

R: Contrôle de Source

Eric Marcon

03 novembre 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

☒ Enable version control interface for RStudio projects

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

SVN executable:
(Not Found)

SSH RSA Key:
(None)

[Using Version Control with RStudio](#)

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 'https://github.com/EricMarcon'. The profile header includes a profile picture of a man with grey hair and a beard, the name 'Eric Marcon', and the username 'EricMarcon'. Below the name is a link to 'Add a bio'. The 'AgroParisTech' organization is listed with a location in Kourou, French Guiana, an email address, and a website link. The 'Organizations' section shows the 'AgroParisTech' logo. The 'Popular repositories' section lists five repositories: 'entropart', 'dbrms', 'SpatDiv', 'ericmarcon.github.io', and 'VMS'. The 'Contributions' section shows a grid of 429 contributions in the last year, with a 'Contribution settings' dropdown menu.

Eric Marcon
 EricMarcon
[Add a bio](#)

AgroParisTech
 Kourou, French Guiana
 eric.marcon@ecofog.gf
<http://www.ecofog.gf/spp.php?>

Organizations
 AgroParisTech

Popular repositories

- entropart**
 Entropy Partitioning to Measure Diversity
 1 star
- dbrms**
 Distance Based Measures of Spatial Structures
 1 star
- SpatDiv**
 Spatially Explicit Measures of Diversity
 1 star
- ericmarcon.github.io**
 Home page
- VMS**
 WMI wrapper for Visual Studio apps
 1 star

429 contributions in the last year

Contribution settings

Learn how we count contributions.

Créer un projet

Contrôle de
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

Contrôle de
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

Control 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

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
	Typo	Eric Marcon <eric.marc
	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

Control 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 des
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 de
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.