



# git

# Versioning avec Git

---

**Baptiste LECOCQ**

Promo B3

Président EPNET

<http://tiste.io>

<http://www.epnet.fr>

@tiste



**Jean-Baptiste VIGNERON**

Promo I5

Parrain EPSILab

<http://www.jbvigneron.fr>

<http://www.epsilab.net>

@jbvigneron



# Sommaire

---

- **Le versioning : la théorie**

- Problématiques
- Intérêts
- Exemples de solutions existantes

- **Présentation de Git**

- Présentation et historique
- Les commandes de base
- Les différents états d'un fichier
- Les branches
- Le GitFlow
- Les plateformes où vous pouvez utiliser Git

# Sommaire

---

- **GitHub**

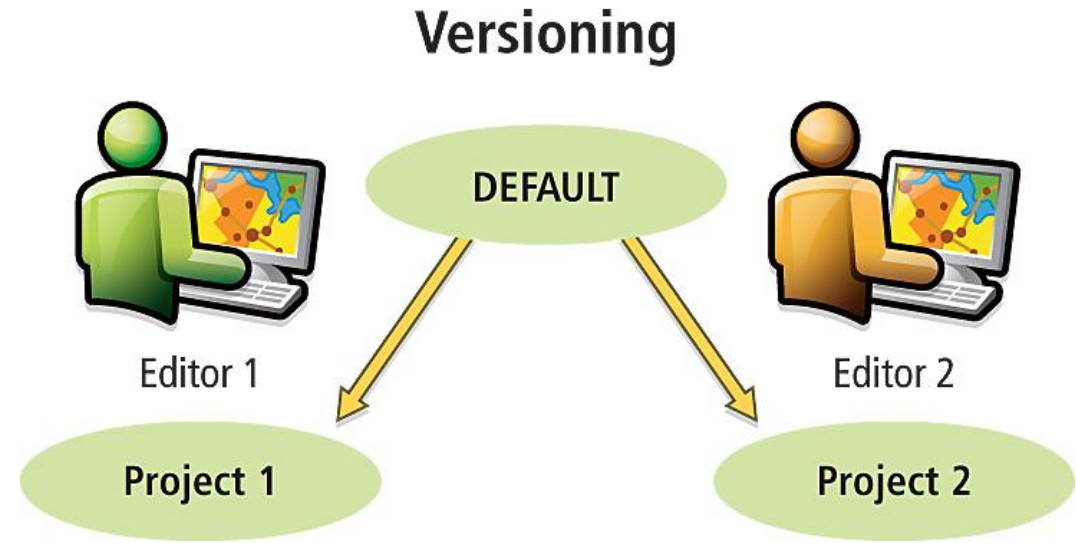
- Présentation du site
- Explication des remotes (push / pull)

- **GitHub pour Windows**

- Présentation et démo

- **Git dans Visual Studio, ça donne ?**

- Démo
- Atelier



# Le versioning : la théorie

---

# Le versioning

---

## Problématiques:

- Une équipe de développement composée de 3 personnes réalise une application. Ils souhaitent s'échanger leur fichiers sources. Comment procéder ?
- Un développeur a écrit du code qui fait planter l'application. Pire encore, un autre développeur a supprimé des fichiers par mégarde. Comment restaurer les fichiers ?
- 2 développeurs ont modifié le même fichier avec des modifications différentes. Comment les fusionner ?

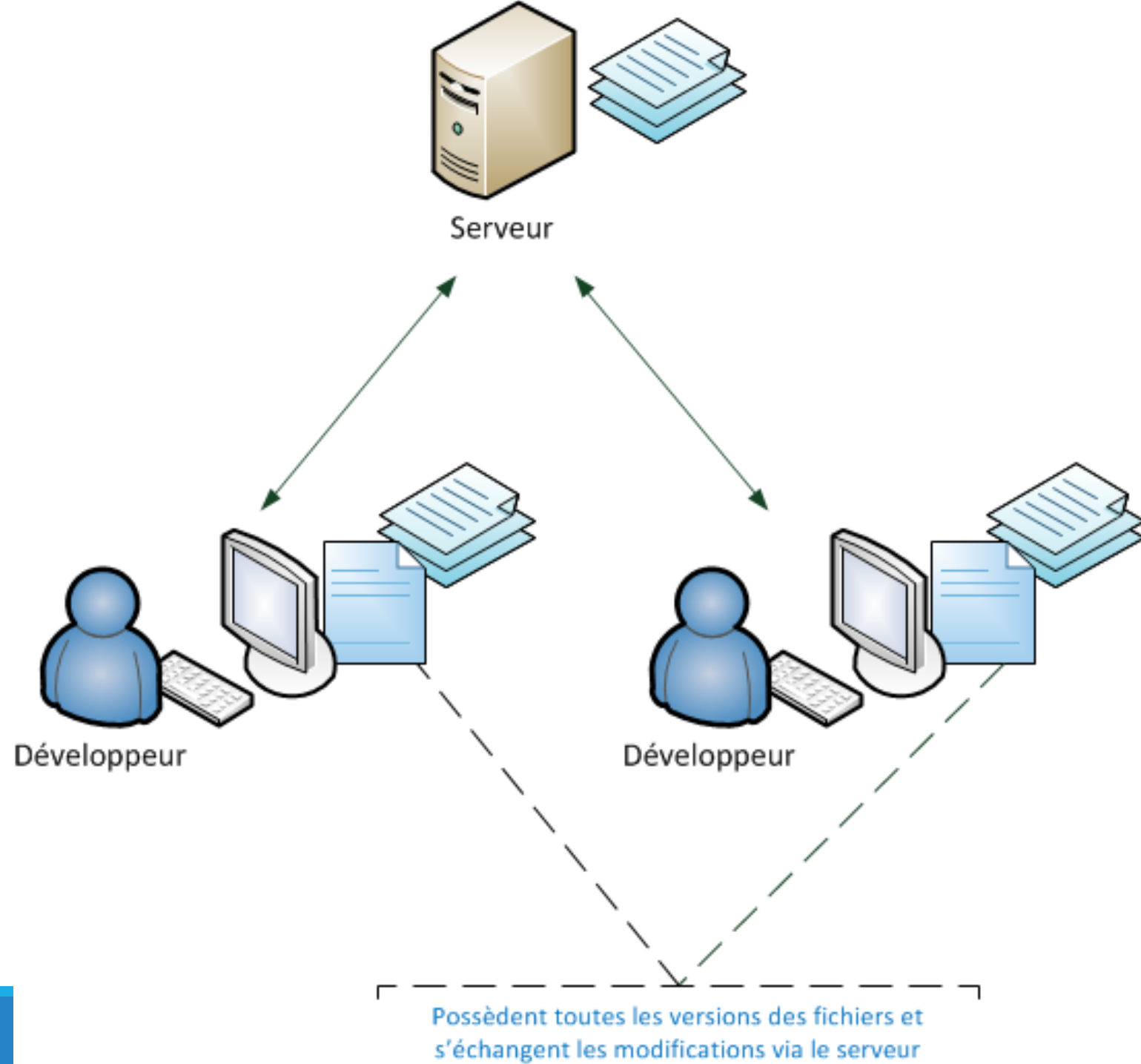
# Le versioning

---

Solution: Le **VERSIONING** !

## - Intérêts

- Partage du code source entre plusieurs développeurs
- Historisation des fichiers
- Gestion des conflits



# Le versioning

---

Exemples de solutions existantes:



**git**





**git**

# Présentation de Git

---

# Historique

---

Création en **2005** par **Linus Torvalds**

Logiciel de gestion de versions **décentralisé**

Disponible pour les systèmes **UNIX** (Mac et Linux) mais également **Windows**

Site officiel: <http://git-scm.com/>

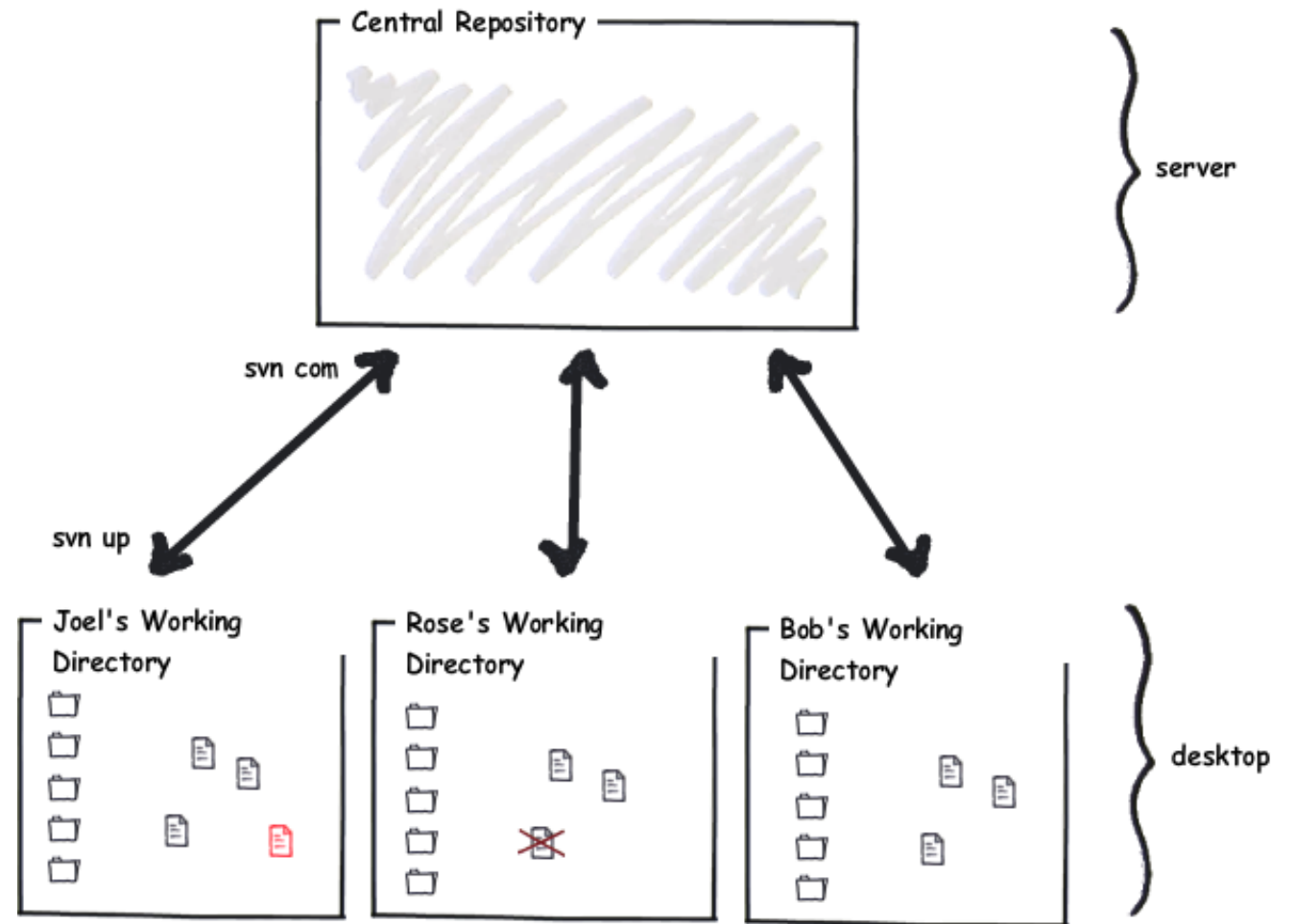


# Architecture centralisée

Exemple avec SVN

Utilisé également par Team Foundation Server

Image extraite de  
<http://hginit.com/00.html>



# Architecture décentralisée

Exemple avec Mercurial

Utilisé également par Git

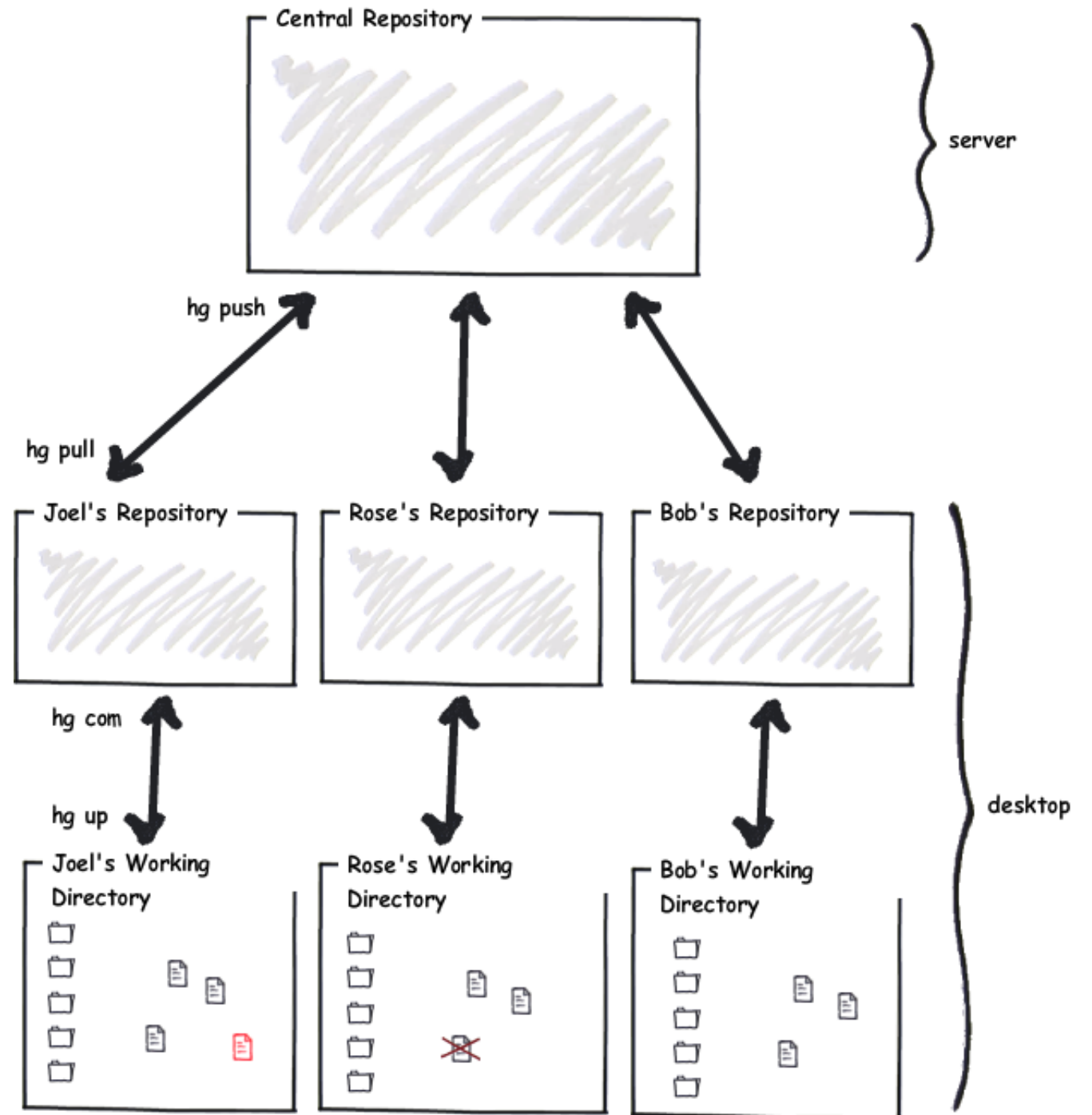
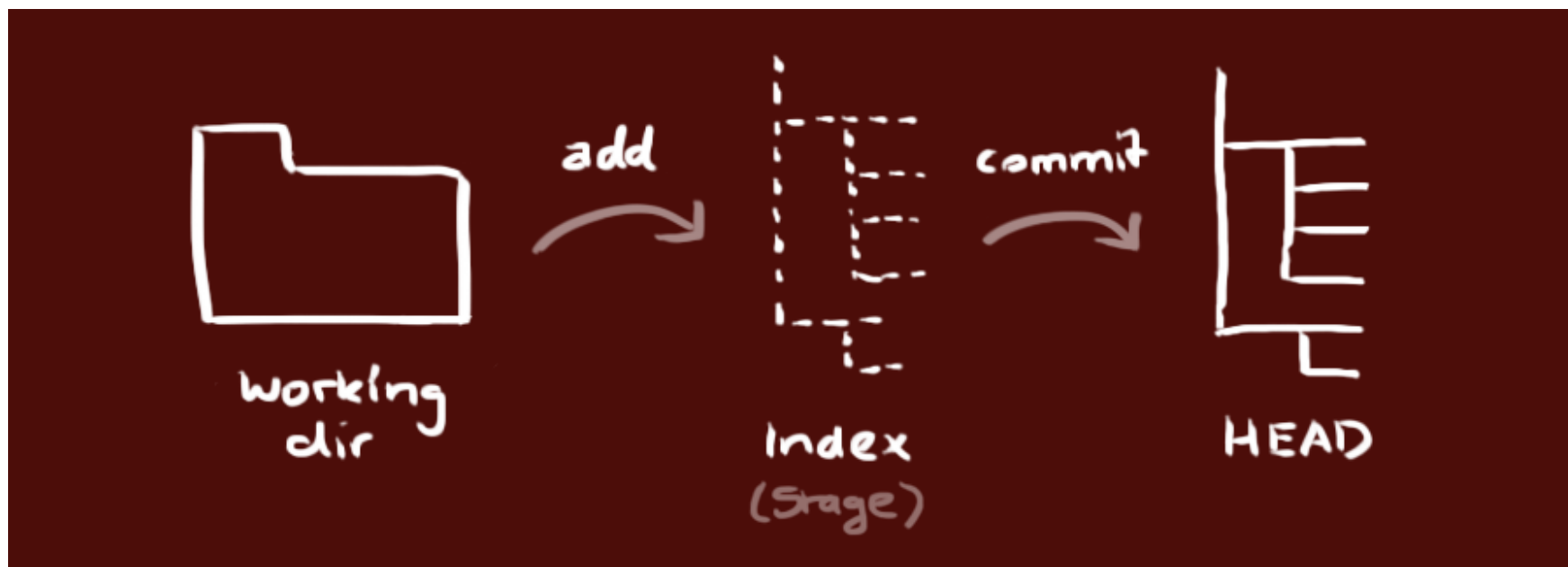
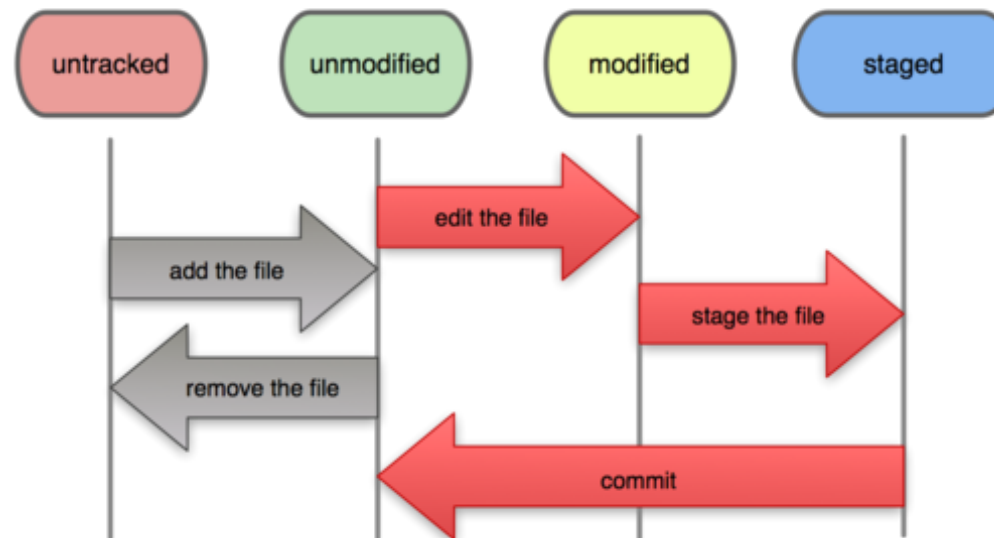


Image extraite de  
<http://hginit.com/00.html>



### File Status Lifecycle



# git init

Initialise un dossier caché .git afin de versionner le dossier courant

# git log

Afficher les informations de tous les commits (un commit représente une version de l'application)

# git status

Afficher l'état de notre git (ce qu'il se passe dans les différentes zones)

```
git diff
```

Afficher les différences entre le dernier commit et les dernières modifications

```
git diff master..develop
```

Afficher les différences de develop par rapport à master

```
git diff --cached
```

Afficher la différenciation entre le dernier commit et les dernières modifications de l'index

```
git add { . | nom_du_fichier }
```

Ajouter les fichiers versionnés ou non à la zone de staging (l'index)

```
git commit -m "{Message du commit}"
```

« Valide » les fichiers = création d'une nouvelle version



# Démo: Application des commandes

---

Création d'un fichier « README.md », le commiter une fois

Utiliser les commandes de bases

- git init
- git status
- git add
- git commit
- git log

# Les branches

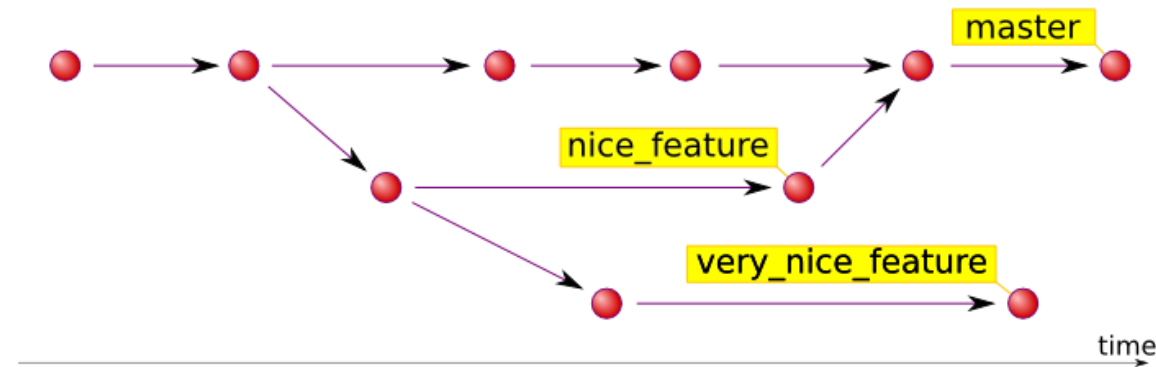
```
git checkout -b develop
```

=

```
git branch develop
```

+

```
git checkout develop
```

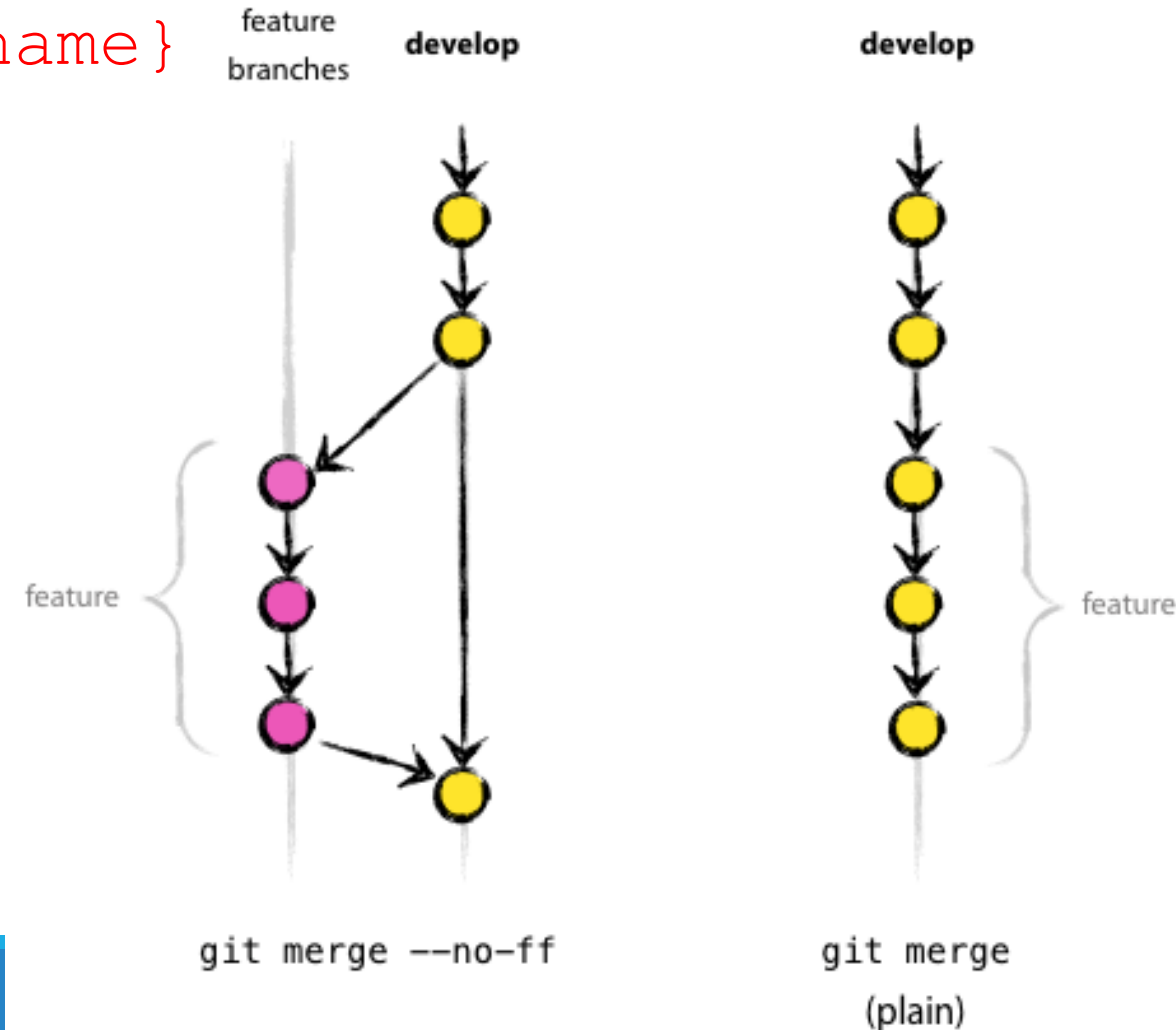


Crée une branche **develop** et s'y déplace

# La fusion entre branches

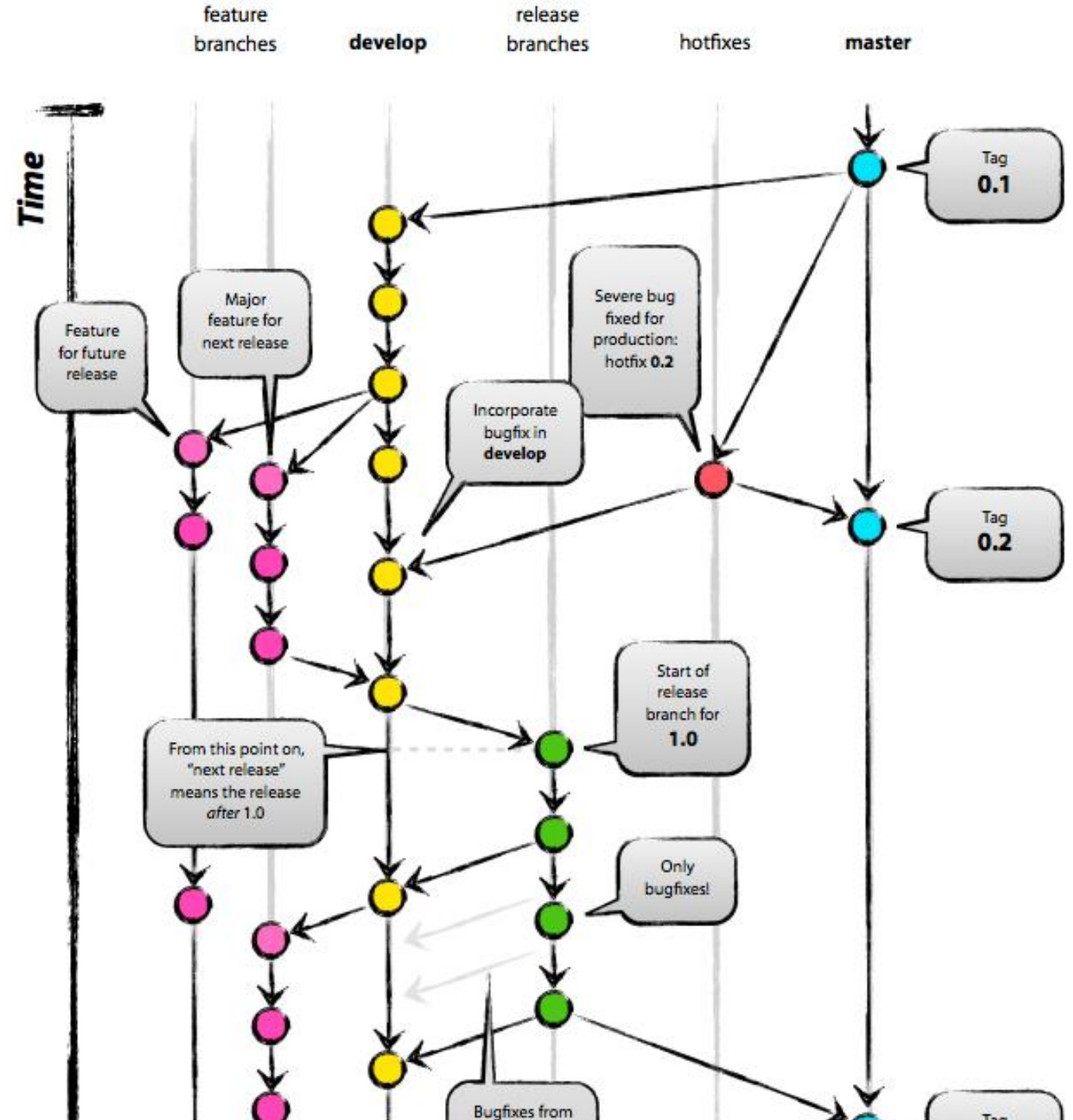
```
git merge [--no-ff] {branchname}
```

Fusionne *branchname* sur la branche courante



# GitFlow

Schéma disponible sur  
<http://nvie.com/posts/a-successful-git-branching-model/>



# Remotes

---

Une remote est un dossier git distant (GitHub, Bitbucket, Assembla, Visual Studio Online, Kiln, custom...)

Multiplicité de celles-ci (défaut : origin)

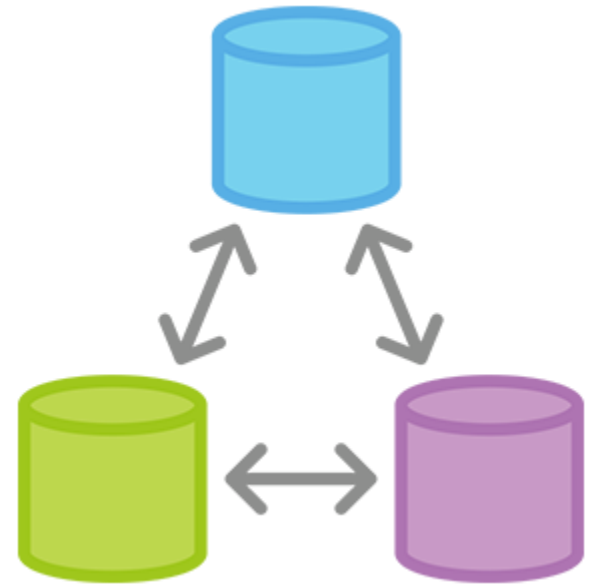


# Du local vers la remote

---

```
git push origin master
```

Pousse les modifications de **master** vers la remote **origin**



```
git clone {url}
```

Récupère le dossier de travail

```
git pull [--rebase]
```

=

```
git fetch
```

+

```
git merge
```

Récupère les modifications et les fusionne avec notre local



# GitHub



# Présentation de GitHub

---

Moyen visuel de contrôle d'un dossier Git (appelé repository)

Réseau social de coding

Outils de gestion de projet

CV en ligne

# Le profil

---

Les meilleurs repos

Les repo auxquels la personne a participé

Les informations sur la personne

Ses organisations (groupes de travail)

Les activités annuelles



## Baptiste Lecocq

tiste

EPSI

Lille, FR

[contact@tiste.io](mailto:contact@tiste.io)

<http://tiste.io>

Joined on Dec 30, 2012

19

followers

21

starred

9

following

### Organizations



[Contributions](#)

[Repositories](#)

[Public Activity](#)

[Edit Your Profile](#)

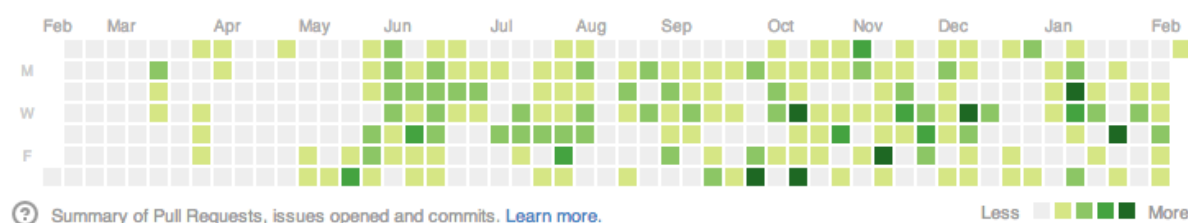
#### Popular repositories

<a href="#">sentencify</a>	Create sentences with array of Active R...	5 ★
<a href="#">paint</a>	Paint-like webapp written in Javascript, ...	2 ★
<a href="#">to_top</a>	A super-light "Back to top" jQuery function	1 ★
<a href="#">main_css_classes</a>	Simple repo to gather all useful css clas...	1 ★
<a href="#">makeitfauve.fr</a>	Create your own FAUVE background	1 ★

#### Repositories contributed to

<a href="#">epnet/beerify</a>	The rails engine which lets you the time ...	6 ★
<a href="#">epnet/EPNet</a>	This is our website, <a href="http://epnet.fr">http://epnet.fr</a>	0 ★
<a href="#">epnet/remisedesdiplomes</a>	This is the webapp regarding graduatio...	0 ★
<a href="#">epnet/ledomainedescascades</a>	Website of "Le Domaine des cascades"	0 ★
<a href="#">epnet/OCafeGourmand</a>	Website of Cafe Gourmand, pub located...	0 ★

#### Your Contributions



748 Total

Feb 16 2013 - Feb 16 2014

Year of Contributions

17 days

June 10 - June 26

Longest Streak

6 days

February 11 - February 16

Current Streak

## Contribution Activity

Period: 1 Week ▾

13 Commits

Pushed 1 commit to [tiste/ReadItLater](#) Feb 16

Pushed 2 commits to [tiste/main\\_css\\_classes](#) Feb 15

Pushed 2 commits to [tiste/betty](#) Feb 14

Pushed 6 commits to [tiste/ReadItLater-old](#) Feb 11 - Feb 13

# Le repository

---

Une interface de navigation dans le projet (raccourci : « t »)

Les langages principaux

Les outils de gestion de projet

Diverses statistiques (contributeurs, activités...)



Sleek, intuitive, and powerful front-end framework for faster and easier web development.

<http://getbootstrap.com>

8,482 commits

11 branches

27 releases

500 contributors



branch: master

bootstrap

regenerate npm-shrinkwrap.canonical.json



cvrebert authored 2 days ago

latest commit accc95ccd6



dist

grunt

3 days ago



docs

Update less.js to v1.6.3.

3 days ago



fonts

Remove execute permission on font files

5 days ago



grunt

Add build-raw-files log; fixes #12603

3 days ago



js

v3.1.1 bump

3 days ago



less

Fixes #11659, #12698, and #12349: Always apply position and z-index f...

5 days ago



test-infra

regenerate npm-shrinkwrap.canonical.json

2 days ago



.editorconfig

Don't re-set indent style for [\*.py]

15 days ago



.gitattributes

add .gitattributes file to enforce Unix newlines within the repo

3 months ago



.gitignore

need customize.js to be tracked for it to work on GitHub Pages

2 months ago



.travis.yml

rm BrowserStack testing from build

5 days ago



CNAME

add cname

7 months ago



CONTRIBUTING.md

Use different coding style for Gruntfile

6 days ago



Gruntfile.js

Switch to the minified JSZip.

4 days ago



LICENSE

missed a file

a month ago



README.md

v3.1.1 bump

3 days ago



\_config.yml

v3.1.1 bump

3 days ago

<> Code

Issues

131

Pull Requests

44

Pulse

Graphs

Network

SSH clone URL

git@github.com:tw



You can clone with [HTTPS](#), [SSH](#), or [Subversion](#).



Clone in Desktop



Download ZIP

# Issues

---

Peut être assimilé à un problème ou à une amélioration

On peut y assigner un membre, des tags (bug, amélioration...), gérer ses états (résolu, ouvert...)

# Milestone

---

Peut être assimilé à un jalon

Il possède une date butoire ainsi qu'un ensemble d'issues

# Pull-request

---

Il s'agit d'une demande de fusion entre une branche et une autre

Répond généralement à une issue

Possède une interface de revue de code ainsi qu'une timeline de commentaires





# GitHub for Windows

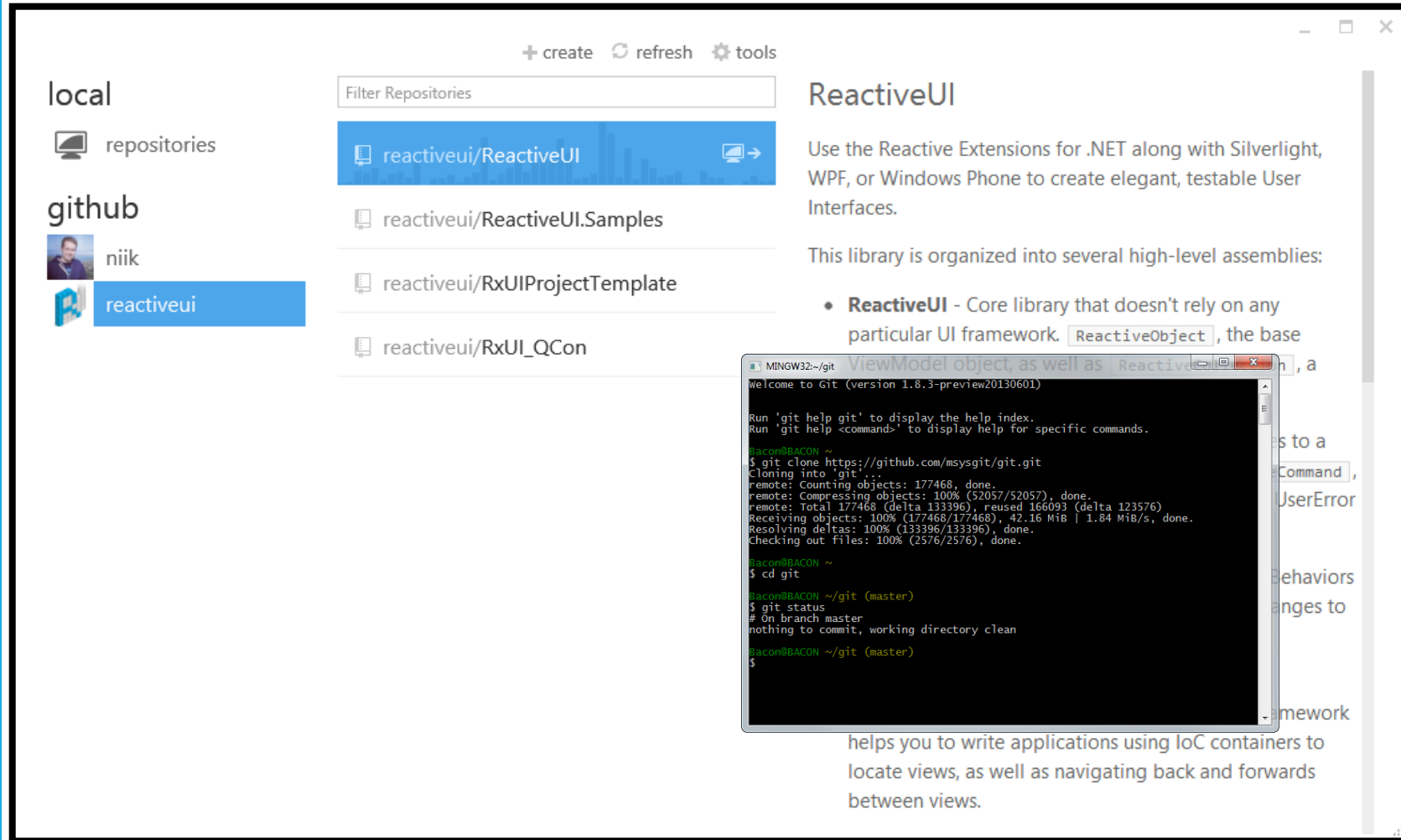
---

# GitHub for Windows

Disponible sur

<http://windows.github.com/>

Compatible avec Windows Vista, 7 et 8



The screenshot displays the GitHub for Windows application interface. On the left, the 'local' sidebar shows 'repositories' and 'github' sections. The 'github' section lists users 'niik' and 'reactiveui', with 'reactiveui' selected. The main area shows a list of repositories for 'reactiveui', including 'ReactiveUI', 'ReactiveUI.Samples', 'RxUIProjectTemplate', and 'RxUI\_QCon'. The 'ReactiveUI' repository is selected, and its details are shown on the right. The details include a description: 'Use the Reactive Extensions for .NET along with Silverlight, WPF, or Windows Phone to create elegant, testable User Interfaces.' and a list of high-level assemblies: 'ReactiveUI - Core library that doesn't rely on any particular UI framework. ReactiveObject, the base ViewModel object, as well as ReactiveCommand, a Command, a UserError, Behaviors, and a framework'. A terminal window is open in the foreground, showing the output of the 'git clone' command and the 'git status' command.

local

repositories

github

niik

reactiveui

Filter Repositories

reactiveui/ReactiveUI

reactiveui/ReactiveUI.Samples

reactiveui/RxUIProjectTemplate

reactiveui/RxUI\_QCon

## ReactiveUI

Use the Reactive Extensions for .NET along with Silverlight, WPF, or Windows Phone to create elegant, testable User Interfaces.

This library is organized into several high-level assemblies:

- ReactiveUI** - Core library that doesn't rely on any particular UI framework. **ReactiveObject**, the base **ViewModel** object, as well as **ReactiveCommand**, a **Command**, a **UserError**, **Behaviors**, and a **framework**.

```
MINGW32-~/git
Welcome to Git (version 1.8.3-preview20130601)

Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.

Bacon@BACON ~
$ git clone https://github.com/msysgit/git.git
Cloning into 'git'...
remote: Counting objects: 177468, done.
remote: Compressing objects: 100% (52057/52057), done.
remote: Total 177468 (delta 133396), reused 166093 (delta 123576)
Receiving objects: 100% (177468/177468), 42.16 MiB | 1.84 MiB/s, done.
Resolving deltas: 100% (133396/133396), done.
Checking out files: 100% (2576/2576), done.

Bacon@BACON ~
$ cd git
$ cd git
Bacon@BACON ~/git (master)
$ git status
# On branch master
nothing to commit, working directory clean
Bacon@BACON ~/git (master)
$
```



Git dans Visual Studio, ça donne ?

---

# En fonction de votre version...

---

## - Visual Studio 2012:

- Installer l'Update 4 <http://www.microsoft.com/fr-fr/download/details.aspx?id=39305>
- Installer Visual Studio Tools for Git  
<http://visualstudiogallery.msdn.microsoft.com/abafc7d6-dcaa-40f4-8a5e-d6724bdb980c>

## - Visual Studio 2013

Intégré nativement

# Atelier !

- Equipes de **2** ou **3** personnes
- Créez chacun votre compte gratuit sur **GitHub**
- Récupérez le projet sur <https://github.com/EPSILab/AtelierGit>
- Créez un repository pour votre équipe
- Ouvrez le fichier **.sln** dans **Visual Studio** et Démarrez !

Le projet consiste à gérer une liste d'étudiants (nom, prénom, classe et date de naissance).

L'affichage des étudiants dans une liste est fait mais il manque les fonctionnalités d'ajout, de modification et de suppression de la base de données.

Voyez avec les membres de votre équipe pour vous répartir les tâches.

Créez 3 branches **feature/add-student**, **feature/edit-student** et **feature/delete-student** puis fusionnez-les. Respectez bien le GitFlow ! Bon courage et gare aux conflits !