

Programmation WEB

Introduction à SVN

Programmation licence

IUT de Fontainebleau

26 septembre 2014

- 1 Introduction
- 2 Notions de Bases
- 3 Cycle d'utilisation
- 4 Gestion des fichiers
- 5 Outils

- 1 Introduction
- 2 Notions de Bases
- 3 Cycle d'utilisation
- 4 Gestion des fichiers
- 5 Outils

SVN ?

Subversion est un logiciel de gestion de sources et de contrôle de versions. Ce type de programmes a plusieurs fonctions, notamment :

- Garder un historique des différentes versions des fichiers d'un projet ;
- Permettre le retour à une version antérieure quelconque ; Revert
- Garder un historique des modifications avec leur nature, leur date, leur auteur... ;
- Permettre un accès souple à ces fichiers, en local ou via un réseau ;
- Permettre à des utilisateurs distincts et souvent distants de travailler ensemble sur les mêmes fichiers.

<https://subversion.apache.org/>

Points forts

- Multiplateforme ;
- Logiciel libre ;
- Centralisée ;
- Utilisation et son administration faciles ;
- Plusieurs modes d'accès distants, dont SSH et WebDAV via Apache.

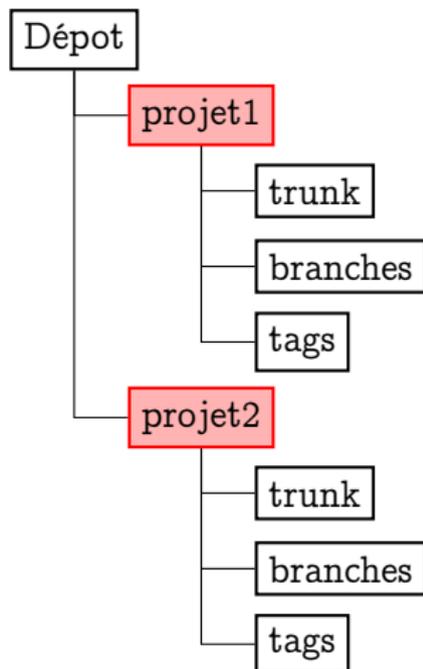
Dépôt

Emplacement central où sont stockées toutes les données relatives aux projets gérés. Le dépôt est accédé via une URL locale ou distant.

Projets

Au sein d'un dépôt se trouvent un ou plusieurs projets. À chaque projet correspond en général un répertoire situé à la racine du dépôt et qui contient lui-même les fichiers et dossiers du projet proprement dit.

Arborescence classique



Copie de travail

La copie de travail est un répertoire situé en local sur le poste de l'utilisateur et qui contient une copie d'une révision donnée des fichiers du dépôt. C'est cette copie qui sert de base de travail et qui est modifiée en local avant d'être importée (sauvegardée) vers le dépôt.

Révisions

Chaque modification faite au dépôt constitue une révision. Le numéro de révision commence à 1 et augmente de 1 à chaque opération. Sa valeur n'a aucune importance, mais c'est un indicateur qui permet de revenir à une version donnée d'un ou plusieurs fichiers.

import

Cette opération consiste à importer dans le dépôt (sur le serveur) des fichiers. Cette opération ne se fait en général qu'une fois par projet.

checkout

Le checkout est l'opération qui consiste à récupérer pour la première fois les fichiers déjà existant au sein d'un projet du dépôt. Cette opération ne se fait en général qu'une fois par projet. Le résultat est une copie de travail.

update

L'update consiste à synchroniser la copie de travail locale avec le dépôt en récupérant la dernière version des fichiers du dépôt. C'est à cette occasion que des conflits de version peuvent apparaître.

commit

Un commit est l'opération inverse d'un update. Elle consiste à mettre à jour le dépôt à partir de la copie de travail locale. Une nouvelle révision est alors créée. Un log (simple message texte contenant une description des modifications effectuées) doit être saisi à cette occasion. À noter que pour qu'un commit soit possible, il faut que la copie de travail corresponde à la dernière version du dépôt (modifications locales exceptées). Si ce n'est pas le cas, il est nécessaire d'effectuer d'abord un update et de résoudre les conflits éventuels avant de réessayer le commit.

Initialisation coté serveur

- Création du dépôt (Serveur)

svn create

```
[root@portable_denis brouillon]# svnadmin create depot  
[root@portable_denis brouillon]# ls depot  
README.txt  conf  db  format  hooks  locks  
[root@portable_denis brouillon]# █
```

- Import (facultatif)

svn import

```
[root@portable_denis brouillon]# svn import -m "init projet" projet/ file:///run  
/media/denis/disk-externe/brouillon/depot/  
Ajout          projet/main.c  
  
Révision 1 propagée.  
[root@portable_denis brouillon]# █
```

Récupération d'une copie de travail

svn checkout

```
[denis@portable_denis disk-externe]$ svn co https://dwarves.iut-fbleau.fr/svn/Test --username monnerat
A   Test/trunk
A   Test/branches
A   Test/tags
A   Test/trunk/essai.txt
A   Test/trunk/toto.html
Révision 3 extraite.
[denis@portable_denis disk-externe]$
```

Synchronisation dépôt \rightsquigarrow copie locale

svn update

```
[denis@portable_denis trunk]$ svn update
Mise à jour de '.' :
A   index.html
Actualisé à la révision 4.
[denis@portable_denis trunk]$
```

Synchronisation copie local \rightsquigarrow dépôt

svn commit

```
[denis@portable_denis trunk]$ svn commit -m "modif index.html"  
Envoi      index.html  
Transmission des données .  
Révision 5 propagée.  
[denis@portable_denis trunk]$ □
```

Récupération Versions antérieures

- La dernière version

svn revert

```
[denis@portable_denis trunk]$ svn revert index.html  
[denis@portable_denis trunk]$
```

- Une révision quelconque

svn update -r numrev

```
[denis@portable_denis trunk]$ svn update -r 2 index.html  
Mise à jour de 'index.html' :  
D    index.html  
Actualisé à la révision 2.  
[denis@portable_denis trunk]$
```

Informations sur une copie

svn info

```
[denis@portable_denis trunk]$ svn update
 Mise à jour de '.' :
A   index.html
Actualisé à la révision 6.
[denis@portable_denis trunk]$ svn info
Chemin : .
Chemin racine de la copie de travail : /run/media/denis/disk-externe/Test
URL : https://dwarves.iut-fbleau.fr/svn/Test/trunk
Relative URL: ^/trunk
Racine du dépôt : https://dwarves.iut-fbleau.fr/svn/Test
UUID du dépôt : f1b0018a-5d1e-4324-b167-ce8765e36b31
Révision : 6
Type de nœud : répertoire
Tâche programmée : normale
Auteur de la dernière modification : monnerat
Révision de la dernière modification : 6
Date de la dernière modification: 2014-09-26 14:29:19 +0200 (ven. 26 sept. 2014)

[denis@portable_denis trunk]$ █
```

Gestion des fichiers

- Ajout

svn add

```
[denis@portable_denis trunk]$ svn add new.php
A      new.php
[denis@portable_denis trunk]$ svn commit -m "ajout fichier new.php"
Ajout      new.php
Transmission des données .
Révision 7 propagée.
[denis@portable_denis trunk]$ █
```

- Suppression

svn delete

```
[denis@portable_denis trunk]$ svn delete new.php
D      new.php
[denis@portable_denis trunk]$ svn commit -m "suppression new.php"
Suppression      new.php

Révision 8 propagée.
[denis@portable_denis trunk]$ █
```

Gestion des fichiers

Renommage

svn mv

```
[denis@portable_denis trunk]$ svn mv essai.txt foo.txt
A      foo.txt
D      essai.txt
[denis@portable_denis trunk]$ svn commit -m ""
Suppression    essai.txt
Ajout          foo.txt

Révision 9 propagée.
[denis@portable_denis trunk]$ █
```

Conflits

La copie a déjà été mise à jour.

```
[denis@portable_denis trunk]$ svn commit -m "commit foo.txt"
Envoi          foo.txt
Transmission des données .svn: E155011: Échec de la propagation (commit), détails :
svn: E155011: Fichier '/run/media/denis/disk-externe/Test/trunk/foo.txt' obsolète
svn: E170004: File '/trunk/foo.txt' is out of date
[denis@portable_denis trunk]$
```

Il faut faire un update et mettre en concurrence les deux versions. Si Subversion ne peut pas résoudre tout seul le conflit :

```
[denis@portable_denis trunk]$ svn commit -m "commit foo.txt"
Envoi          foo.txt
Transmission des données .svn: E155011: Échec de la propagation (commit), détails :
svn: E155011: Fichier '/run/media/denis/disk-externe/Test/trunk/foo.txt' obsolète
svn: E170004: File '/trunk/foo.txt' is out of date
[denis@portable_denis trunk]$ svn update
Mise à jour de '.' :
C   foo.txt
Actualisé à la révision 10.
```

Résolution

```
[denis@portable_denis trunk]$ ls  
foo.txt  foo.txt.mine  foo.txt.r15  foo.txt.r16  index.html  toto.html  
[denis@portable_denis trunk]$ █
```

`foo.txt.mine` copie du fichier tel qu'il se trouvait dans votre copie de travail, en local, avant de faire l'update. C'est la version que vous souhaitiez "commiter" avant de détecter un conflit

`foo.txt.r15` version du fichier pour la révision 15, c'est à dire lors de votre dernier update. C'est la version qui a servi de base pour les deux utilisateurs du dépôt qui ont travaillé en parallèle

`foo.txt.r16` version du fichier pour la révision 16, c'est à dire la version actuellement dans le dépôt. Il s'agit de la version modifiée par un autre utilisateur, "commitée" avant votre update et dont le contenu est à l'origine du conflit

`foo.txt` version qui, en quelque sorte "résume" les trois autres en faisant apparaître les différences entre versions au sein d'un seul fichier

On résout à la main en éditant le fichier.

```
<<<<<<< .mine  
  
Denis  
=====  
Pierre  
>>>>>>> .r16  
foo.txt (END)
```

et on marque le conflit résolu avant de "commiter".

```
[denis@portable_denis trunk]$ less foo.txt  
[denis@portable_denis trunk]$ vim foo.txt  
[denis@portable_denis trunk]$ svn resolved foo.txt  
Conflit sur 'foo.txt' résolu  
[denis@portable_denis trunk]$ svn commit -m "resolution"  
Envoi          foo.txt  
Transmission des données .  
Révision 17 propagée.  
[denis@portable_denis trunk]$
```

Informations

svn help	aide de svn
----------	-------------

svn info	informations sur la copie de travail
----------	--------------------------------------

svn log	historique
---------	------------

svn blame	historique ligne par ligne (révisions et auteurs)
-----------	---

svn status	état de la copie depuis le dernier update
------------	---

svn list	liste d'un dépôt distant
----------	--------------------------

- Traditionnellement, le repertoire principal (la branche principale, le tronc) est placé dans le repertoire *trunk*.
- On peut créer des branches de développements différentes, placées usuellement dans le repertoire *branches*.

Pour créer une branche, on copie le tronc (souvent) dans une nouvelle branche :

svn copy

```
[denis@portable_denis Test]$ svn copy trunk branches/mabranche
A      branches/mabranche
[denis@portable_denis Test]$ svn commit -m "fork mabranche"
Ajout      branches/mabranche

Révision 22 propagée.
[denis@portable_denis Test]$ ls branches/mabranche/
foo.txt  index.html  toto.html
[denis@portable_denis Test]$ █
```

Pour changer de branche :

svn switch

```
[denis@portable_denis mabranche]$ svn switch https://dwarves.iut-fbleau.fr/svn/T  
est/branches/mabranche  
À la révision 22.  
[denis@portable_denis mabranche]$ █
```

Sur le même principe, on peut créer des Tags, c'est à dire des instantanés de l'état d'un dépôt.

On peut également fusionner des branches avec l'option `svn merge`.

Hooks Scripts

Sur le serveur, des *scripts* personnalisées peuvent être exécutés juste avant ou après un commit (il y a d'autres événements). très pratique, par exemple pour :

- Mettre la version en production automatiquement.
- Envoyer un mail.
- indenter les sources automatiquement avant le commit.
- ...

Clients svn

- La console et la commande svn.
- TortoiseSVN
- ...

Il existe de nombreux outils graphiques, ou de plugins pour éditeurs de texte ou navigateurs de fichiers.

Un exemple sous linux avec rapidsvn :
<http://rapidsvn.tigris.org/>

The screenshot shows a file manager window with a sidebar containing a tree view of an SVN repository. The main pane displays a table of repository items.

Nom	Révision	Dernière ré	Auteur	Statut	Stat
.	22	22	monnerat		
branches	22	22	monnerat		
tags	22	1	USVN		
trunk	22	17	monnerat		

Below the file manager, a terminal window displays the following output:

```

Executer : recuperer
Erreur : Erreur pendant l'exécution de l'action : Unexpected HTTP status 405 'Method Not Allowed' on '/svnTest'

Additional errors:
Échec de la requête PROPFIND sur '/svnTest' : 405 Method Not Allowed
Prêt

Exécuter : Récupérer
Ajouté: /home/denis/SVN/SVN/tags
Ajouté: /home/denis/SVN/SVN/trunk
Ajouté: /home/denis/SVN/SVN/branches
Ajouté: /home/denis/SVN/SVN/branches/mabranche
Ajouté: /home/denis/SVN/SVN/trunk/foo.txt
Ajouté: /home/denis/SVN/SVN/trunk/toto.html
Ajouté: /home/denis/SVN/SVN/trunk/index.html
Ajouté: /home/denis/SVN/SVN/branches/mabranche/foo.txt
Ajouté: /home/denis/SVN/SVN/branches/mabranche/toto.html
Ajouté: /home/denis/SVN/SVN/branches/mabranche/index.html
Mis à jour: /home/denis/SVN/SVN
Mise à jour...
Prêt
    
```