

Subversion - initiation

Guillaume Allègre - François Gannaz

SILECS

2007-2008 - Rev: 262

Subversion - Présentation 1/2

- ▶ Gestion des versions d'un projet
 - ▶ issu du développement : fichiers **textes** et binaires
 - ▶ archive les versions *successives* d'un projet
 - ▶ stocke les versions *parallèles* d'un projet (branches)
 - ▶ facilite le travail collaboratif *asynchrone*
 - ▶ sans verrou (contr. RCS), sans validation (contr. [git...](#))
 - ▶ centralisé (clients - serveur)

- ▶ Le contexte - avant
 - ▶ WebDAV : *Web-based Distributed Authoring and **V**ersioning*
 - ▶ Les ancêtres
 - ▶ RCS : Revision Control System
 - ▶ CVS : Concurrent Versioning System
 - ▶ Les bases : commandes Unix [diff](#) et [patch](#)
 - ▶ La communauté <http://www.tigris.org>

Subversion - Présentation 2/2

- ▶ Des “extensions” à Subversion
 - ▶ *SVN utilities* : base des développements futurs...
 - ▶ Trac : SVN + gestion de tickets (Python)
<http://trac.edgewall.org/>
 - ▶ SVK : “SVN décentralisé” (Perl)
<http://svk.bestpractical.com/>
 - ▶ FSVS : FileSystem VerSioning <http://fsvs.tigris.org/>
- ▶ Les références
 - ▶ Le site <http://subversion.tigris.org/>
 - ▶ l'aide en ligne : `svn help ...`
 - ▶ la référence : *Version Control with Subversion* aka “SVN Book”
<http://svnbook.red-bean.com/>
 - ▶ la FAQ : <http://subversion.tigris.org/faq.html>
 - ▶ tutoriel L. Meister en français
<http://www.codiciel.fr/gestion/subversion/>

Rappel : diff, patch...

- ▶ La commande `diff`
 - ▶ calcule la différence entre deux fichiers/répertoires
 - ▶ `diff (-b -u) fic1 fic2` : entre deux fichiers
 - ▶ `diff -R -b -u Rep1 Rep2` : entre deux répertoires
- ▶ La commande `patch`
 - ▶ applique le diff à un fichier/répertoire
 - ▶ éventuellement de façon intelligente (*fuzz*, contexte)
 - ▶ `patch -p0 < patchfile`
- ▶ Compléments
 - ▶ patchutils : `lsdiff`, `grepdiff`...
 - ▶ `xdelta` : un diff pour les binaires

Diff simple

- ▶ Récupérer archive :
`http://hulotte/form-svn/projet.tgz`
- ▶ Extraction du diff
 - ▶ Modification d'un fichier simple
 - ▶ une interface graphique : `tkdiff`
 - ▶ `diff` simple : `-b, -w`
 - ▶ `diff` contextuel : `-c, -u`
- ▶ Application du résultat
 - ▶ application "à la main"
 - ▶ application par `patch`
 - ▶ Emacs : mode `ediff`, `epatch`
 - ▶ `diff` + application combinés : `sdiff -o fic fic1 fic2`

Diff récursif (ou multi-fichiers)

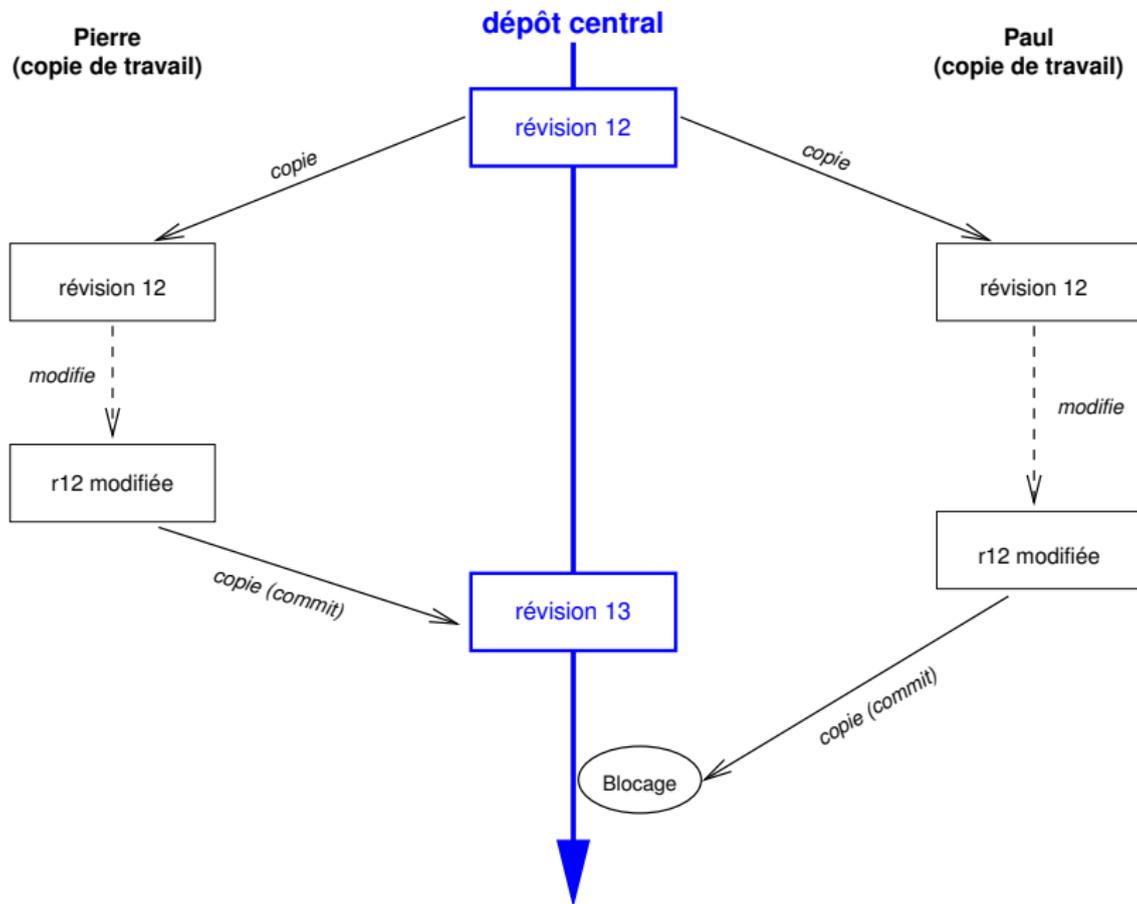
- ▶ Extraction du diff
 - ▶ Modification de plusieurs fichiers du répertoire
 - ▶ `diff -r Rep1 Rep2` : récursif
- ▶ Application du résultat
 - ▶ application par `patch -pN <`
 - ▶ sécurité : `--dry-run`, `--backup`
 - ▶ utilitaire `lsdiff`
 - ▶ gestion des imprécisions (*fuzz*) : `--fuzz=N`,
`--ignore-whitespace`

Subversion - Généralités

- ▶ Les interfaces utilisateur
 - ▶ ligne de commande ; ex : `svn log -v`
 - ▶ mode SVN intégré à l'éditeur : Emacs, vim, Eclipse...
 - ▶ interface graphique indépendante : eSVN, RapidSVN, TortoiseSVN...
 - ▶ interface web : CVSweb (Perl), WebSVN (php)...
 - ▶ intégré aux plateformes SourceForge, GForge...

- ▶ Évolution rapide mais stable
 - ▶ 1.5.0 (19 juin 2008)
 - ▶ 1.4.0 (sep. 2006)
 - ▶ 1.3.0 (jan. 2006)
 - ▶ 1.2.0 (mai 2005)
 - ▶ 1.1.0 (sep. 2004)
 - ▶ 1.0.0 (fév. 2004)
 - ▶ 0.17 (jan. 2003) → 0.37 (jan. 2004)

- ▶ a progressivement remplacé CVS dans la communauté LL/OS

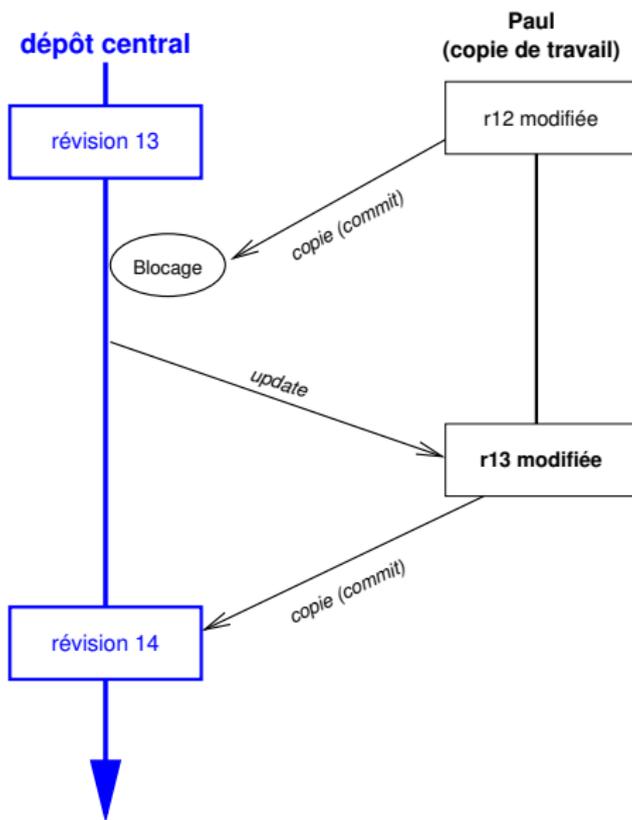


Exemple d'une copie de travail

- ▶ `svn checkout svn ://hulotte/initiation-svn rep-local`
- ▶ `svn info`
- ▶ `svn log (-v, -rNNN)`
- ▶ `vim index.php ...`
- ▶ `svn status (-u)`
- ▶ `svn diff (-cNNN)`
- ▶ `svn commit (-m 'message')`
- ▶ `svn update`

`svn help` : aide en ligne

Travailler sans conflit



Cycle usuel

- ▶ Initialisation d'une copie locale
`svn checkout`

- ▶ Cycle
 1. Synchroniser avec le dépôt :
`svn update`
 2. Modifier la copie locale...
 3. Vérifier les modifications avant diffusion :
`svn status (-u) / svn diff`
 4. Répercuter les changements :
`svn commit (-m)`

Opérations sur les fichiers

- ▶ `svn add` : ajoutera un fichier au dépôt
- ▶ Variantes des commandes unix usuelles :
 - ▶ `svn rm`
 - ▶ `svn cp` : *cheap copies* (copies a minima)
 - ▶ `svn mv`
 - ▶ `svn mkdir`

Ces modifications sont uniquement sur la copie locale.

Seul `svn commit` modifie le dépôt subversion.
(exception : `svn cp` dans certains cas particuliers)

Commandes utiles

- ▶ Utilisation courante :
 - ▶ `svn status (-u)`
 - ▶ `svn diff`
 - ▶ `svn log`

- ▶ Utilisation moins fréquente :
 - ▶ `svn revert` : annuler les modifications locales
 - ▶ `svn ls`
 - ▶ `svn cat`

- ▶ Autres :
 - ▶ `svn blame` : connaître les auteurs d'un fichier
 - ▶ `svn merge` : fusionner deux branches (`svn diff + patch`)

Mise en pratique : annulation de modification

- ▶ Annuler les modifications locales
`svn revert fichier`
- ▶ Annuler une modification propagée (révision)
 - ▶ Principe : elle restera dans l'historique !
 - ▶ par `svn update -rNNN` → impasse
 - ▶ par `svn cat`
 - ▶ par `svn diff ...`
 - ▶ par `svn merge (-c -NNN)`

Gestion d'un conflit de modifications

- ▶ apparition d'un conflit : lors d'un `update`
 - ▶ U \Rightarrow update (mise à jour) : OK
 - ▶ G \Rightarrow merge (fusion) : OK
 - ▶ C \Rightarrow conflit : cf fichiers créés

- ▶ modification manuelle + `svn resolved`

- ▶ abandon du fichier :
 - ▶ `rm fichier` puis `svn up fichier`
 - ▶ `svn revert fichier`

Organisation canonique du dépôt

- ▶ Numéro de révision global
- ▶ Structure recommandée pour les répertoires :
 - ▶ /trunk/
 - ▶ /branches/
 - ▶ /tags/

Subversion utilise des *cheap copies* (copies paresseuses).

- ▶ un fichier copié est simplement marqué comme tel
- ▶ il est réellement dédoublé en cas de modification...
- ▶ ... ou presque : gestion des diffs

Mise en pratique

- ▶ Exercice...
 - ▶ Créer une branche (`svn cp URL URL (exception!)`)
 - ▶ Passer dedans (`svn switch`)
 - ▶ Modifier un (des) fichier(s)
 - ▶ `commit`
 - ▶ *backport* dans le tronc ?

- ▶ Variante
 - ▶ sans `switch` ?

Propriétés des fichiers : exemples...

Substitution : “marquer” les fichiers

En ligne de commande :

```
svn proplist index.php
```

```
svn propset svn:keywords "ID Date" index.php
```

Dans index.html, insérer :

```
<p>Dernière modification de ce fichier : $Date$</p>
```

Ignorer certains fichiers

```
svn propset svn :ignore -F Fichier Rép
```

Administration : gestion d'un dépôt

L'outil d'administration : `svnadmin`

- ▶ Création du dépôt : `svnadmin create Chemin`
- ▶ Sauvegarde : `svnadmin dump` \longleftrightarrow `svnadmin load`
Possibilité de filtrer le dump : `svndumpfilter`

Auxiliaire : `svnlook`

- ▶ `svnlook tree (--show-ids) Chemin`
- ▶ `svnlook info Chemin`

Les protocoles d'accès : 2 cas de figure

Un dépôt local

- ▶ `svn co file :///depot`
⇒ utilisateurs et permissions système

Un accès réseau : 3 solutions

- ▶ Serveur svnservice : `svn ://depot`
⇒ comptes ad-hoc, auth. faible, permissions fines
- ▶ SSH+svnservice :
`svn+ssh ://pierre@serveur.domaine/depot`
⇒ auth. ssh, comptes système, permissions grossières
- ▶ Apache2 + WebDav : `https ://serveur.domaine/depot`
⇒ auth. Apache(...), comptes ad-hoc, permissions fines.

L'accès svnserve

svnserve simple (sans ssh)

- ▶ Configuration du dépôt
 - ▶ `vim DEPOT/conf/svnserve.conf`
 - ▶ `vim DEPOT/conf/passwd`
- ▶ Lancement du service
 - ▶ `svnserve -d`
 - ▶ option `-r /var/lib/svn`
 - ▶ définitif : ajout à `/etc/services` et `/etc/inetd.conf`
`svn stream tcp nowait svnuser /usr/bin/svnserve`
`svnserve -i`

svn+ssh

`svnserve -t` temporaire lancé par `svn ...`

pas de démon ⇒ rien à configurer.

L'accès Apache / WebDAV

- ▶ Prérequis : Apache 2 + libapache2-svn (mod_dav_svn)
- ▶ Configuration de base

```
LoadModule dav_module .../mod_dav.so
LoadModule dav_svn_module .../mod_dav_svn.so
```

```
<Location /svnrepos>
  DAV svn
  SVNParentPath /var/lib/svn
</Location>
```

- ▶ Authentification basique

```
AuthType Basic
AuthName "Depot SVN"
AuthUserFile /etc/apache2/dav_svn.passwd
Require valid-user
```

```
sudo htpasswd -b -c /etc/apache2/dav_svn.passwd
USER PASSWORD ...
```

Administration : les pannes courantes

- ▶ Mauvaises permissions sur le dépôt
 - ▶ réglages droits Apache/WebDAV
 - ▶ droits pour `svn+ssh` : : encapsulation dans un wrapper
⇒ *SVN Book, 6.6 Supporting Multiple Access Methods*
- ▶ Incohérence du dépôt
 - ▶ théoriquement inutile depuis la v.1.2
 - ▶ `svnadmin recover /chemin/depot`
 - ▶ puis vérifier les permissions

Crédits

► Licence

Ce document est sous licence *Creative Commons, Attribution-Share alike 3.0*

<http://creativecommons.org/licenses/by-sa/3.0/>

Conformément à la philosophie du logiciel libre, vous êtes encouragés à réutiliser, reproduire et modifier ce document, sous les conditions de la licence.

J'accepte volontiers les remarques, suggestions d'améliorations, corrections et contributions à ce document.

► Crédits

- François Gannaz, 2006-2007 pour la version initiale
- Guillaume Allègre, 2007-2008 pour la version actuelle
- © SiLECS SARL, contact@silecs.info