

# **Versiebeheer met Git**

VAKOVERSCHRIJDEND  
EINDPROJECT

ACADEMIEJAAR 2012–2013

DEEL 1

# Versiebeheer?

# Waarom versiebeheer?

- Doel: geschiedenis van code bijhouden
- Dropbox, SkyDrive ... ?
  - Historiek *per bestand* is onvoldoende
  - Teams verliezen snel overzicht
- Beter: op gecontroleerde momenten *snapshots* van werkend geheel

# Versiebeheersystemen

- Veel keuze: *CVS, Subversion, BitKeeper, Darcs, Bazaar, Mercurial, Git ...*
- Interactie doorgaans via command line
- Ook ondersteund door IDE's

DEEL 2

# Enkele concepten

# Graafgebaseerd versiebeheer



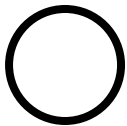
lege map



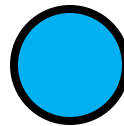
Dog.java



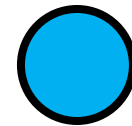
Animal.java  
Dog extends Animal



aanmaak **repository**



eerste **commit**



tweede commit



# Graafgebaseerd versiebeheer



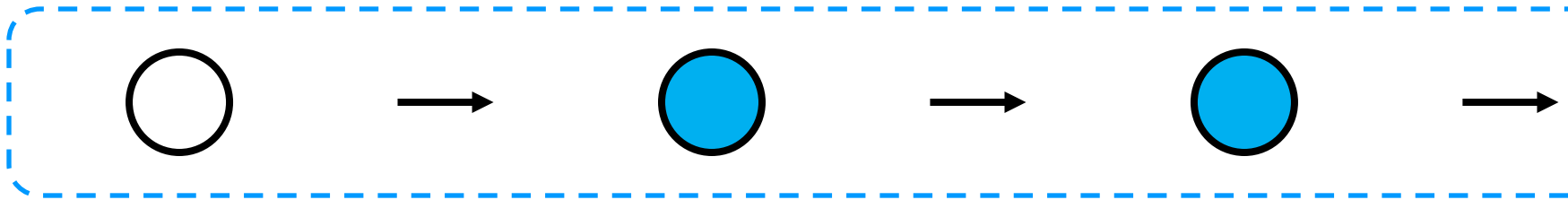
lege map



Dog.java

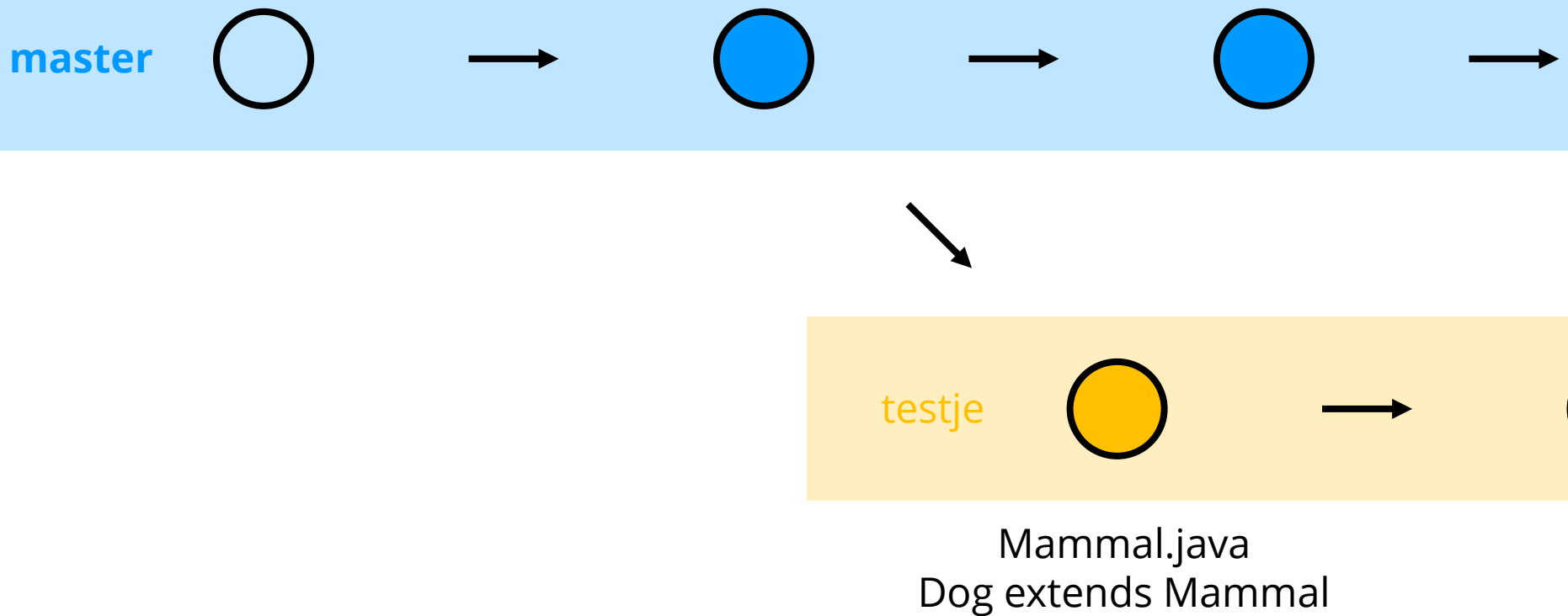


Animal.java  
Dog extends Animal



gerichte, lusloze **graaf** (zie verder Algoritmen)

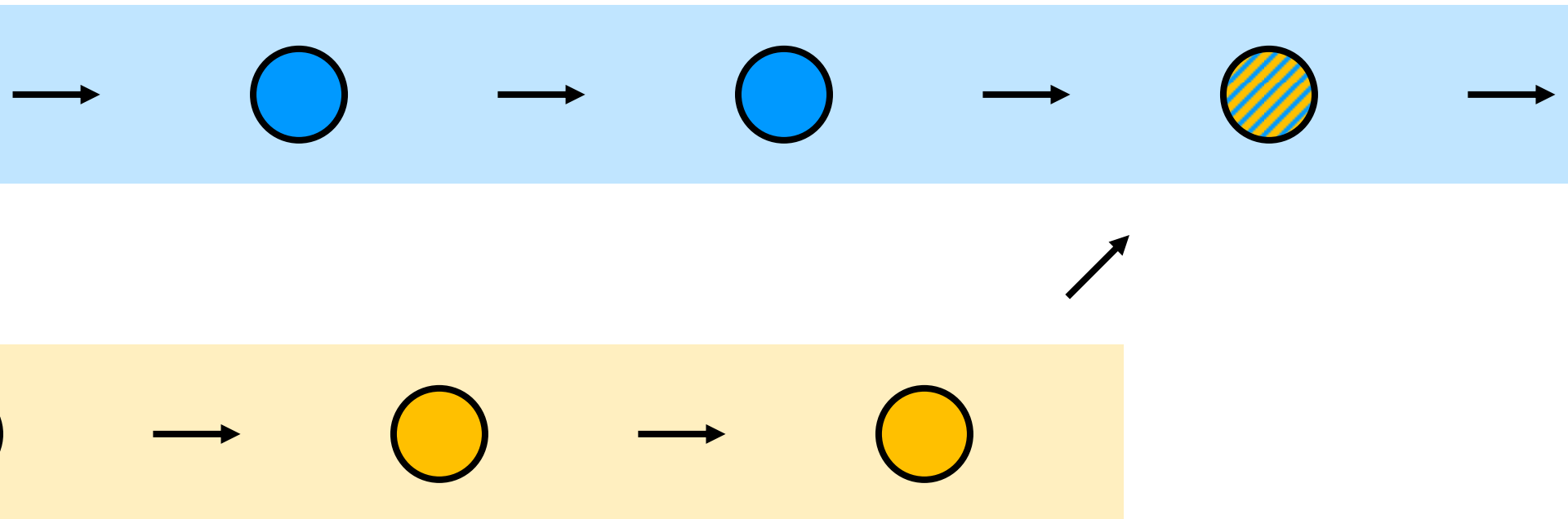
# Branches



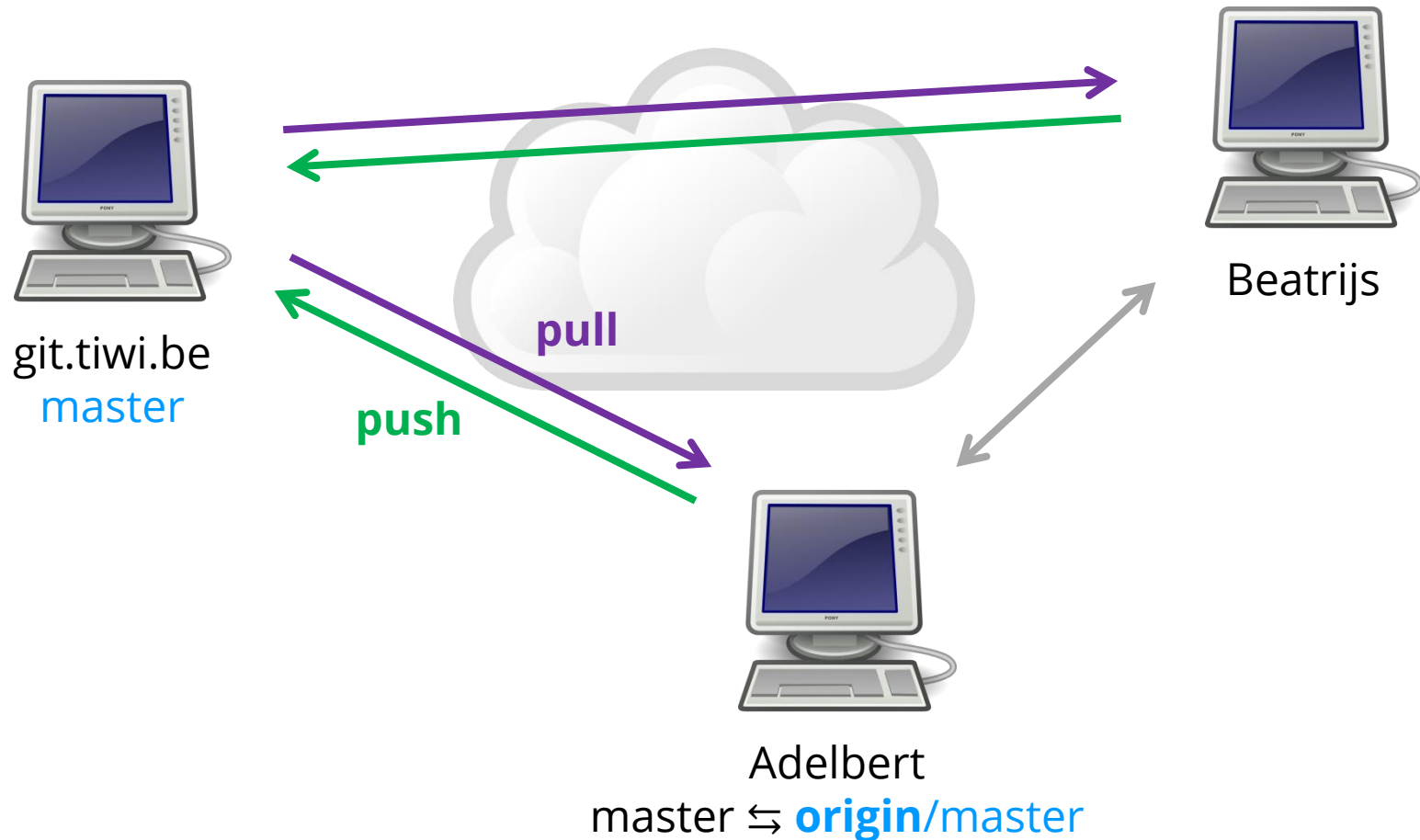


# Merges

**merge commit**  
Mammal extends Animal



# Gedistribueerd versiebeheer



DEEL 3

\$ git help

# Basisscenario

- `git clone "Tim De Pauw@git.tiwi.be:/git/2012-2013/vop/team00" vop`  
# Eenmalig: repository downloaden
- `cd vop`
- `git status` / `git log` / `gitk` / ... # Historiek bekijken
- `vim readme.txt` # Bestand aanmaken (of bewerken)
- `git add readme.txt` # Git op de hoogte brengen
- `git rm Dummy.java` # Ander bestand verwijderen
- `git commit -m "Readme toegevoegd, Dummy verwijderd"`
- `git pull` # Code van collega's integreren
- `git push` # Eigen code uploaden

# Conflicten

- Bij *git pull* blijkt dat iemand een bestand wijzigde, maar jij dat lokaal ook deed
- Tenzij automatisch opgelost door Git:
  - `git status` # Welke bestanden?
  - `vim Dog.java` # Manueel aanpassen
  - `git add Dog.java`
  - `git commit -m "Conflict in Dog.java opgelost"`

DEEL 4

# Epiloog

# Git met stijl

- Zo weinig mogelijk binaire bestanden (dus o.a. geen *.class*-bestanden)
- Enkel commits met compileerbare, werkende code
- Duidelijke commit messages
- Zinvolle mappenstructuur

# En nu ...

- Zie *doc*-map in repository
- Read the fantastic manual
- Vragen staat vrij



# Vragen?