

# Git : introduction

**Achref El Mouelhi**

Docteur de l'université d'Aix-Marseille  
Chercheur en programmation par contrainte (IA)  
Ingénieur en génie logiciel

`elmouelhi.achref@gmail.com`



# Plan

- 1 Introduction
- 2 Installer Git
- 3 Créer un repository Git (dépôt)
- 4 Configurer Git
- 5 Installer ungit

# Git

## Git

- Logiciel de gestion de version créé en Avril 2005 par **Linus Torvald** : fondateur du Kernel Linux
- Mai 2013 : 36% des professionnels utilisent **Git** comme logiciel de versioning
- Avril 2013 : **Github** déclare avoir 3.5 millions d'utilisateurs
- Mai 2019 : **Github** déclare avoir 37 millions d'utilisateurs et plus de 100 millions de dépôts (repositories)
- Mai 2018, **Github** est acheté par **Microsoft**

## Pendant la réalisation d'un projet informatique

- Il y aura plusieurs versions de notre logiciel
- On partage le code avec d'autres développeurs
- On le modifie souvent (c'est évident sinon le développement n'avance pas)
- Des fois, on dispose d'une version qui tourne correctement
- **Mais**, après une (petite) modification, plus rien ne fonctionne

# Git

## On a besoin soit de

- récupérer la dernière version qui fonctionnait
- savoir
  - qu'est ce qui a été modifié ?
  - qui a modifié ?
  - pourquoi ?
  - quand ?
- publier (partager) une version valide du projet
- organiser son projet par version
- ...

## Deux modèles de versioning (**VCS** : **V**ersion **C**ontrol **S**ystem)

- **Modèle centralisé** : le code du logiciel est géré par un serveur central  
Exemple : **SVN**, **CVS**
- **Modèle distribué** : tous les développeurs ont accès au code sans passer par un serveur  
Exemple : **Git**, **Mercurial**, **Bazaar**

# Git

## En quelques mots

- Syntaxe proche de **Shell Linux**
- Basée sur des **commit** (une version valide du code)
- Permettant de retrouver un fichier supprimé, une ancienne version modifiée...
- Acceptant toute extension de fichier **.html**, **.css**, **.js**, **.java**, **.php**, **.cshtml**, **.asp...**
- Utilisé par des sites web de partage
  - **Github** : <https://github.com/>
  - **BitBucket** : <https://bitbucket.org/>
  - **GitLab** : <https://about.gitlab.com/>
- Gestion de projet de taille importante

# Git

## Installation selon le système d'exploitation

- Sous **Linux** : via le gestionnaire de paquet en faisant `(sudo) apt-get install git`
- Sous **MAC** : via homebrew `(brew install git)`
- Sous **Windows** : le nouveau lien depuis l'achat par Windows  
<https://gitforwindows.org/>



# Git

## Installation selon le système d'exploitation

- Sous **Linux** : via le gestionnaire de paquet en faisant `(sudo) apt-get install git`
- Sous **MAC** : via homebrew `(brew install git)`
- Sous **Windows** : le nouveau lien depuis l'achat par Windows  
<https://gitforwindows.org/>

Pendant l'installation, choisissez un éditeur de texte que vous maîtrisiez (**Sublime text...**)

## Ajouter **Sublime text 3** aux variables d'environnement de **Windows**

- Dans la zone de recherche de **Windows 10**, chercher **Système**
- Aller dans **Paramètres système avancés** et choisir **Variables d'environnement**
- Dans **Variables système**, sélectionner **PATH** puis cliquer sur **Modifier**
- Cliquer sur **Nouveau**, ajouter le chemin vers le dossier d'installation de **Sublime Text 3** (C:\Program Files\Sublime Text 3)

# Git

## Ajouter **Sublime text 3** aux variables d'environnement de **Windows**

- Dans la zone de recherche de **Windows 10**, chercher **Système**
- Aller dans **Paramètres système avancés** et choisir **Variables d'environnement**
- Dans **Variables système**, sélectionner **PATH** puis cliquer sur **Modifier**
- Cliquer sur **Nouveau**, ajouter le chemin vers le dossier d'installation de **Sublime Text 3** (C:\Program Files\Sublime Text 3)

## Pour lancer sous **Windows**

Chercher **Git Bash**

Quelques commandes **Unix**

- `pwd` : imprimer le chemin d'accès
- `ls` : lister le contenu du répertoire courant
- `cd` : changer de répertoire
- `mkdir` : créer un répertoire
- `rm` : supprimer un répertoire ou un fichier
- `touch` : créer un fichier
- `echo` : écrire dans la console ou dans un fichier
- `head` : afficher le contenu d'un fichier dans la console
- ...

# Git

## Un dépôt (repository)

Un répertoire de travail

- géré par Git
- contenant les éléments à partager ou à 'commiter'

# Git

## Comment créer un dépôt ?

- créer un répertoire
- se positionner dedans
- déclarer ce répertoire comme un dépôt

© Achref EL M...

# Git

## Comment créer un dépôt ?

- créer un répertoire
- se positionner dedans
- déclarer ce répertoire comme un dépôt

## Techniquement

- `mkdir firstGit`
- `cd firstGit`
- `git init`

# Git

**Pour désactiver la coloration dans la console (par défaut activée)**

```
git git config --global color.ui false
```

© Achref EL MOUELHI ©



# Git

**Pour désactiver la coloration dans la console (par défaut activée)**

```
git config --global color.ui false
```

**Pour désactiver la coloration dans la console (par défaut activée)**

- `git config --global color.diff auto`
- `git config --global color.status auto`
- `git config --global color.branch auto`
- `git config --global color.interactive auto`

# Git

## Configuration du compte **Git**

- `git config --global user.name "monNom"`
- `git config --global user.email "mon@email"`

© Achref EL MOUETRI

# Git

## Configuration du compte **Git**

- `git config --global user.name "monNom"`
- `git config --global user.email "mon@email"`

**Pour consulter la liste de configurations (et vérifier les modifications)**

```
git config --list
```

# Git

## Configuration du compte **Git**

- `git config --global user.name "monNom"`
- `git config --global user.email "mon@email"`

## Pour consulter la liste de configurations (et vérifier les modifications)

```
git config --list
```

## Pour vérifier la valeur d'une propriété de configuration

```
git config user.name
```

# Git

## Ungit

- Application qui permet d'utiliser **git** en mode graphique
- Disponible sous **Node.js** (il faut utiliser le gestionnaire de paquet npm)

© Achref EL MOU

# Git

## Ungit

- Application qui permet d'utiliser **git** en mode graphique
- Disponible sous **Node.js** (il faut utiliser le gestionnaire de paquet npm)

## Installation et mise en place

- Exécuter la commande `npm install -g ungit` (l'option `g` pour globale, c'est-à-dire pour tous les projets)
- Se positionner dans votre (futur) repository et exécuter `ungit`