

Introduction à GIT



Everytime I do a git pull
& there is a conflict





Capgemini



Sheikoop



Président actuel



Président 2022-2023

MIAGE
RÉSEAU
DES MIAGE
DE FRANCE
Grenoble

Responsable P&C M1 MIAGE

MIAGE
RÉSEAU
DES MIAGE
DE FRANCE
Grenoble

Master 2023



zenika

MIAGE
RÉSEAU
DES MIAGE
DE FRANCE
Grenoble

UGA
IM²AG
UFR
Université
Grenoble Alpes

Ingénieur Logiciel



Nicolas Marinho



ATRALTECH

On s'y met ?





git

Son histoire



En avait marre de CVS



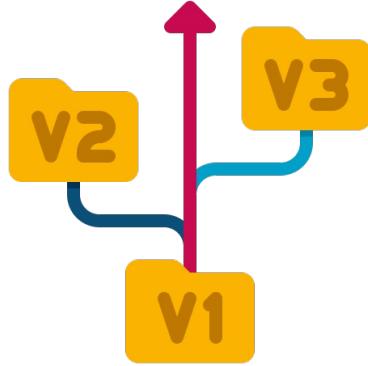
Créé de manière libre



Linus Torvalds



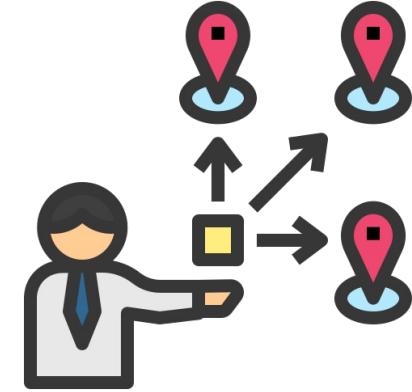
À quoi ça sert



Versionning



Travail collaboratif
en parallèle

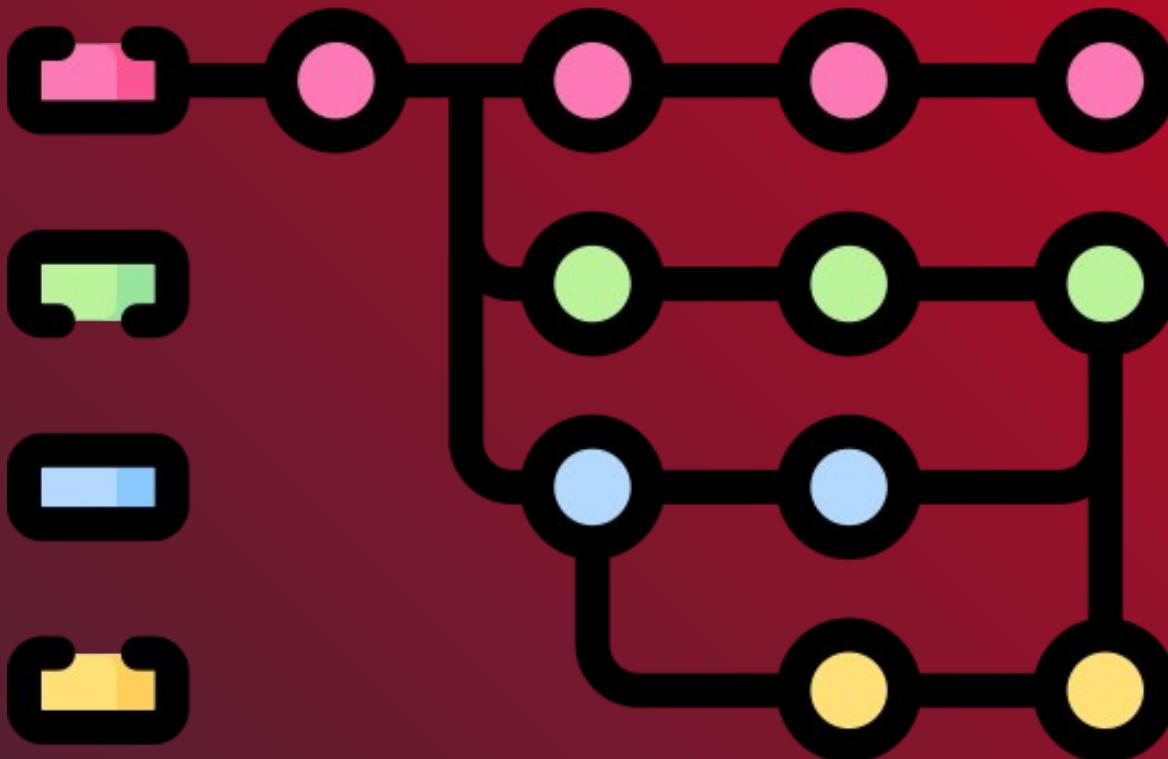


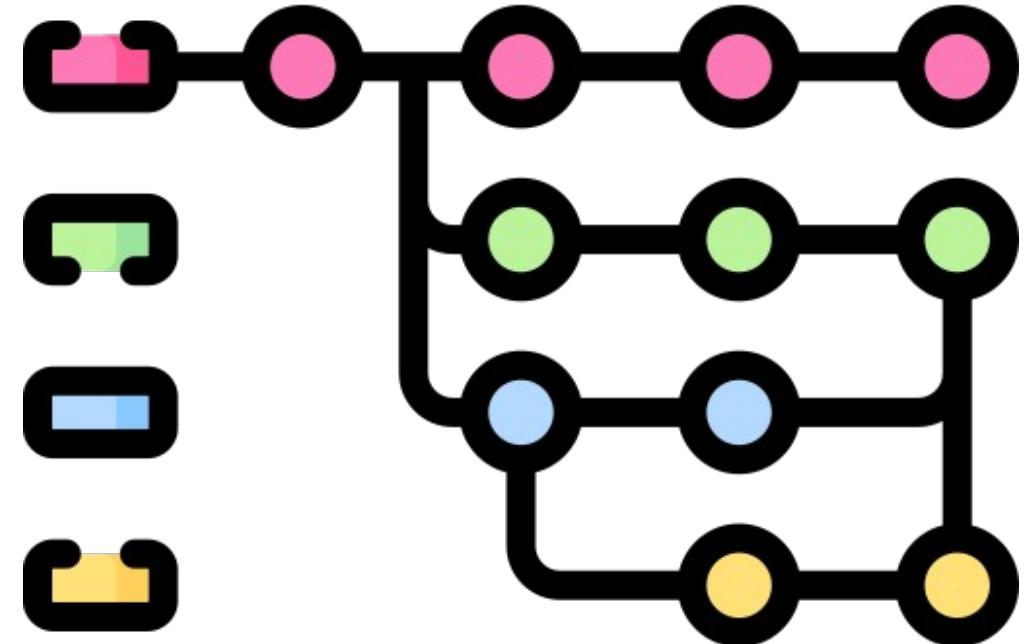
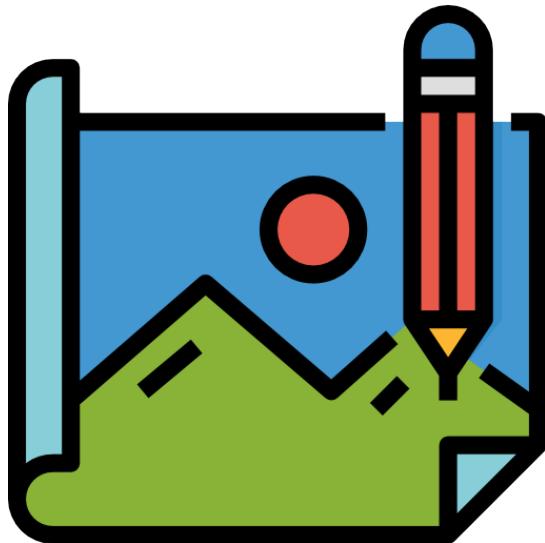
Distribuer son
code



Une grande histoire

Le branching



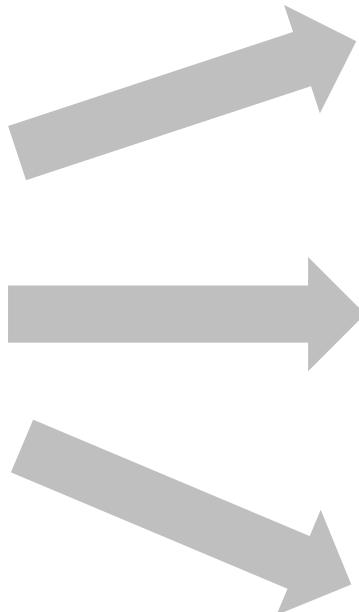


Une représentation
graphique du versionning

Un repository



Un ensemble de fichiers



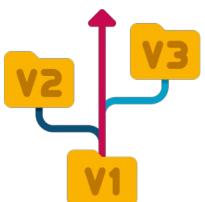
Suivis



git

clone <https://git/mon-repo.git>

Distribués



Versionnés

Un commit



Un commit



Ensemble de lignes modifications



Réunies sous un identifiant unique

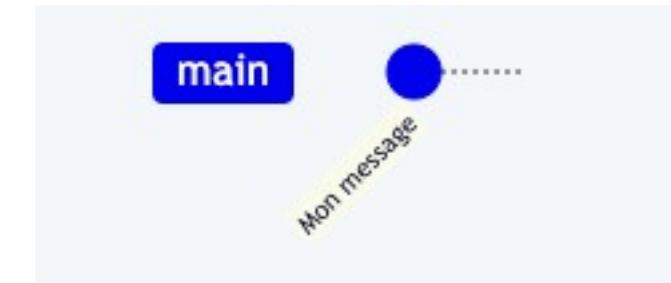


Permet donc de versionner des changements

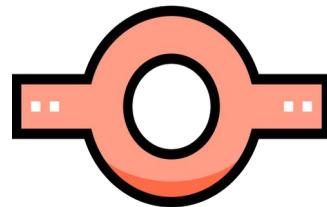
```
git commit -m "mon message"
```



Git graph vision



Le status

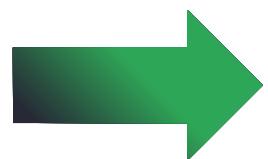


Permet de voir
l'état

Repository local

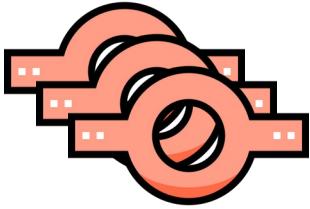
Ensemble de
modifications

git
status

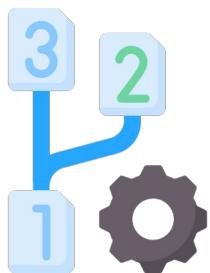


```
→ airport-si git:(main) ✘ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   pom.xml
    modified:   src/main/kotlin/fr/uga/miage/m1/data/entities/UserEntity.kt
```



Ensemble de commits



Dans un ordre précis

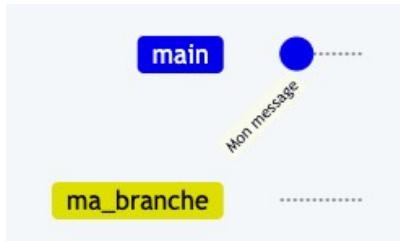
Permet



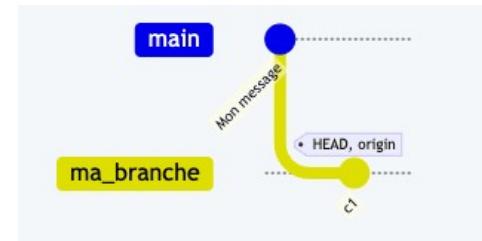
Travail à part du code fonctionnel

```
git branch ma_branche
```

```
git checkout ma_branche
```



Git graph vision

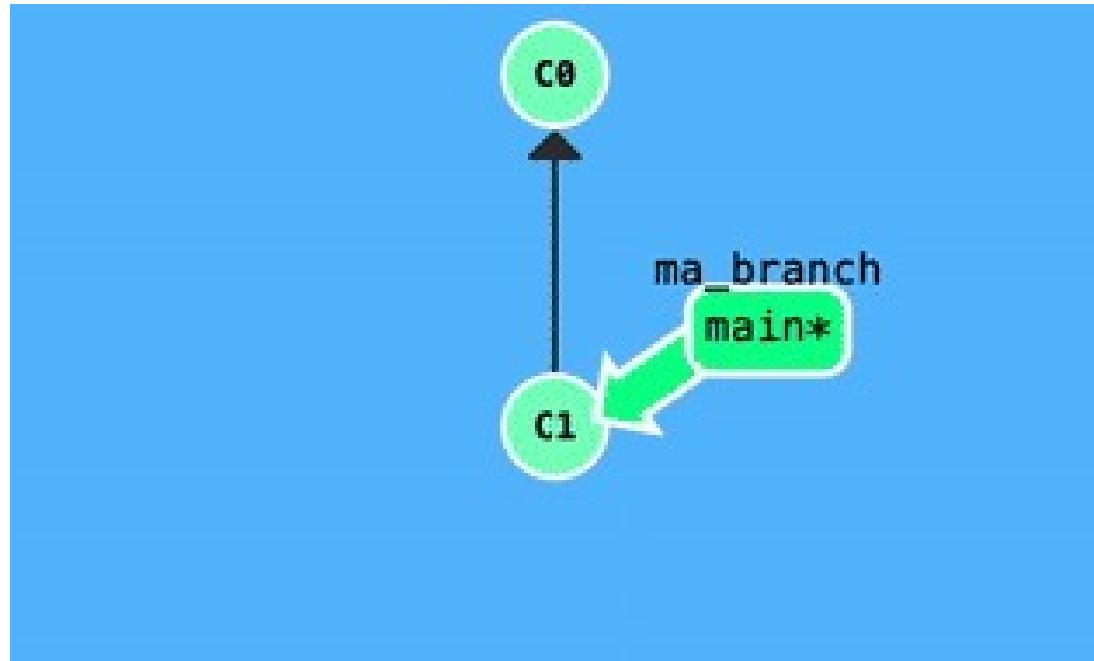


Les Branches

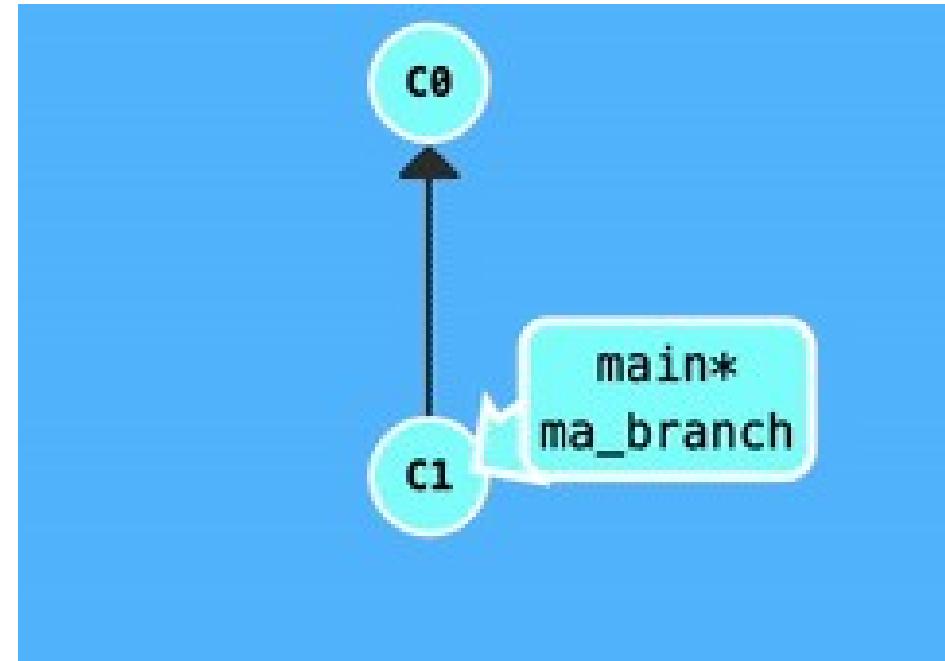


Git graph vision

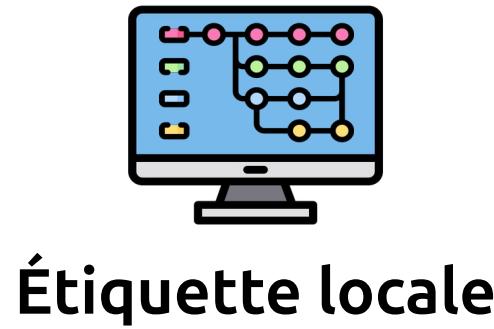
`git branch ma_branch`



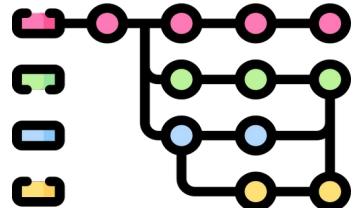
`git checkout ma_branch`



Les étiquettes



Étiquette locale



Une copie locale



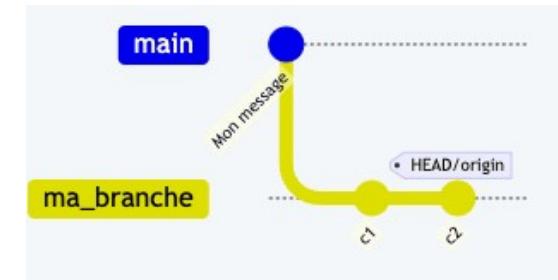
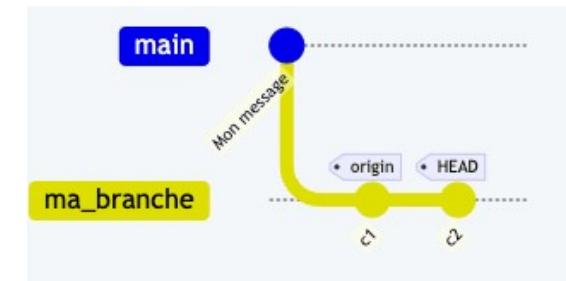
Étiquette en ligne
(remote)



Toutes les modifications
poussées



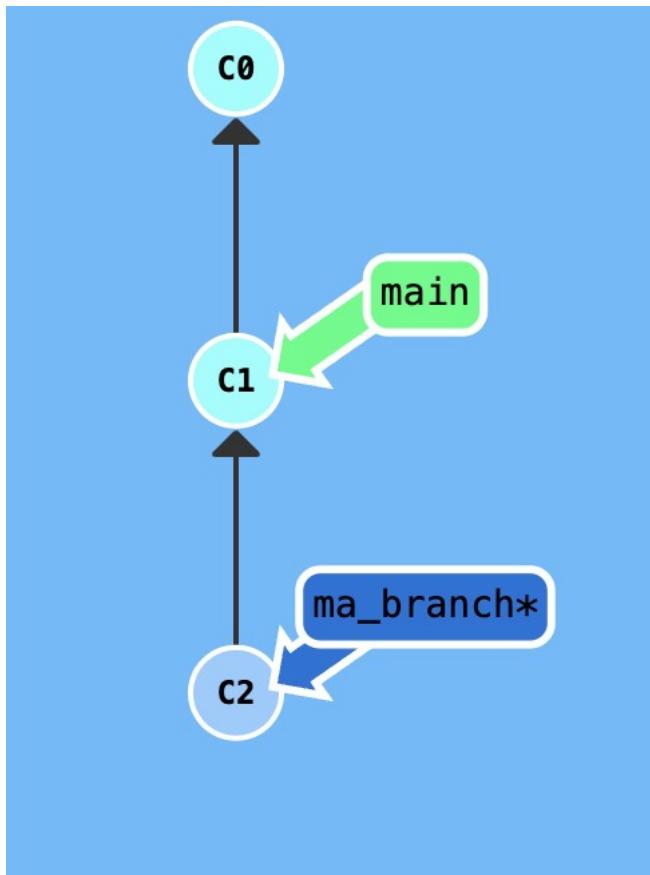
Git graph vision



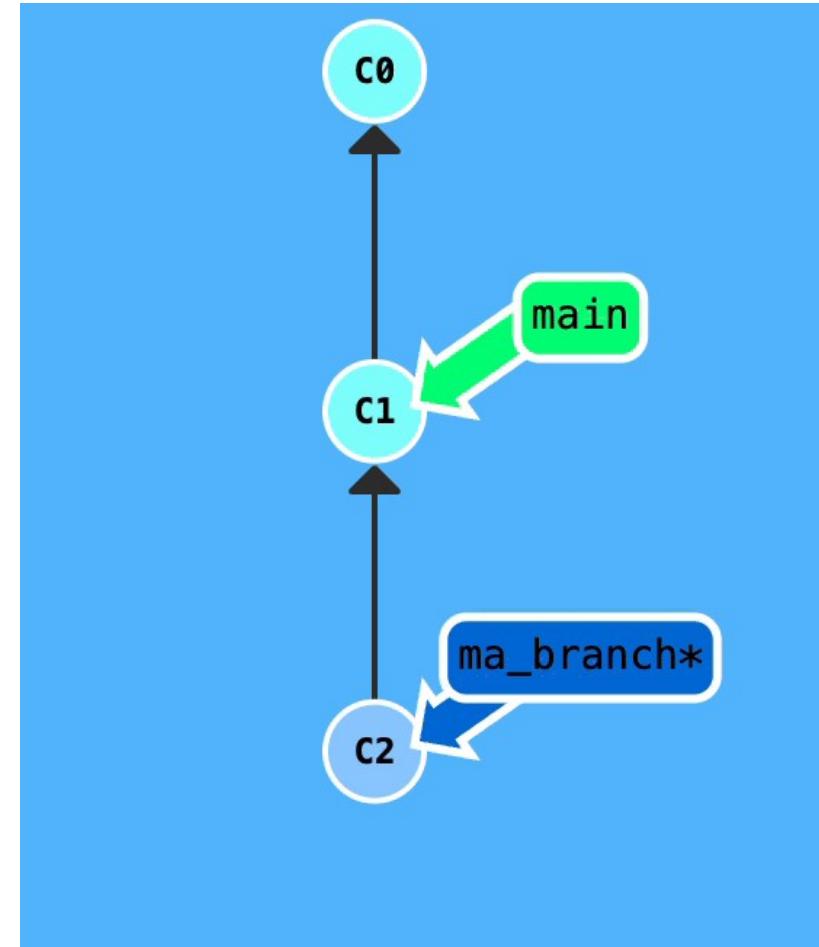
Les étiquettes



Git graph vision



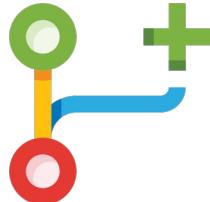
`git checkout C0`



Les merges



Les merges



Une branche
source

Injecter les modifications dans une autre

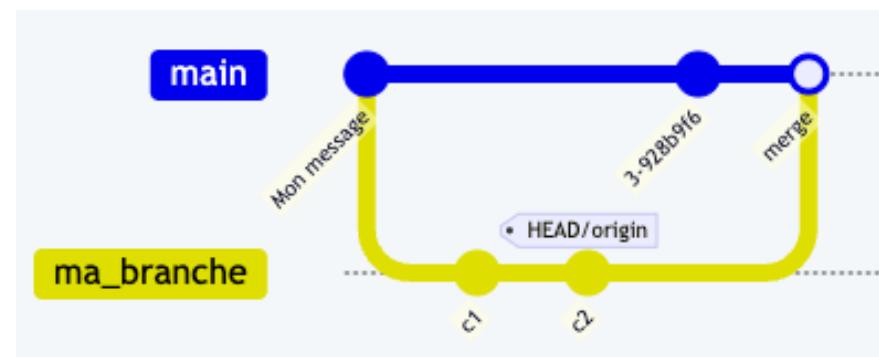


Une branche
destination

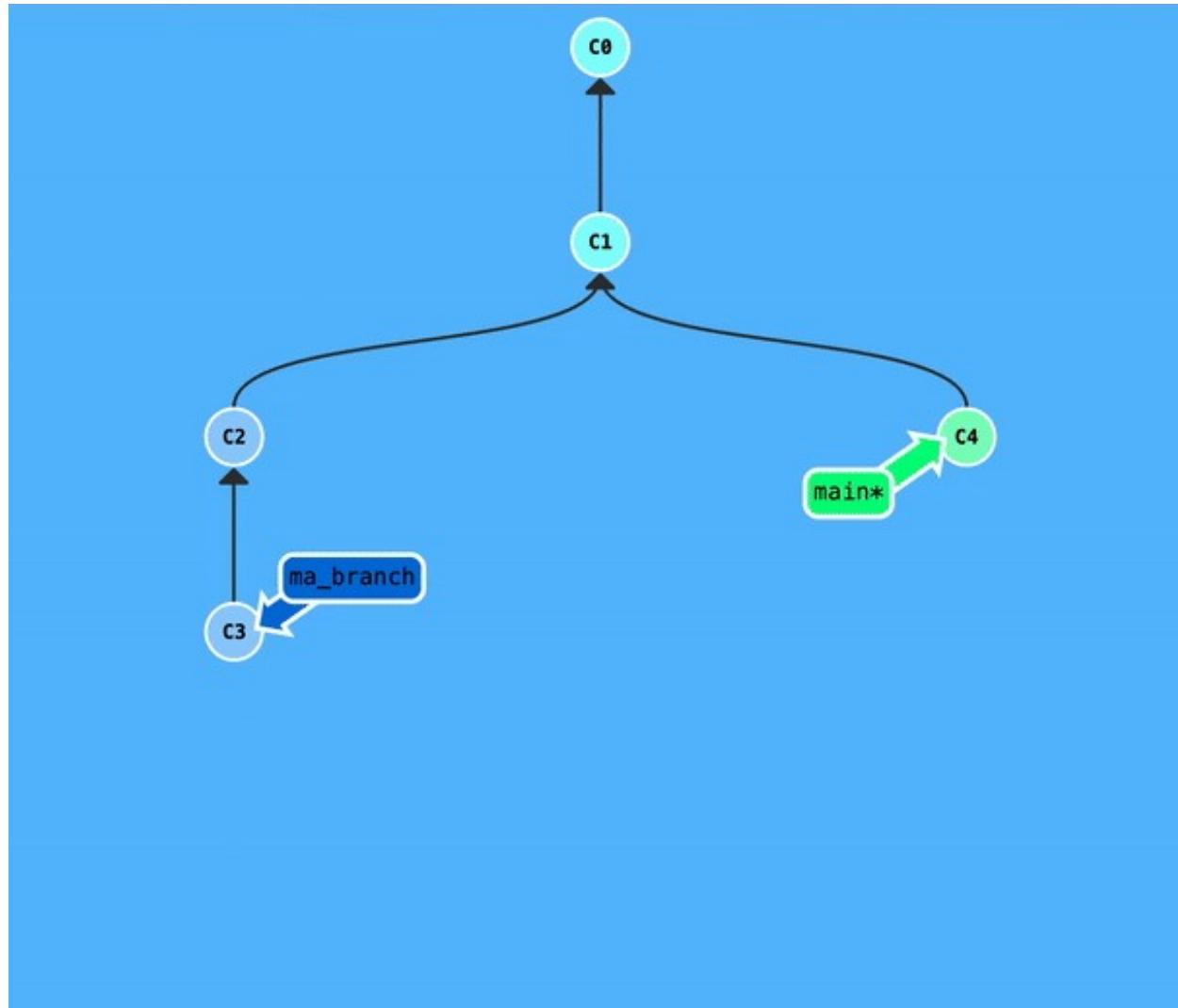
git merge



Git graph vision



Les merges



Les rebases



Les rebases



Ramener tous les commits sur la tête



Une branche à
rebase

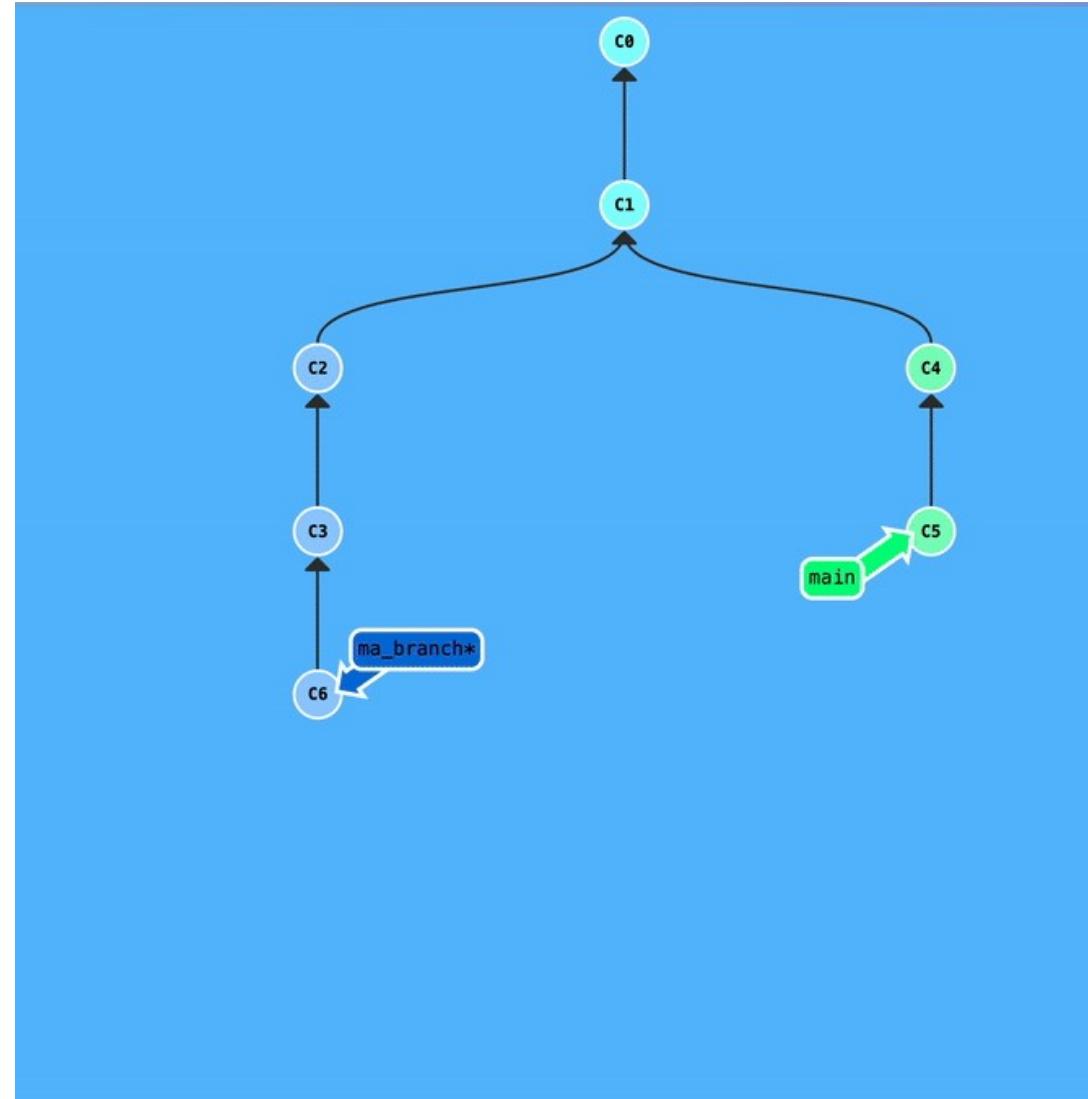
Une branche
De base

git rebase
main

Les rebases



Git graph vision



Les conflits



Le plus chiant



TRELOU LAAAAAAA

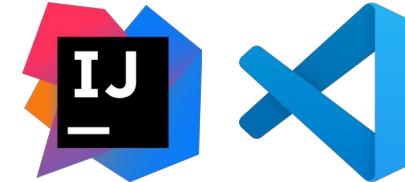
Les conflits

2 méthodes



Par les fichiers

```
<<<<<< HEAD
  int[] liste = {1, 2, 3, 4, 7, 9, 0, 9};
=====
  int[] liste = {1, 2, 3};
>>>>> origin/merge/merge-1
```



Par les IDE



Des outils pour

Les conflits



Par les IDE

Merge Revisions for /Users/nicolas.marinho/fac/M1-MIAGE/temp/merge-git/src/main/java/fr/uga/miage/l3/merge/git/Task.java

3 changes, 1 conflict.

Changes from main

```

package fr.uga.miage.l3.merge.git;

public class Task {

    public void execute() {
        int[] liste = {1, 2, 3};
        int somme = 0;
        for (int nombre : liste) {
            somme += nombre;
        }
        System.out.println("Le nombre de nombres est : " + liste.length);
        System.out.println("La somme des nombres est : " + somme);
    }
}

```

Changes from origin/merge/merge-2

```

package fr.uga.miage.l3.merge.git;

import java.util.List;

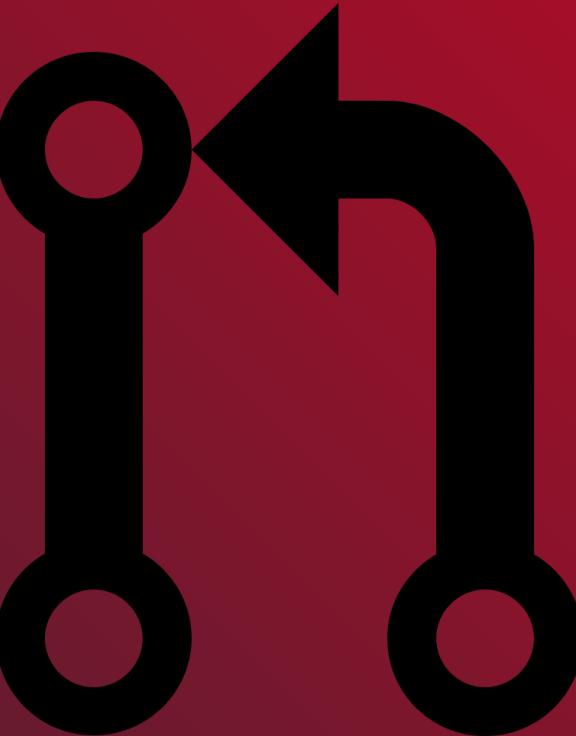
public class Task {

    public void execute() {
        List<Integer> liste = List.of(1, 2, 3, 4, 5);
        int somme = 0;
        for (int nombre : liste) {
            somme += nombre;
        }
        System.out.println("La somme des nombres est : " + somme);
    }
}

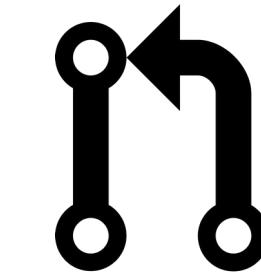
```

Accept Left Accept Right Cancel Apply

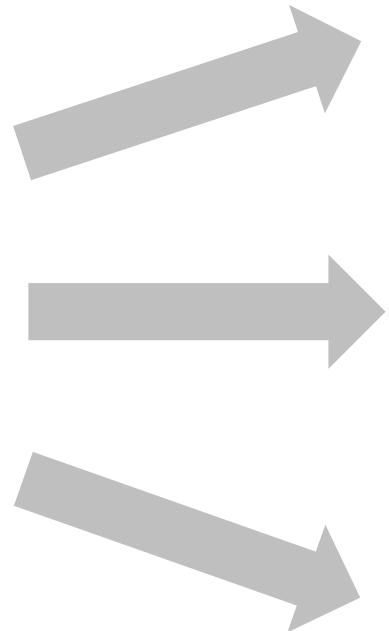
Une pull request



Une pull request



**Une pull
request**

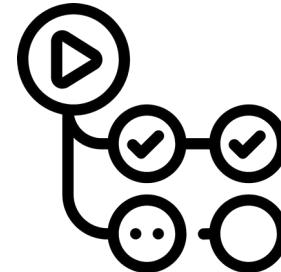


Rebase or merge

A une branche source et destination



La validation par les pères



La validation par les pères



A screenshot of a GitHub pull request page. The title is "Feedback #1". The top navigation bar includes "Code", "Issues", "Pull requests" (2), "Actions", "Projects", "Security", "Insights", and "Settings". A warning message at the top states: "⚠ GitHub users are [now required](#) to enable two-factor authentication as an additional security measure. Your activity on GitHub includes you in this requirement. You will need to enable two-factor authentication on your account before January 01, 2026, or be restricted from account actions." with a "Enable 2FA" button.

The main content shows a summary of the merge: "github-classroom wants to merge 64 commits into [feedback](#) from [main](#)". Below this are tabs for "Conversation" (0), "Commits" (64), "Checks" (3), and "Files changed" (151). The "Files changed" tab is selected, showing a list of files. The list includes ".DS_Store" (Binary, 46 KB) which is noted as "Binary file not shown.", and ".github/workflows/pr-dev-back.yml" (44 lines). The content of ".github/workflows/pr-dev-back.yml" is displayed:

```
@@ -0,0 +1,44 @@
1 + # This workflow will build a Java project with Maven, and cache/restore any dependencies to improve the
2 + # workflow execution time
3 +
4 + # This workflow uses actions that are not certified by GitHub.
5 + # They are provided by a third-party and are governed by
6 + # separate terms of service, privacy policy, and support
7 + # documentation.
8 +
9 + name: PR dans dev Backend
10 +
11 + on:
12 +   workflow_dispatch:
13 +
14 +   pull_request:
15 +     branches: [dev]
16 +     paths:
```

Ajout d'un fichier au commit

git add mon_fichier

Pousser les commits sur la remote

git push

Merge une branche

git merge branche-à-merge

Créer un commit avec un message

git commit -m « mon message »

Récupérer les commit distant

git pull

Changer de branche

git checkout branche_destination

Voir le status

git status

Créer une branche

git branch nom_branche

Rebase une branche

git rebase branche-de-base

Conclusion

