

# Git basis

Koen Verbruggen  
WP Antwerp 18 jan 2016

# Whoami ?

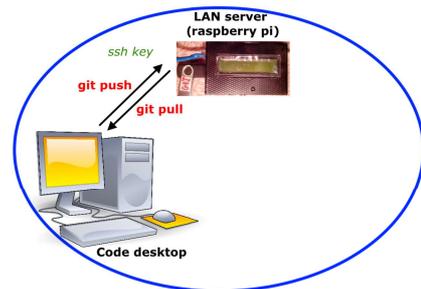
- \* NGO - social profit
- \* Website / ICT / training / evenement



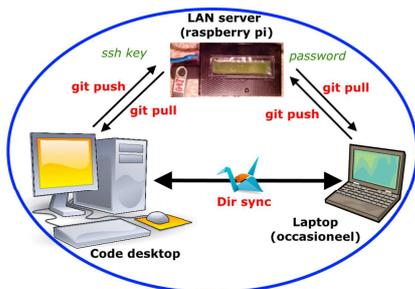
# Git basis – TOC

- 1.Introductie
- 2.De 3 meest gebruikte commando's
- 3.Vanaf het begin
- 4.Branching / merging
- 5.Veranderingen bekijken

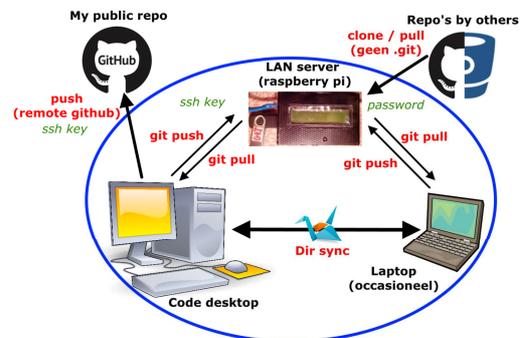
# Introductie – Mijn setup 1



# Introductie – Mijn setup 2

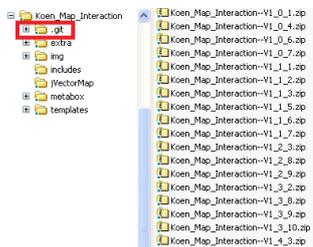


# Introductie – Mijn setup 3



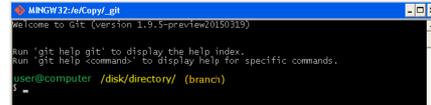
## Introductie – versie beheer

### Git ↔ manueel



## Introductie – Git bash

### Gebruik van de commandline tool (sh.exe)



- Enige optie met **ALLE** git commando's
- Faciliteert effectief begrip van commando's
- Elke computer met git heeft git bash

## 3 meest gebruikte commando's

- 1.git add .
- 2.git commit -am "documentatie"
- 3.git push origin master

1bis : git add --all  
> bij gewiste / verplaatste bestanden

## Vanaf het begin – Server ?

(commerciële) service



online privat  
(e.g. VPS)



lokaal netwerk (e.g. raspberry pi)

## Vanaf het begin – config (1x)

- git config --global user.name "vn an"
- git config --global user.email "email"
  
- git config --list

## Vanaf het begin – commando's

- 1.Server (ssh)  
git init --bare test.git
- 2.Lokaal (git bash)  
git clone ssh://user@host:/home/user/test.git

geen server :

- i.mkdir test
- ii.cd test
- iii.git init

## 3 meest gebruikte commando's

- 1.git add . OF git add --all
- 2.git commit -am "documentatie"
- 3.git push origin master

herhaal ad infinitum



## Vanaf het begin – remote

- Automatisch (host1 = "origin" remote) :  
git clone ssh://user1@host1:/home/user1/test.git  
> git push **origin** master
- Manueel (host2 = "remote-name" remote) :  
1.cd repo-name  
2. git remote add remote-name ssh://user2@host2:/path/repo-name.git  
> git push **remote-name** master



## Git branching / merging

- Kracht + complexiteit van git
- Branching strategie nodig  
> Vele werkwijzes
- Merging strategie nodig  
> Vele werkwijzes  
> **Verschillende commando sequenties**



## Git branching

- Overzicht : git branch
- Wisselen : git checkout branch-name
- Nieuwe : git checkout -b branch-name  
– Maakt nieuwe branch  
– Gaat naar nieuwe branch
- Delete : git branch -d branch-name  
> *branch must be fully merged (-D)*



## Git merging

- # make a branch the master
- 1.git checkout new-master
  - 2.git merge -s ours master
  - 3.git checkout master
  - 4.git merge new-master
  - 5.git commit -am "documentation"



## Veranderingen – geschiedenis

- # bekijk commits (exit "q")
- git log
  - git log -n 5
  - git log --after=YYYY-MM-DD
  - git log file-name.ext
  - git log --name-status



## Veranderingen – vergelijken

- # vergelijk bestanden (exit "q")
- # SHA hash : 40 hexadecimaal (6 eerste)
- git diff
- git diff SHA1 SHA2
- git diff file-name.ext
- git diff --name-status



## Veranderingen – teruggaan

- # laatste commit ongedaan maken
  - # werkt door nieuwe commit te maken
  - # vorige → **juist** → **fout** → **juist** → volgende
- 1.git revert HEAD
  - 2.git push origin master



## Veranderingen – undelete

- # herstel per ongeluk gewiste bestanden
  - # *toon overzicht van (D) bestanden*
- 1.git checkout
- # *specifiek bestand terughalen*
- 2.git checkout file-name.ext



## Vragen

