

SVN

F. Langrognat



PLAN

1 Principales commandes SVN

2 Cycle de travail

3 Autres commandes SVN

- add, rm, mv
- Récupérer une version plus ancienne
- log, status
- Gestion des conflits

PLAN

1 Principales commandes SVN

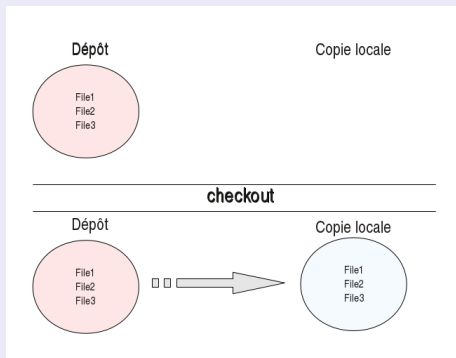
2 Cycle de travail

3 Autres commandes SVN

- add, rm, mv
- Récupérer une version plus ancienne
- log, status
- Gestion des conflits

checkout

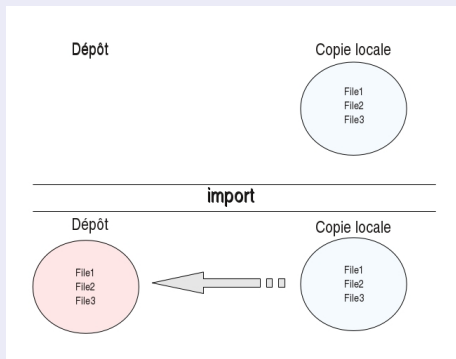
Opération qui consiste à récupérer (pour la première fois) les fichiers du dépôt dans son répertoire de travail (copie de travail).



import

Opération inverse du checkout.

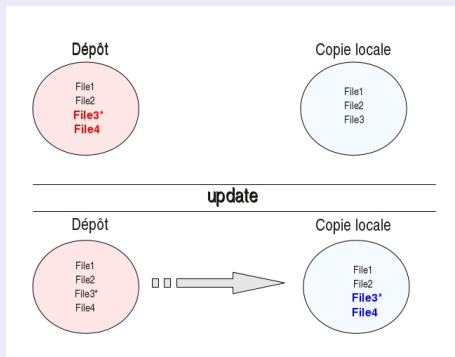
Elle consiste à placer dans le dépôt des fichiers locaux déjà existants pour y créer un nouveau projet.



update

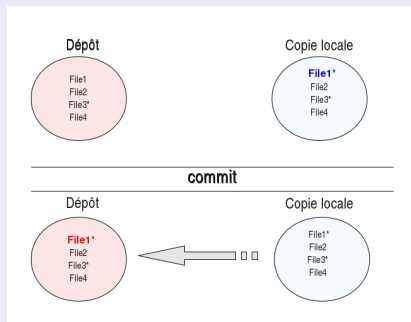
Synchronisation de la copie de travail locale avec le dépôt en récupérant la dernière version des fichiers du dépôt.

Note : c'est à cette occasion que des conflits de version peuvent apparaître.



Opération inverse d'un update.

Mise à jour du dépôt à partir de la copie de travail locale. Une nouvelle révision est alors créée.

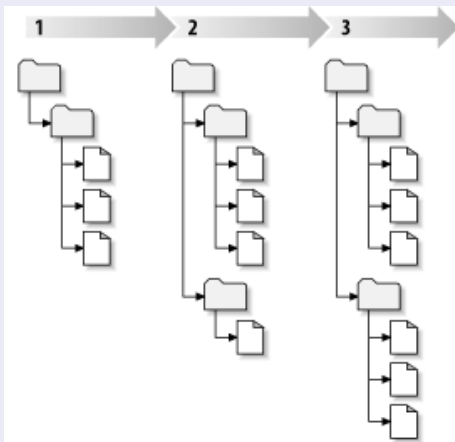


Pour qu'un commit soit possible, il faut que la copie de travail corresponde à la dernière version du dépôt (modifications locales exceptées). Si ce n'est pas le cas, il est nécessaire d'effectuer d'abord un update et de résoudre les conflits éventuels avant de réessayer le commit.

Notion de révision

Révision

Chaque modification faite au dépôt constitue une **révision**.
Le numéro de révision commence à 1 et augmente de 1 à chaque opération.
Les révisions successives constituent l'**historique** du projet



PLAN

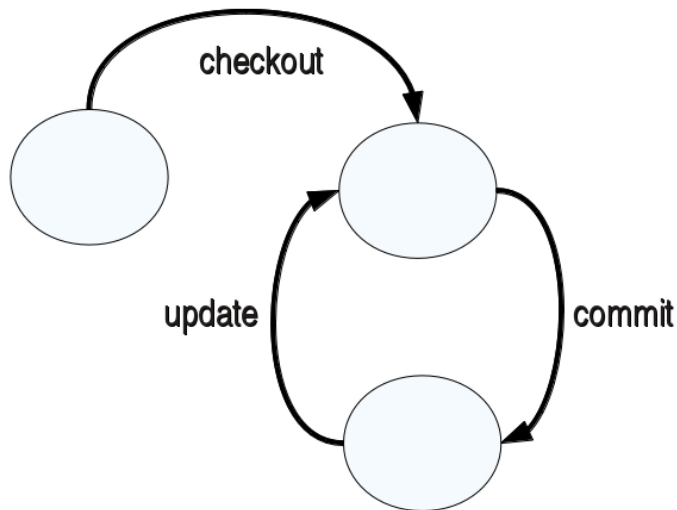
1 Principales commandes SVN

2 Cycle de travail

3 Autres commandes SVN

- add, rm, mv
- Récupérer une version plus ancienne
- log, status
- Gestion des conflits

Cycle de travail



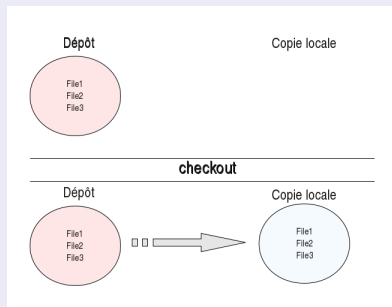
Créer sa copie de travail à partir d'un dépôt

checkout

```
$ svn co https://...<cheminVersLeDepot>  
  (ou svn+ssh:...)  
A  File1  
A  File2  
A  File3  
Checked out revision 36.
```

La copie de travail est créée avec

- répertoire `.svn`
- les 3 fichiers File1, File2, File3



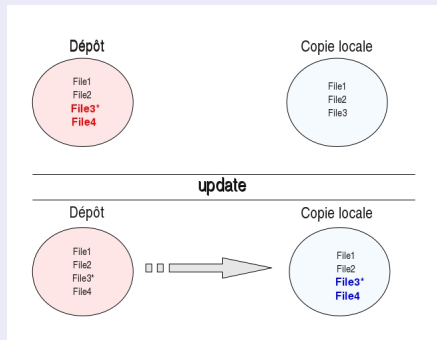
Mettre à jour sa copie de travail

update

```
$ svn update
U  File3
A  File4
Updated to revision 37.
```

Mise à jour de la copie de travail

- Nouvelle version du fichier File3
- Nouveau fichier File4



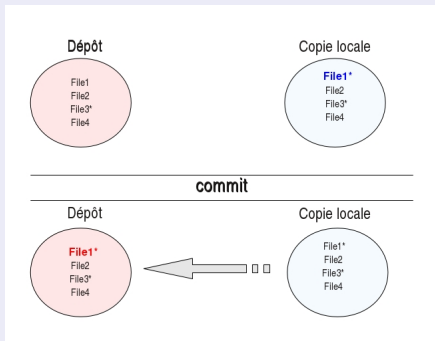
Propager - Mise à jour de ses modifications vers le dépôt

commit

```
$ svn commit -m "Modification du fichier File1"  
Sending          File1  
Transmitting file data .  
Committed revision 38.
```

Mise à jour du dépôt

- Nouvelle version du fichier File1



PLAN

1 Principales commandes SVN

2 Cycle de travail

3 Autres commandes SVN

- add, rm, mv
- Récupérer une version plus ancienne
- log, status
- Gestion des conflits

PLAN

1 Principales commandes SVN

2 Cycle de travail

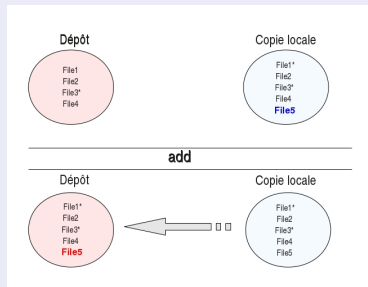
3 **Autres commandes SVN**

- **add, rm, mv**
- Récupérer une version plus ancienne
- log, status
- Gestion des conflits

Ajout d'un fichier

svn add

```
$ svn add File5  
A File5  
  
$ svn commit -m "Ajout du fichier File5"  
Adding File5  
Transmitting file data .  
Committed revision 39.
```



Suppression d'un fichier

svn rm

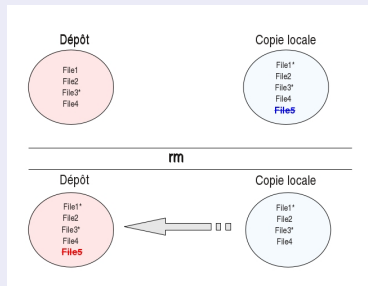
```
$ svn rm File5
```

```
D File5
```

```
$ svn commit -m "Suppression du fichier File5"
```

```
Deleting File5
```

```
Committed revision 40.
```



Déplacement-Renommage d'un fichier

svn mv

```
$ svn mv File4 Fichier4
```

```
A      Fichier4
```

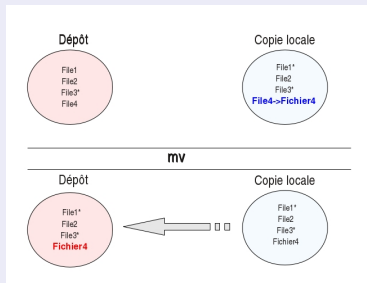
```
D      File4
```

```
$ svn commit -m "Renommage d'un fichier"
```

```
Deleting      File4
```

```
Adding        Fichier4
```

```
Committed revision 41.
```



PLAN

1 Principales commandes SVN

2 Cycle de travail

3 **Autres commandes SVN**

- add, rm, mv
- **Récupérer une version plus ancienne**
- log, status
- Gestion des conflits

Récupérer une version plus ancienne

svn update ... avec options

- Récupérer une version d'une révision donnée

```
$ svn update -r35
U  File1
....
Updated to revision 35.
```

- Récupérer une version d'une date donnée

```
$ svn update -r {"2013-12-30"}
U  File2
....
Updated to revision 31.
```

PLAN

1 Principales commandes SVN

2 Cycle de travail

3 **Autres commandes SVN**

- add, rm, mv
- Récupérer une version plus ancienne
- **log, status**
- Gestion des conflits

log (1)

Journal complet du projet

```
$ svn log
```

```
-----  
r43 | Florent | 2014-03-10 22:56:19 ... | 1 ligne  
Modification File2.
```

```
-----  
r42 | Julien | 2014-03-10 21:21:19 ... | 2 lignes  
Ajout commentaire sur File1
```

```
....
```

log (2)

Journal sur un fichier

```
$ svn log File2
```

```
-----  
r43 | Florent | 2014-03-10 22:56:19 ... | 1 ligne  
Modification File2.
```

```
-----  
r40 | Florent | 2014-03-10 20:21:19 ... | 2 lignes  
Ajout commentaire File2
```

```
....
```

Connaître le statut de la copie local

```
$ svn status  
A File6  
D File2  
M File3  
? File7
```


PLAN

1 Principales commandes SVN

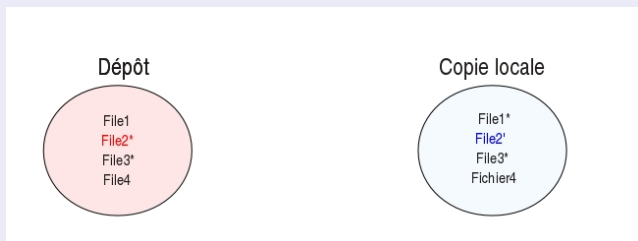
2 Cycle de travail

3 **Autres commandes SVN**

- add, rm, mv
- Récupérer une version plus ancienne
- log, status
- **Gestion des conflits**

Gestion des conflits (1)

Situation de conflit



Le même fichier (File2) a été modifié

- Sur le dépôt (File2*)
- En local (File2')

Gestion des conflits (2)

Tentative de commit

```
$ svn commit -m "modification File2"
Sending          File2
svn: Commit failed (details follow):
svn: Your file or directory 'File2' is probably
    out-of-date
svn:
The version resource does not correspond to the resource
within the transaction.  Either the requested version
resource is out of date (needs to be updated), or the
requested version resource is newer than the
transaction root (restart the commit).
```

Il faut donc déjà **mettre à jour la copie de travail** : update

Gestion des conflits (2)

Cas simple

Les modifications ne concernent pas les mêmes parties du fichier :

fusion automatique

```
$ svn update
G File2
Updated to revision 42.
```

La fusion de File* et File2' a été réalisée automatiquement.

On peut maintenant faire un commit

```
$ svn commit -m "Modification du fichier File2"
Sending          File2
Transmitting file data .
Committed revision 43.
```

Gestion des conflits (3)

Cas compliqué

Les modifications concernent les mêmes parties du fichier ; la fusion automatique n'est pas possible

fusion manuelle

```
$ svn update
C  File2
Updated to revision 43.
```

Il faut résoudre **manuellement** le conflit.

Gestion des conflits (4)

Cas compliqué...suite

Pour résoudre le conflit, on dispose dans le répertoire de travail de :

- **File2.mine** : version du fichier tel qu'il se trouvait dans la copie de travail, en local, avant de faire l'update. C'est la version que l'on souhaitait « commiter » avant de détecter un conflit ;
- **File2.r42** : version du fichier pour la révision 42, c'est à dire lors du dernier update. C'est la version qui a servi de base pour les deux utilisateurs du dépôt qui ont travaillé en parallèle ;
- **File2.r43** : version du fichier pour la revision 43, c'est à dire la version actuellement dans le dépôt. Il s'agit de la version modifiée par un autre utilisateur, « commitée » avant l'update, et dont le contenu est à l'origine du conflit.
- **File2** : il s'agit d'une version qui, en quelque sorte « résume » les trois autres en faisant apparaître les différences entre versions au sein d'un seul fichier.

Gestion des conflits (5)

Cas compliqué...suite

Pour résoudre le conflit, il faut éditer le fichier File2 et choisir la version à conserver :

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

On indique que le conflit est résolu et on peut “committer”

```
$ svn resolved File2
Resolved conflicted state of File2

$ svn commit -m "Modification de File2"
Sending          File2
Transmitting file data .
Committed revision 44.
```

SVN



F. Langrognat