

Introduction à GIT

le **cnam**

Pierre Courtieu

- ★ gérer les versions d'un répertoire
- ★ édition collaborative contrôlée
- ★ fichiers textes principalement

Édition ~~simultanée~~ désynchronisée car :

- ★ erreurs de compilation pas encore corrigées
- ★ bugs pas encore corrigés
- ★ on va prendre un café
- ★ pas de connexion pendant qq temps

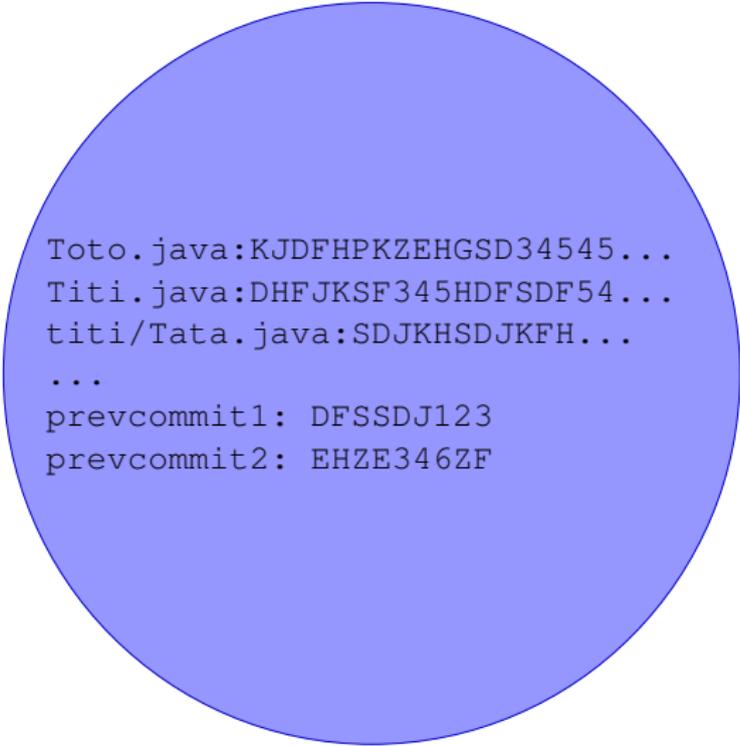


On ne “publie” que des versions qui marchent

- ★ on publie quand ça marche
- ★ que font les autres lorsque je publie ?
- ★ ils doivent se mettre à jour
- ★ ils ont eux aussi des modifications en cours
- ★  réconciliations de versions (merge)
- ★ problème compliqué
- ★ GIT & co : solution puissante mais parfois non triviale

- ★ archive (repository)
- ★ version d'une archive (commit)
- ★ répertoire de travail (working directory)
- ★ staging area
- ★ branches (non traitées ici mais très utile)

- ★ “photo” de l’état de l’archive à un moment
- ★ + réf. commit(s) “précédent(s)”



```
Toto.java:KJDFHPKZEHGSD34545...  
Titi.java:DHFJKSF345HDFSDF54...  
titi/Tata.java:SDJKHSDJKFH...  
...  
prevcommit1: DFSSDJ123  
prevcommit2: EHZE346ZF
```

- ★ “photo” de l’état de l’archive à un moment
- ★ + réf. commit(s) “précédent(s)”



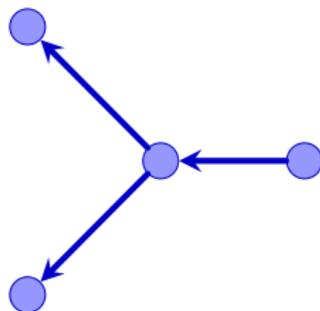
- ★ archive (repository) = historique de toutes les “photos”



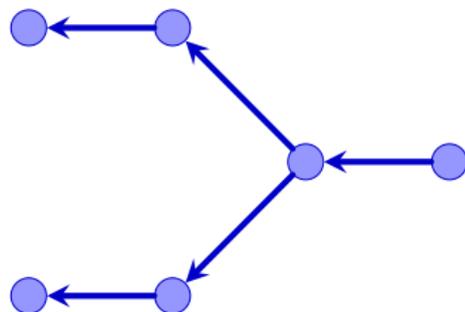
- ★ archive (repository) = historique de toutes les “photos”



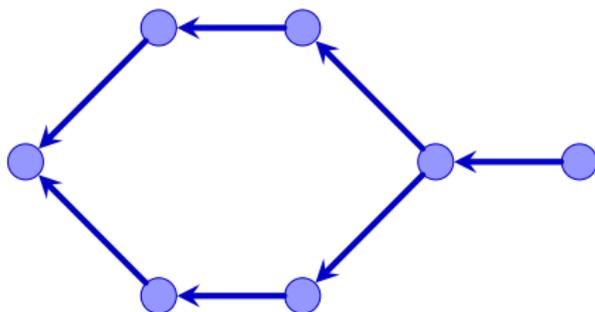
- ★ archive (repository) = historique de toutes les “photos”



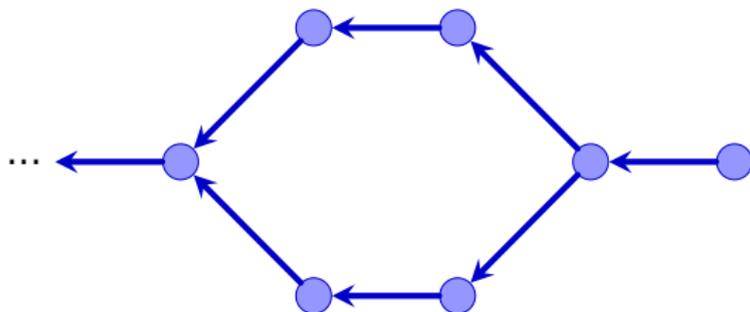
- ★ archive (repository) = historique de toutes les “photos”



- ★ archive (repository) = historique de toutes les “photos”



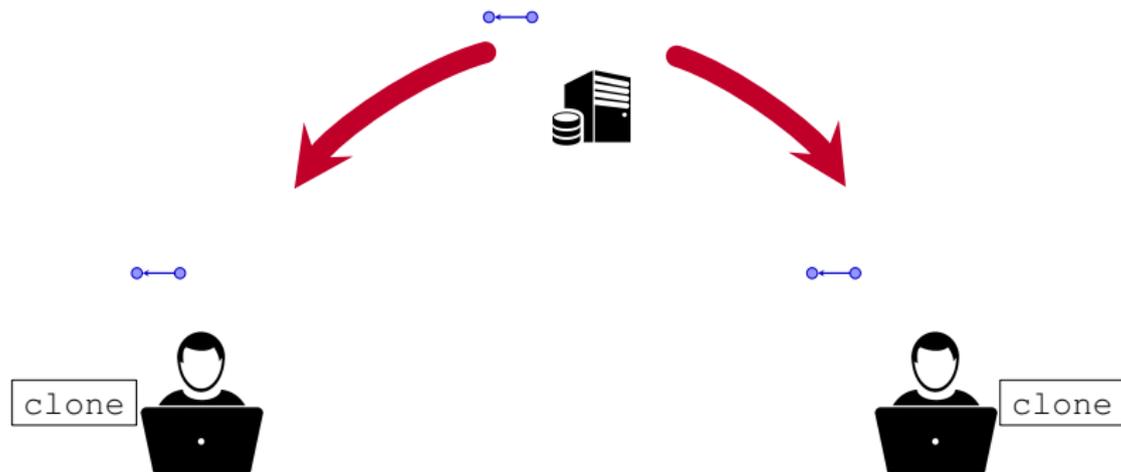
- ★ archive (repository) = historique de toutes les “photos”



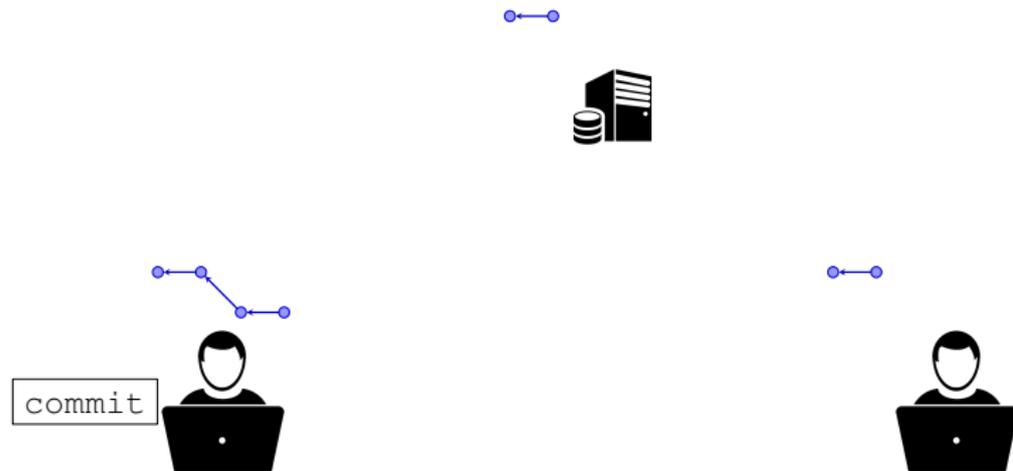
Chaque utilisateur possède son propre repository complet !

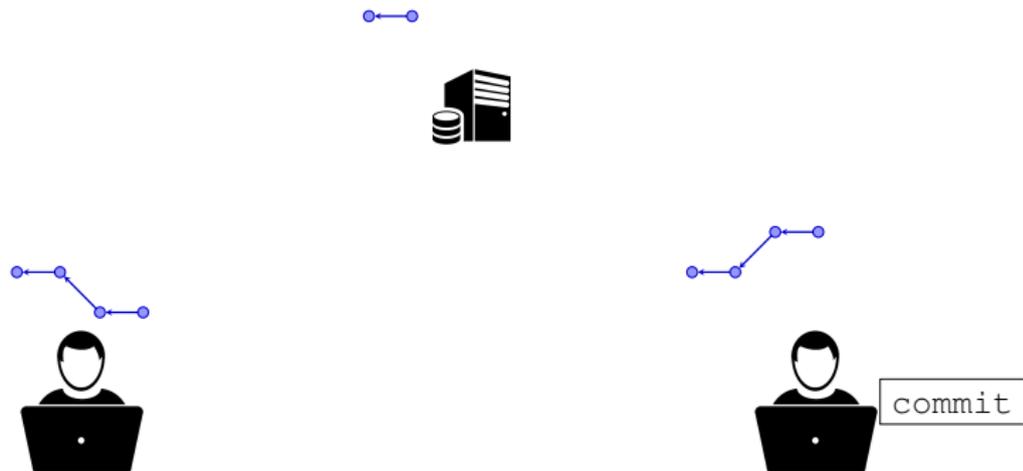
- ★ tout l'historique, tous les commits
- ★ mise à jour = échange de commits avec une autre archive
- ★ entre utilisateurs ou avec un serveur au milieu

Scénario classique : Une archive centrale

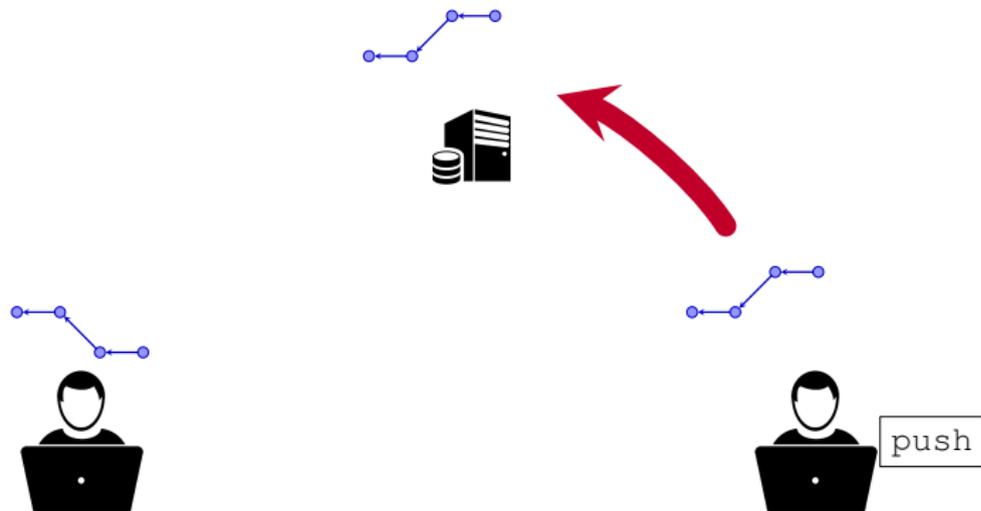


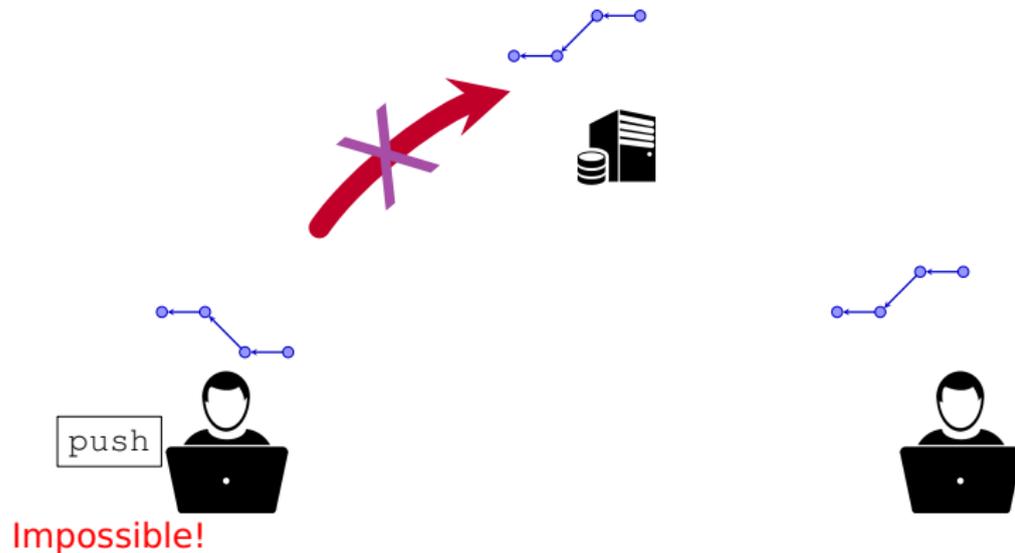
Scénario classique : Une archive centrale





Scénario classique : Une archive centrale

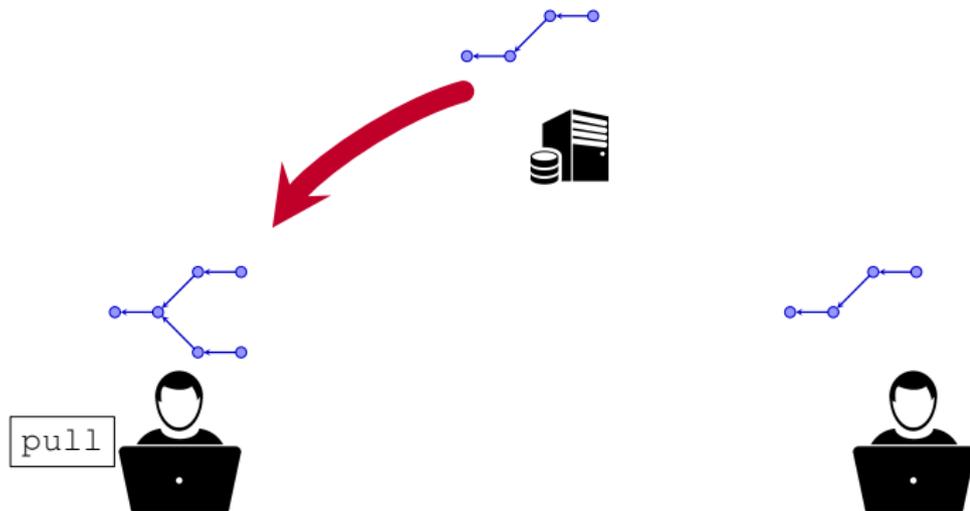




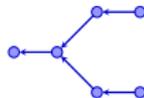
Impossible!



On ne peut pas pousser un commit sur un arbre qui a changé



D'abord se mettre à jour



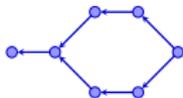
pull



merge!



parfois arbitrer les divergences



commit

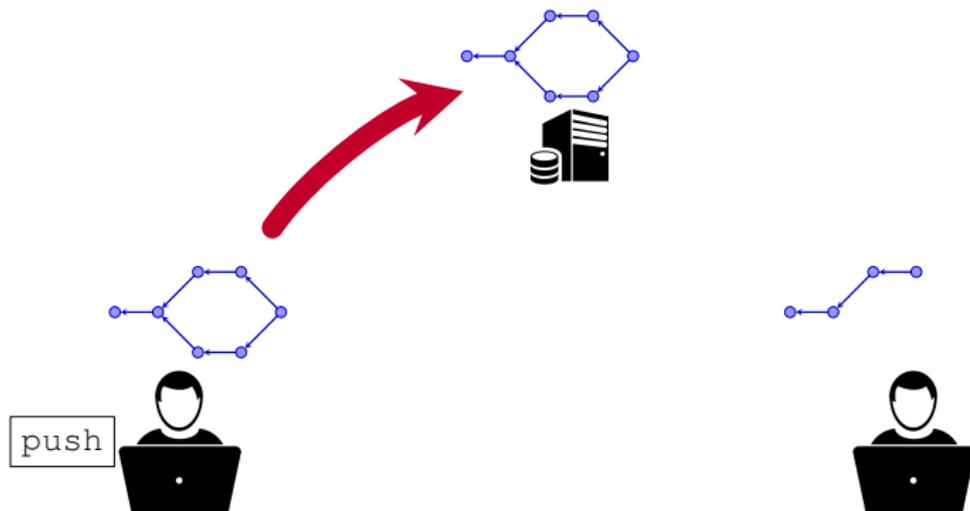


merge!

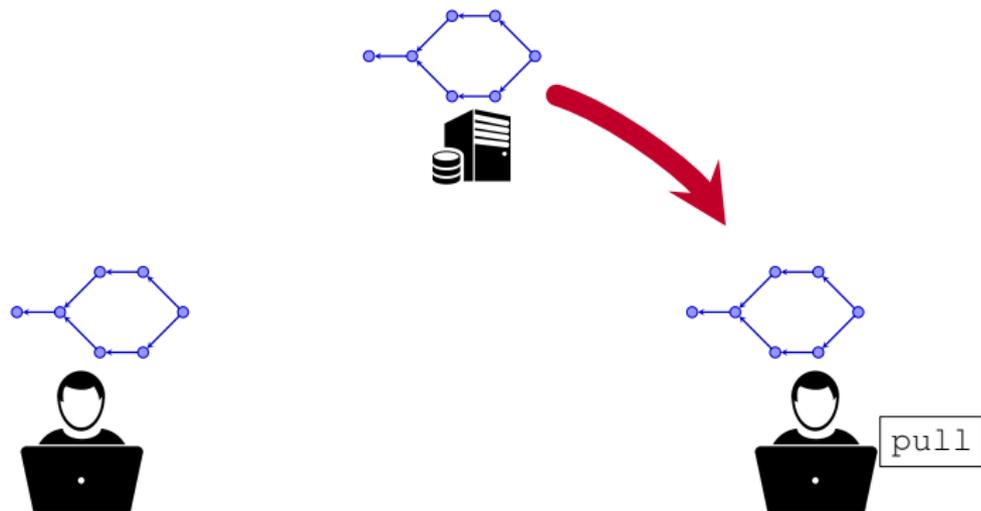


parfois arbitrer les divergences **PAR UN NOUVEAU COMMIT.**

Scénario classique : Une archive centrale



Scénario classique : Une archive centrale



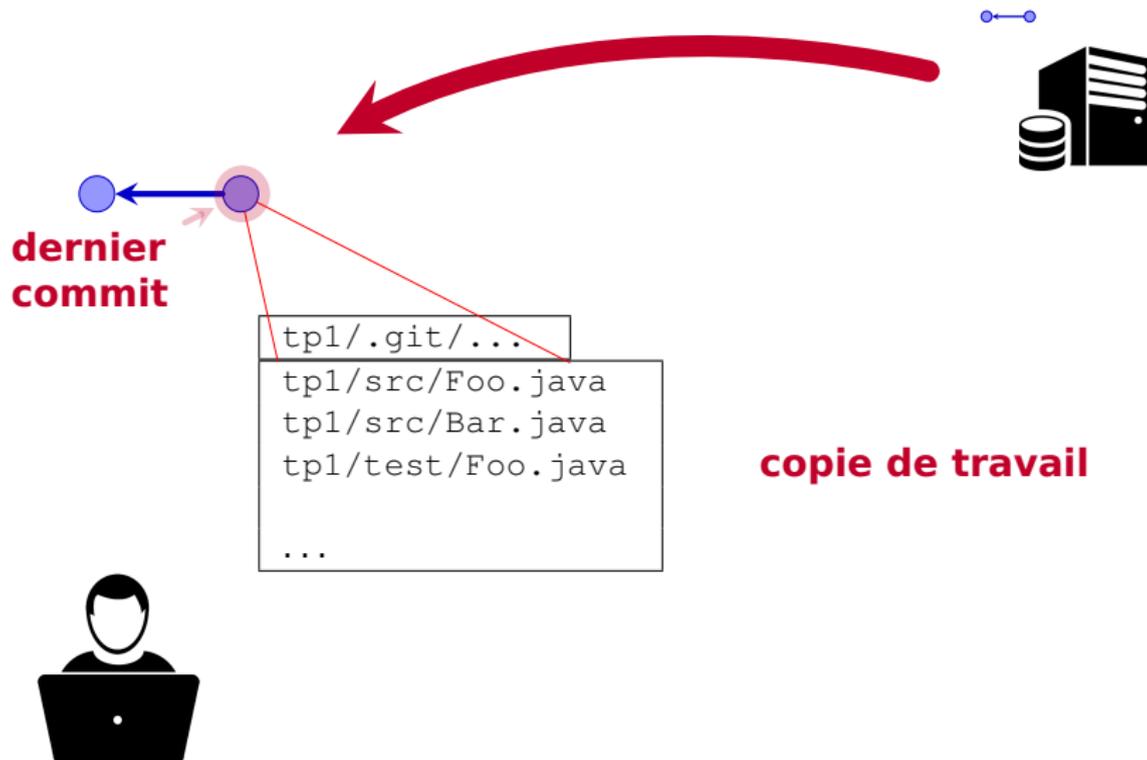


```
tp1/.git/...
```

archive complète



```
git clone https://gitlab.../xxx.git
```





```
tp1/.git/...  
tp1/src/Foo.java  
tp1/src/Bar.java  
tp1/test/Foo.java  
...
```

édition des fichiers



`netbeans/vscode/emacs/...`



```
tp1/.git/...  
tp1/src/Foo.java  
tp1/src/Bar.java  
tp1/test/Foo.java  
...
```

édition des fichiers



`netbeans/vscode/emacs/...`



```
tp1/.git/...  
tp1/src/Foo.java  
tp1/src/Bar.java  
tp1/test/Foo.java  
tp1/test/Oof.java  
...
```

ajouts des fichiers



```
netbeans/vscode/emacs/...
```



```
tp1/.git/...  
tp1/src/Foo.java  
tp1/src/Bar.java  
tp1/test/Foo.java  
tp1/test/Oof.java  
...
```

préparation commit



```
git add src/Foo.java src/Bar.java src/Oof.java
```



```
tp1/.git/...
```

archive modifiée

```
tp1/src/Foo.java  
tp1/src/Bar.java  
tp1/test/Foo.java  
tp1/test/Oof.java  
...
```

commit



```
git commit -m "Correction Bar + tests"
```



```
tp1/.git/...  
tp1/src/Foo.java  
tp1/src/Bar.java  
tp1/test/Foo.java  
tp1/test/Oof.java  
...
```

push



`git push`

Démo sans netbeans

- ★ Menu dédié au VC, disponible à 2 endroits :
 - ★ menu général `Teams`
 - ★ menu contextuel du PROJET : `Git`
- ★ `clone, pull, push` : **sous-menu** `Remote`
- ★ `add + commit` **en une seule étape** : `commit`.



TOUJOURS sélectionner le PROJET d'abord

Démo avec netbeans

Un projet git contient un ou plusieurs projet(s) Netbeans

Règles **absolues** :

- ★ **JAMAIS un projet git dans un autre projet git.**
- ★ **JAMAIS un projet Netbeans dans un projet Netbeans.**
- ★ **JAMAIS plusieurs projets git dans un projet Netbeans.**

tps/.git/...

tps/tp1/src/Foo.java
tps/tp1/src/Bar.java
tps/tp1/test/Foo.java
tps/tp1/test/Oof.java
tps/tp1/pom.xml
...
...

Projet Netbeans

Projet git

tps/tp2/src/XXX.java
tps/tp2/src/YYY.java
tps/tp2/testTestXXX.java
tps/tp2/test/TestYYY.java
tps/tp2/pom.xml
...

Projet Netbeans

NON : un projet git dans un autre

tps/.git/... 

tps/tp1/src/Foo.java
tps/tp1/src/Bar.java
tps/tp1/test/Foo.java
tps/tp1/test/Oof.java
tps/tp1/pom.xml
tps/tp1/.git/... 

...

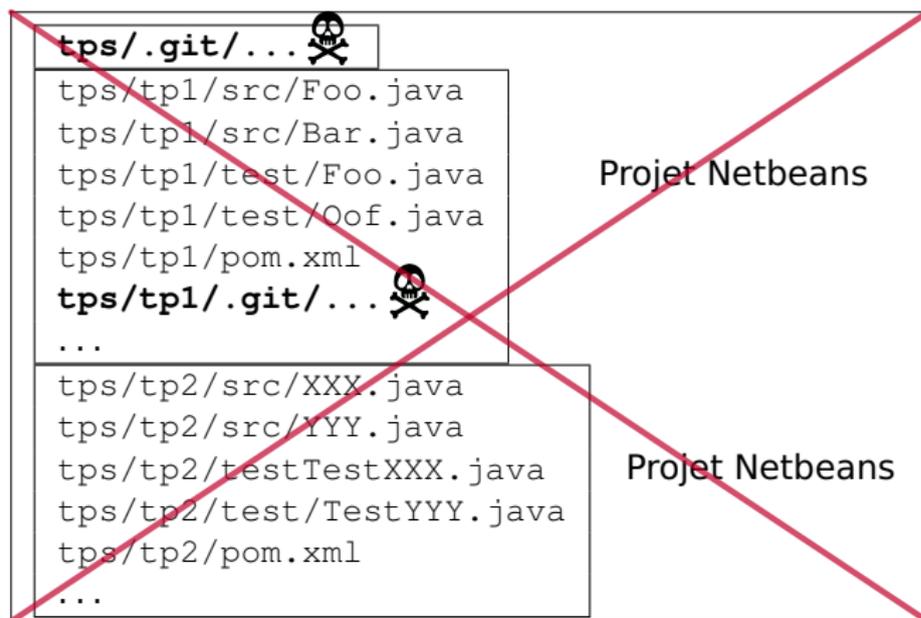
Projet Netbeans

Projet git

tps/tp2/src/XXX.java
tps/tp2/src/YYY.java
tps/tp2/testTestXXX.java
tps/tp2/test/TestYYY.java
tps/tp2/pom.xml
...

Projet Netbeans

NON : un projet git dans un autre



Fin