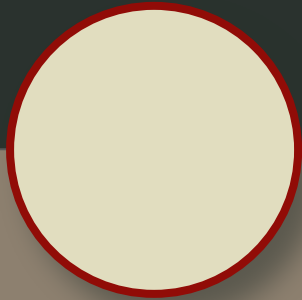


Intermediate Github/Gitlab



Javier Moldón

IAA-CSIC (Granada)

SKA Regional Centre Training Event Series
Hands-on Containerization
February 07, 2022

Objectives

What we will do

Show 3 practical cases on how to contribute to a repo

We will use a fork of [SDC1-solution](#)

Interaction with collaborators through issues

Creating and managing git branch

2 branch merge examples: with and without conflicts



A/myrepo



A



A/myrepo



Fork



B/myrepo



A



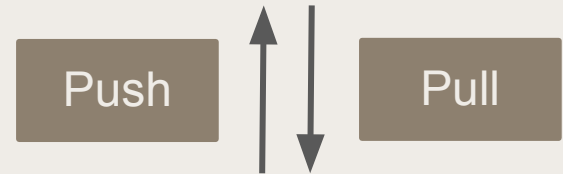
B



A/myrepo



B/myrepo



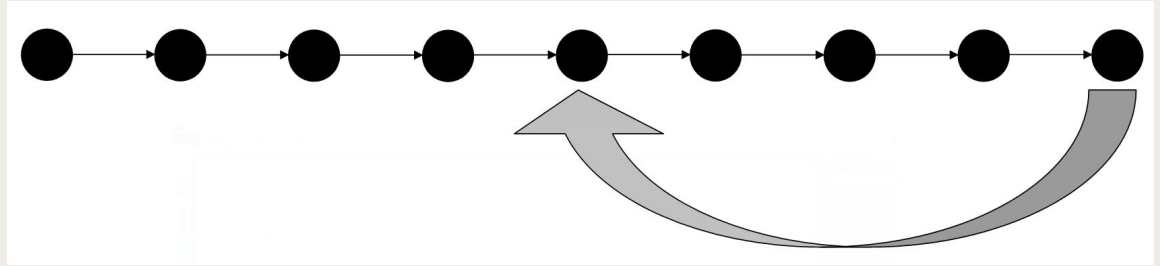
A



B

Linear progression of commits

It is easy to navigate back in time to any point

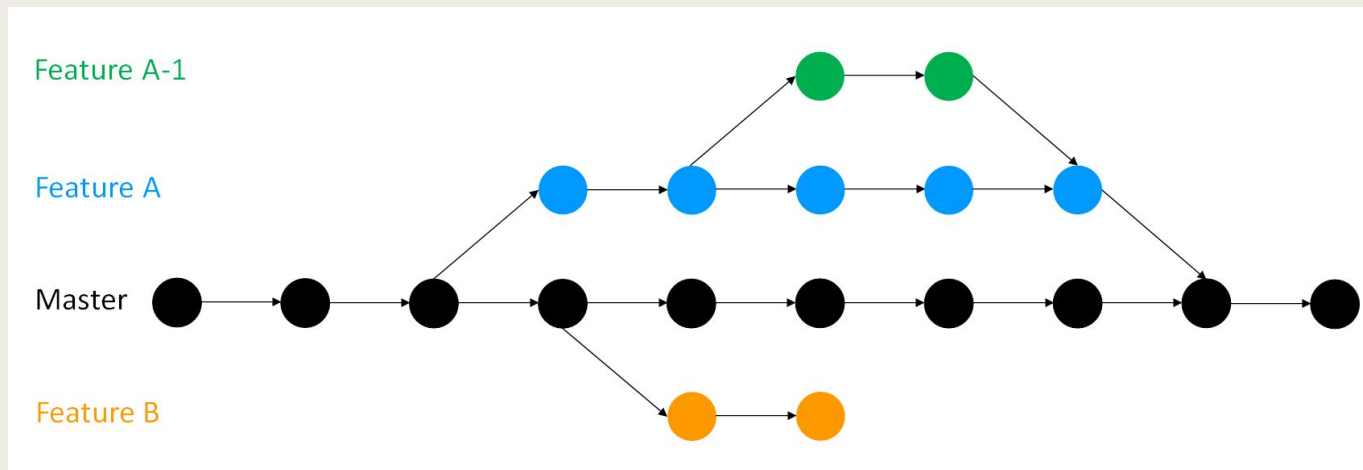


Git branches

Easy to collaborate

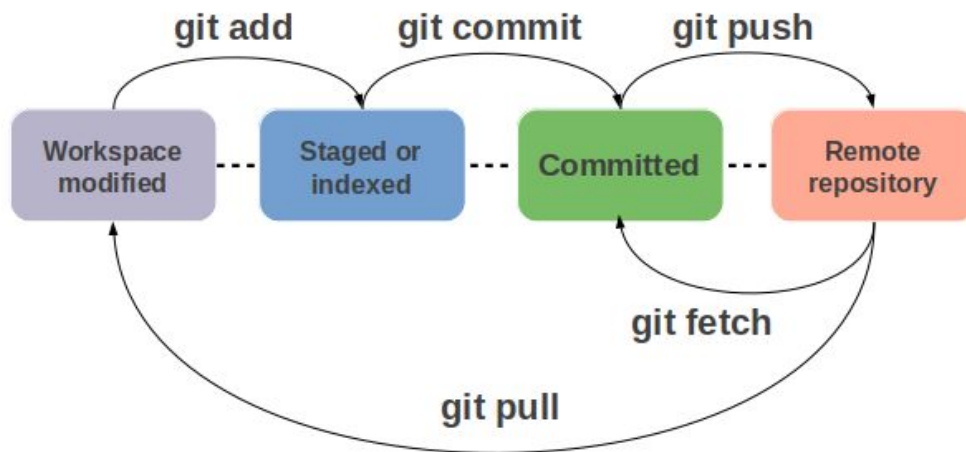
Check and verify
new content

Requires
pull/merge requests

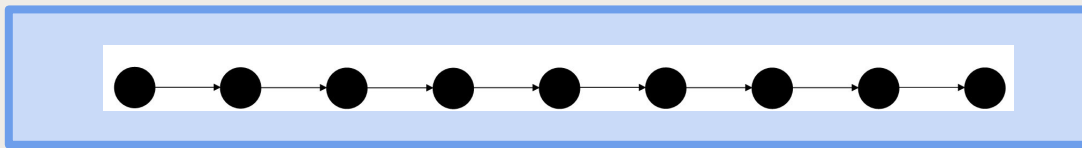


Basic git cycle

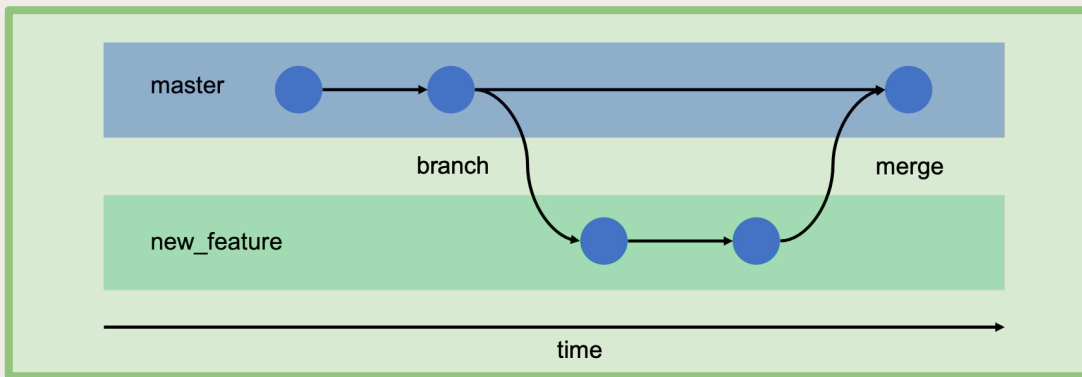
- It's the git life-cycle



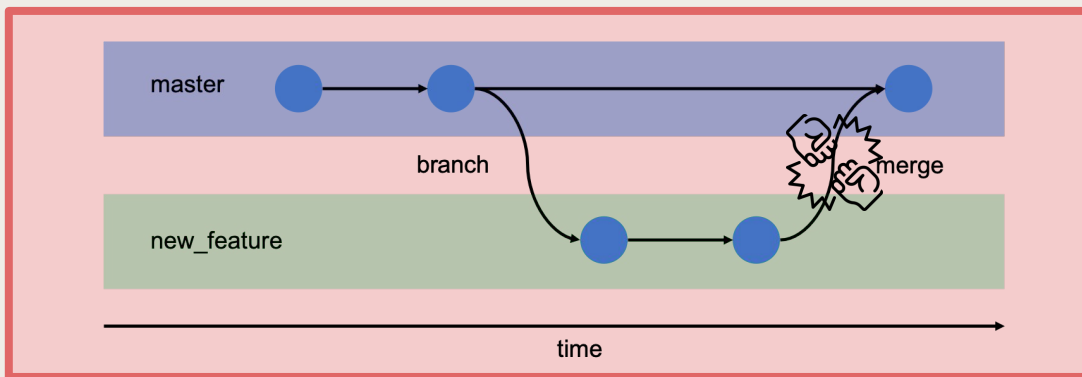
1



2



3



Resources

Some useful links

- [git - the simple guide](#)
- The Turing Way chapter on [version control](#). Very detailed book, check it out!
- [An Intro to Git and GitHub for Beginners](#)
- [Learn Git](#). This is a very nice resource.
- [NBIS - Introduction to Git](#)
- [Github Hello World](#) example
- The [Git Book](#). A very detailed guide of everything
- [Try Git](#). Very nice and funny interactive "Try me" resource.
- [Git for beginners: The definitive practical guide](#). Good quick reference for commands.
- [Learn git branching](#). Interactive practical examples
- [Google](#) (you will need it) and [stackoverflow](#).