

Praktinis darbas su Subversija

Saulius Gražulis

2009 ruduo

Vilniaus universitetas, Matematikos ir informatikos fakultetas
Informatikos institutas

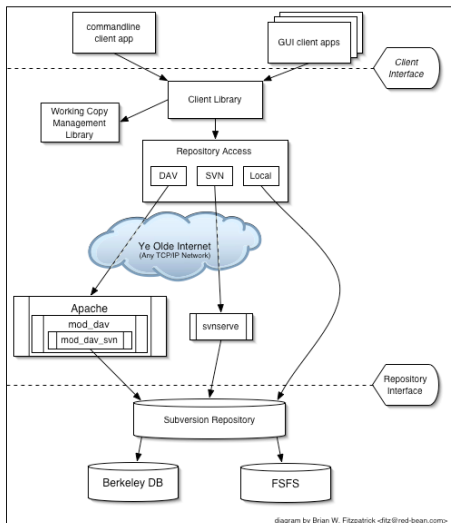


Ši skaidrių rinkinį galima kopijuoti, kaip nurodyta Creative Commons
[Attribution-ShareAlike 4.0 International](https://creativecommons.org/licenses/by-sa/4.0/) licenzijoje



Subversijos architektūra¹

Subversija sukonstruota pagal kliento-serverio architektūrą



¹ Šis paveikslėlis paimtas iš Ben Collins-Sussman, Brian W. Fitzpatrick, C. Michael Pilato *Version Control with Subversion* e-knygos, <http://svnbook.red-bean.com/>, 2009-10-27

- *x sistemos dokumentacija man nukreipia mus į vidinę Subversijos dokumentacijos (help) sistemą

```
man svn
svn help
```

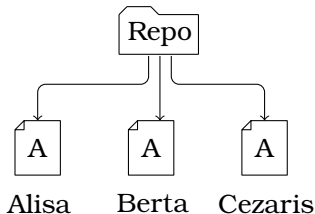
- Skaitykime klaidų pranešimus – Subversija bando paaiškinti, kas nutiko ir kaip pataisyti situaciją:

```
saulius@vanagas proj2/ > rm -rf python/
saulius@vanagas proj2/ > svn revert -R .
Failed to revert 'python' -- try updating instead.
saulius@vanagas proj2/ > svn update
A      python
Updated to revision 12.
```

- Use the Source, Luke! – Subversija yra atviro kodo, laisva programa.

Darbinės kopijos sukūrimas

Prieš pradėdant darbą, reikia atsikelti informaciją iš repozitorijos



- Kiekvienas projekto dalyvis įvykdo komandą, panašią į žemiau parašytą:

```
svn co svn://repo.project.lt/projektas/trunk projektas
```

Repozitorijų adresai

Repozitorija pasiekama nurodant jos URL

- Repozitoriją galime pasiekti skirtingais protokolais:
 - Naudojant failų sistemą tiesiogiai (file):
file://localhost/path/to/repo/
file:///path/to/repo/
 - Subversijos nuosavu protokolu (svn):
svn://server.addr.org/path/to/repo/
 - Subversijos protokolu per SSH tunelį (svn+ssh):
svn+ssh://server.addr.org/path/to/repo/
 - Per naudotojo apibrėžtus tunelius (svn+rsh ir kt.)²:
svn+rsh://server.addr.org/path/to/repo/
svn+ssh2222://server.addr.org/path/to/repo/
 - HTTP protokolu (http, https):
http://server.addr.org/path/to/repo/

² tuneliai apibrėžiami konfigūracijos faile `~/.subversion/config`

Repozitorijos sukūrimas

Kiekvienas gali sukurti Subversijos repozitoriją!

- Repozitorija sukuriamą komanda `svnadmin create`

```
mkdir ~/svn-repositories  
svnadmin create ~/svn-repositories/my-project
```

- Komanda `svnadmin` taip pat naudojama kitiems administravimo darbams (tikrinimui, informacijos perkėlimui į kitą repozitoriją ir t.t.). Smulčiau žr.:

```
svnadmin help
```

Projekto importas

Pradėti naudoti versijų kontrolę niekada nevēlu

- Importuoti projektą į Subversiją galime dviem būdais:
 - naudodami specialią komandą `svn import`:

```
cd ~/projektas
svn import svn://repo.server.dom/project/trunk
```

Importuojant šiuo būdu, projekto direktorija *nepavirsta* darbine kopija; darbinę kopiją reikia išsikelti kitoje vietoje!

- naudojant `svn co` į darbinę direktoriją:

```
cd ~/projektas
svn co svn://repo.server.dom/project/trunk/ .
```

Direktorija `~/projektas/` pavirsta darbine kopija. Geriausiai tinka naujai sukurtam, tuščiam `svn://svn.server.dom/project/trunk/`.

Darbo su versijų kontrolės sistema ciklas

“Update every morning, commit every hour!”

- Standartinis programos kūrimo mikrociklas papildomas vienu žingsniu:

Edit → Compile → Link → Run/Test > Commit
↑

```
cd projektas
svn up
edit somefile.txt
svn diff
make
make test
svn ci

edit anotherfile.txt
svn diff
svn diff -r2
svn revert anotherfile.txt
```


Failų pridėjimas ir ištrynimasis

Įkelta į repozitoriją – kaip iškalta akmeny...

- Naujus failus galime įkelti į repozitoriją komanda `svn add`:

```
svn add naujas-failas.txt
svn ci
```

- Failus iš darbinės kopijos ištriname komanda `svn rm` (sinonimas `svn del`):

```
svn del nereikalingas-failas.txt
svn ci
```

- **Pastaba:** failas ištrinamas iš darbinės kopijos, bet repozitorijoje informacija lieka. Neįkėlinėkite į repozitoriją konfidencialios informacijos (slaptažodžių, asmeninių laiškų, nesklebtinos dokumentacijos ir pan).

- Pakeistų failų sąrašą galime pamatyti komanda `svn status` (sinonimas `svn st`):

```
saulius@vanagas proj2/ > svn st
?      strange.txt
D      python
A      new.pl
M      source.py
A +    source.pl
```

- Komanda `svn diff` parodys, kokie pasikeitimai įvyko po paskutinio įkėlimo (`commit`) arba tarp nurodytų revizijų:

```
svn diff source.pl
svn diff
svn diff -r10:22
```

Pakeitimų atstatymas

Subversija duoda mums neribotą “undo”!

- Padarytų pakeitimų galite atsisakyti komanda `svn revert`, kol jų neįkėlėte į repozitoriją:

```
svn revert pakeistas-failas.txt
svn revert -R .
```

- Laikintai atstatyti kokios nors senos revizijos failų medį ar atskirus failus leidžia komanda `svn update` (sinonimas `svn up`):

```
svn update -r123 failas.txt
svn up -rPREV
```

Komanda `svn up` be revizijos numerio atstatys pačią naujausią reviziją (`-rHEAD`).

- Atšaukti kurios nors revizijos pakeitimus padeda `svn merge`

```
svn merge -r321:123 failas.txt; svn ci
svn merge -rHEAD:PREV .; svn ci
```

Failų kopijavimas, pervadinimas

Iš kur mes atėjome ir kurlink einame?

- Failus kopijuoti ir pervadinti geriausia specialiomis subversijos komandomis:

```
svn cp failas1.txt failas2.txt
svn mv senas-vardas.c naujas-vardas.c
svn ci
```

Panaudojus šias komandas, repozitorijoje lieka informacija apie tai, koks failas buvo nukopijuotas ar pervadintas, ir nuorodos į senus šių failų žurnalinius įrašus.

- Standartinės *x komandos veikia, bet prarandama istorija:

```
cp failas1.txt failas2.txt
mv senas-vardas.c naujas-vardas.c
svn add failas2.txt naujas-vardas.c
svn del senas-vardas.c
svn ci
```

Darbas su direktorijomis

- Direktorija sukuriama specialia komanda:

```
svn mkdir nauja-direktorija
svn ci
```

- Direktorijos kopijuojamos, trinamos, pervadinamos (rename) bei perkeliamos (move) tomis pat komandomis, kaip ir paprasti failai:

```
svn cp esanti-dir nauja-dir
svn mv buvusi-dir kita-dir
svn rm sena-dir
svn ci
```

- ištrintai direktorijai atstatyti gali neužtekti komandos `svn revert`, gali tekti naudoti `svn update`

Žurnaliniai įrašai

Kompiuteriai saugo tik tą informaciją, kurią įvedė žmonės

- Kiekvieno revizijos įkėlimo (checkin, commit) metu Subversijos klientas paleidžia redaktorių ir prašo įvesti žurnalinį įrašą (log record).
- Naudojamą redaktorių galima nustatyti argumentu (option) `--editor-cmd` arba nustatčius aplinkos kintamuosius `EDITOR` arba `SVN_EDITOR`:

```
svn ci --editor-cmd=nano
export SVN_EDITOR=nano
svn ci
```

- Žurnalinį įrašą taip pat galima paimti iš failo arba iš komandų eilutės:

```
svn ci -F change-log.txt
svn ci -m "Removing unused code from 'math.c'."
```

- Žurnalą galima peržiūrėti komanda `svn log`:

```
svn log
svn log -r12456:12500 vienas-failas.txt
```

- Komanda `svn log -v` parodo ne tik žurnalinius įrašus, bet ir pakeistų (modifikuotų, pridėtų, ištrintų, nukopijuotų ar pervadintų) failų sąrašą.
- Žurnalas – tai žmogaus parašyta projekto istorija. Netingėkit pasakyti, ką padarėt, kitiems projekto dalyviams!
- Tas kitas dalyvis būsite Jūs pats ar pati po kelių mėnesių ar kelių metų.