživatel pracuje. Git má lokální verzi repozitáře v adresáři .git

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 5 / 63

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Získání lokální kopie - checkout

- Vytvoření lokální kopie verzovaných souborů
- Adresářová struktura většinou obsahuje pomocné soubory s informacemi o verzi souborů
- Při změnách modifikujeme lokální kopii příslušné verze



Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 8 / 63

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Přehled témat

Základní pojmy verzování souborů

SVN - Subversion – vybrané pokročilé vlastnosti

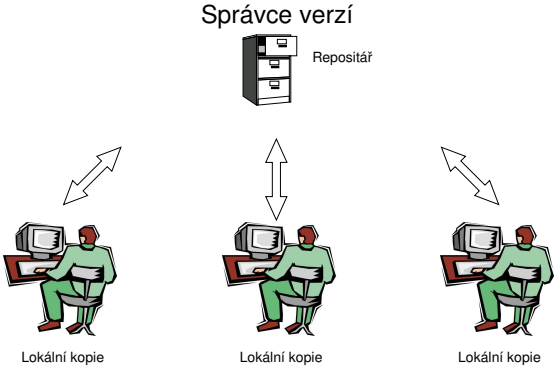
Distribuované verzovací systémy – základní filozofie

Verzování

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 2 / 63

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Základní schema



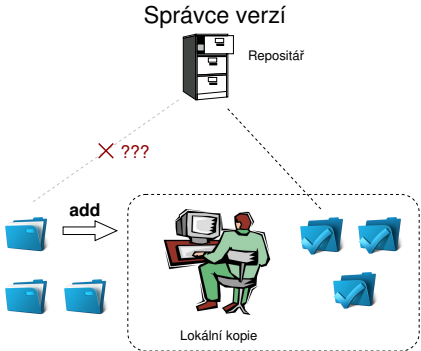
Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 6 / 63

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Získání lokální kopie - add

- Soubory je nutné zařadit do správy verzí

Před přidáním neví správce verzí, zdali chceme soubor verzovat nebo nikoliv.

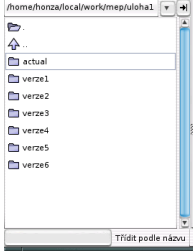


Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 9 / 63

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Proč používat správce verzí

- Zálohování starších verzí pro „jistotu“
- Vyzkoušení nového směru bez ztráty původní verze
- Jak distribuovat soubory při více členěm týmu?



Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 4 / 63

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Správce verzí

- Sada nástrojů (příkazů) pro interakci s repozitářem a lokálními soubory (kopiemi)

Nástroj je zpravidla příkaz nebo ikona ke kliknutí či položka v menu.

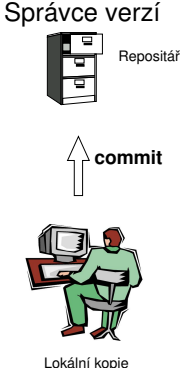
- Lokální příkazy nebo v případě repozitáře též „serverová“ služba

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 7 / 63

VCS SVN - Subversion Distribuované verzovací systémy – DVCS Verzování

Potvrzení změn - commit

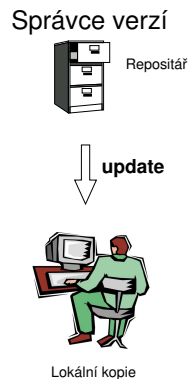
- Žádost o přijmutí lokálních modifikací jako nové verze
- Správce verzí vytvoří nejbližší vyšší verzi
- V repozitáři není novější verze, než lokální kopie, jinak:
 - Aktualizace lokální kopie na aktuální novou verzi
 - Řešení konfliktů
- Je slušné změny komentovat



Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 10 / 63

Aktualizace - update

- Aktualizace lokální kopie na novou verzi
- Pokud jsou změny verzovaných souborů v souladu s lokálními modifikacemi probíhá sloučení změn (merge)
- Jinak řešení konfliktů



Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 11 / 63

Řešení konfliktů

- **Správce verzí** nezabraňuje vzniku konfliktů, ale nabízí **nástroje pro jejich řešení**
- Konflikt vzniká převážně současnou modifikací stejného místa v souboru
- Konfliktům lze zabránit vhodným rozčleněním projektů na moduly a případně vyhrazení práv modifikací jednotlivým vývojářům.

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 12 / 63

Sloučený soubor s konfliktem

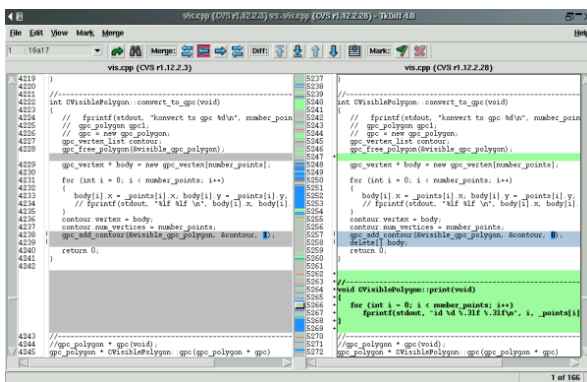
```

1169     fprintf(stdout, "%d [%.31f, %.31f]\n", i,
1170     }
1171     <<<<<< vis.cpp
1172     G=12*cities.number;
1173     //G=12.41*4+0.06;
1174     =====
1175     G=12.41*cities.number+0.06;
1176     >>>>>> 1.12.2.48
1177     separate = false;
1178     return 0;
1179 }
1180
1181 //-----
1182 int CMap::coords_size(double * min_x, double * m
1183 {

```

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 13 / 63

Vizualizace rozdílů



Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 14 / 63

Značkování verzí - tag

- Správce verzí zachycuje historii vývoje jednotlivých souborů
- Konkrétní stav repozitáře, lze označit (tag) např. Release_1.0
- Tag - symbolické jméno pro konkrétní verzi
- Pro aktuální verzi se používá značka HEAD

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 15 / 63

Větvě - branch

Umožňují:

- Paralelní vývoj
- Postupný přechod na novější technologie
- Odkoušení nových přístupů

Používané větvě:

- CURRENT, TRUNK - hlavní vývojová větev
- STABLE - stabilní vývojová větev

Commit do STABLE větvě by neměl narušit činnost ostatních vývojářů.

- Hlavní vývojová větev může být zároveň stable větví

V případě rozvětveného vývoje lze s výhodou využít slučování větví (branch merge)

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 16 / 63

Příklad více větví



Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 17 / 63

CVS - Concurrent Version System

- Jeden z prvních rozšířených systémů verzování
Pro lokální verzování např. souborů v /etc lze použít ještě jednodušší systém rcs
- Každý soubor je číslován zvlášť
- Verze lze označit (tag), existují dynamické a statické tagy
- Rozlišuje mezi adresářem a souborem
- Ze serveru jsou posílány změnové soubory (diff), na server jsou však uploadovány celé soubory
- Implicitně pracuje s textovými soubory, binární soubory je nutné označit
- Náročnější slučování větví
- Obtížný přesun adresářů či přejmenování souborů

Trocha historie

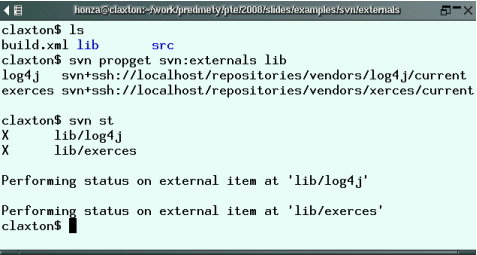
Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 18 / 63

Správce verzí

- Source Code Manager – SCM
- Existuje celá řada nástrojů pro správu verzí jako proprietární i jako svobodný software
https://en.wikipedia.org/wiki/Comparison_of_version_control_software
- Není jen CVS, Git, nebo Subversion (SVN)
 - Bazaar – **bzr**
 - Darcs – **darcs**
 - Mercurial – **hg**
 - Perforce – *proprietary software*
 - Plastic SCM – *proprietary software*
<https://www.plasticscm.com>
 - Surround SCM – *proprietary software*
<http://www.seapine.com/surround-scm/overview>

Je dobré mít povědomí o existujících řešeních, jejich možnostech a také omezeních. Rozhled a znalosti základních principů nám mohou pomoci si správně vybrat.

Jan Faigl, 2016 A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů 19 / 63

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
svn:externals - dokaz do jiného repozitáře			
■ Kombinace zdrojů z více repozitářů. ■ Každý řádek definuje jeden adresář a jeho zdroj. Příklad - knihovny třetích stran 			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	30 / 63	

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
Tags - značky - 1/2			
■ Explicitní označení stavu repozitáře v čase. ■ Realizovány kopií adresáře (mechanismus levných kopií). Příklad - značka <i>version1</i> <pre>1 \$ svn info 2 Path: . 3 URL: https://comrob/svn/pte1430/ 4 Revision: 1 5 Node Kind: directory 6 Last Changed Rev: 1 7 \$ ls 8 dijkstra 9 \$ ls dijkstra 10 build.xml src 11 \$ ls dijkstra/src/ 12 Dijkstra.java 13 \$ svn mkdir -m "Create tags directory" \ 14 https://comrob/svn/pte1430/tags 15 Committed revision 2. 16 \$ svn copy -m "Create tag of Dijkstra" \ 17 https://comrob/svn/pte1430/dijkstra \ 18 https://comrob/svn/pte1430/tags/version1 19 Committed revision 3.</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	33 / 63	

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
Příklad adresářové struktury			
Celý projekt <pre>/dijkstra /dijkstra/trunk /dijkstra/branches /dijkstra/tags /dijkstra/trunk/src /dijkstra/branches/linear /dijkstra/branches/heap_naive /dijkstra/tags/submission /dijkstra/tags/static_arrays /dijkstra/branches/linear/src</pre> Každý modul <pre>/common /gui /readlog /polygon_filter /common/trunk /common/branches /common/tags /gui/trunk /gui/branches /gui/tags /readlog/trunk /readlog/branches /readlog/tags</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	36 / 63	

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
Vlastnosti vs Log zprávy			
■ Vlastnosti lze s výhodou využít pro jednoznačné nastavení konkrétní proměnné. ■ Systémové vlastnosti (proměnné) obsahují údaje, které není nutné psát do zpráv při <i>commitu</i>. ■ Přes výše uvedené může být vhodné informace uvést i ve zprávě, neboť vlastnosti se relativně obtížně prohledávají. ■ Formát log zprávy lze předepsat pro snadnější parsování <i>log-message templating</i>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	31 / 63	

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
Tags - značky - 2/2			
Příklad - značka <i>version1</i> - aktualizace pracovního adresáře 2/2 <pre>1 \$ svn up 2 A tags 3 A tags/version1 4 A tags/version1/src 5 A tags/version1/src/Dijkstra.java 6 A tags/version1/build.xml 7 Updated to revision 3. 8 # výpis adresářové struktury 9 \$ svn ls --depth infinity 10 dijkstra/ 11 dijkstra/build.xml 12 dijkstra/src/ 13 dijkstra/src/Dijkstra.java 14 tags/ 15 tags/version1/ 16 tags/version1/build.xml 17 tags/version1/src/ 18 tags/version1/src/Dijkstra.java</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	34 / 63	

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad I. 1/6			
Historie revizí: - vytvoření adresářové struktury, - první verze souborů, - nová větev pro lineární prohledávání <pre>\$ svn copy dijkstra/trunk dijkstra/branches/linear \$ svn commit</pre> - implementace načítání v trunk <pre>\$ svn ci -m "Implement load_graph" Adding dijkstra/trunk/src/load_graph.c Adding dijkstra/trunk/src/load_graph.h Transmitting file data .. Committed revision 4.</pre> - nová větev pro haldu <pre>\$ svn copy dijkstra/trunk dijkstra/branches/heap A dijkstra/branches/heap \$ svn ci -m "Create heap branch" Adding dijkstra/branches/heap Committed revision 5.</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	37 / 63	

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
Log zprávy			
■ Obsah zprávy by měl vystihovat podstatu změny. ■ Rozsáhlé změny by měly být dostatečně komentovány. ■ Zprávy mají také charakter komunikace mezi vývojáři. ■ Zprávy lze mimo jiné využít pro vytváření souboru <i>ChangeLog</i>. ■ Většina příkazů svn má možnost výstupu do xml včetně svn log. Příklad XML výstupu <pre>\$ svn -r 62:62 log --xml <?xml version="1.0"?> <log> <logentry revision="62"> <author>kordam2</author> <date>2008-12-10T19:59:36.941186Z</date> <msg>mala uprava.</msg> </logentry> </log></pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	32 / 63	

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
Adresářová struktura			
■ Větvě i značky jsou adresáře. ■ V repozitáři může být více projektů. ■ Projekt je složen z více modulů, každý modul může mít několik vývojových větví. Dva základní přístupy: ■ Větvě (značky) pro celý projekt. ■ Větvě (značky) pro každý modul.			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	35 / 63	

VCS	SVN - Subversion	Distribuidoané verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad I. pokračování 2/6			
- implementace lineárního vyhledávání, <pre>\$ svn add dijkstra/branches/linear/src/linear.? A dijkstra/branches/linear/src/linear.c A dijkstra/branches/linear/src/linear.h \$ svn ci -m "Implementation of linear search" Adding dijkstra/branches/linear/src/linear.c Adding dijkstra/branches/linear/src/linear.h Transmitting file data .. Committed revision 6.</pre> - přidání lineárního vyhledávání do trunk, <pre>\$ svn merge dijkstra/trunk@3 dijkstra/branches/linear@6 dijkstra/trunk --- Merging differences between repository URLs into 'dijkstra/trunk': A dijkstra/trunk/src/linear.h A dijkstra/trunk/src/linear.c \$ vim dijkstra/trunk/src/dijkstra.c \$ svn ci -m "Use linear search" Sending dijkstra/trunk Sending dijkstra/trunk/src/dijkstra.c Adding dijkstra/trunk/src/linear.c Adding dijkstra/trunk/src/linear.h Transmitting file data . Committed revision 7.</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	38 / 63	

VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování	VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování	VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad I. pokračování 3/6				Adresářová struktura - příklad I. pokračování 4/6				Adresářová struktura - příklad I. pokračování 5/6			
8. implementace haldy, <pre>\$ svn ci -m "Heap has been implemented and tested" Adding dijkstra/branches/heap/src/heap.c Adding dijkstra/branches/heap/src/heap.h Transmitting file data .. Committed revision 8.</pre> 9. použití haldy v trunk, <pre>\$ svn merge dijkstra/trunk@5 dijkstra/branches/heap@8 dijkstra/trunk --- Merging differences between repository URLs into 'dijkstra/trunk': A dijkstra/trunk/src/heap.h A dijkstra/trunk/src/heap.c \$ vim dijkstra/trunk/src/dijkstra.c \$ svn ci -m "Replace linear search by heap" Sending dijkstra/trunk Sending dijkstra/trunk/src/dijkstra.c Adding dijkstra/trunk/src/heap.c Adding dijkstra/trunk/src/heap.h Transmitting file data . Committed revision 9.</pre>				10. označení odevzdávaného trunk, <pre>\$ svn copy dijkstra/trunk dijkstra/tags/submission A dijkstra/tags/submission \$ svn propset status TOREVIEW dijkstra \$ svn ci -m "dijkstra has been finished" Sending dijkstra Adding dijkstra/tags/submission Adding dijkstra/tags/submission/src Adding dijkstra/tags/submission/src/dijkstra.c Adding dijkstra/tags/submission/src/heap.c Adding dijkstra/tags/submission/src/heap.h Committed revision 10.</pre>				Přepínání mezi větvemi. <pre>\$ svn info Path: . URL: https://comrob/svn/pte1430/dijkstra Repository Root: https://comrob/svn/pte1430 Revision: 10 Last Changed Rev: 10 \$ svn switch https://comrob/svn/pte1430/dijkstra/trunk D trunk A src/heap.h D branches A src/main.c D tags A src/load_graph.h A src A src/linear.c A src/linear.h A src/dijkstra.c A src/dijkstra.h A src/heap.c A src/load_graph.c U . Updated to revision 10 \$ svn info Path: . URL: https://comrob/svn/pte1430/dijkstra/trunk Repository Root: https://comrob/svn/pte1430 Revision: 10 Node Kind: directory Last Changed Rev: 9</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		39 / 63	Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		40 / 63	Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		41 / 63
VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování	VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování	VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad I. pokračování 6/6				Adresářová struktura - příklad II. 1/4				Adresářová struktura - příklad II. pokračování 2/4			
Přepnutí na větev heap a zpět na trunk. <pre>\$ svn switch https://comrob/svn/pte1430/dijkstra/branches/heap D src/linear.h D src/linear.c U src/dijkstra.c U . Updated to revision 10 \$ svn info Path: . URL: https://comrob/svn/pte1430/dijkstra/branches/heap Repository Root: https://comrob/svn/pte1430 Revision: 10 Last Changed Rev: 8 \$ svn switch https://comrob/svn/pte1430/dijkstra/trunk A src/linear.h A src/linear.c U src/dijkstra.c U . Updated to revision 10. \$ svn info Path: . URL: https://comrob/svn/pte1430/dijkstra/trunk Repository Root: https://comrob/svn/pte1430 Revision: 10 Last Changed Rev: 9</pre>				Jak vytvořit lokální adresářovou strukturu pro sestavení projektu z více modulů? <pre>common/ \$ svn co https://comrob/svn/sensors common/branches/ A sensors/gui common/trunk/ A sensors/gui/trunk common/trunk/errors.h A sensors/gui/trunk/cairoPainter.c common/trunk/logging.h A sensors/gui/trunk/gui_window.h gui/ A sensors/gui/trunk/cairoPainter.h gui/branches/ A sensors/gui/trunk/gui_window.c gui/trunk/ A sensors/gui/branches gui/trunk/cairoPainter.c A sensors/common gui/trunk/cairoPainter.h A sensors/common/trunk gui/trunk/gui_window.c A sensors/common/trunk/errors.h gui/trunk/gui_window.h A sensors/common/trunk/logging.h readlog/ A sensors/common/branches readlog/branches/ A sensors/readlog readlog/trunk/ A sensors/readlog/trunk readlog/trunk/laser_scan.c A sensors/readlog/trunk/laser_scan.h readlog/trunk/laser_scan.h A sensors/readlog/branches Checked out revision 2.</pre>				Přepnutí jednotlivých modulů na trunk. <pre>\$ cd sensors \$ svn switch https://comrob/svn/sensors/gui/trunk gui D gui/trunk D gui/branches A gui/cairoPainter.c A gui/gui_window.h A gui/cairoPainter.h A gui/gui_window.c Updated to revision 2. \$ svn switch https://comrob/svn/sensors/common/trunk common D common/trunk D common/branches A common/errors.h A common/logging.h Updated to revision 2. \$ svn switch https://comrob/svn/sensors/readlog/trunk readlog D readlog/trunk D readlog/branches A readlog/laser_scan.h A readlog/laser_scan.c Updated to revision 2.</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		42 / 63	Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		43 / 63	Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		44 / 63
VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování	VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování	VCS	SVN - Subversion	Distribuované verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad II. pokračování 3/4				Adresářová struktura - příklad II. pokračování 4/4				Adresářová struktura - příklad III. 1/5			
<pre>\$ svn info Path: . URL: https://comrob/svn/sensors Repository Root: https://comrob/svn/sensors Revision: 2 Last Changed Rev: 2 \$ svn update At revision 4 \$ svn -v log -l 2 ----- r4 standa 2007-10-14 01:10:36 (Sat, 14 Oct 2007) 1 line Changed paths: A /gui/branches/glPainter/glPainter.c A /gui/branches/glPainter/glPainter.h M /gui/branches/glPainter/guiWindow.c implemented glPainter ----- r3 standa 2007-10-13 18:11:13 (Fri, 13 Oct 2007) 1 line Changed paths: A /gui/branches/glPainter (from /gui/trunk:2) Create glPainter branch -----</pre>				<pre>\$ svn switch https://comrob/svn/sensors/gui/branches/glPainter gui A gui/glPainter.c A gui/glPainter.h U gui/guiWindow.c Updated to revision 4. \$ svn info common Path: common URL: https://comrob/svn/sensors/common/trunk Repository Root: https://comrob/svn/sensors Revision: 4 Last Changed Rev: 2 \$ svn info gui Path: gui URL: https://comrob/svn/sensors/gui/branches/glPainter Repository Root: https://comrob/svn/sensors Revision: 4 Last Changed Rev: 4 \$ svn info readlog/ Path: readlog URL: https://comrob/svn/sensors/readlog/trunk Repository Root: https://comrob/svn/sensors Revision: 4 Last Changed Rev: 2</pre>				■ Samostatný repositář pro knihovny třetích stran. <pre>\$ svn --depth infinity ls https://comrob/svn/vendors log4j/ log4j/1.2.13/ log4j/1.2.13/log4j.jar log4j/1.2.9/ log4j/1.2.9/log4j.jar log4j/current/ log4j/current/log4j.jar xerces/ xerces/2.6.2/ xerces/2.6.2/xercesImpl.jar xerces/2.6.2/xml-apix.jar xerces/2.7.1/ xerces/2.7.1/xercesImpl.jar xerces/2.7.1/xml-apix.jar xerces/current/ xerces/current/xercesImpl.jar xerces/current/xml-apix.jar</pre> ■ V projektu nastavíme property svn:externals.			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		45 / 63	Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		46 / 63	Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů		47 / 63

VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad III. pokračování 2/5			
■ Aplikace pro načítání xml. <pre>\$ svn --depth infinity ls https://comrob/svn/pte1430 task2_xml/ task2_xml/trunk/ task2_xml/trunk/libs/</pre> ■ V projektu používáme knihovny <i>log4j</i> a <i>xerces</i>, v <i>trunk</i> budeme používat verze 1.2.9 a 2.6.2. <pre>log4j https://comrob/svn/vendors/log4j/1.2.9 xerces https://comrob/svn/vendors/xerces/2.6.2</pre> ■ Nastavíme property <i>svn:externals</i>. <pre>svn propedit svn:externals https://comrob/svn/pte1430/task2_xml/trunk/ libs -m "Set vendors libs" Set new value for property 'svn:externals' on 'https://comrob/svn/pte1430/task2_xml/trunk/libs' Committed revision 2.</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	48 / 63	

VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad III. pokračování 5/5			
■ Přepneme na novou větev. <pre>svn switch https://comrob/svn/pte1430/task2_xml/ branches/new_vendor_libs pte1430/task2_xml/ D pte1430/task2_xml/trunk A pte1430/task2_xml/libs Fetching external item into 'pte1430/task2_xml/libs/log4j' A pte1430/task2_xml/libs/log4j/log4j.jar Updated external to revision 4. Fetching external item into 'pte1430/task2_xml/libs/xerces' A pte1430/task2_xml/libs/xerces/xml-apix.jar A pte1430/task2_xml/libs/xerces/xercesImpl.jar Updated external to revision 4. Updated to revision 5. \$svn info pte1430/task2_xml/libs/xerces/ Path: pte1430/task2_xml/libs/xerces URL: https://comrob/svn/vendors/xerces/2.7.1 Repository Root: https://comrob/svn/vendors Revision: 4 Last Changed Rev: 3</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	51 / 63	

VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Distribučované verzovací systémy – (DVCS)			
■ Nemají centrálně uložený repozitář. ■ Každý vývojář si v podstatě udržuje vlastní <i>lokální</i> repozitář. ■ Velmi časté používání vývojových větví. ■ Výsledná verze je vlastně sestavením příslušných vývojových větví jednotlivých vývojářů. ■ Příklad existujících DVCS: ■ BitKeeper – <i>proprietary software</i> ■ Bazaar – bzr ■ Perforce – <i>proprietary software</i> ■ Git – git ■ Darcs – darcs ■ Mercurial - hg ■ Plastic SCM – <i>proprietary software</i>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	55 / 63	

VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad III. pokračování 3/5			
■ Provedeme <i>checkout</i>. <pre>\$ svn checkout https://comrob/svn/pte1430 A pte1430/task2_xml A pte1430/task2_xml/trunk A pte1430/task2_xml/trunk/libs Fetching external item into 'pte1430/task2_xml/trunk/libs/log4j' A pte1430/task2_xml/trunk/libs/log4j/log4j.jar Checked out external at revision 4. Fetching external item into 'pte1430/task2_xml/trunk/libs/xerces' A pte1430/task2_xml/trunk/libs/xerces/xml-apix.jar A pte1430/task2_xml/trunk/libs/xerces/xercesImpl.jar Checked out external at revision 4. Checked out revision 2.</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	49 / 63	

VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Hooks			
■ Při interakci s repozitářem je možné definovat skripty, které provedou příslušnou operaci. ■ Adresářová struktura repozitáře (na serveru) obsahuje adresář hooks. ■ Pro konkrétní událost je volán skript s konkrétním jménem: např. post-commit, pre-commit. ■ Skript je volán s definovanými argumenty, např. cesta k repozitáři a číslo revize. ■ Pro přístup k repozitáři lze použít program svnlook. ■ Po vytvoření repozitáře, např. příkazem svnadmin, obsahuje adresář šablony pro jednotlivé skripty (hooks).			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	52 / 63	

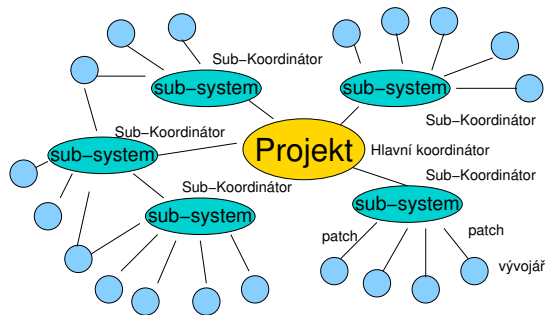
VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Vlastnosti			
■ Můžeme verzovat i bez připojení k síti. ■ Centrální repozitář je často nahrazen zodpovědným vývojářem. ■ Vyžadují hlubší porozumění struktuře projektu. ■ Vhodnost nasazení záleží na povaze projektu (modelu vývoje).			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	56 / 63	

VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Adresářová struktura - příklad III. pokračování 4/5			
■ Přechod na nové verze knihoven provedeme v samostatné větvi. <pre>\$ svn mkdir https://comrob/svn/pte1430/task2_xml/branches -m "Create branch directory" Committed revision 3. \$ svn copy https://comrob/svn/pte1430/task2_xml/trunk https://comrob/svn/pte1430/task2_xml/branches/new_vendor_libs/ -m "Create branch for new libs" Committed revision 4.</pre> ■ Použijeme verze 1.2.13 a 2.7.1. <pre>#log4j https://comrob/svn/vendors/log4j/1.2.13 #xerces https://comrob/svn/vendors/xerces/2.7.1 \$ svn propedit svn:externals https://comrob/svn/pte1430/task2_xml/branches/new_vendor_libs/lib</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	50 / 63	

VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Příklad - post-commit hook			
<pre>#!/bin/sh REPOSITORY="\$1" REV="\$2" SVNLOOK=/usr/local/bin/svnlook CHANGES='\$SVNLOOK changed \$REPOSITORY grep -e'^_U[[:space:]]+' uloha [[::digit:]]/\$' STUDENTS_MAILS='\$SVNLOOK pg \$REPOSITORY mails / tr '\n' ' ' for i in \$CHANGES do STATUS='\$SVNLOOK pg \$REPOSITORY status \$i' case \$STATUS in TOREVIEW) #send mail ;; CHECKED) if [-n "\$STUDENTS_MAILS"] #mails are not set then RESULT='\$SVNLOOK log \$REPOSITORY' #send mail with \$RESULT fi ;; esac done</pre>			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	53 / 63	

VCS	SVN - Subversion	Distribučované verzovací systémy – DVCS	Verzování
Některé distribuované verzovací systémy			
■ BitKeeper - http://www.bitkeeper.com. ■ darcs - David's Advanced Revision Control System - napsaný v haskell http://darcs.net. ■ Monotone - http://monotone.ca. ■ Mercurial - http://www.selenic.com/mercurial/wiki ■ SVK - založeno na subversion filesystem knihovně http://svk.bestpractical.com/view/HomePage. ■ git - vytvořen pro potřeby vývoje jádra Linux http://git-scm.com			
Jan Faigl, 2016	A0B36PR2 – Přednáška 11: Úvod do verzovacích systémů	57 / 63	

Model vývoje s velkým počtem vývojářů



GIT - základní vlastnosti

- Lokální repositář.
- Efektivní pro rozsáhlé projekty.
- Soubory jsou uloženy jako objekty v databázi (INDEX).
- SHA1 otisk souboru slouží jako identifikátor souboru.
- Low-level operace nad databází jsou zapouzdřeny uživatelsky přívětivějším rozhraním.
- Výrazná podpora pro vývojové větve.
- Podpora pro aplikování patch setů, např. z mailu.
- Základní příkazy <http://git-scm.com/documentation>
- Git - SVN Crash Course <http://git-scm.com/course/svn.html>

Základní použití

- Vytvoření kopie repositáře pro udržování vlastních změn; git clone.
- Sledování jiných repositářů; git remote, git fetch.
- Vlastní vývoj a lokální verzování; git add, git status, git log, git merge, git branch, git checkout.
- Publikování změn do jiného (vzdáleného) repositáře; git push.

GIT - základní příkazy

git init	svnadmin create repo
git clone url	svn checkout url
git add file	svn add file
git commit -a	svn commit
git pull	svn update
git status	svn status
git log	svn log
git rm file	svn rm file
git mv file	svn mv file
git tag -a name	svn copy repo/trunk repo/tags/name
git branch branch	svn copy repo/trunk repo/branches/branch
git checkout branch	svn switch repo/branches/branch

Co všechno verzovat?

- Verzování zdrojových kódů programů.
- Verzování knihoven třetích stran.
- Verzování dokumentů (text/binární).
 - File and Directory Layout for Storing a Scientific Paper in Subversion
<http://blog.plesslweb.ch/post/6628076310/file-and-directory-layout-for-storing-a-scientific>
- Verzování jako „zálohování”.
Repositář je na serveru zpravidla uložena na zálohovaném diskovém systému.
- Verzování jako prostředek sdílení.

Užívat rozumně!