

Projet Informatique

Gestion de version et de configuration : de svn à git

Philippe Collet

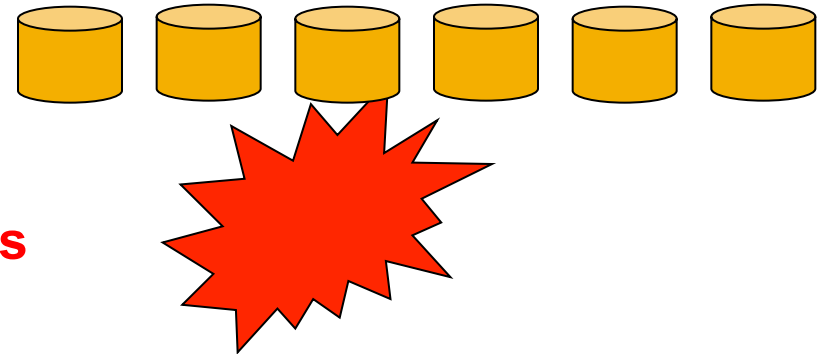
Licence 3 Informatique – S5

2014-2015

<http://deptinfo.unice.fr/twiki/bin/view/Linfo/ProjetInfo201415>

Motivations

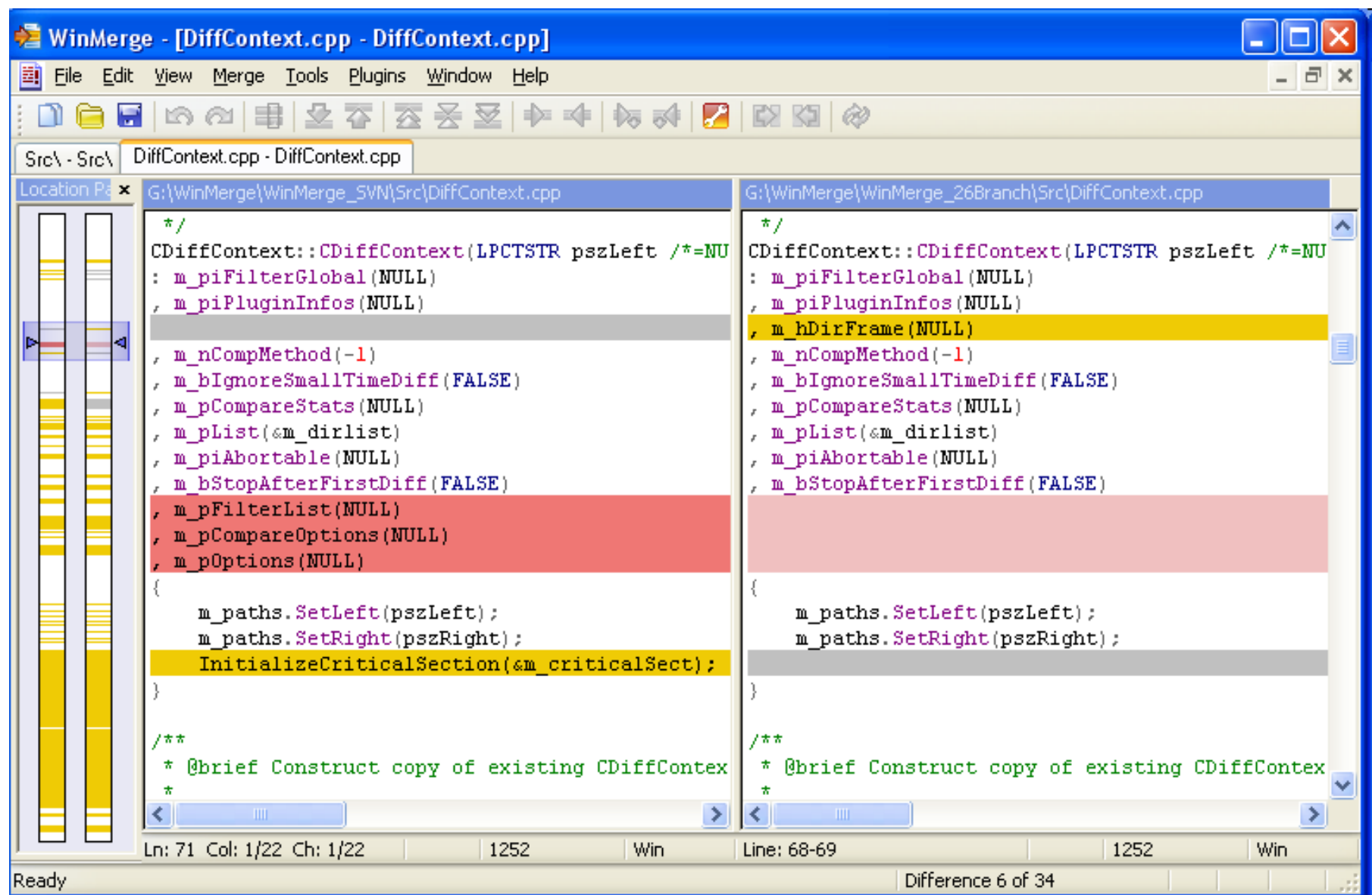
- ❑ Quand on modifie des sources :
 - Des bugs apparaissent parfois (souvent !)
- ❑ On pourrait sauver chaque version de chaque fichier modifié...
 - Ou ne stocker que les différences !
- ❑ Et quand on est plusieurs à modifier
 - ☞ **Savoir qui modifie quoi**
 - ☞ **Ne rien écraser**
 - ☞ **Fusionner si on modifie à plusieurs**
 - ☞ **Revenir en arrière (un bug...)**
 - ☞ **Gérer des développements en //**



Principe de la différenciation

- ❑ **Outil diff**
- ❑ **Différences entre 2 fichiers d'après**
 - **Ligne de début/de fin**
 - **Insertion/Suppression d'une ou plusieurs lignes**
- ❑ **Facilité de détection et de construction d'un patch**
- ❑ **Pas de détection des lignes modifiées**
 - **Traitées comme suppression + insertion**

Différenciation : illustration



Historique

- ❑ **SCCS** (livré avec Unix dès Vx, Bell labs programmer workbench, fusionné en 1983)
- ❑ **RCS** (W. Tichy 1985)
- ❑ **CVS** (B. Berliner 1989)
 - Support dans beaucoup d'environnements...
- ❑ **Subversion** (subversion.tigris.org)
 - Bonne gestion des modifications de l'arborescence des répertoires
 - Installation et maintenance simplifiée
- ❑ **Visual Source Safe** : The Microsoft Way
- ❑ **ClearCase** (Rational) : L'usine de gestion de traçabilité
- ❑ **Git, Mercurial...**

Ce que SVN n'est pas...

- ☐ Un système de construction (makefile, ant...)
- ☐ Un système de gestion de projet (Ms-project)
- ☐ Un substitut à la communication entre développeurs (ex: conflit *sémantique*)
- ☐ Un système de contrôle du changement (bug-tracking, ChangeLog)
- ☐ Un système de tests automatisés
- ☐ Un système fondé sur un processus particulier

Commande(s) SVN

- ❑ `svn subcommand [switches] [cmd_args]`
 - Commande de base coté client
 - **subcommand : obligatoire**
 - **switches : options spécifiques à la sous-commande**
 - **cmd_args : arguments de la sous-commande**

```
svn checkout http://svn.c.net/rep/svn/trunk subv
```

- ❑ `svnadmin subcommand [switches] [cmd_args]`
 - Administration de la base

Base (ou dépôt) svn

❑ Locale (accéder directement par le client) :

- `file://`

❑ Accédée à travers Apache 2 (WebDAV)

- `http://`
- `https://` (SSL encryption)

❑ Accédée par le protocole spécifique « svn » (possibilité de passer par ssh)

- `svn://` (nécessité d'avoir un serveur svnserve)
- `svn+ssh://` identique à `svn://`, mais *tunneling ssh* (et pas de serveur)

Administrer une base SVN

❑ Créer une base SVN

- `svnadmin create /chemin/vers/referentiel`
- Par défaut format de stockage FSFS (autre format Berkeley-DB moins performant, conservé pour compatibilité)



administrateur

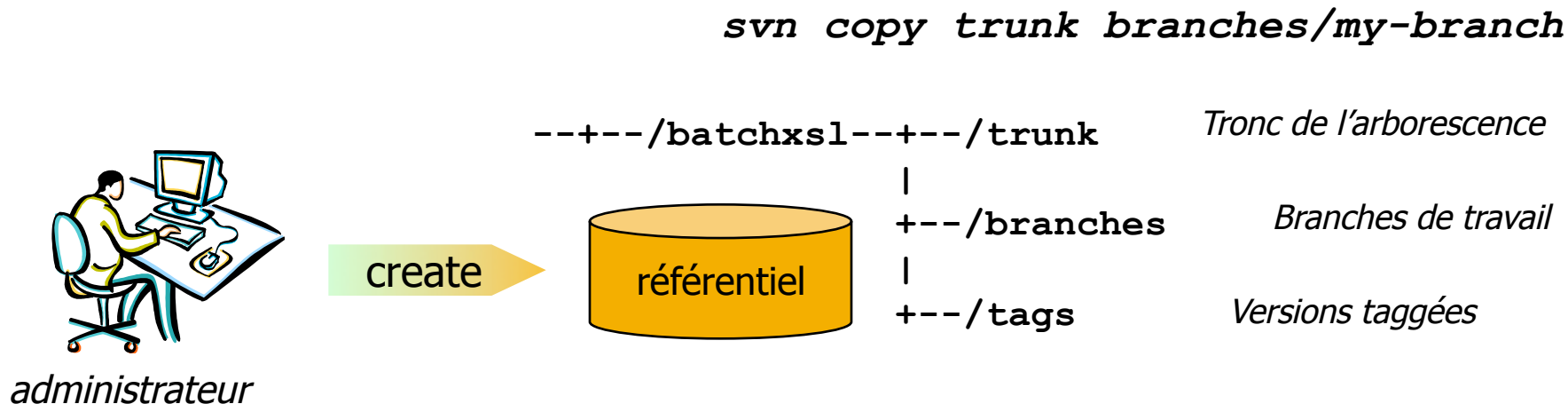
create



- conf : répertoire des fichiers de config
- dav : répertoire spécifique à mod_dav_svn
- db : les données (pas directement « lisibles »)
- format : un fichier avec un seul entier donnant le numéro de version des hooks de traitement
- hooks : répertoire des scripts de hook
- locks : répertoire des verrous de subversion
- README.txt : des infos sur les autres répertoires

Administrer une base SVN

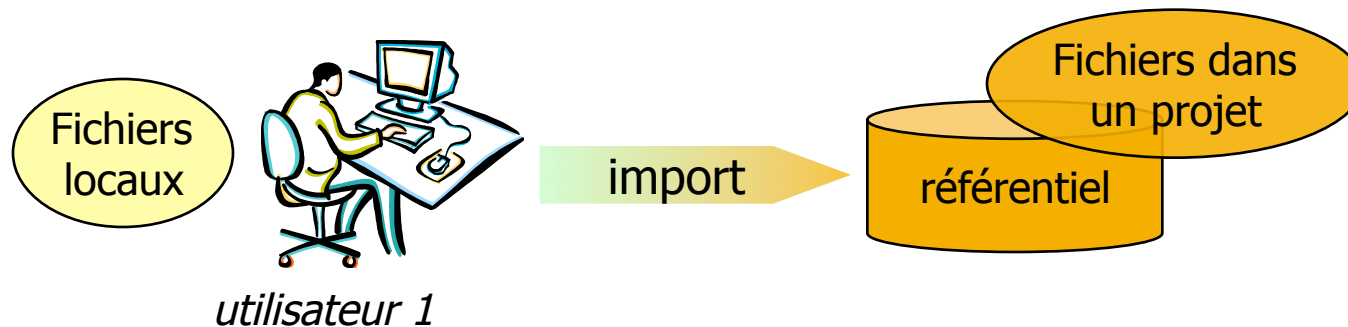
- ❑ Au sein d'une base se trouvent un ou plusieurs projets.
- ❑ À chaque projet correspond en général un répertoire situé à la racine du dépôt et qui contient lui-même les fichiers et dossiers du projet.
- Organiser les répertoires :



Importer des sources

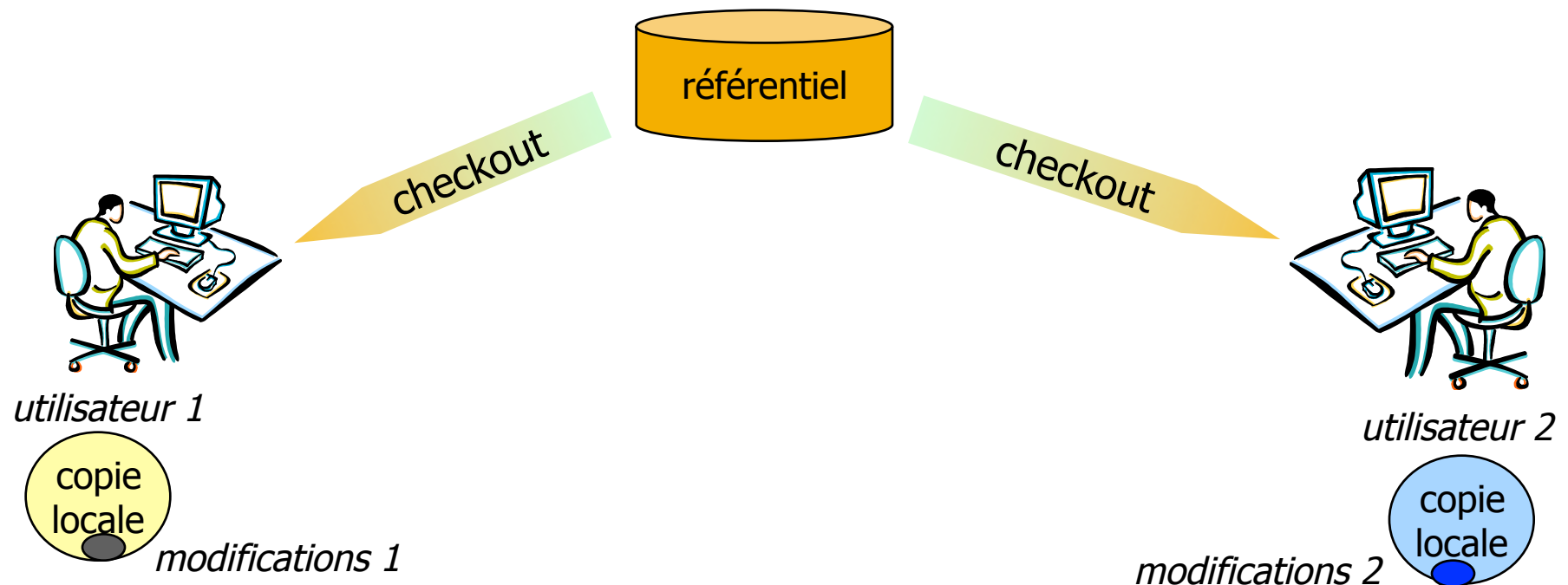
❑ Importer des sources

- `svn import rep_local /chemin/vers/referentiel [options]`
- `svn import myTree file:///usr/local/svn/newrepos/batchxml/trunk -m "Initial import"`



Récupérer une copie locale des sources

- ❑ `svn checkout chemin/vers/referentiel/et/projet [options]`
- ❑ `svn checkout http://svn.collab.net/repos/svn/trunk`



Propager ses changements

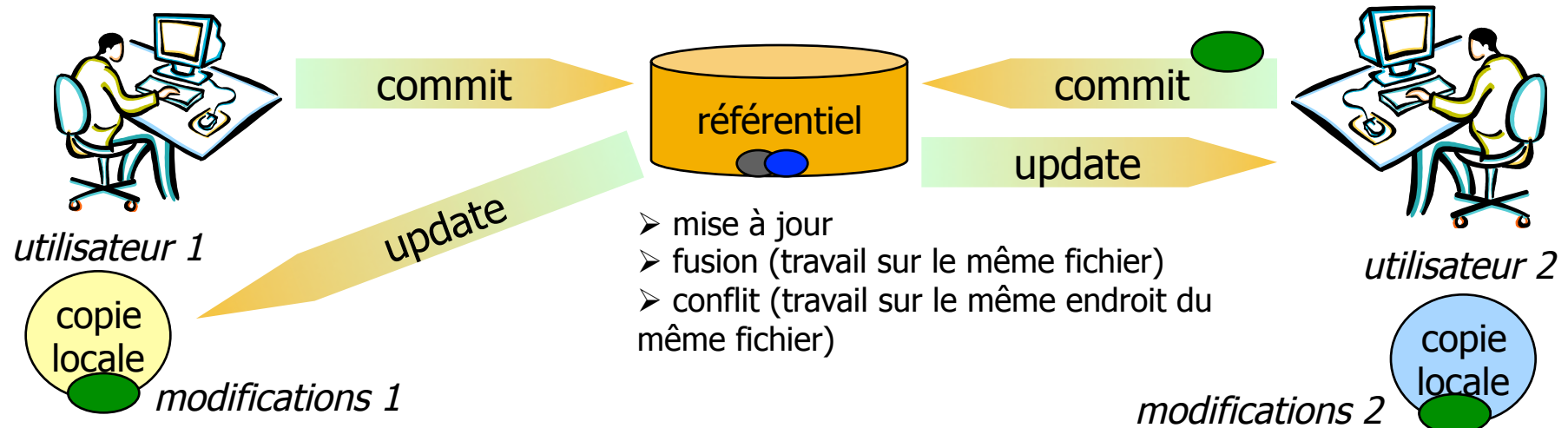
Mettre à jour par rapport à la base

❑ Propagation de vos changements

■ `svn commit`

❑ Récupération de nouvelles mises à jour

■ `svn update`



Signification des sorties SVN pour *update* et *checkout*

- ❑ U file : votre répertoire a été mis à jour
- ❑ A file : fichier ajouté à votre copie privée, sera propagé après commit
- ❑ D file : fichier effacé... définitivement après commit
- ❑ C file : conflit détecté lors de fusion
- ❑ G file : fusion effectuée (car pas de conflit)

Quelques commandes et options

- ❑ **Ajouter un fichier/répertoire :** `svn add`
 - + commit
- ❑ **Retirer un fichier/répertoire :** `svn delete`
 - + commit
- ❑ **Copie des fichiers/répertoires :** `svn copy`
 - + commit
- ❑ **Déplacer des fichiers/répertoires :** `svn move`
 - + commit

Quelques commandes et options (suite)

- ☐ **Liste des répertoires dans le référentiel :**
 - `svn list`
- ☐ **Affichage des messages de commit :**
 - `svn log`
- ☐ **Mes modifications locales (pas de connection au référentiel) :**
 - `svn status`
- ☐ **Visualiser les différences :**
 - `svn diff`
- ☐ **Revenir en arrière (undo) :**
 - `svn revert`
- ☐ **Indiquer qu'un conflit est résolu sur un fichier :**
 - `svn resolved sandwich.txt`

Illustration : TortoiseSVN

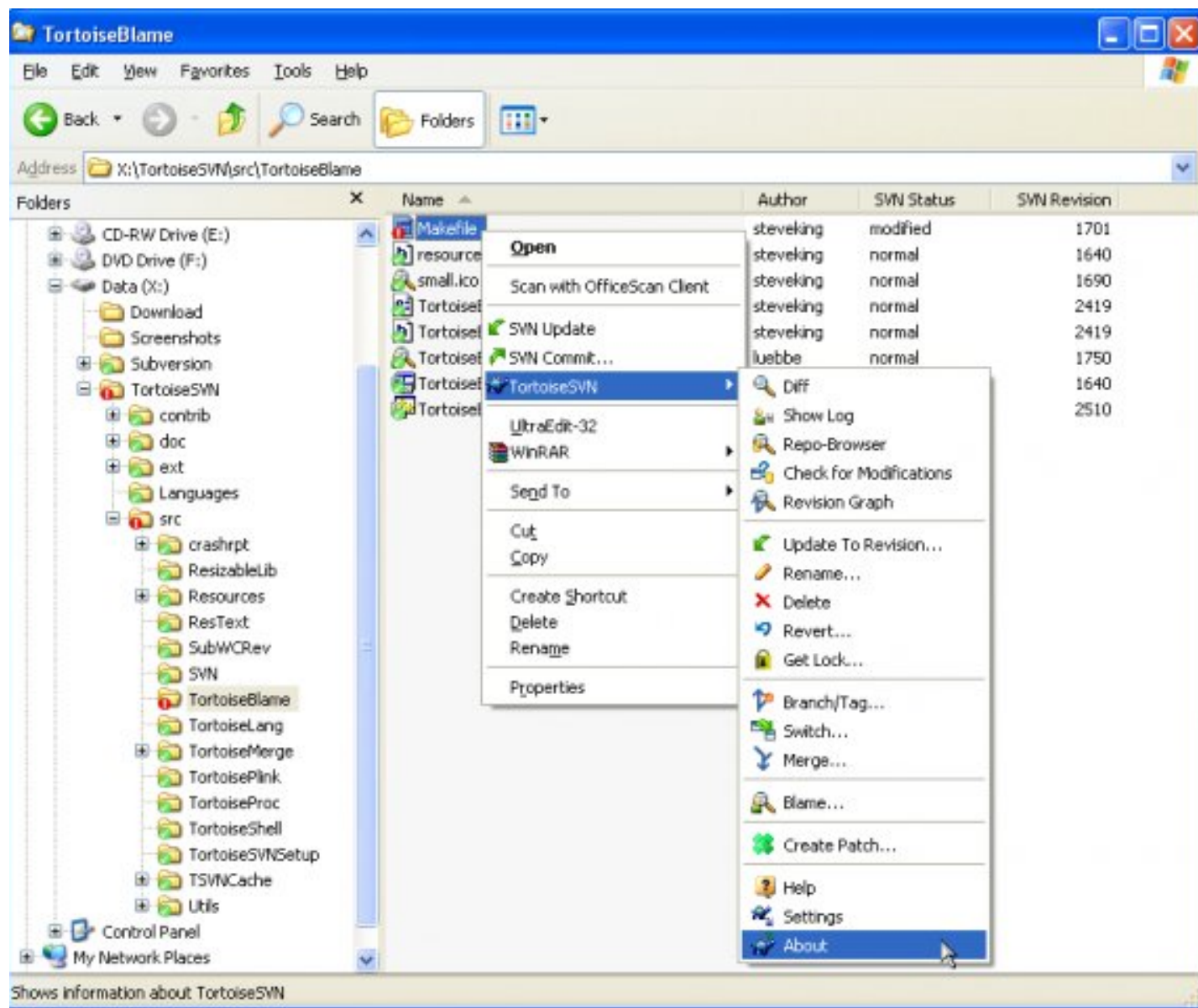


Illustration : plugin subclipse dans Eclipse

Team Synchronizing - SVNClientManager.java - Eclipse Platform

File Edit Navigate Search Project Run Window Help

Synchronize SVN (/core)

core

- src/org/tigris/subversion/s
- INewInterface.java
- messages.properties
- SVNClientManager.java
- src/org/tigris/subversion/s
- ResourceStatus.java
- src/org/tigris/subversion/s
- SVNStatusSyncInfo.java
- .classpath (1500)
- build.properties (1500)
- javasvn.jar (1500)
- jsch.jar (1500)
- plugin.xml (1500)
- svnClientAdapter.jar (1500)

Java Structure Compare

- Compilation Unit
- Import Declarations
- SVNClientManager
- startup(IProgressMonitor)

Java Source Compare

Local File

```
}  
try {  
    CmdLineClientAdap  
} catch (SVNClientExc  
    // removed some c  
    // the quick-diff  
}
```

Remote File (1500)

```
javaSvnExcept  
}  
try {  
    CmdLineClient.  
    adapterAvaila  
} catch (SVNClie  
    // if an exce  
    // already re  
    cliException :
```

Tasks Problems SVN Resource History

SVN Resource History (SVNClientManager.java)

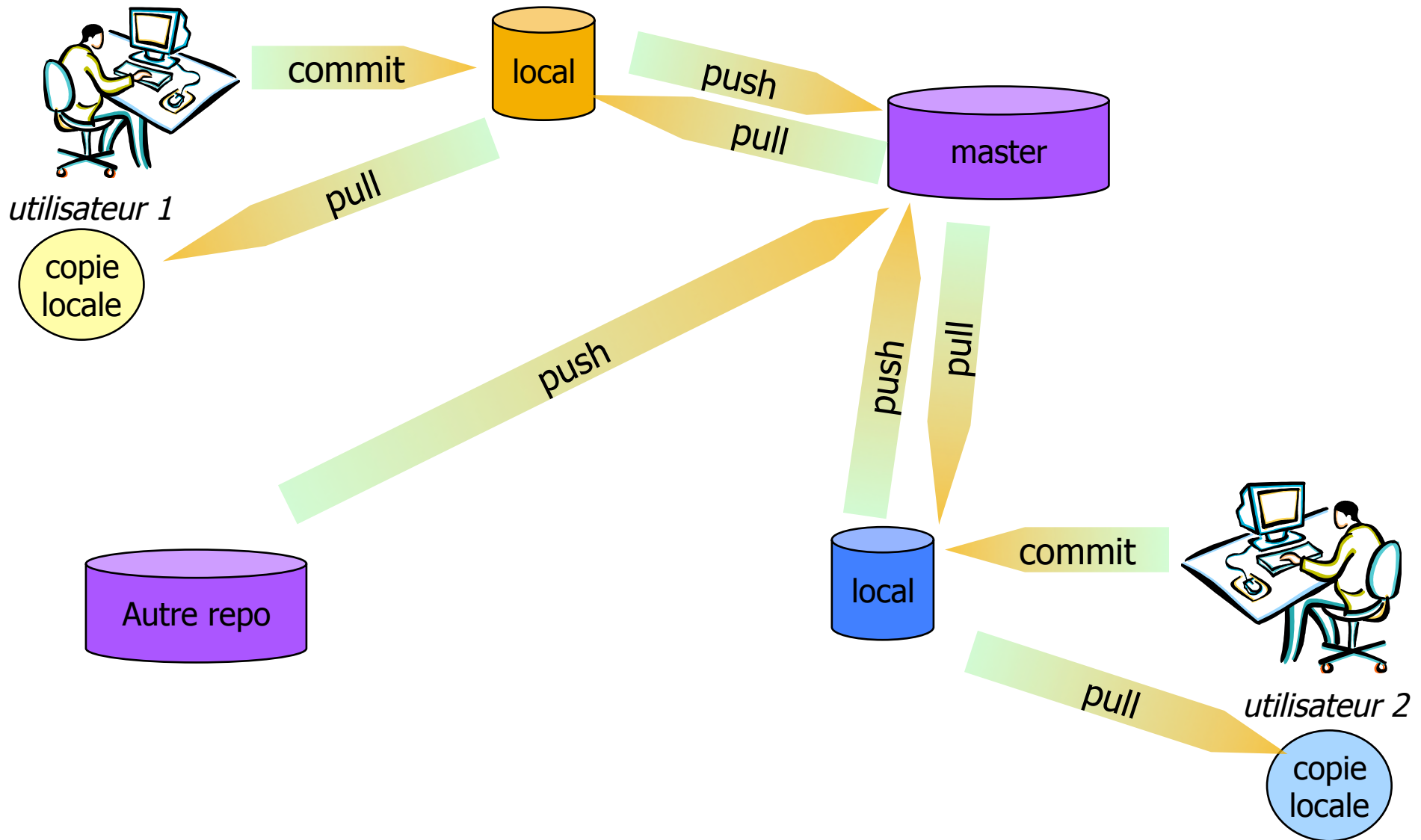
| Revision | Date | Author | Comment |
|----------|---------------|---------|---|
| 1493 | 8/3/05 11:... | markhip | Added support for JavaSVN as an adapter choi... |
| *1420 | 6/28/05 8... | markhip | Setting svn:eol-style on files that are mis... |

- Synchronize with Repository
- Commit...
- Update
- Create Patch...
- Apply Patch...
- Show in Resource History
- Show Properties
- Set Property...
- Add Keywords...
- Add to Version Control
- Add to svn:ignore
- Branch/Tag...
- Switch...
- Merge...
- Configure Branches/Tags
- Copy...
- Export...
- Edit conflicts
- Revert...
- Mark as Deleted
- Mark Resolved
- Show pending operations
- Cleanup
- Disconnect...
- Share Project...

Centralisation vs. Distribution

- ❑ **SVN: pas d'accès à une base partagée sans connexion réseau**
- ❑ **GIT (et Mercurial, Bazaar):**
 - **« Distributed » Revision control**
 - ◆ Plus de serveur central (si le serveur tombe, le service est indisponible)
 - ◆ Chacun à une base locale, il existe une ou plusieurs bases distribuées sur des serveurs (une base est la base maître)
 - **On peut donc :**
 - ◆ commiter en local, sans que les autres développeurs soient tenus informés
 - ◆ puis « pousser » quand le moment est opportun vers la base maître
 - ◆ Updater depuis la base maître
 - **Avantages / inconvénients :**
 - ◆ Si la base maître tombe, on clone sa base locale sur une autre base
 - ◆ Complexe à utiliser : parfois on oublie de « pousser » vers la base...

Git : illustration



- ❑ <http://git-scm.com/>
- ❑ **Gestion de version distribuée open-source**
 - gestion des branches et des merges
- ❑ **Chaque copie de repository distribué est autonome**
 - gère son propre historique et ses versions
 - peut-être utilisé hors connexion indépendamment du repository original
 - Les sources sont publiables d'un repository vers un autre (push)
 - Le partage et la réintégration des sources (pull) est très simple

Principales commandes Git

- ❑ ***git init*** : crée un nouveau dépôt
- ❑ ***git clone*** : clone un dépôt distant
- ❑ ***git add*** : ajoute les nouveaux objets depuis le dernier commit. Les objets précédents restent inchangés
- ❑ ***git commit*** : marquer les changements en local
- ❑ ***git branch*** : crée une nouvelle branche de développement
- ❑ ***git merge*** : fusionne plusieurs branches de développement
- ❑ ***git push*** : publie le commit vers un repository distant
- ❑ ***git pull*** : récupère depuis un repository distant

Client Git : SourceTree (Mac, Windows)

The screenshot displays the SourceTree application window. The top toolbar includes icons for View, Commit, Checkout, Reset, Stash, Add, Remove, Add/Remove, Fetch, Pull, Push, Branch, Merge, Tag, Create Patch, Apply Patch, and Settings. The left sidebar shows the FILE STATUS (Working Copy), BRANCHES (7.x-1.x), TAGS, REMOTES (origin), and STASHES. The main area is divided into three panes: a Graph view, a commit details pane, and a diff view. The commit details pane shows the commit hash e774eadbbd690ac3a67748703fb197b8fcc33749, the author, the date (October 23, 2011), and the commit message: "Digidog: added 8.x tag to module info (this is an early 8.x shot, please don't use this - for maintainers only)". The diff view shows the changes to the file link.info, with a hunk of 6 lines. The diff highlights the addition of the core = 8.x line and the removal of the core = 7.x line.

FILE STATUS
Working Copy

BRANCHES
7.x-1.x

TAGS

REMOTES
origin
4.7.x-1.x
5.x-1.x
6.x-1.x
6.x-2.x
7.x-1.x
8.x-1.x
HEAD
master

STASHES

All Branches | **Current Branch** | **Show Remote Branches** | **Date Order** | **Jump to:**

Graph | **Description** | **Commit** | **Author** | **Date**

Uncommitted changes

origin/6.x-2.x Issue #1010850 by jcfiala, quartsize, raman2385, Bevan, eporama, das-peter, smit... 6d2d904 Digidog... Oct 28, 2011 9:...

HEAD origin/7.x-1.x 7.x-1.x Issue #1010850 by jcfiala, quartsize, raman2385, Bevan, epo... 649a430 Digidog... Oct 28, 2011 5:...

origin/8.x-1.x Digidog: added 8.x tag to module info (this is an early 8.x shot, please don't use thi... e774ead Digidog... Oct 23, 2011 9:...

7.x-1.0 Issue #1318850 by Digidog: Fixed completion of special chars for function link_validate_... 2b16004 Digidog... Oct 23, 2011 7:...

Issue #1199806 by ss81, jcfiala, Digidog: Fixed fatal error when the link URI is equal to page URL. cd53224 Digidog... Oct 23, 2011 3:...

Issue #1307788 Feature request by nmc, tars16, Digidog: Fixed Add custom maxlength to link title text afd93c4 Digidog... Oct 23, 2011 2:...

Commit: e774eadbbd690ac3a67748703fb197b8fcc33749
[e774ead]
Author:
Date: October 23, 2011 9:56:29 PM GMT+02:00
Labels: origin/8.x-1.x

Digidog: added 8.x tag to module info (this is an early 8.x shot, please don't use this - for maintainers only)

Context: 3 Lines | **Diff Parent** | **Show Whitespace** | **External Diff**

link.info
Modified file, 1 lines added, 1 lines removed

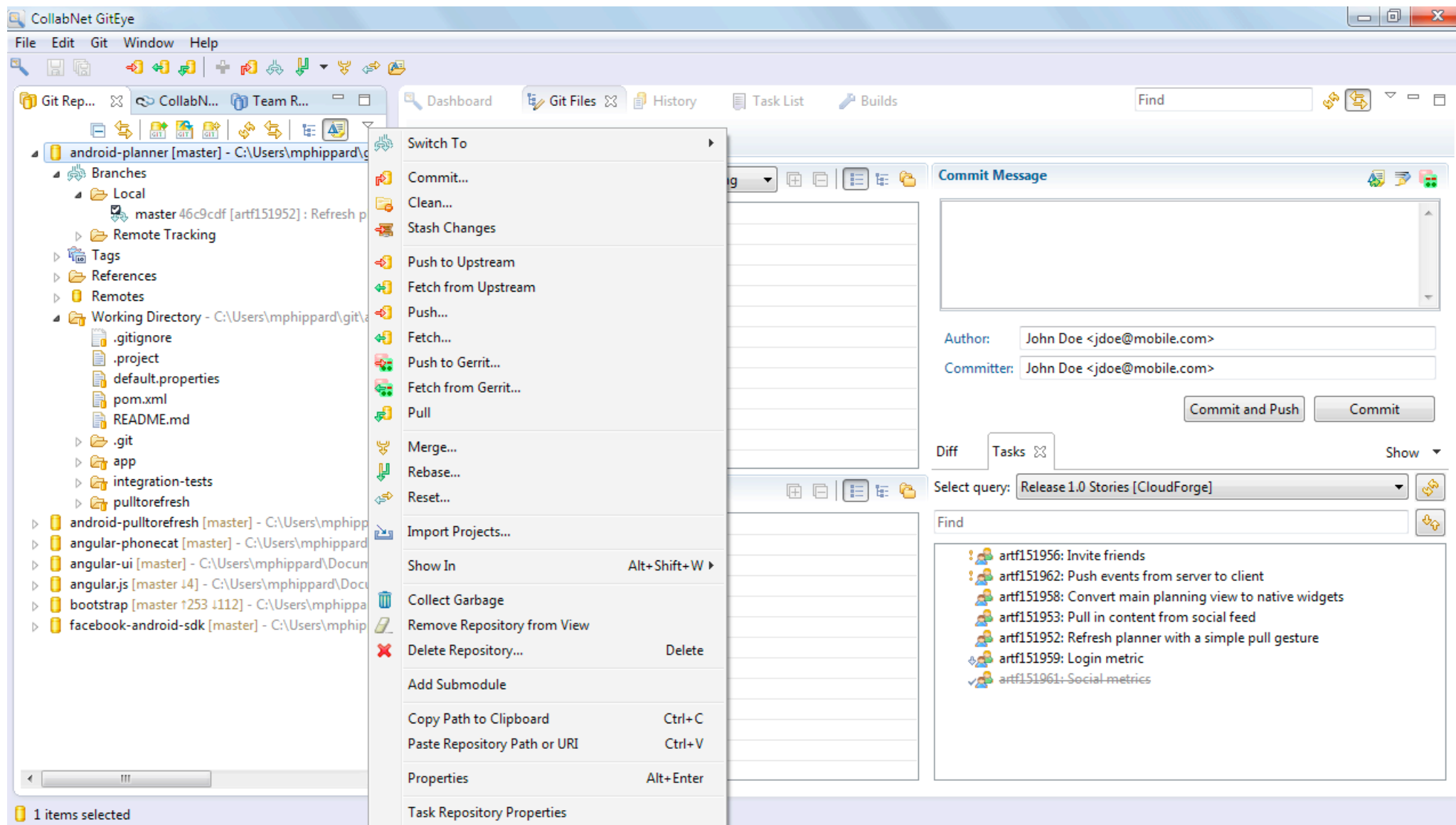
Hunk 1 : Lines 1-6

```
1 1 name = Link
2 2 description = Defines simple link field types.
3 - core = 7.x
3 + core = 8.x
4 4 package = Fields
5 5
6 6 files[] = link.module
```

7.x-1.x 1 Modified 1 Not Tracked

ATLASSIAN

Client Git : GitEye (Mac, Windows, **Linux**)



Attention !

❑ Ne pas utiliser EGit sous Eclipse

- Effectue des commandes dans votre dos
- Se plante
- Met le bazar dans tous les repositories



❑ Git

- Pro-Git Book : <http://git-scm.com/book/fr>
- <http://www.cheat-sheets.org/saved-copy/git-cheat-sheet.pdf>
- http://www.git-tower.com/files/cheatsheet/Git_Cheat_Sheet_grey.pdf
- <http://git-scm.com/>

❑ SVN

- The SVN Book: <http://svnbook.red-bean.com/>
- <http://subversion.tigris.org/>

