



Migration d'une base de code subversion vers git

Geoffrey Bachelet – PMSIpilot

Forum PHP 2010

PMSIpilot

"Le groupe PMSIpilot conçoit, édite et distribue des logiciels étudiés spécialement pour les établissements de santé."



The banner features a group of five healthcare professionals (three men and two women) in white coats, smiling. To their right is the text "Innovations" followed by a paragraph: "2 à 3 nouveaux logiciels par an, plus de 70 000 utilisateurs en France et des implantations en cours au Canada et dans plusieurs pays d'Europe. Le groupe PMSIpilot conçoit, édite et distribue des logiciels étudiés spécialement pour les établissements de santé." Below this is a login section with "Identifiant : [input]" and "Mot de passe : [input]". To the right of the login fields is a small icon of a padlock. Below the login section is a blue banner for "Qualifact" with the text "Vérifier et contrôler la chaîne de production de votre facturation hospitalière." and an image of people working at a computer. The PMSIpilot logo is visible on the right side of the banner.

Innovations

2 à 3 nouveaux logiciels par an, plus de 70 000 utilisateurs en France et des implantations en cours au Canada et dans plusieurs pays d'Europe. Le groupe PMSIpilot conçoit, édite et distribue des logiciels étudiés spécialement pour les établissements de santé.

Identifiant : Mot de passe :

Qualifact

Vérifier et contrôler la chaîne de production de votre facturation hospitalière.

PMSIpilot



PMSIpilot

- 8 logiciels interdépendants
- 15 développeurs
- 250 000+ lignes de code
- 6.5G de dépôt subversion
 - 24 000+ révisions
 - Fichiers binaires



Pourquoi ?

- Flexibilité des branches
- Gestion d'un gros projet
- Facilités de backport
- Fonctionnalités « bonus »

Flexibilité des branches

- Coût de création d'une branche très faible
- Branches locales
- « feature branches »

Gestion d'un gros projet

- Gère les sources du noyau Linux
 - 17 000+ révisions
- Quelques surprises quand même
 - Fichiers binaires volumineux

Facilité des backports

- Une branche par version
- « git cherry-pick »
- « git rebase [--interactive] »



Bonus

- « git stash »
- « git add --interactive »
- « git rebase --interactive »
- « git commit --amend »
- Commit en mode déconnecté



Migration du dépôt

git-svn

Problèmes avec git-svn

- C'est long (7h d'import)
- Gère mal les tags

Solutions !

- Importer un dépôt local (`file:///path/to/svn`)
- Convertir les « branches tags » en vrais tags
 - <http://gitready.com/advanced/2009/02/16/convert-git-svn>
 - <http://blue-dwarf.de/wp/2008/08/30/converting-git-svn-ta>
 - <http://progit.org/book/ch8-2.html>
 - ...



Ne pas oublier !

```
git svn init --stdlayout
```



Hébergement du serveur

Par un tiers ou en interne ?



Tiers

- Pas de serveur à gérer
- Services autour du dépôt
- Large choix
 - <https://git.wiki.kernel.org/index.php/GitHosting>
 - <http://stackoverflow.com/questions/109440/best-git-repo>



Interne

- Contrôle des données
- Flexibilité
- Interfaçage avec l'existant



Gitosis

- Authentification des développeurs par clés RSA
- Gestion des droits par utilisateur et par dépôt
 - Lecture seulement
 - Lecture / écriture



Problème rencontré

Fichiers binaires volumineux

Fichiers binaires volumineux

- Consommation mémoire lors du « repack »
- Problème connu
 - <http://www.google.com/search?q=git+fatal+out+memory>
 - 223 000 résultats





Solution ?

- Avoir une version de git à jour
- « git repack -adf »
 - Echoue sur notre serveur (4Go de RAM, 32bits)



Solution !

git help repack

--window

--window-memory

--max-pack-size



Réécrire l'histoire

git filter-branch



Réécrire l'histoire

```
git filter-branch --index-filter 'git rm bigfile' -- --all
```



Réécrire l'histoire

git filter-branch --index-filter 'git rm bigfile' --**--all**

Réécrire l'histoire

- <http://progit.org/book/ch9-7.html>
- `git help filter-branch`
- Beaucoup d'outils dans git
 - `git count-objects`
 - `git rev-list`
 - `git verify-pack`
 - etc





Réécrire l'histoire

N'en faites pas trop quand même.



Réécrire l'histoire

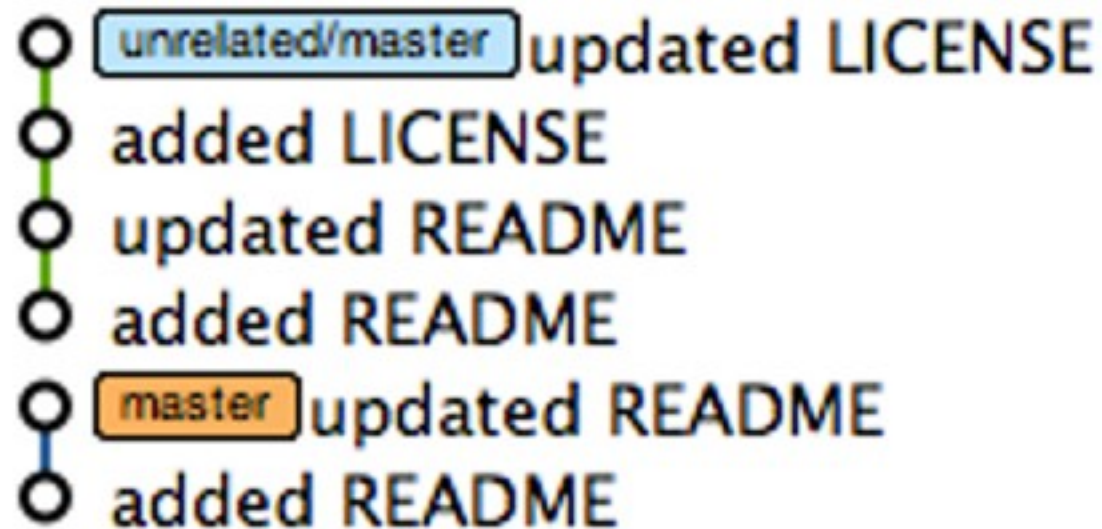
Les grafts



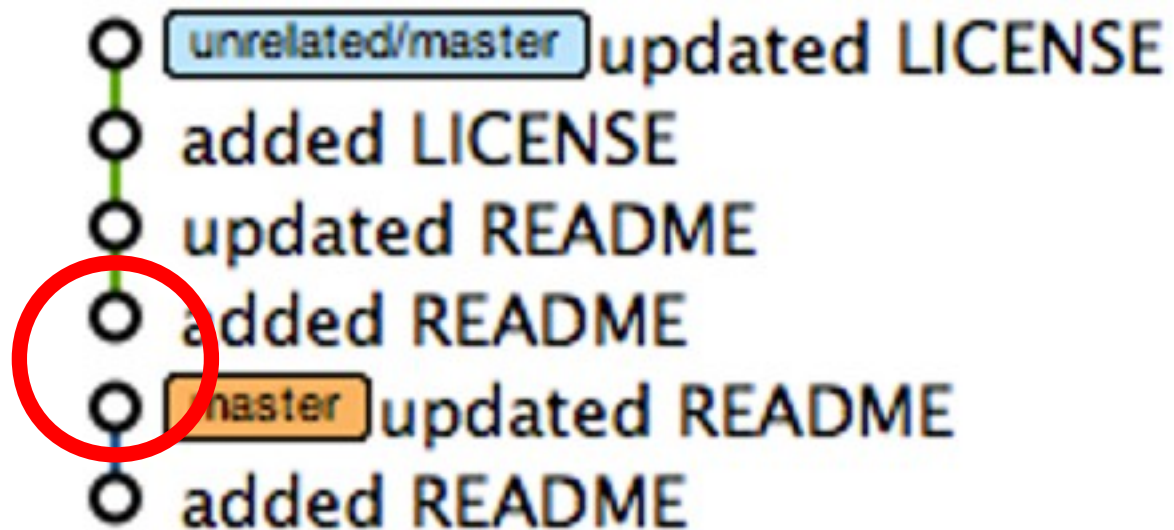
Les grafts

Deux dépôts, zéro commits communs.

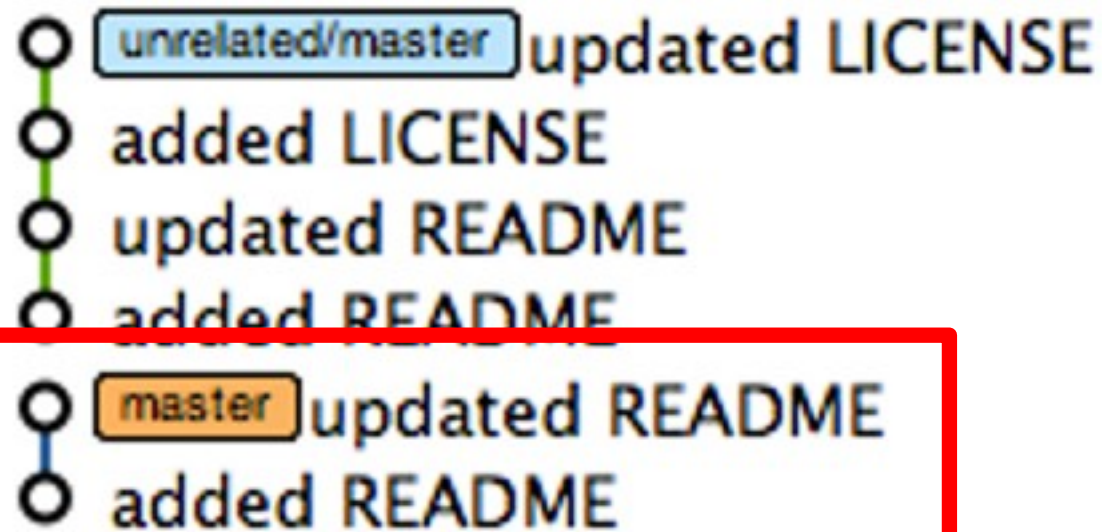
Les grafts



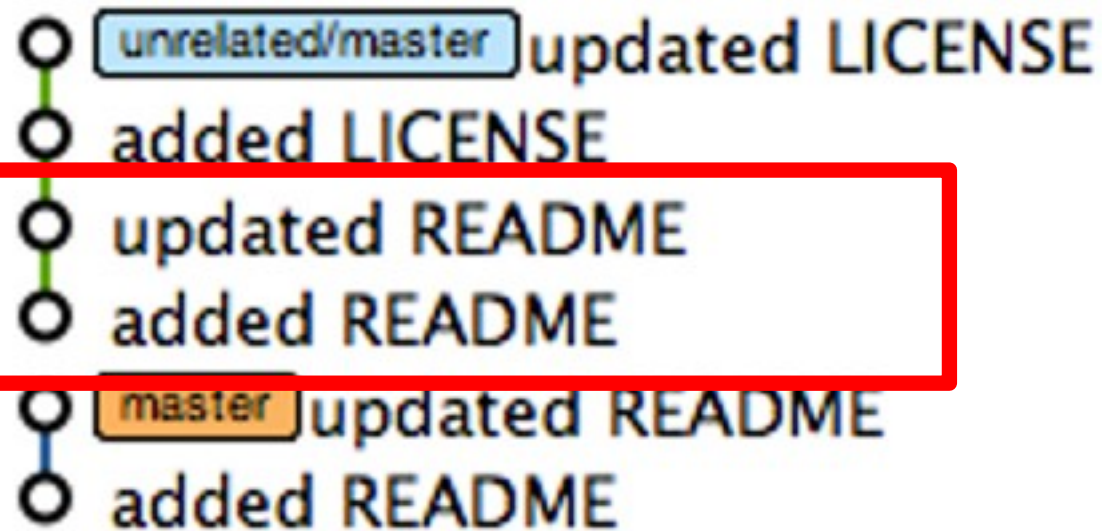
Les grafts



Les grafts



Les grafts





Les grafts

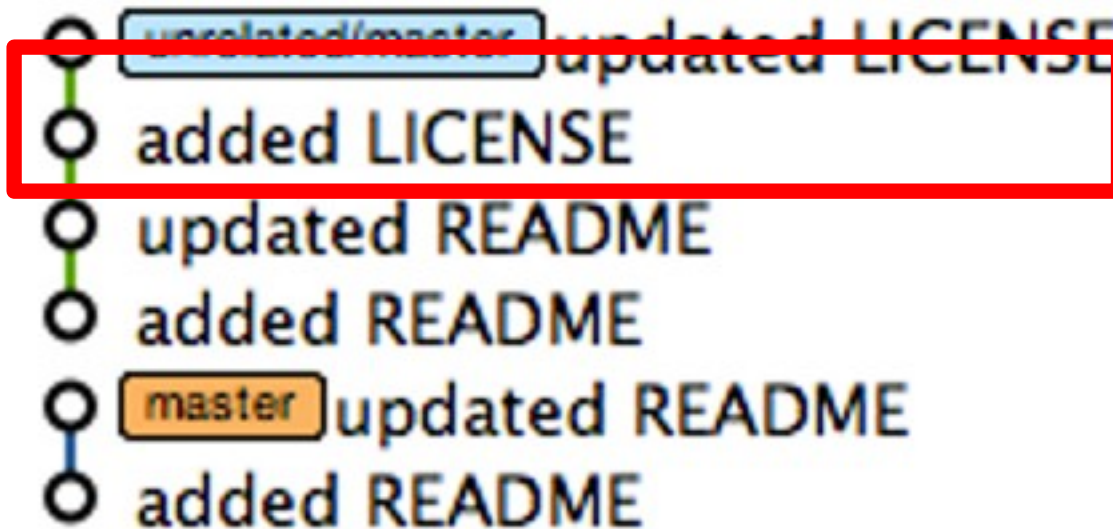
« .git/info/grafts »



Les grafts

<commit SHA1> <parent SHA1>

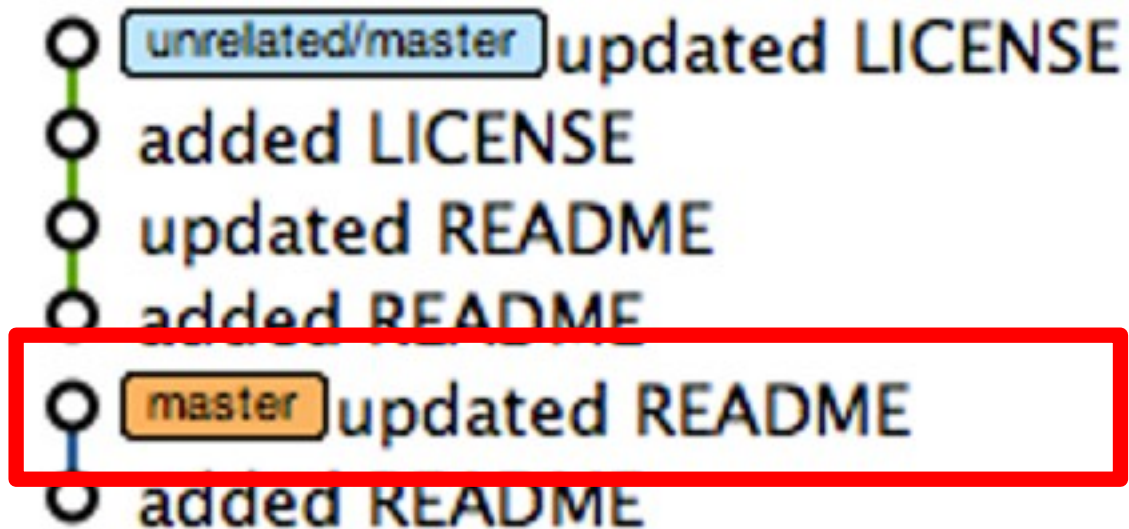
Les grafts



dc5b6cfa



Les grafts



746f0b6b



Les grafts

<commit>	<parent>
dc5b6cfa	746f0b6b

Les grafts

- `unrelated/master` updated LICENSE
- added LICENSE
- `master` updated README
- added README





Les grafts

`git help filter-branch`

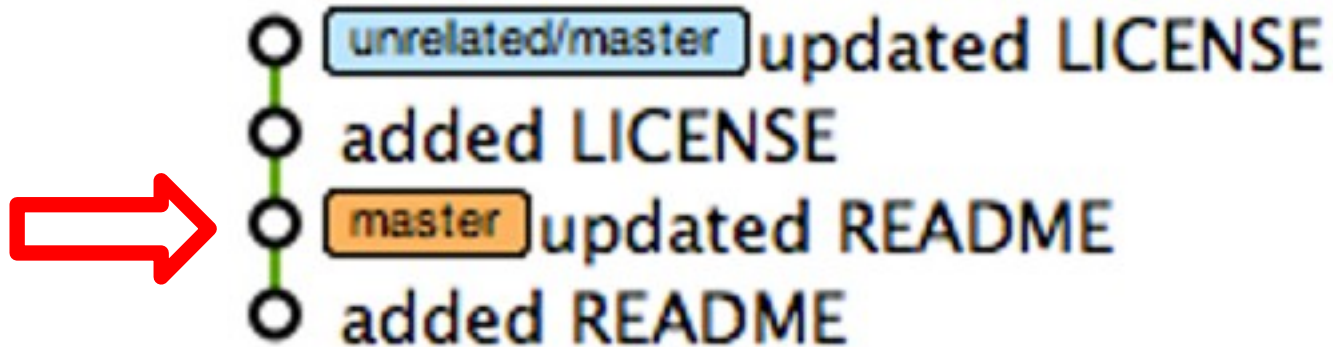
NOTE: This command honors `.git/info/grafts`. If you have any grafts defined, running this command will make them permanent.



Les grafts

« git filter-branch -- dc5b6cfa..unrelated-master »

Les grafts

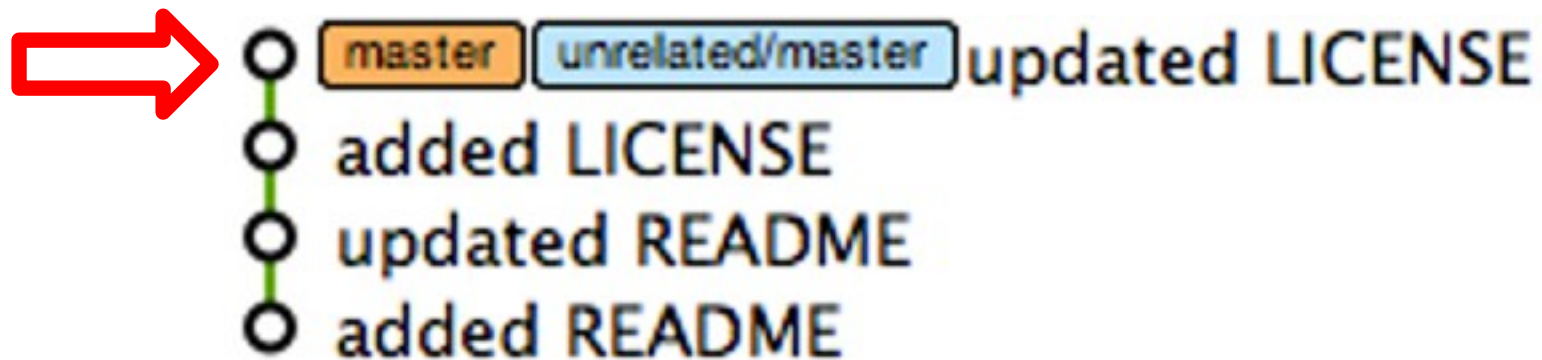




Les grafts

git reset --hard unrelated-master

Les grafts





BLACK MAGIC VODOO



Les développeurs



Les développeurs

- Nouveau workflow
 - Similaire à l'ancien pour commencer
- Nouveaux logiciels
 - git, gitk, gitx
 - Redmine
 - PHPStorm



Les développeurs

- Formation interne
 - Réunions techniques
 - Référents internes
 - Auto-formation



Auto-formation

- <http://progit.org/>
- <http://git-scm.com/documentation>
- <http://help.github.com/>
- <http://gitref.org/>
- etc.





Problème rencontré

Beaucoup de nouveaux concepts

Nouveaux concepts

- Fonctionnement en mode distribué
- Confusion entre « remote » et « branche »
- Confusion entre « commit » et « push »
- Confusion sur le « pull » et le « merge »
- Vocabulaire différent
 - « revert » vs « checkout » / « reset »





Nouveaux concepts

Oubliez tout ce que vous croyez savoir.



Vraiment.



Questions ?



C'est fini !

- Merci à l'AFUP
- Merci à PMSIpilot (on recrute !)
- Merci à vous !