

**Macoun' II**



# Git Crash-Kurs

Tobias Günther

# I. Repository

I a) Neues Repo erstellen



| b) Existierendes Repo clonen



## **2. Basis-Workflow**

**Working Directory**  
„Real Files“

**Staging Area**  
„The Index“

**.git Directory**  
„The local Repo“

**Working Directory**  
„Real Files“

**Staging Area**  
„The Index“

**.git Directory**  
„The local Repo“



**untracked**

**Working Directory**  
„Real Files“



**unmodified**

**Staging Area**  
„The Index“

**.git Directory**  
„The local Repo“

**Working Directory**  
„Real Files“



**modified**

**Staging Area**  
„The Index“

**.git Directory**  
„The local Repo“

**Working Directory**  
„Real Files“



**Staging Area**  
„The Index“



**staged**

**.git Directory**  
„The local Repo“

staged

=

im nächsten Commit

# Warum Staging Area?

# Bananen-Milchshake

# Bananen-Milchshake

## Mixer



# Bananen-Milchshake

Mixer + Milch



# Bananen-Milchshake

Mixer



+ Milch



+ Tomate



**Schmeckt wie ein Bug.**

Versuchen Sie ein  
“Rollback” der Tomate!

“Mixen” in der  
Versionskontrolle heißt...

**...Rollback wird schwierig.**

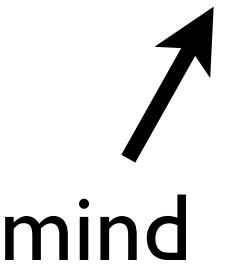
...schwer nachvollziehbar.

Ihre Kollegen werden Sie hassen.

**VC-Grundregel:**  
**“Commit only related changes!”**

| Feature = | Commit

| Feature = | Commit



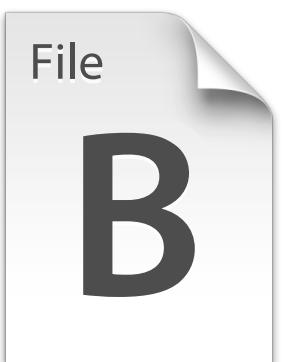
mind

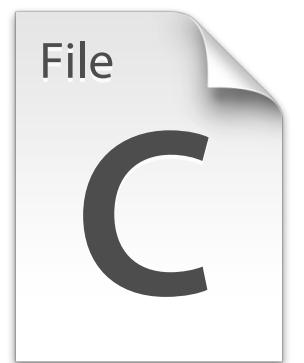
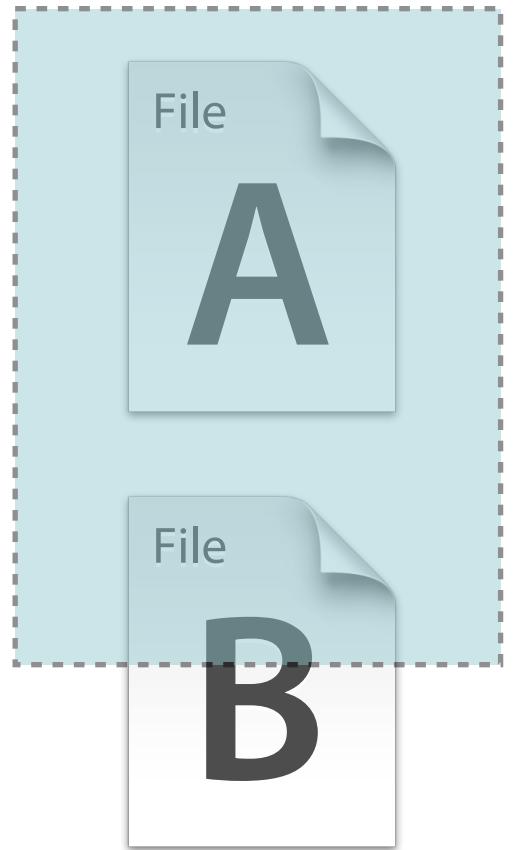
| Bugfix = | Commit

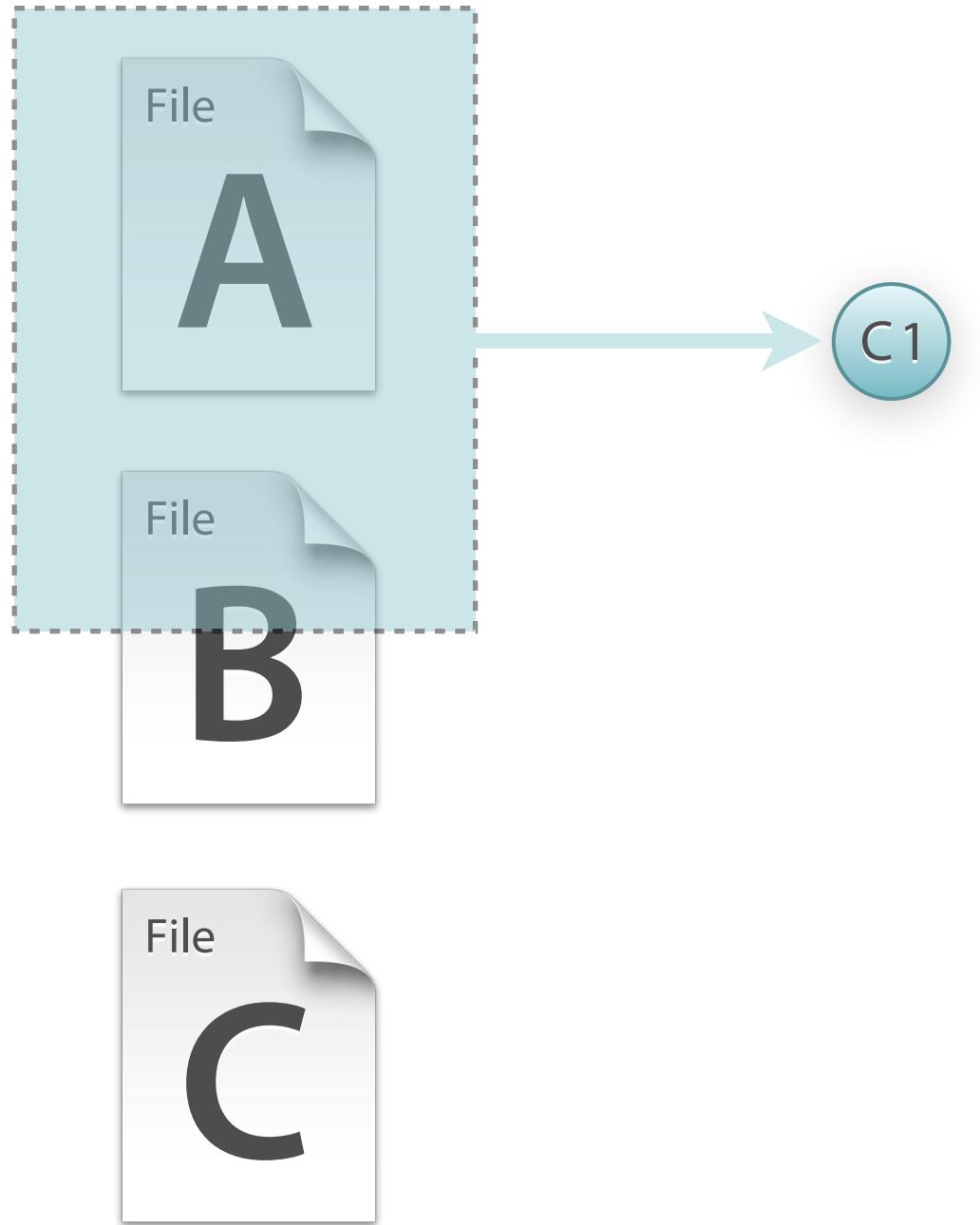
**“Staging Area”**

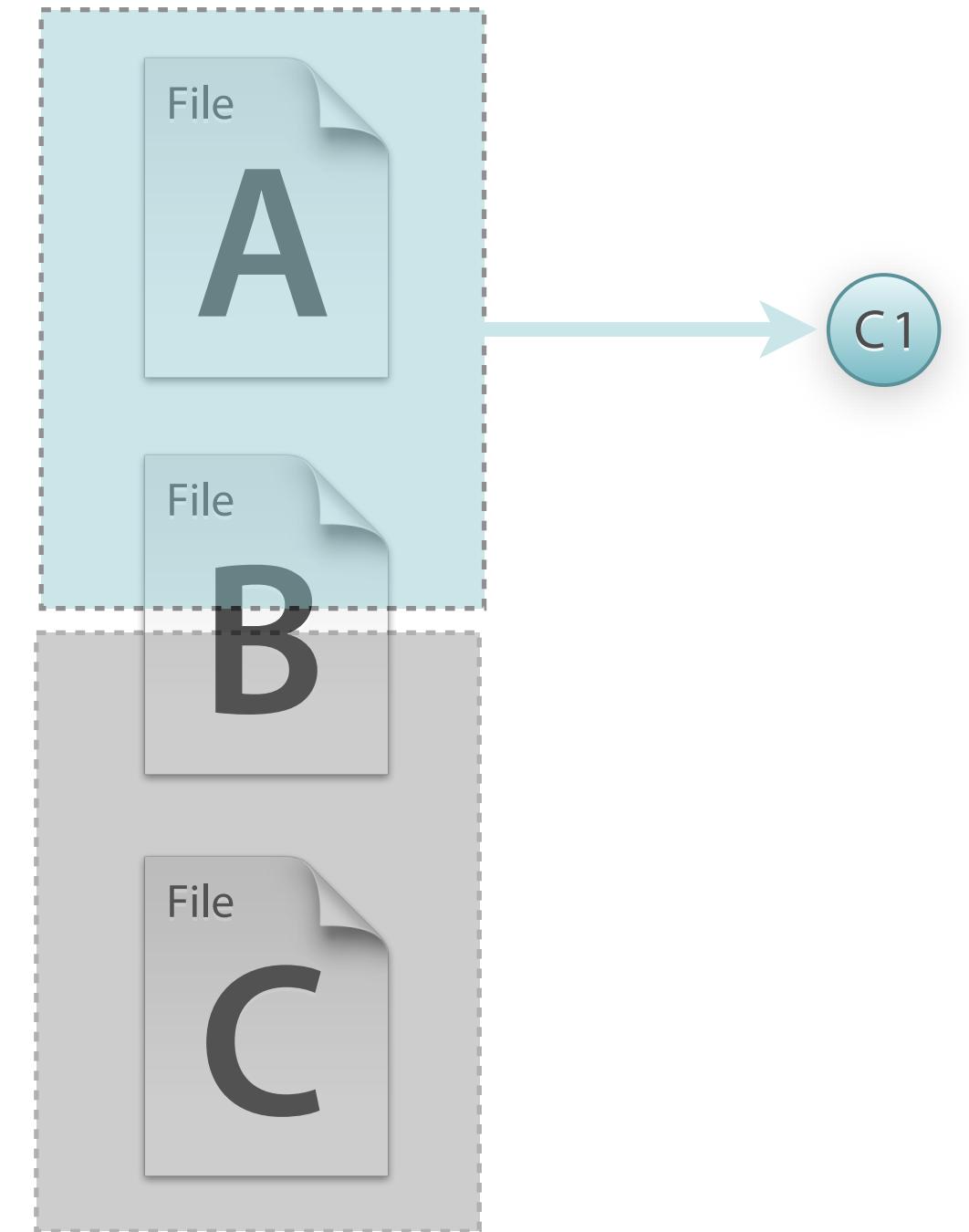
“Staging Area”  
-oder-

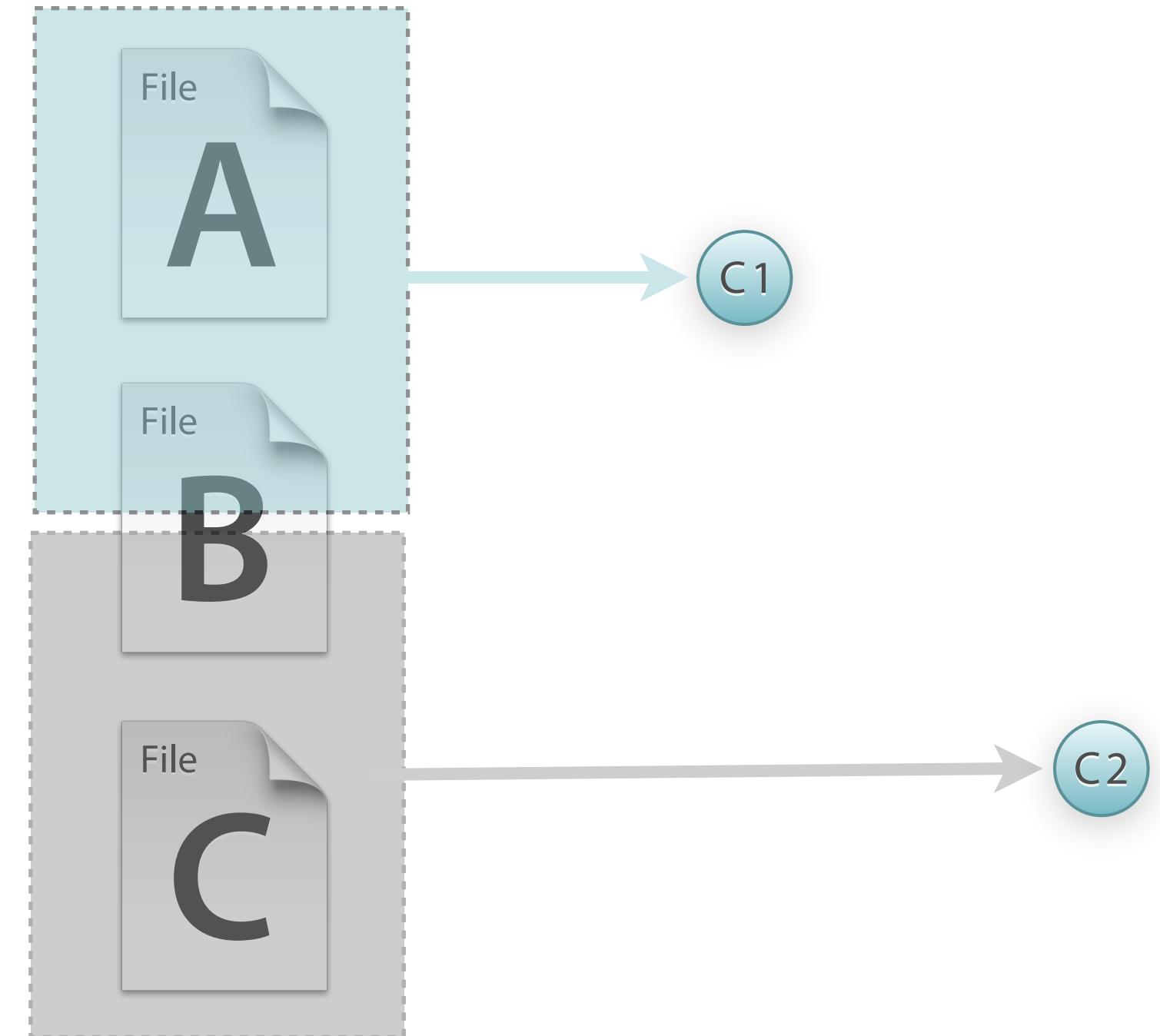
Commits granular  
zusammenstellen



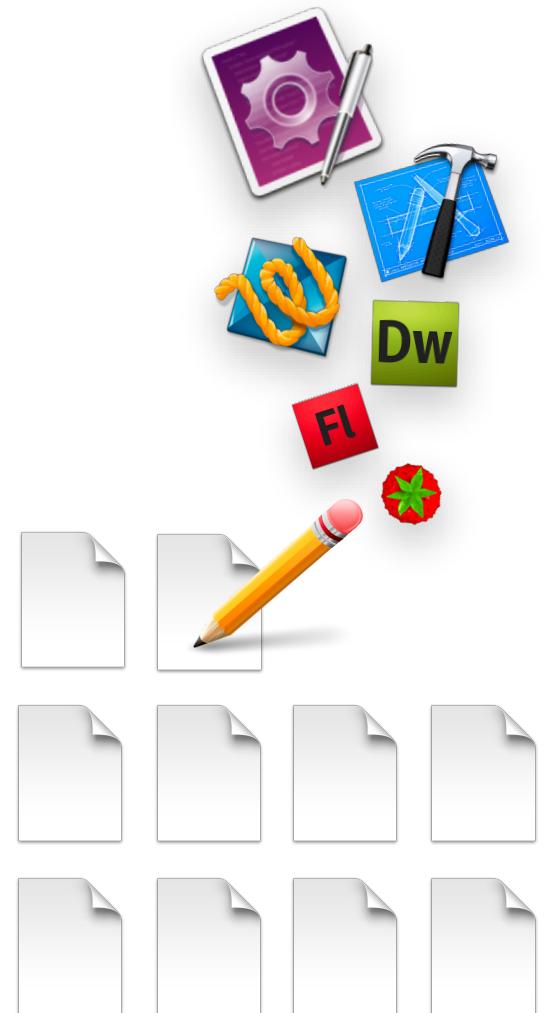








**Working Directory**  
„Real Files“



**Staging Area**  
„The Index“

**.git Directory**  
„The local Repo“

## Working Directory „Real Files“



## Staging Area „The Index“

## .git Directory „The local Repo“

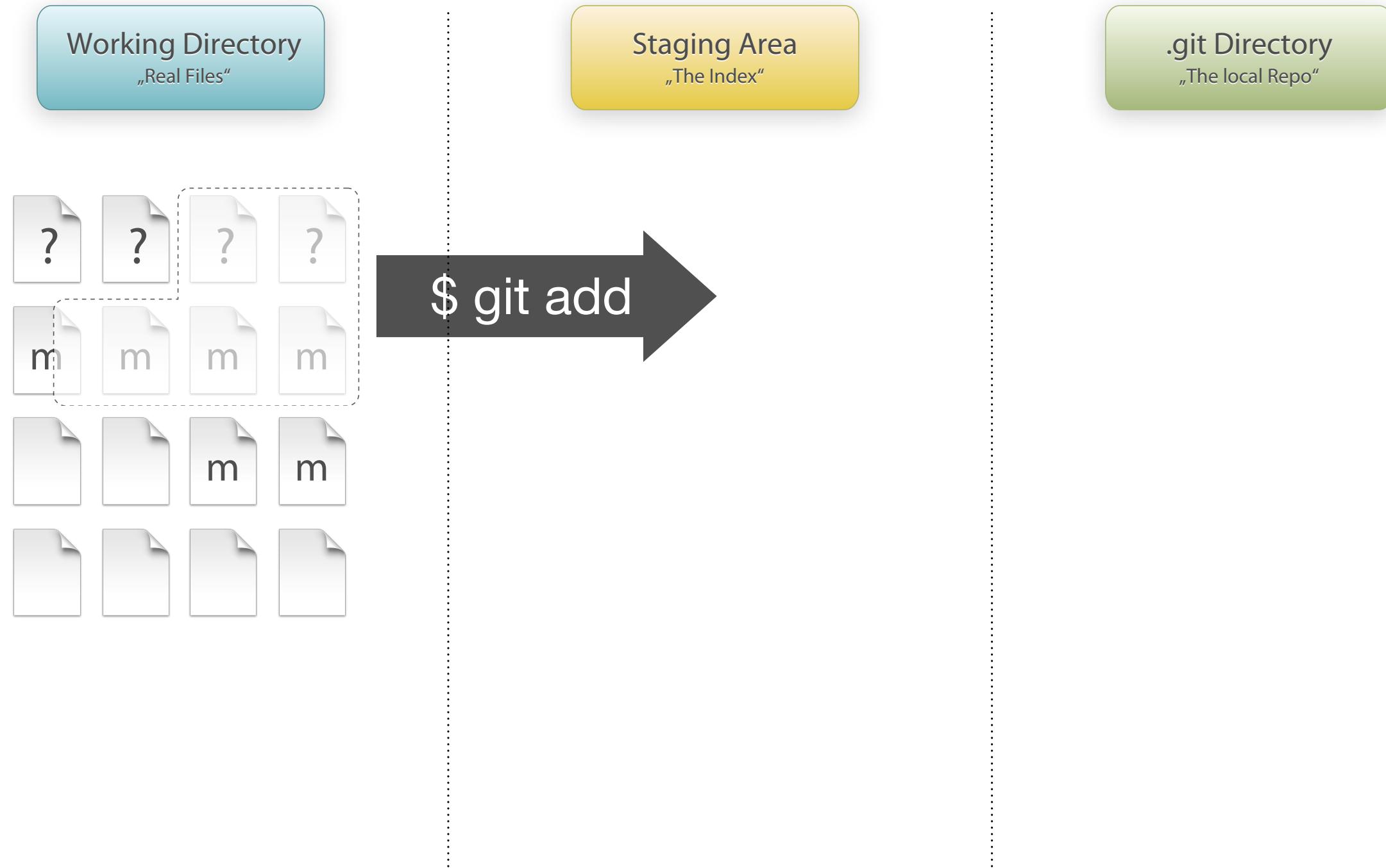
**Working Directory**  
„Real Files“



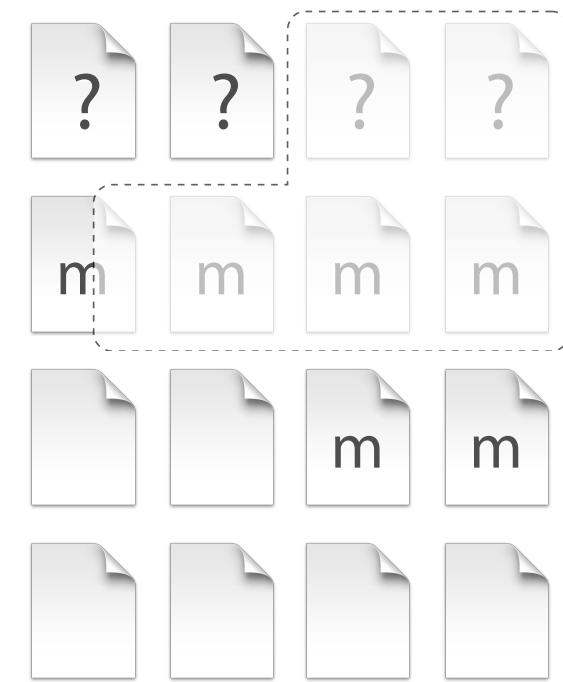
**Staging Area**  
„The Index“

**.git Directory**  
„The local Repo“

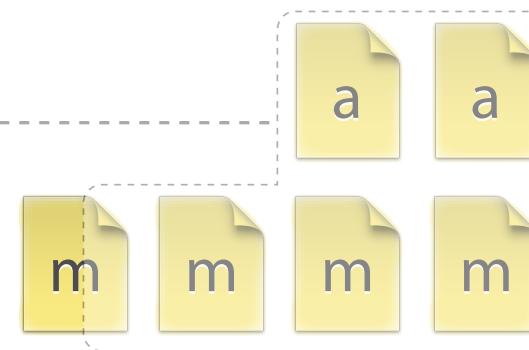
\$ git status  
Changed (but not staged):  
f5, f6, f7, f8, f11, f12  
Untracked files:  
f1, f2, f3, f4



## Working Directory „Real Files“



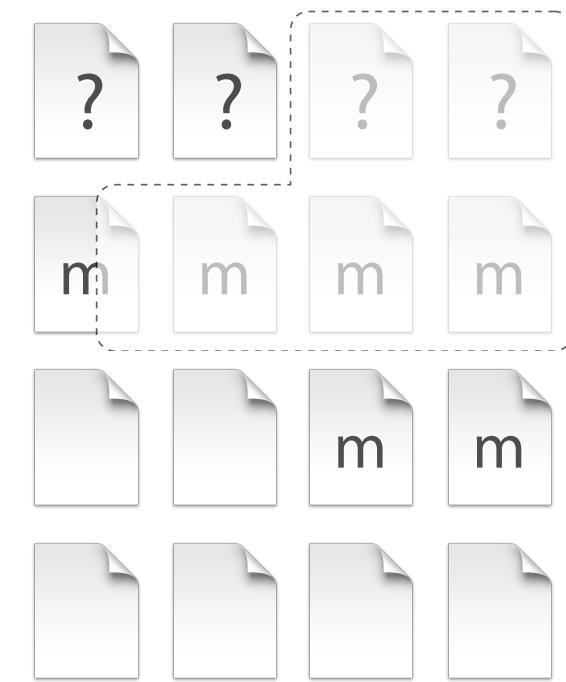
## Staging Area „The Index“



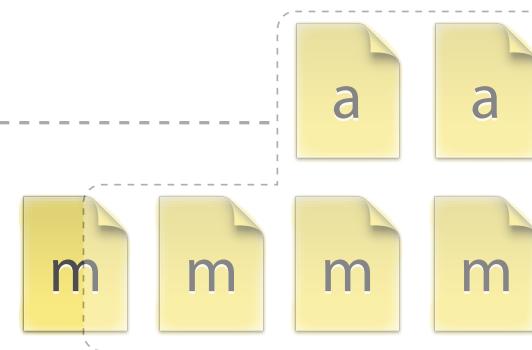
## .git Directory „The local Repo“



## Working Directory „Real Files“

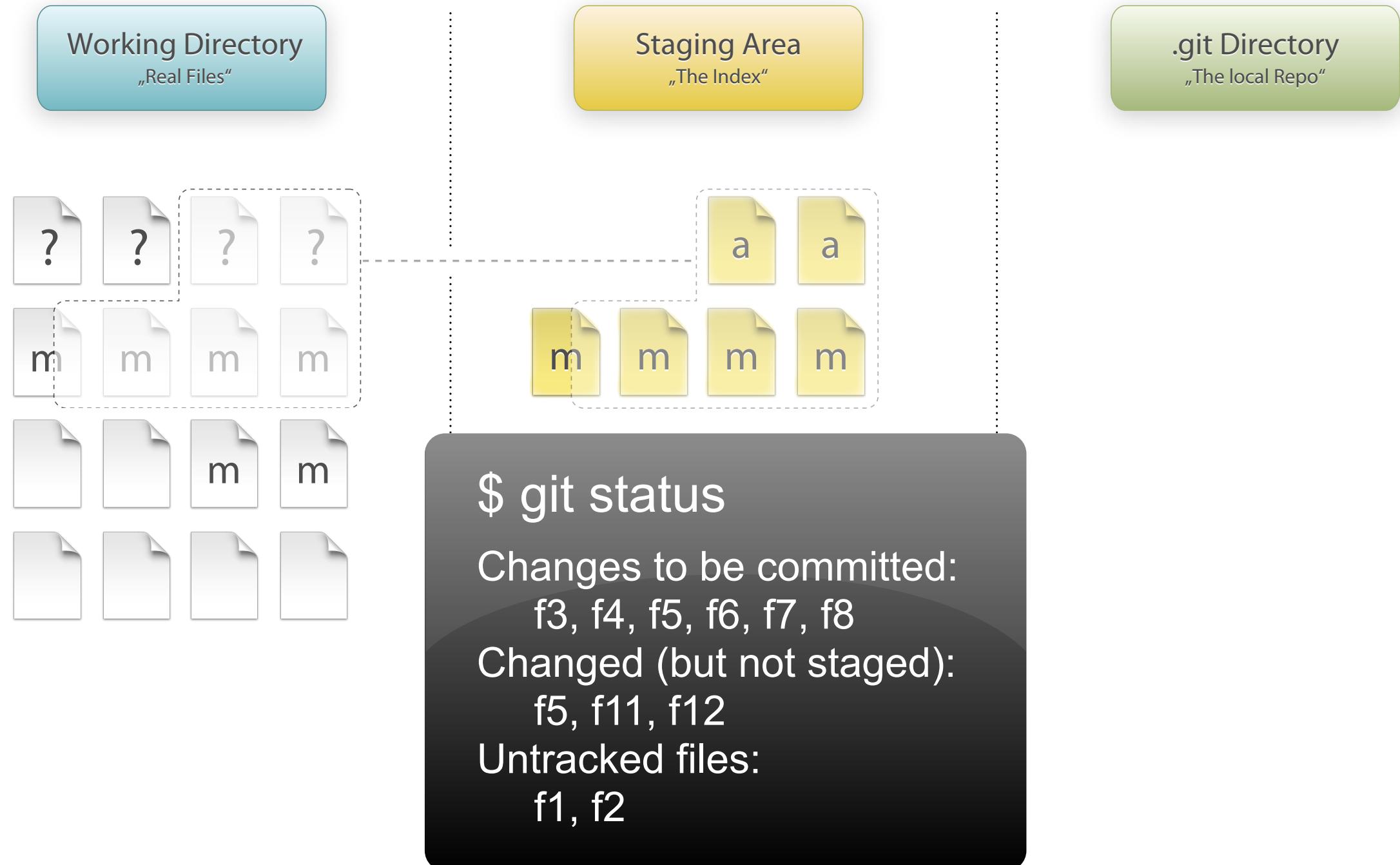


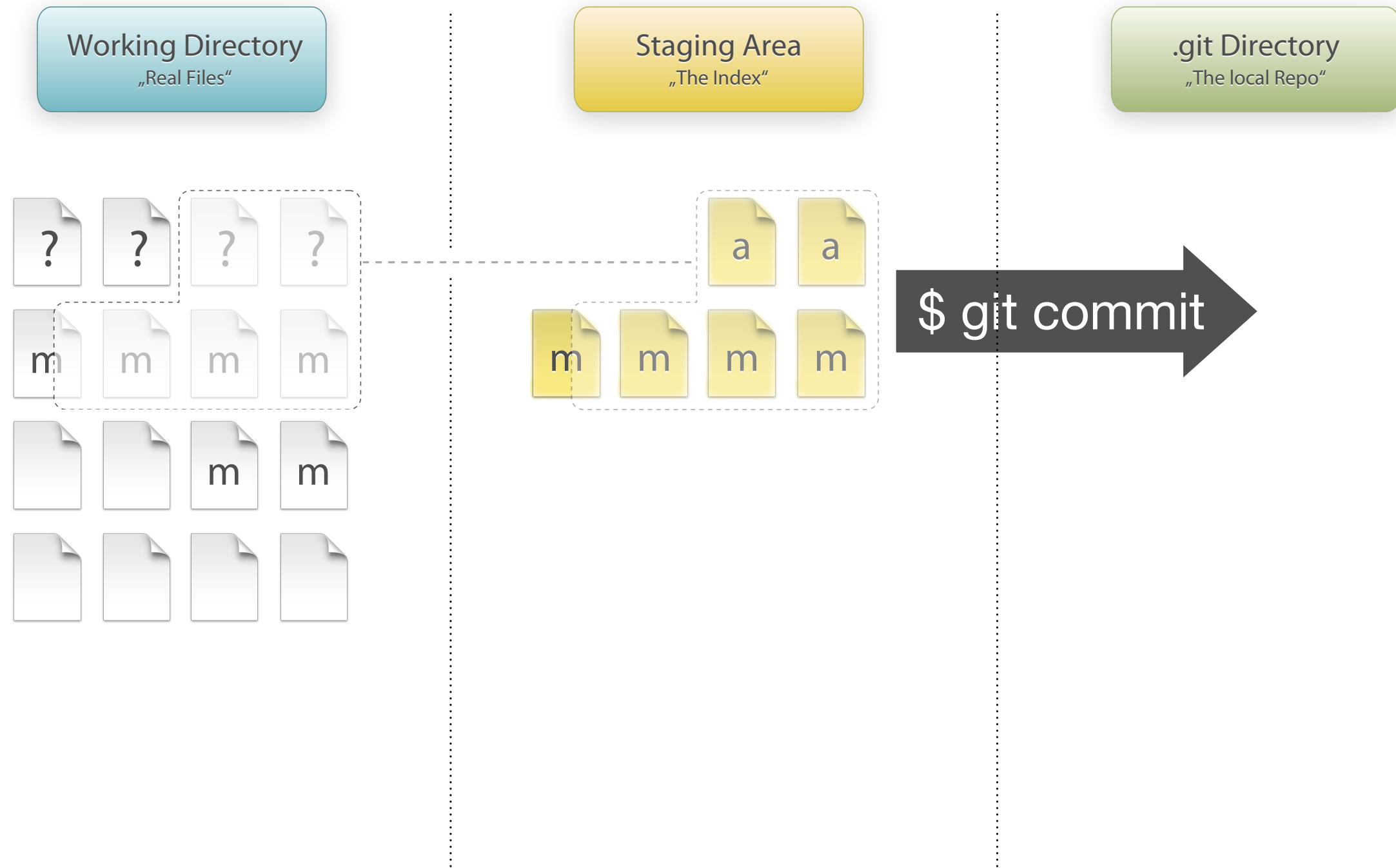
## Staging Area „The Index“



## .git Directory „The local Repo“







**Working Directory**  
„Real Files“



**Staging Area**  
„The Index“

**.git Directory**  
„The local Repo“

```
$ git status
Untracked files:
  f1, f2
Changed (but not staged):
  f5, f11, f12
```

`git commit`

`git commit -a`

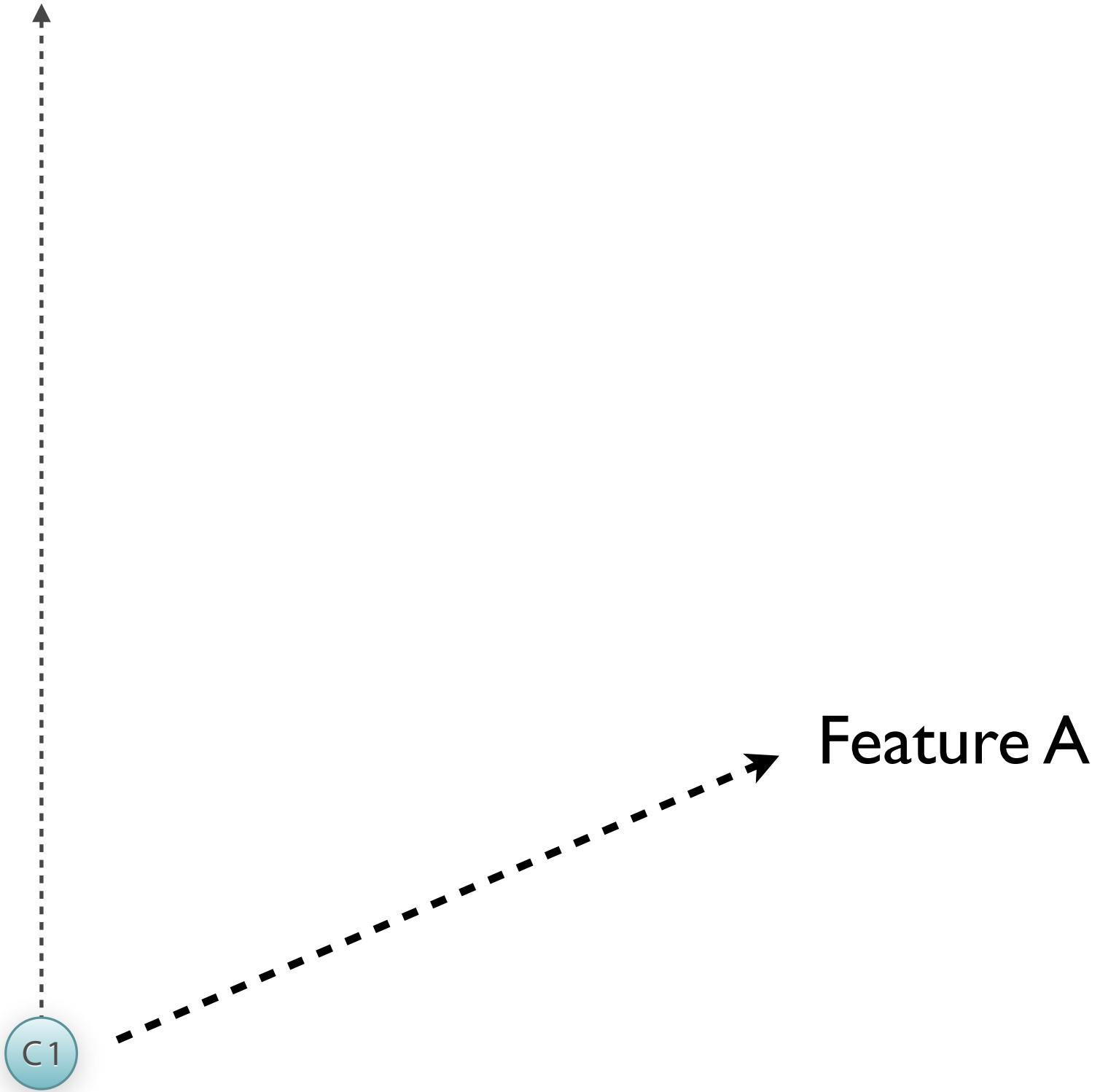


# **3. History**



# **4. Branching & Merging**

# **Parallele Entwicklung an mehreren Themen**

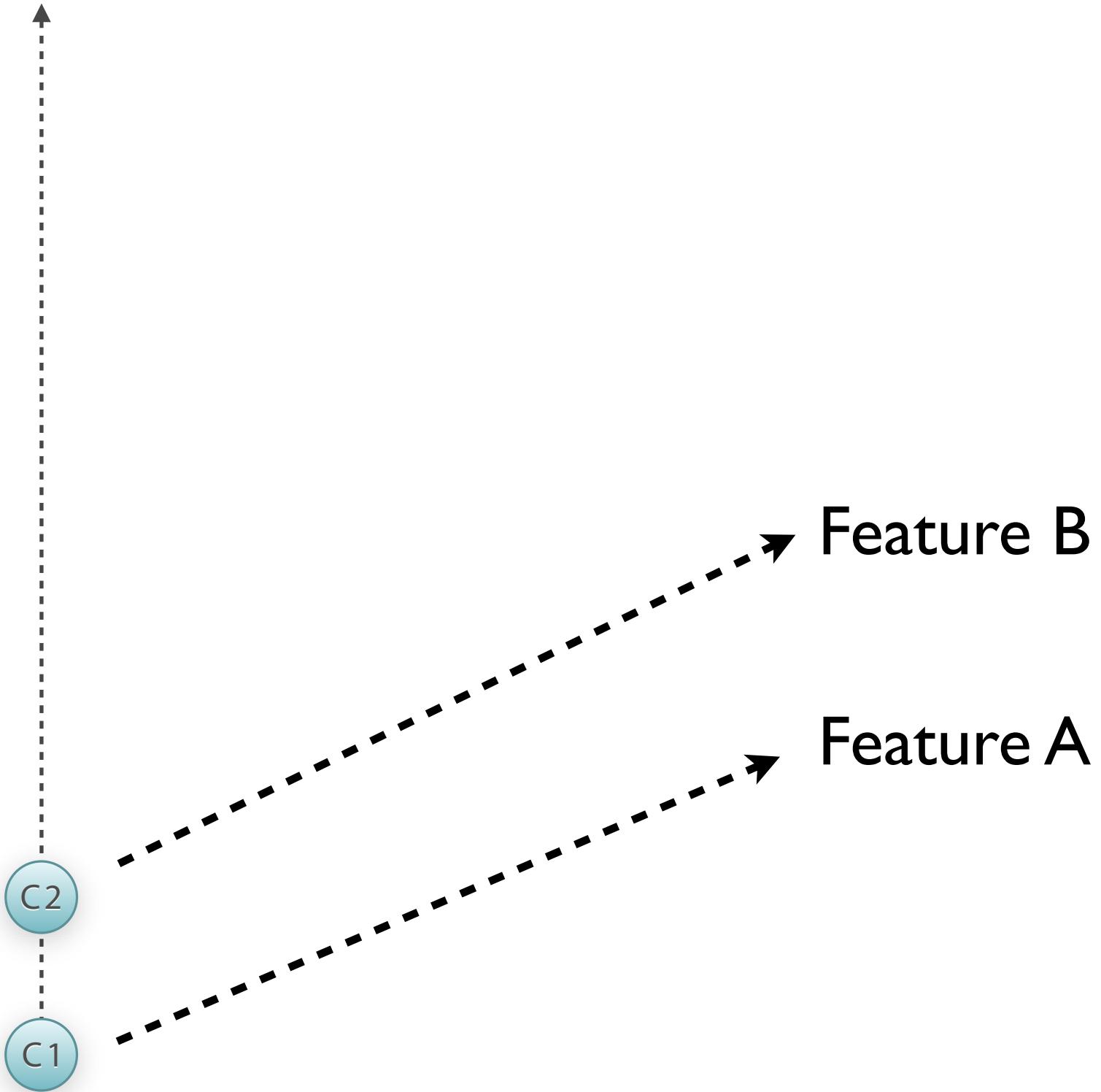


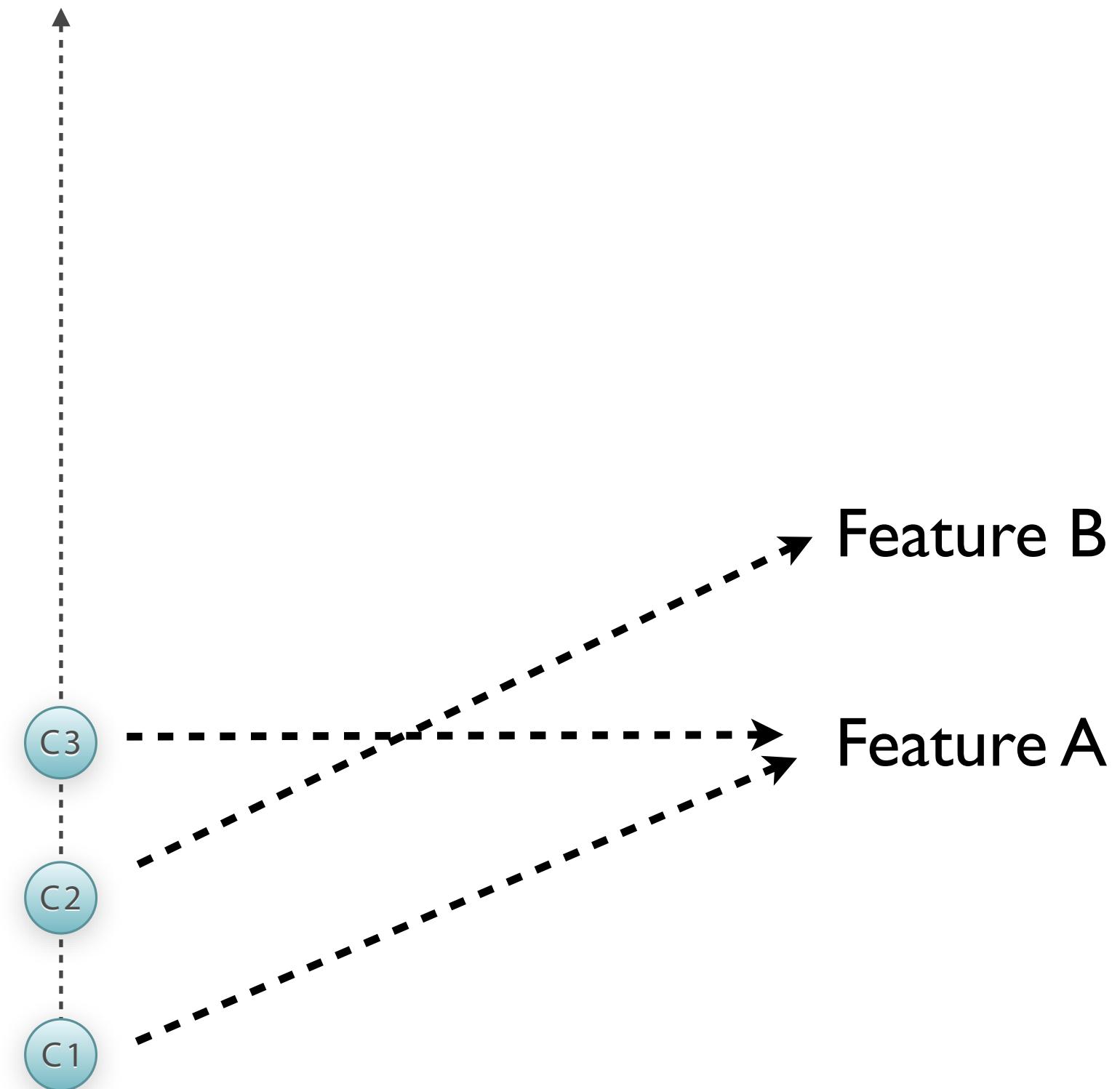
Feature A

**Feature B**

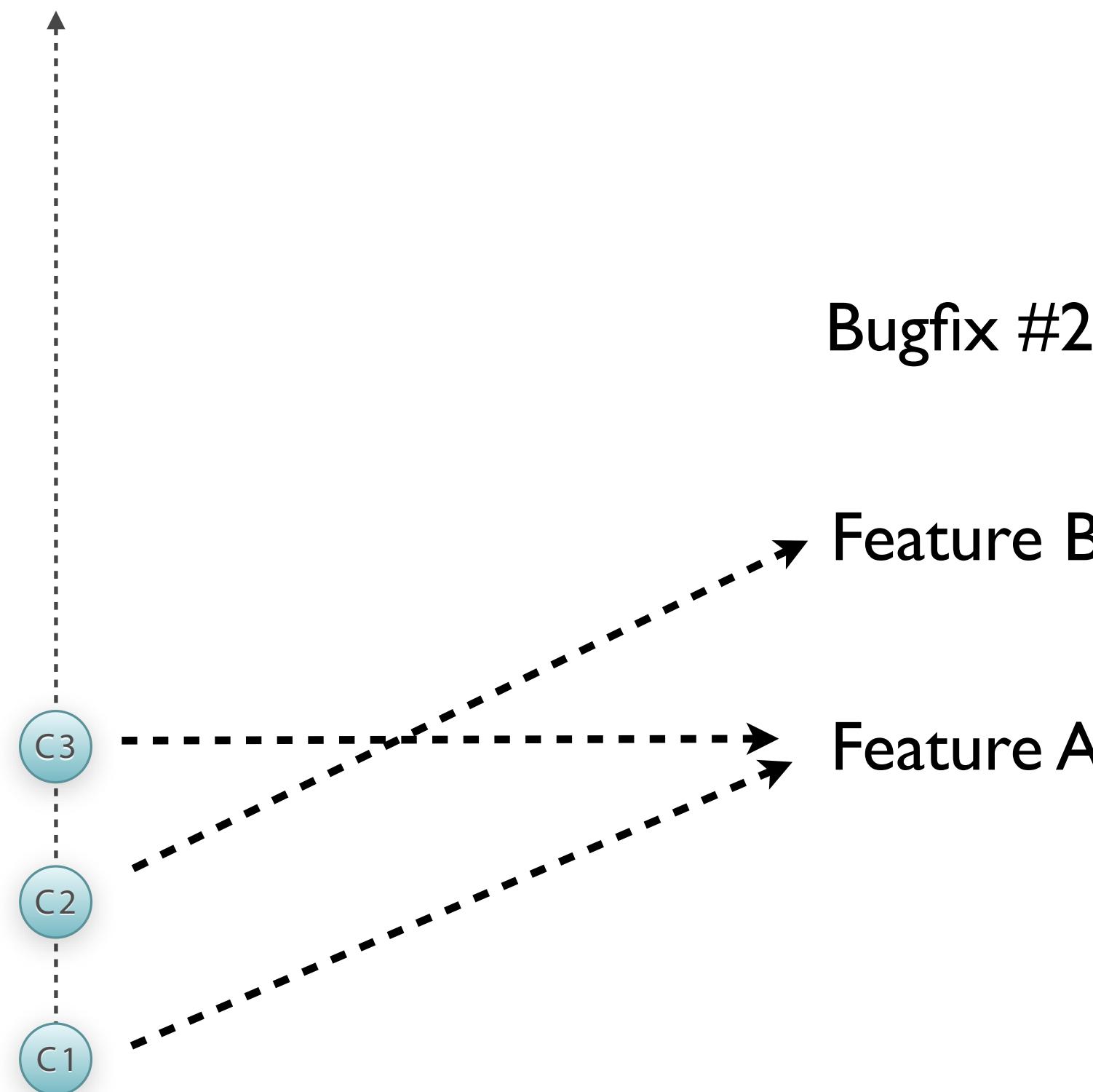
**Feature A**

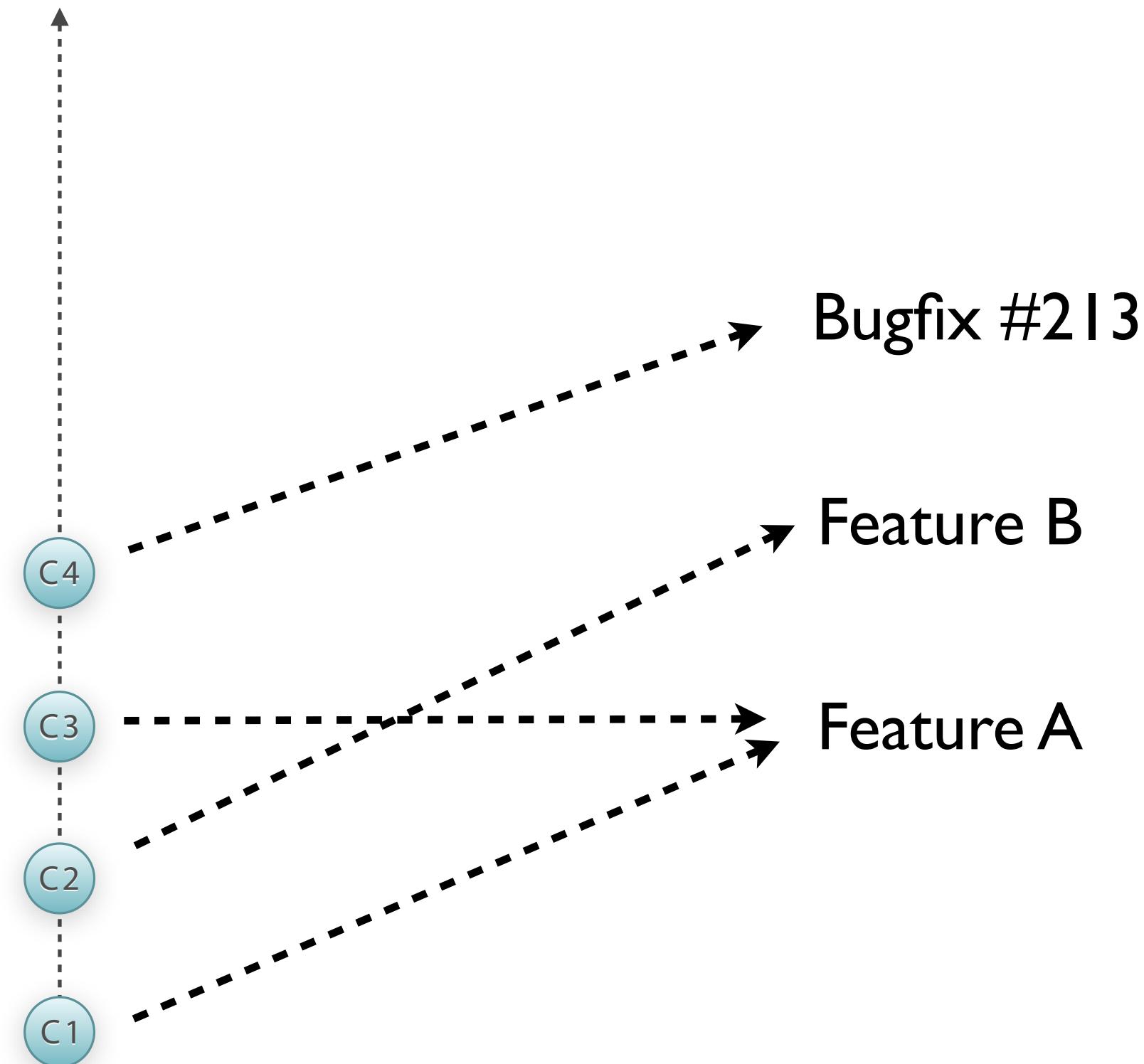
C1

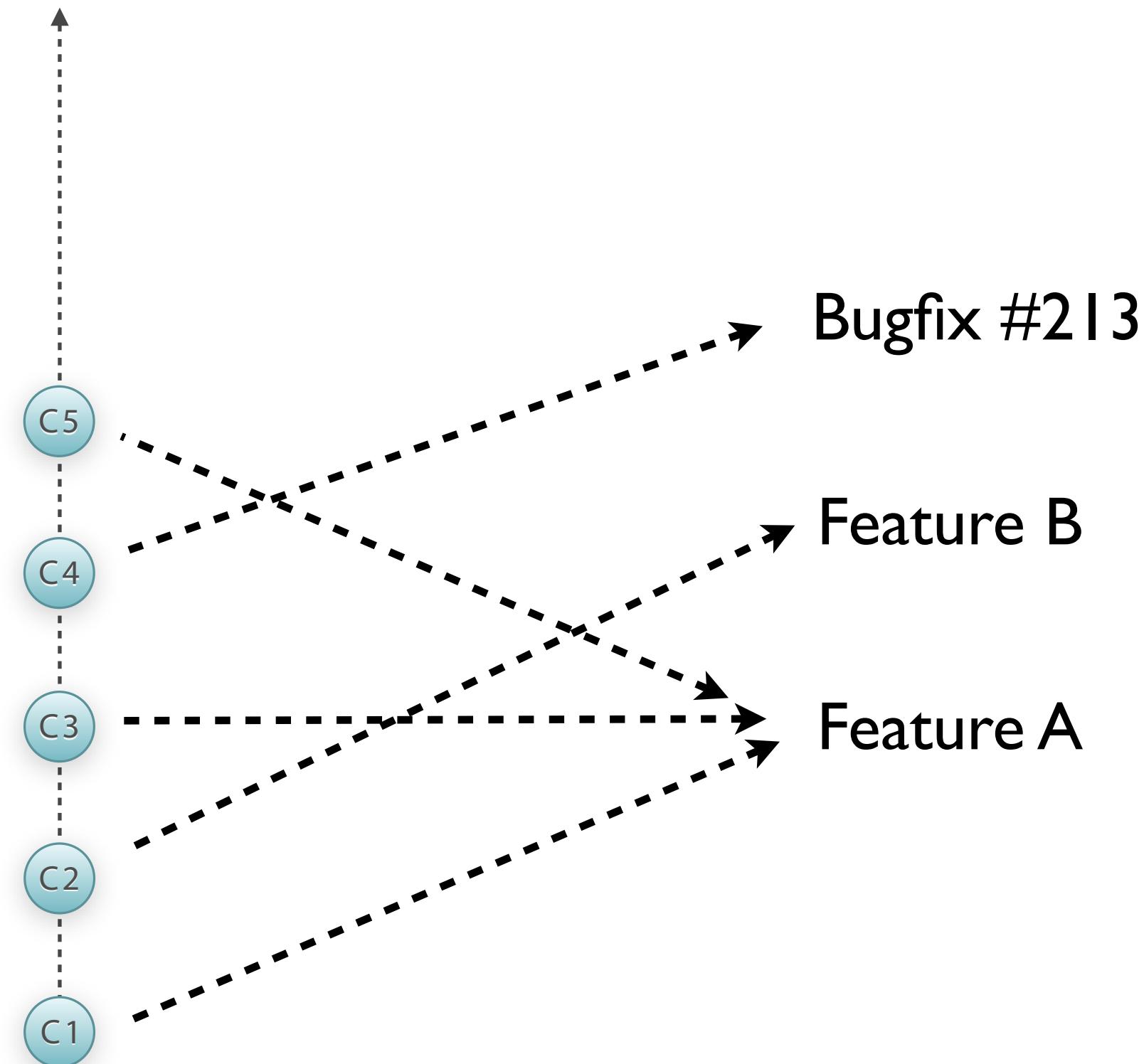


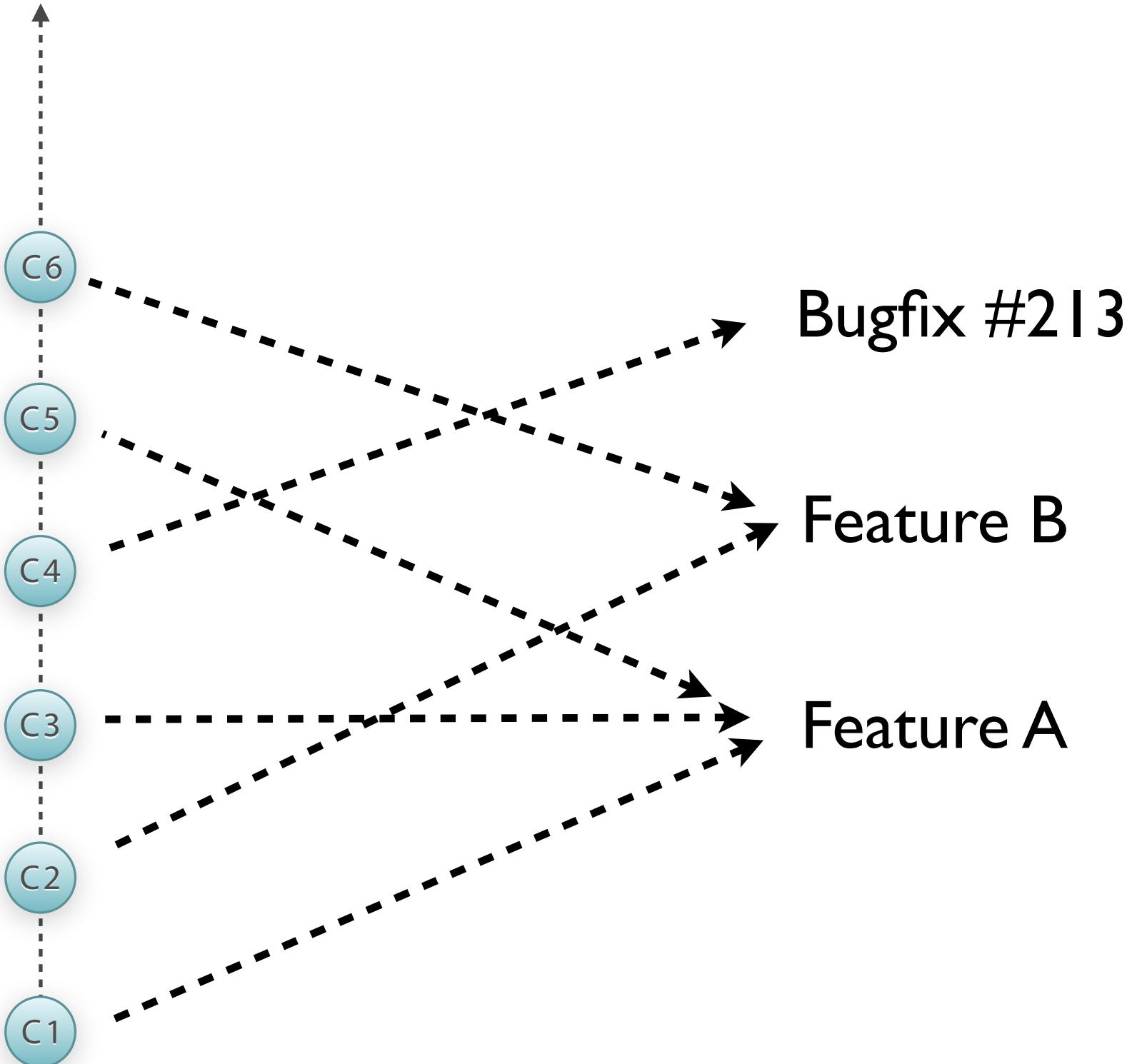


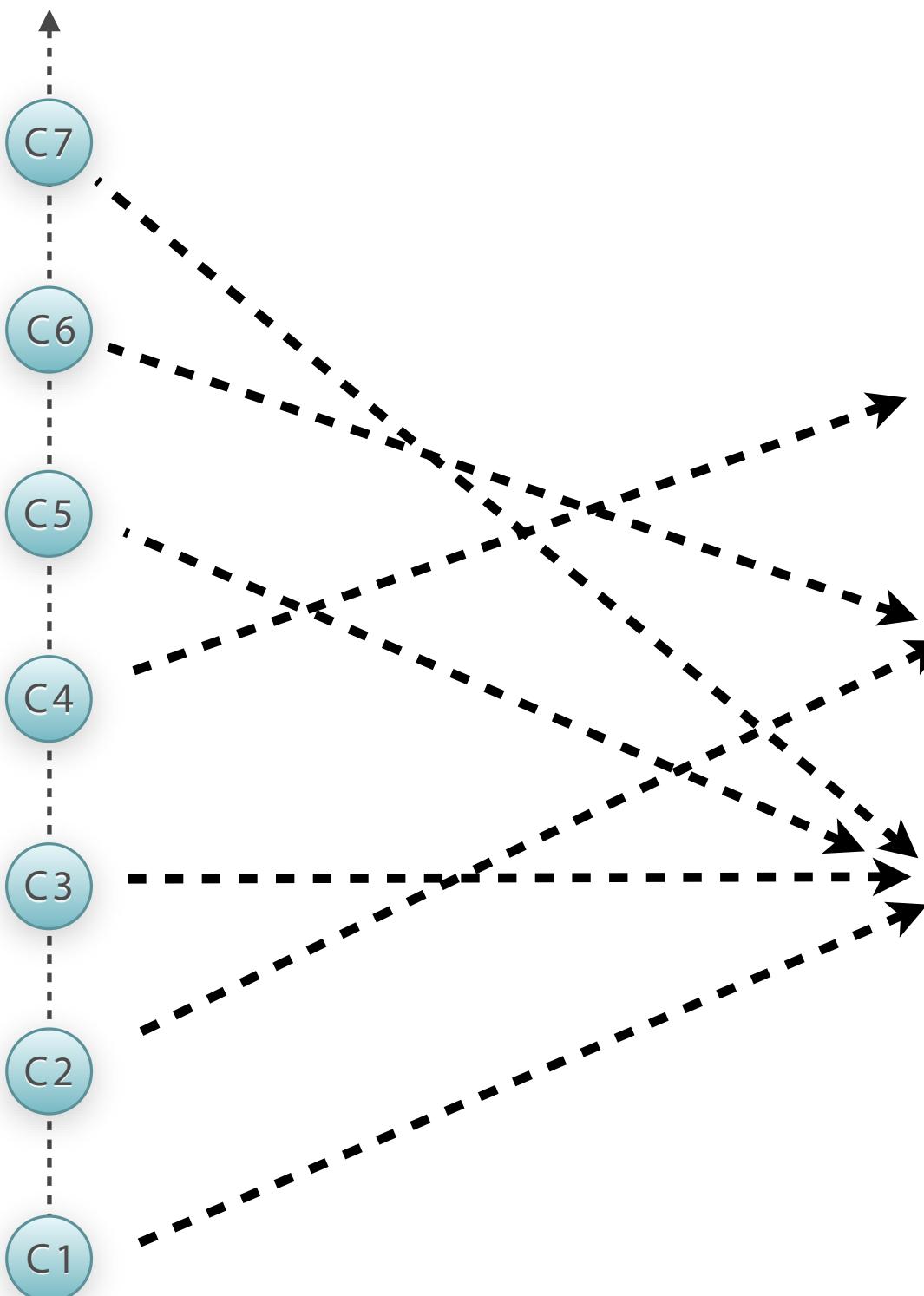
**Bugfix #213**











Bugfix #213

Feature B

Feature A

Nicht gut.



**Code aus unfertigen  
Features separat halten!**

Anforderungen ändern sich -  
Feature wird obsolet

**Proof of Concept schlägt fehl  
Feature unrealisierbar**

Feature A

C1

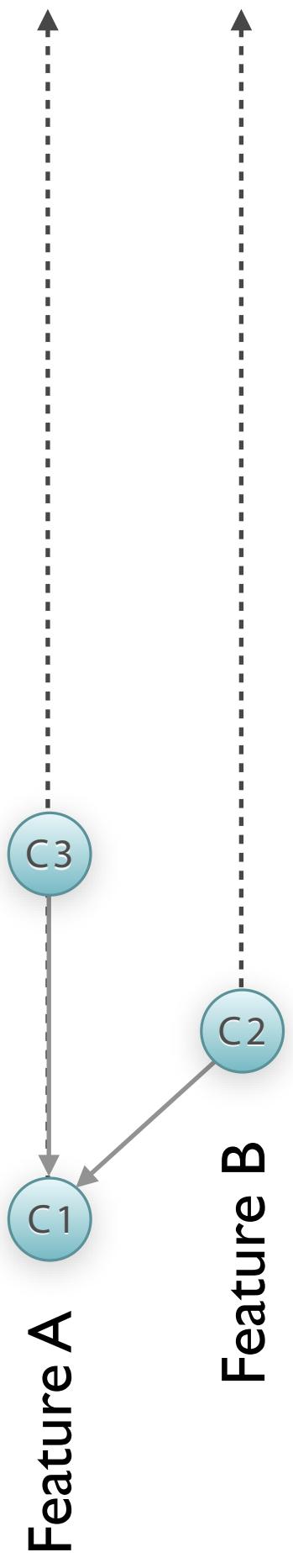




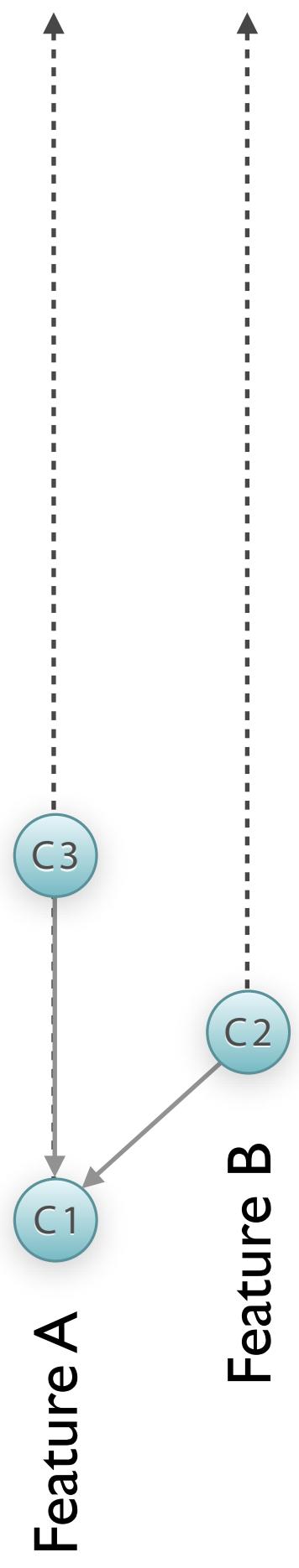
Feature A

Feature B



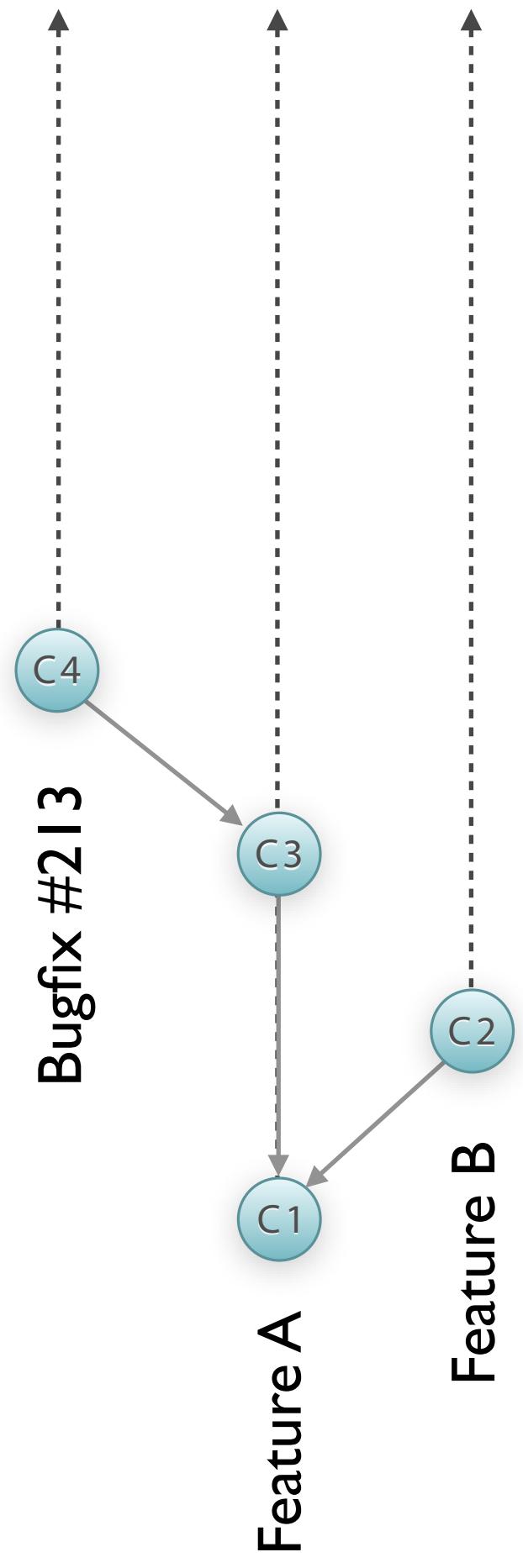


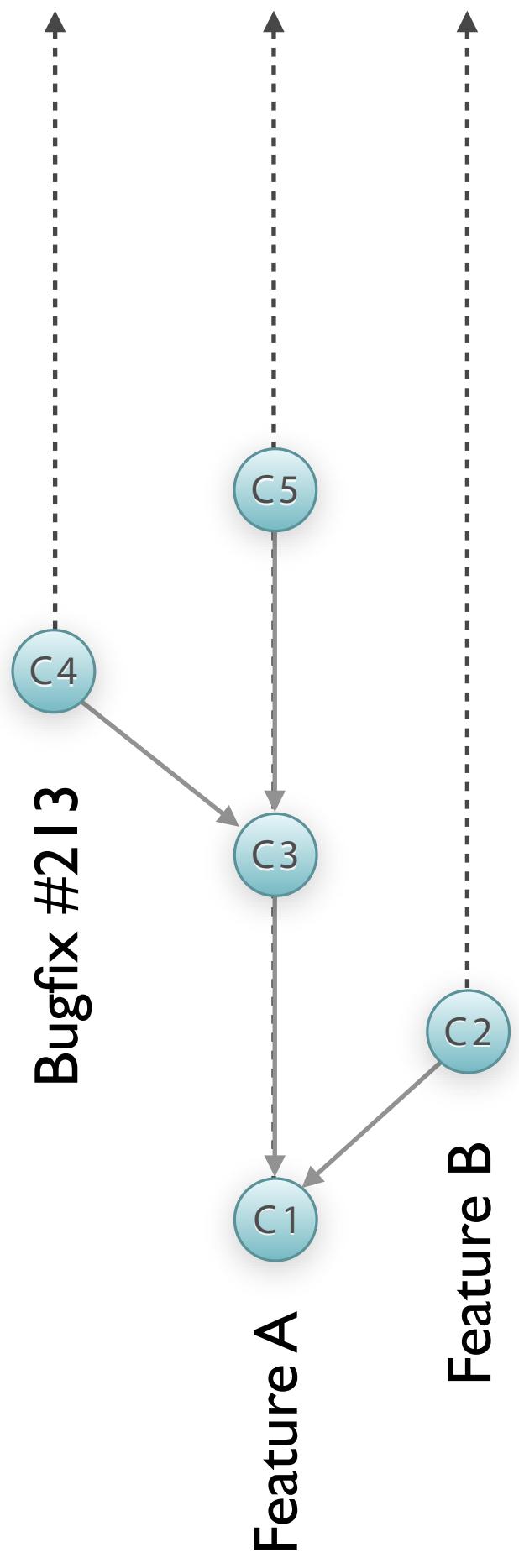
Bugfix #2 | 3

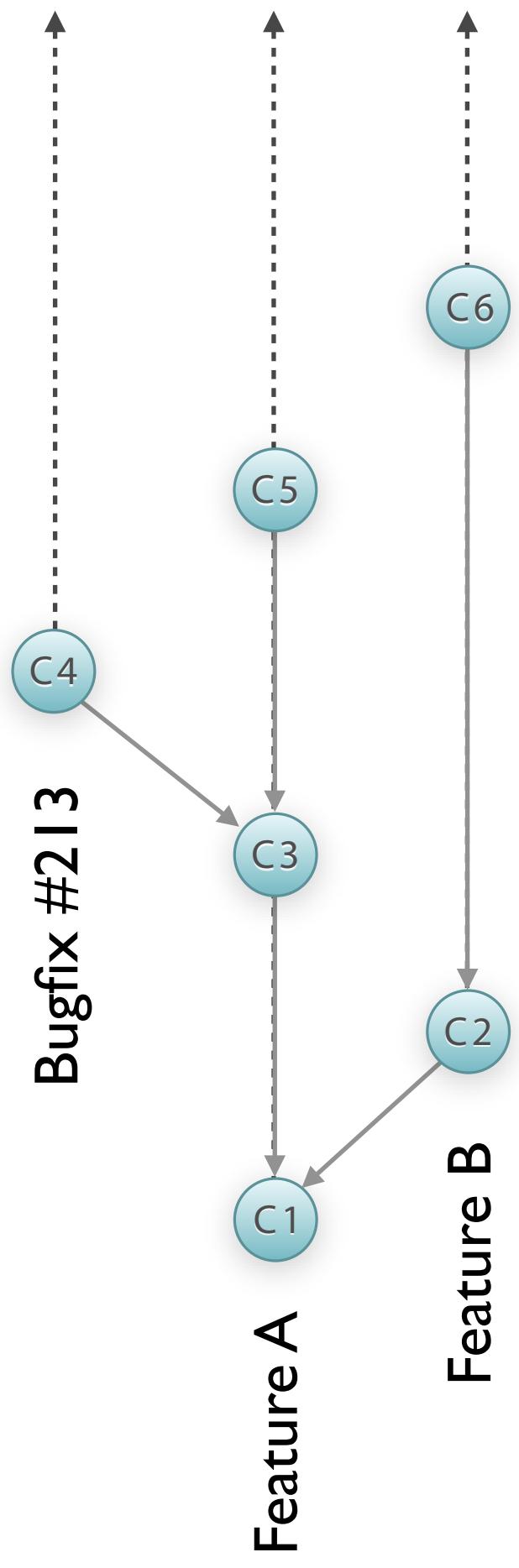


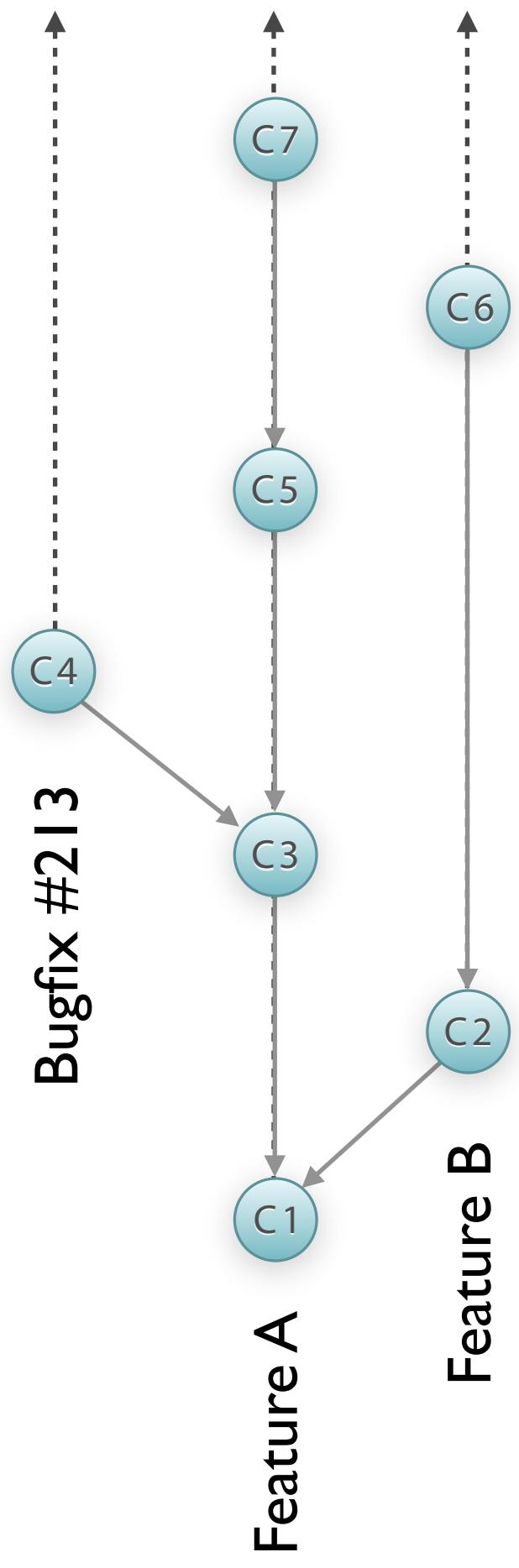
Feature A

Feature B









“SVN und Mercurial kennen  
Branches auch - nicht nur Git!”

“SVN und Mercurial kennen  
Branches auch - nicht nur Git!”

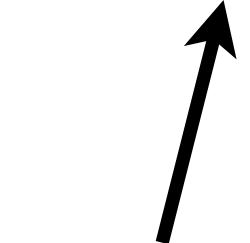


Apfel

Birne



“SVN und Mercurial kennen  
Branches auch - nicht nur Git!”



Apfel

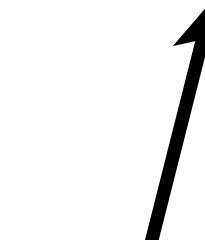
Wassermelone



Birne

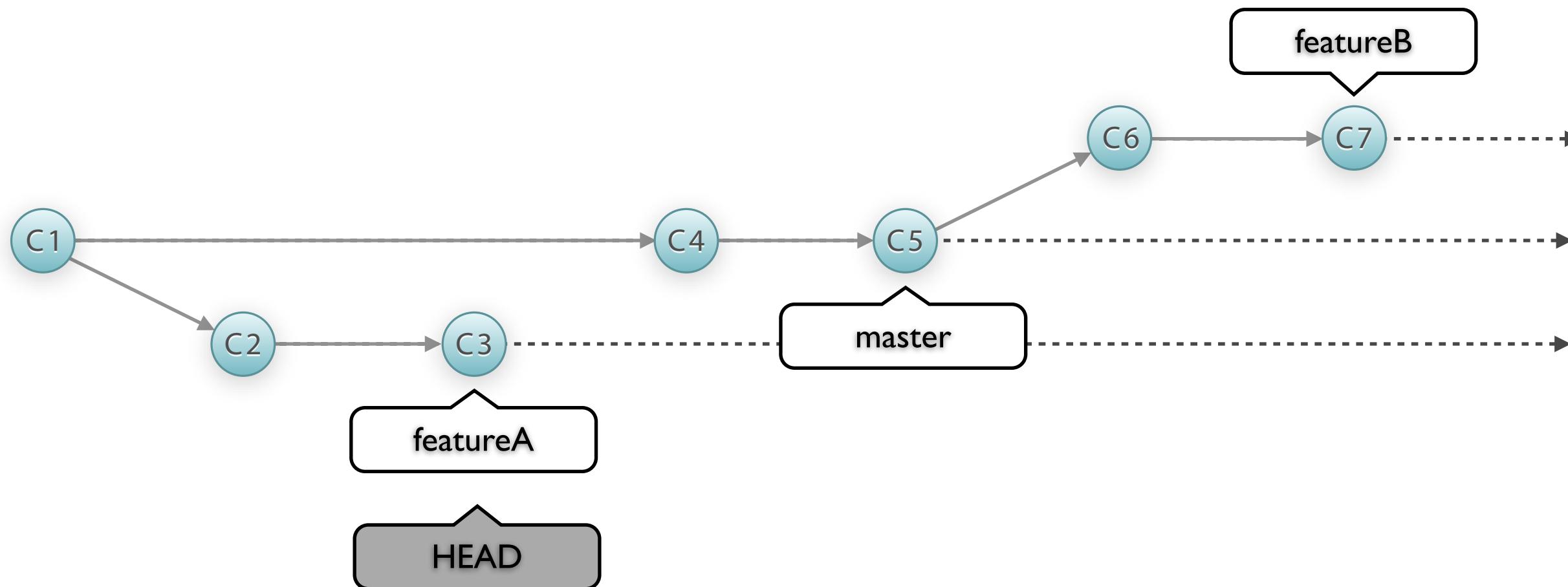


“SVN und Mercurial kennen  
Branches auch - nicht nur Git!”

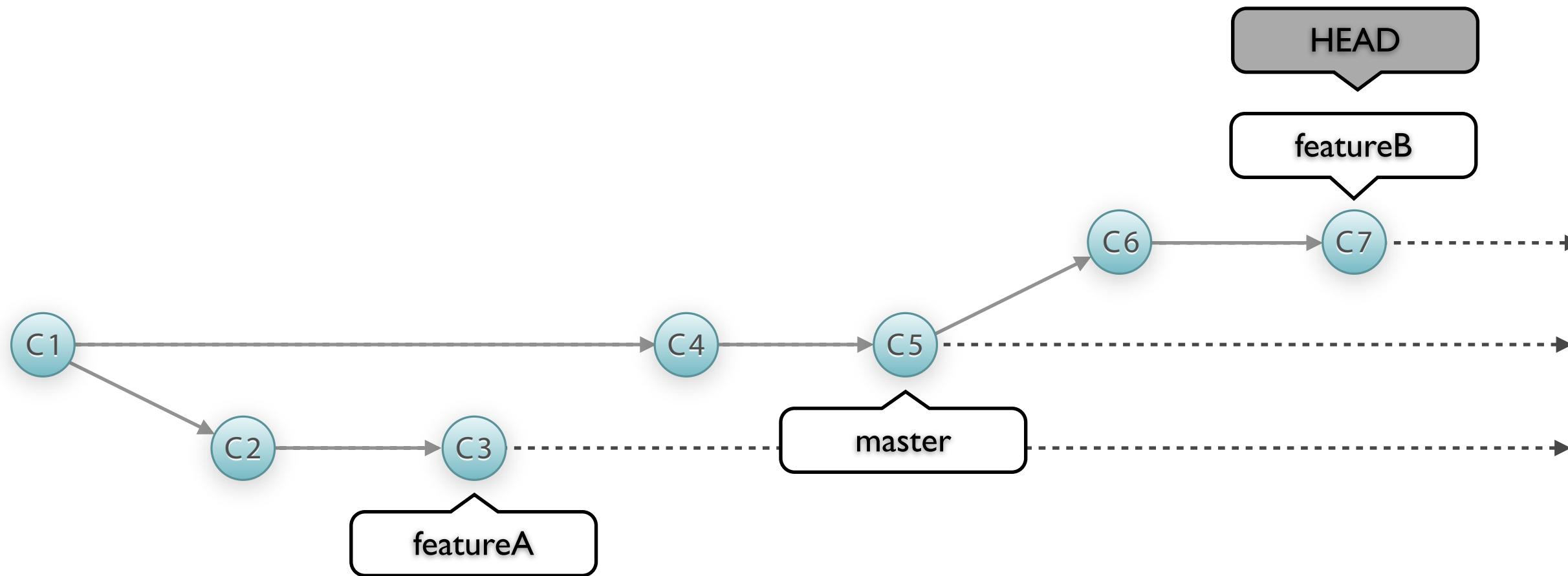


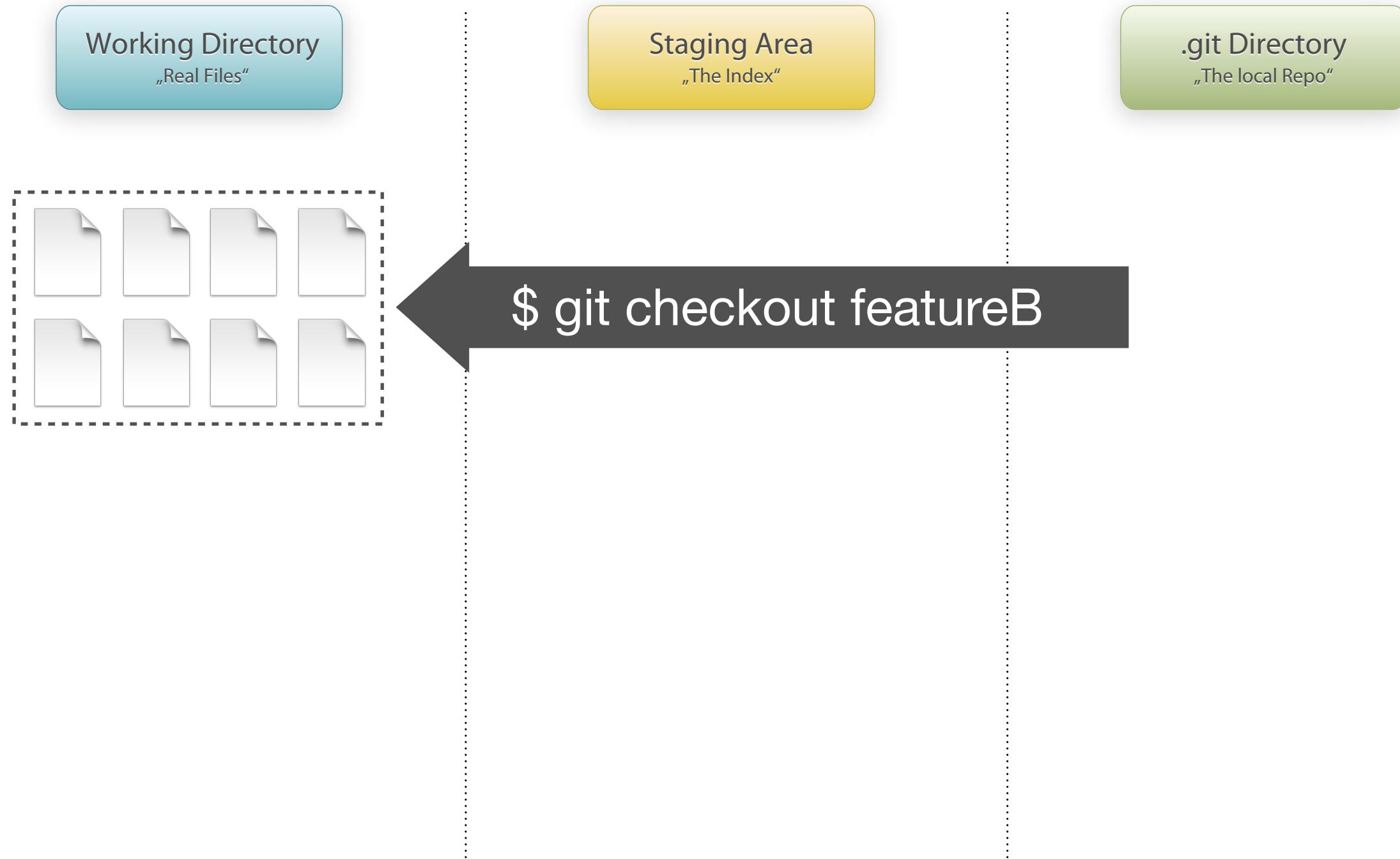
Apfel

# HEAD



# Checkout



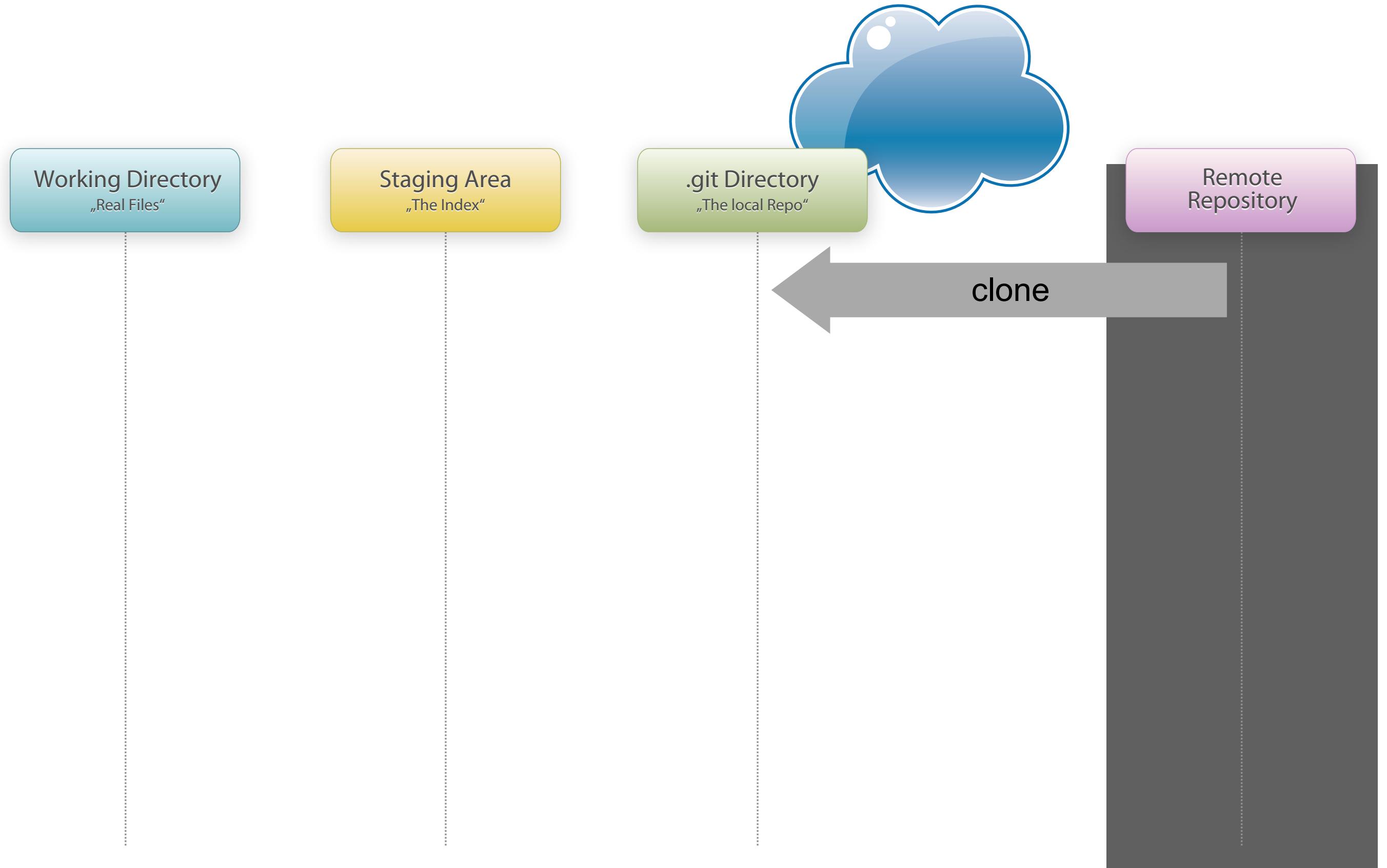


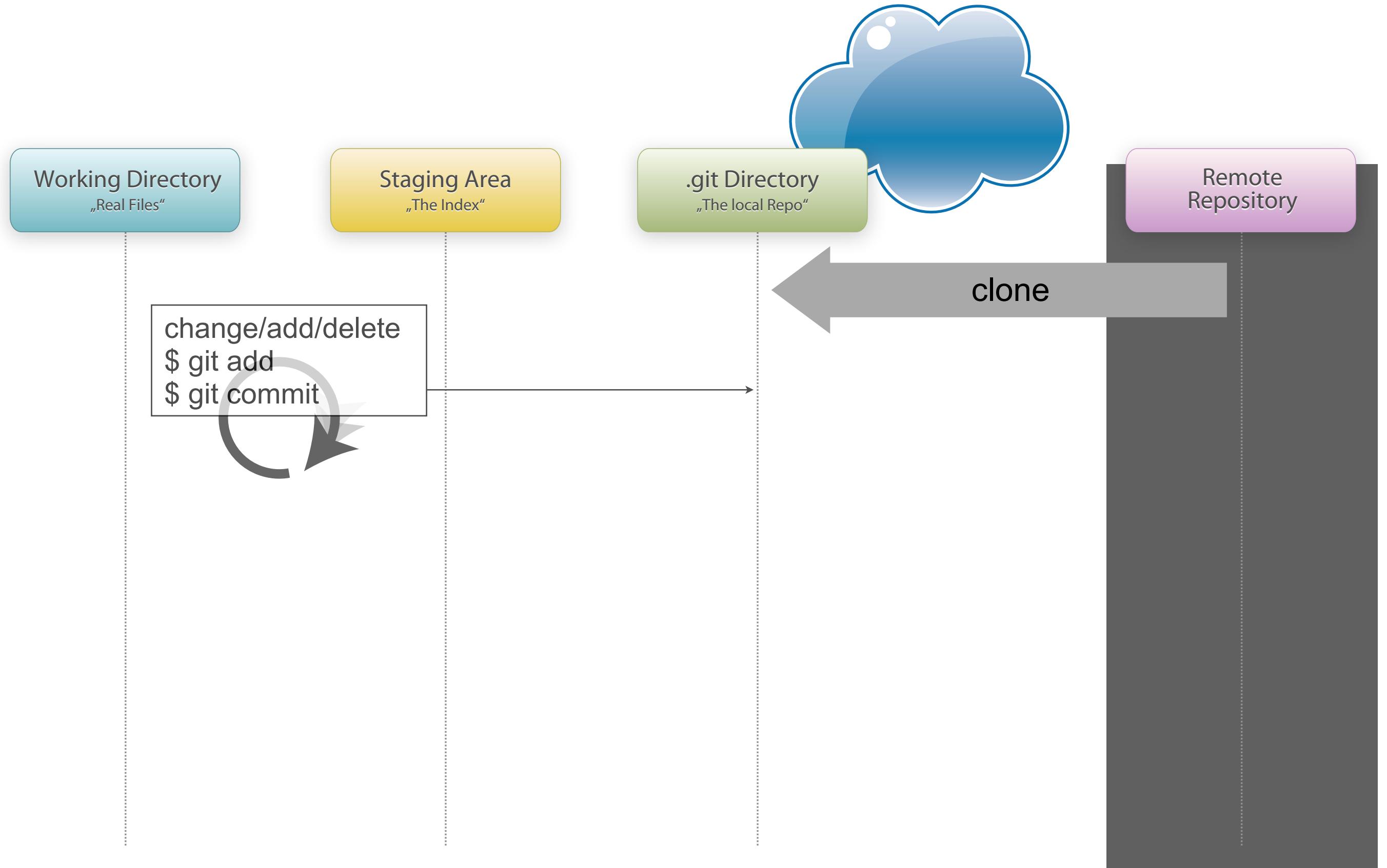


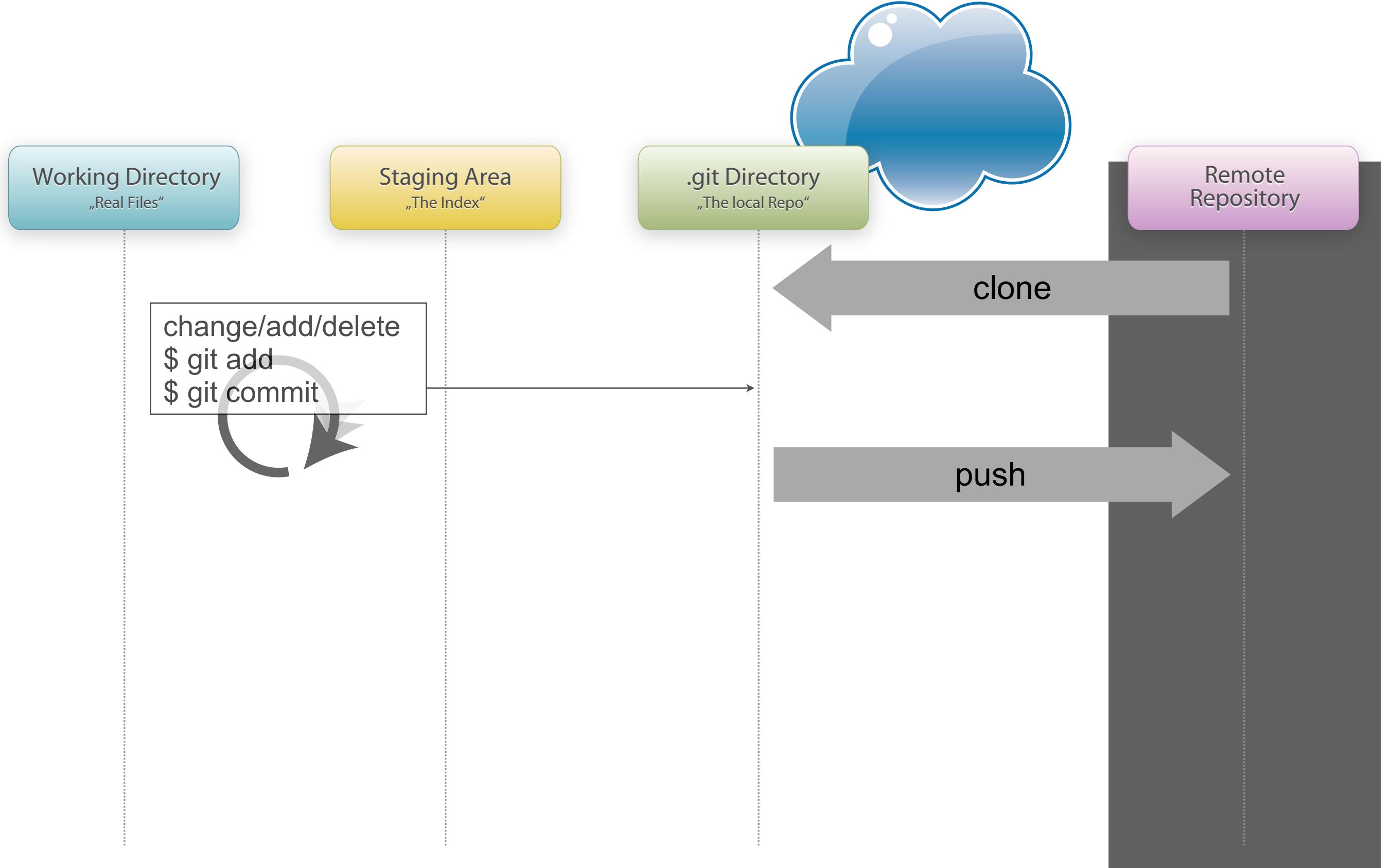
# 5. Stash

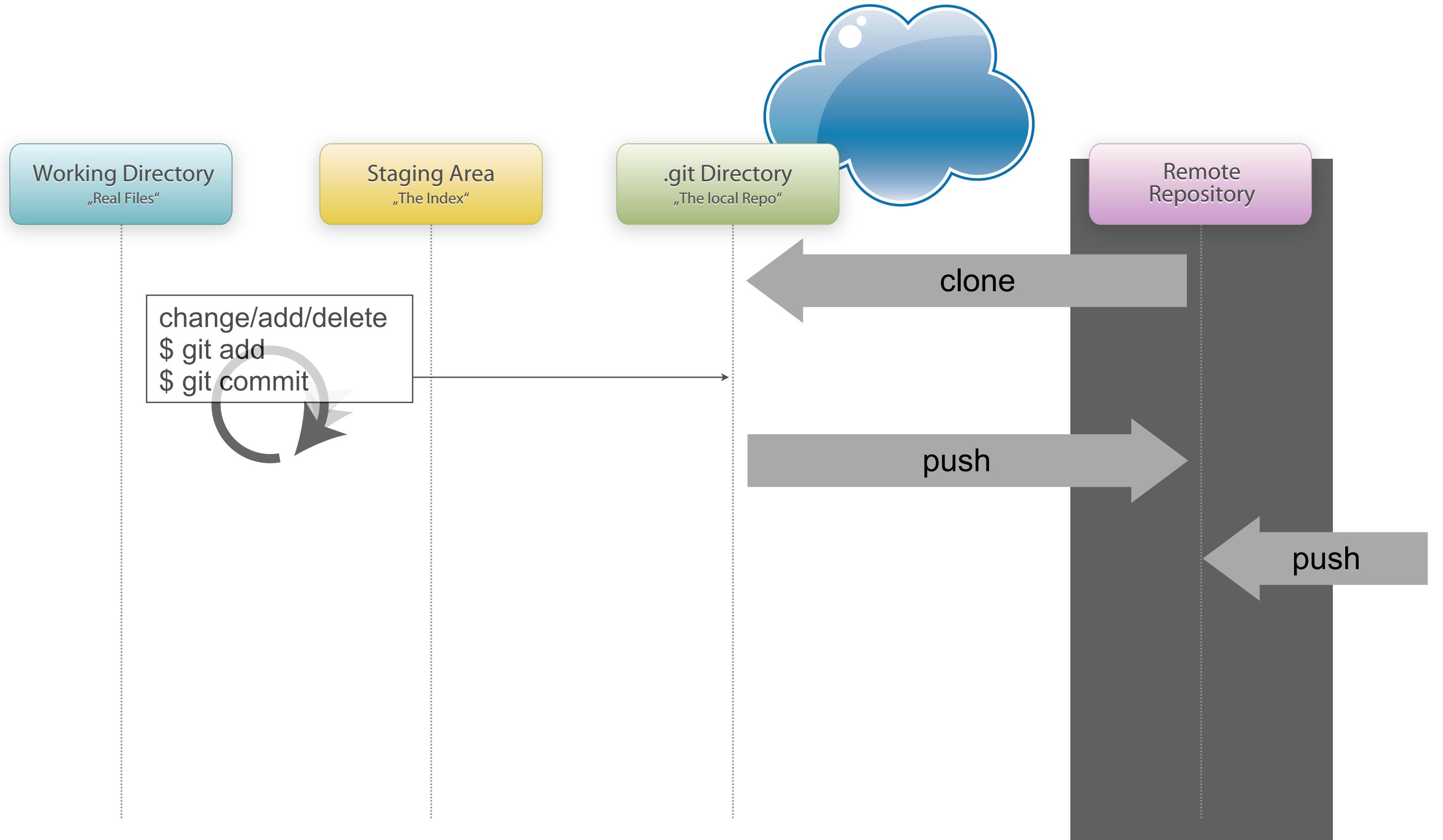


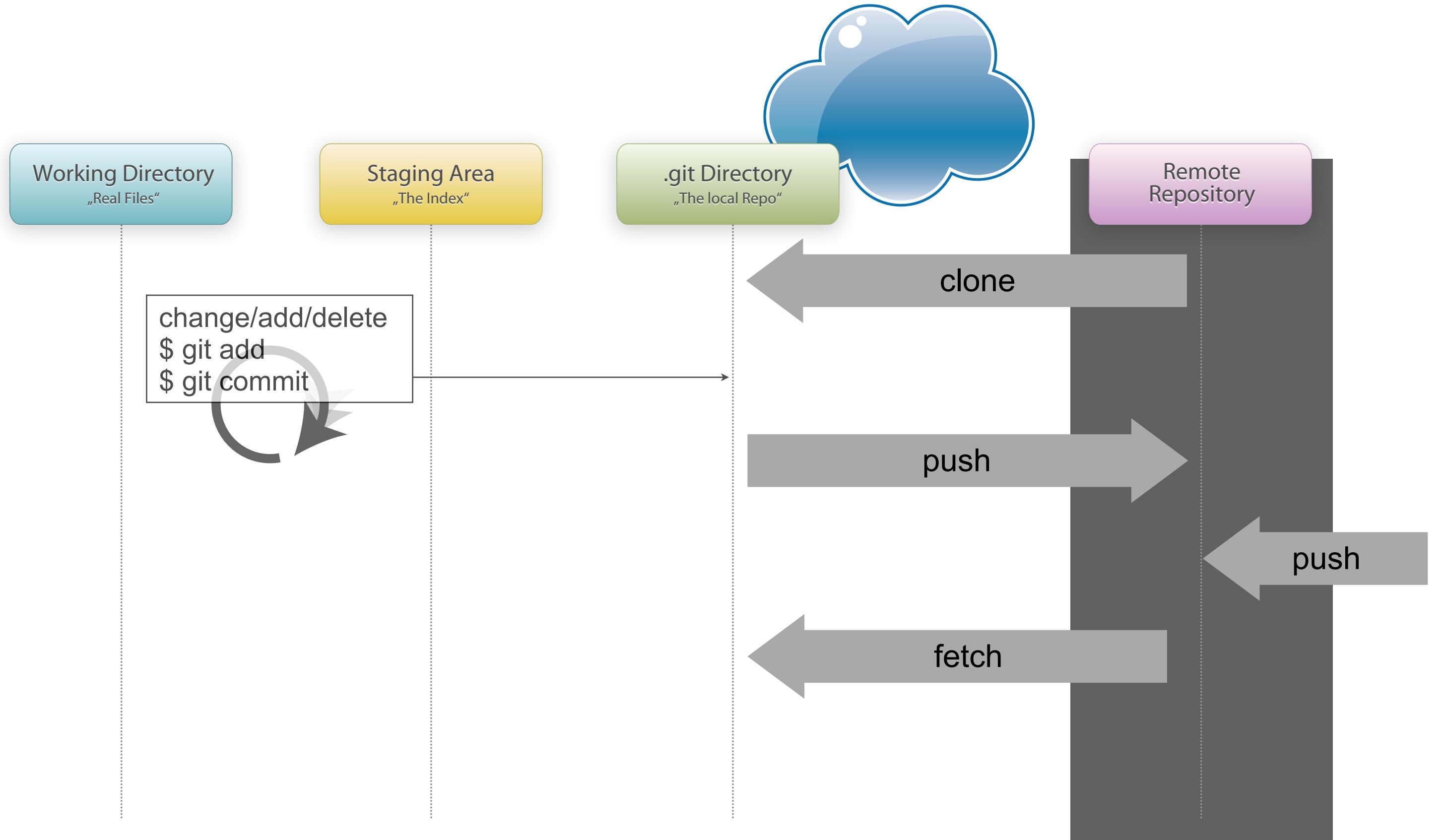
# 6. Remotes

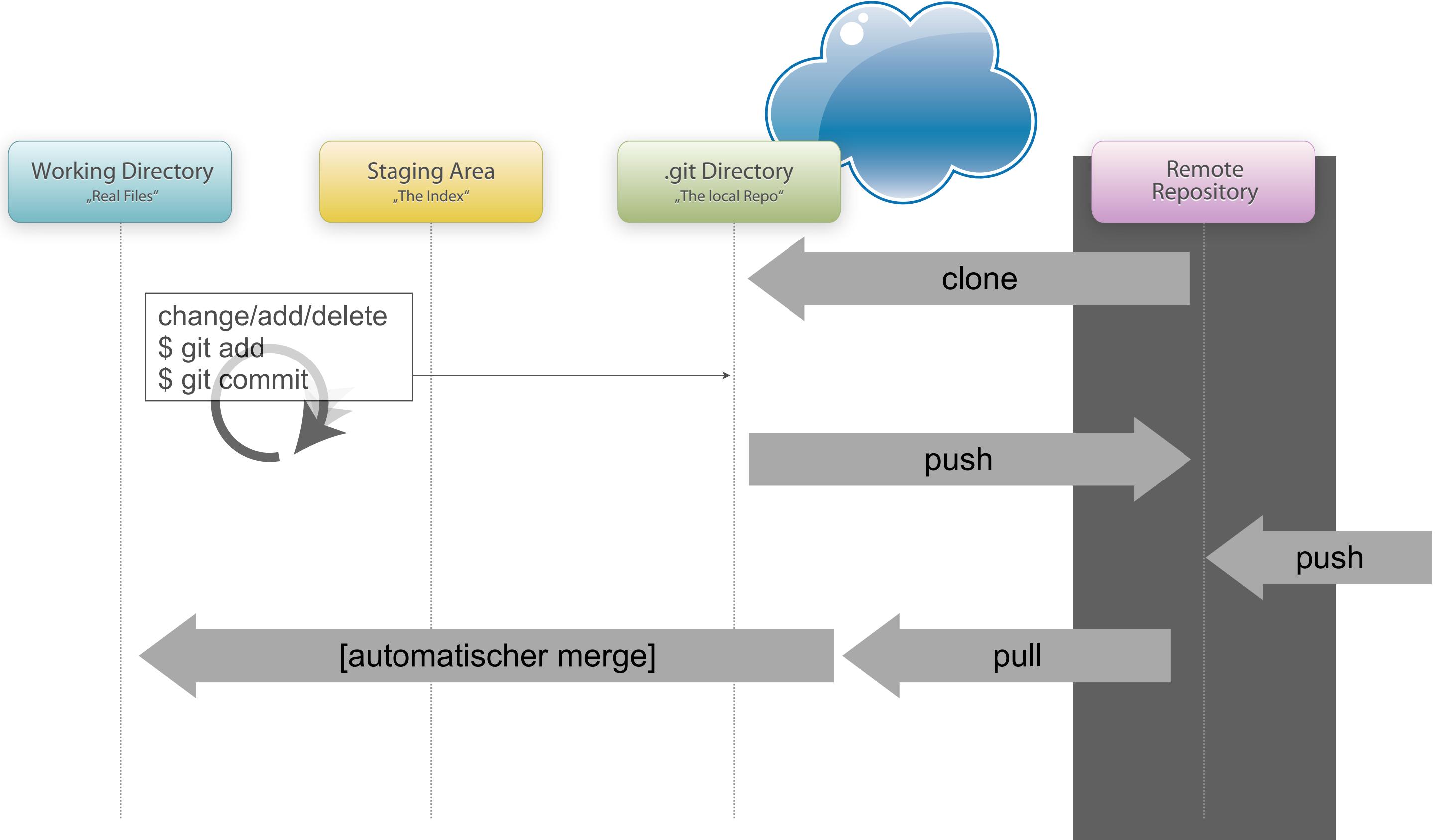












git fetch

git pull

git push



# 7. Chunk & Line Staging

Macoun-Aktion (10%)  
2. bis 10. Okt.  
Code: MACOUN10

**Vielen Dank**