

# R: Contrôle de Source

Eric Marcon

25 December 2020

# Principes

git est un gestionnaire de sources :

- Suivi des modifications : bien plus qu'une sauvegarde !
- Possibilité de revenir en arrière ;
- Plusieurs versions concurrentes : branches.

GitHub est une plateforme pour la collaboration basée sur Git.

- Possibilité de développer à plusieurs : fin des pièces jointes !
- Présentation des résultats : pages GitHub ;
- Intégration continue: Travis.

# Mise en place

## RStudio doit détecter Git

Options

**General**

☒ Enable version control interface for RStudio projects

Git executable:  
C:/Program Files/Git/bin/git.exe Browse...

SVN executable:  
(Not Found) Browse...

SSH RSA Key:  
(None) Create RSA Key...

[Using Version Control with RStudio](#)

**Git/SVN**

**Publishing**

**Terminal**

OK Cancel Apply

Sinon, l'installer.

## Ouvrir un compte sur GitHub

The screenshot shows the GitHub profile of Eric Marcon. The browser address bar displays the URL <https://github.com/EricMarcon>. The profile header includes a search bar, navigation links (Pull requests, Issues, Marketplace, Explore), and a user profile section with a photo, name (Eric Marcon), and bio (EricMarcon). Below the profile, there are sections for 'Popular repositories' and 'Organizations'. The 'Popular repositories' section lists four repositories: **entropart** (Entropy Partitioning to Measure Diversity), **dbens** (Distance Based Measures of Spatial Structures), **SpatDlv** (Spatially Explicit Measures of Diversity), and **ericmarcon.github.io** (Home page). The 'Organizations' section shows the **AgroParisTech** organization. At the bottom, there is a 'Contributions' section showing 429 contributions in the last year, represented by a calendar grid with green squares indicating contribution days.

# Créer un projet

Contrôle des  
Sources

Eric Marcon

Principes

Mise en place

Pratique

Branches

Fork et Pull  
Request

A partir de rien :

- Dans GitHub :
  - *New Repository*
  - Choisir le nom (pas de caractères spéciaux)
  - Ajouter un *.gitignore* approprié.
- Copier l'URL à partir de *Clone or Download*
- Dans RStudio : nouveau projet à partir de Git, coller l'URL.



# Créer un projet

Contrôle de  
Sources

Eric Marcon

Principes

Mise en place

Pratique

Branches

Fork et Pull  
Request

A partir d'un projet RStudio existant :

- Passer le projet sous contrôle de version :
  - *Tools /Version Control /Project Setup...*
  - Sélectionner *Git*.
- Créer un dépôt sur GitHub, récupérer son URL :  
<https://github.com/MonCompte/MonDepot.git>
- Dans le Terminal de RStudio, exécuter :

```
git remote add origin https://github.com/MonCompte/MonDepot.git  
git push -u origin master
```

# Opérations élémentaires : filtrage

Contrôle de  
Sources

Eric Marcon

Principes

Mise en place

Pratique

Branches

Fork et Pull  
Request

Les fichiers modifiés sont dans la fenêtre Git de RStudio.

Utiliser `.gitignore` pour masquer les fichiers non suivis.

- Créer un fichier R avec une ligne de code et l'enregistrer.

# Opérations élémentaires : commit

Control des  
Sources

Eric Marcon

Principes

Mise en place

Pratique

Branches

Fork et Pull  
Request

Après chaque séance de travail, livrer le résultat (*Commit*)

Sélectionner les fichiers à livrer.

Saisir un message clair : résumé sur la première ligne.

# Opérations élémentaires : synchroniser

Control of  
Sources

Eric Marcon

Principes

Mise en place

Pratique

Branches

Fork et Pull  
Request

Pousser (*Push*) ses modifications vers GitHub pour les rendre publiques.

Tirer (*Pull*) pour récupérer les modifications sur GitHub.

# Opérations élémentaires : collaborer

Contrôle de  
Sources

Eric Marcon

Principes

Mise en place

Pratique

Branches

Fork et Pull  
Request

Déclarer un collaborateur.

Travailler à deux ou plus sur le même fichier.

Contenu d'une séance de travail :

- Tirer ;
- Modifier ;
- Livrer ;
- Pousser.

# Conflits

Contraintes  
Sources

**Eric Marcon**

Principes

**Mise en place**

Pratique

Branches

Fork et Pull  
Request

L'information élémentaire est la ligne.

Modifications contradictoires = conflit.

Minimiser les conflits : dans un texte, une phrase = une ligne.

En cas de conflit, trancher.

# Suivi du projet

Eric Marcon

Principes

Mise en place


Pratique

Branches

Fork et Pull  
Request

## Icône en forme d'horloge dans la fenêtre Git

RStudio: Review Changes

Changes History master (all commits) 

Langage	Eric Marcon <eric.marc
Préparation	Eric Marcon <eric.marc
Merge branch 'master' of https://github.com/EricMarcon/C	Eric Marcon <eric.marc
<b>Typo</b>	<b>Eric Marcon &lt;eric.marc</b>
Set theme jekyll-theme-slate	Eric Marcon <eric.marc
Présentation Langage	Eric Marcon <eric.marc
Plan	Eric Marcon <eric.marc

Contrôle des  
Sources

**Eric Marcon**

Principes

Mise en place

**Pratique**

Branches

Fork et Pull  
Request

**Pratique**



# Modifier le projet Cours-R-Introduction

Contrôle de  
Sources

**Eric Marcon**

Principes

Mise en place

**Pratique**

Branches

Fork et Pull  
Request

Ajouter un fichier README.md

Prendre en compte, valider et pousser.

# Générer un conflit

Contrôle de  
Sources

**Eric Marcon**

Principes

Mise en place

**Pratique**

Branches

Fork et Pull  
Request

Editer la même ligne de README.md :

- en ligne sur GitHub ;
- localement.

Tirer, constater le conflit, le résoudre.

# Collaborer avec son voisin

Contrôle de  
Sources

**Eric Marcon**

Principes

Mise en place

**Pratique**

Branches

Fork et Pull  
Request

Le voisin de gauche invite celui de droite sur GitHub.

Les deux modifient le projet.

- Bien penser à tirer avant de modifier.
- Pousser rapidement pour limiter les conflits.

Contrôle de  
Sources

**Eric Marcon**

Principes

Mise en place

Pratique

**Branches**

Fork et Pull  
Request

Branches

# Intérêt

Contrôle de  
Sources

Eric Marcon

Principes

Mise en place

Pratique

Branches

Fork et Pull  
Request

Modifier le projet sans perturber son état stable.

Application :

- développer une nouvelle fonctionnalité ;
- la tester, corriger les bugs ;
- la rendre visible quand elle est terminée.

# Créer une branche

Contrôle de  
Sources

**Eric Marcon**

Principes

Mise en place

Pratique

**Branches**

Fork et Pull  
Request

Cliquer sur New Branch

Travailler dans la nouvelle branche (pull, commit, push).

# Fusionner

Contrôle de  
Sources

**Eric Marcon**

Principes

Mise en place

Pratique

**Branches**

Fork et Pull  
Request

Se placer dans la branche master.

Exécuter:

```
git merge branche_a_fusionner
```

# Fork et Pull Request



# Fork

Control of  
Sources

**Eric Marcon**

Principes

Mise en place

Pratique

Branches

**Fork et Pull  
Request**

Objectif : modifier le dépôt d'un autre

Fourcher: création d'une copie du dépôt

Commencer une nouvelle branche, la modifier.

# Pull Request

Contrôle de  
Sources

Eric Marcon

Principes

Mise en place

Pratique

Branches

Fork et Pull  
Request

Demande d'intégration de la branche: *Pull Request*.

Sur GitHub.

Dialogue possible.

Si acceptation, fusion de la branche et suppression.