Real-time Conflict Awareness for Distributed Version Control Systems

Master thesis presentation

Fabian Gremper

ETH zürich
Chair of Software Engineering

Supervisors
Martin Nordio
H.-Christian Estler
Prof. Bertrand Meyer

Presentation Overview

- Motivation
- Features
- Introduction to Git
- Setup
- Architecture
- Live demonstration
- Future Work
- Conclusion

Motivation

- *CloudStudio*
  — awareness information and conflict detection
- New implementation from ground up

Main differences

- Separate layer on top of *any* existing Git repository
- Public API
- Robust

Features

What information is interesting and useful?

- What are people working on?
(branches, files, content)
- Real-time look at other people’s work
- Are there going to be merge conflicts?
- We know that some branches are going to be merged together at some point. What are the conflicts there?

Features

Ecosystem

- User accounts
- Access control and management features
- API
- Web interface

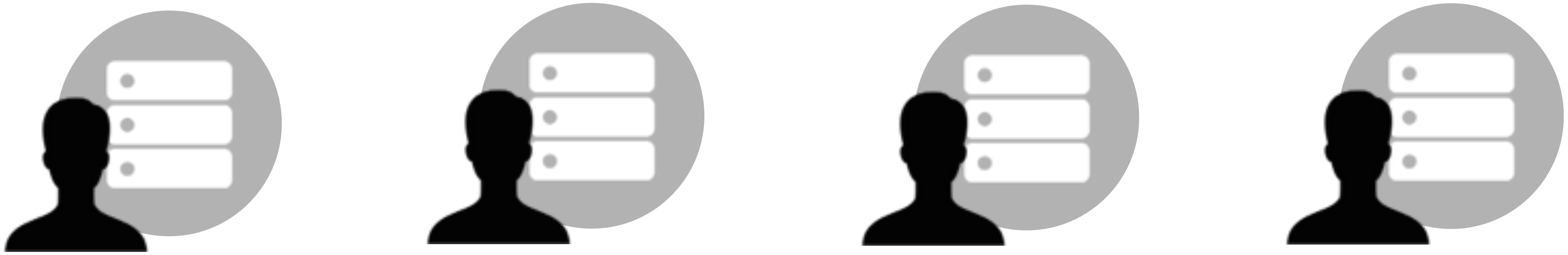
Git

Introduction

Git

Introduction

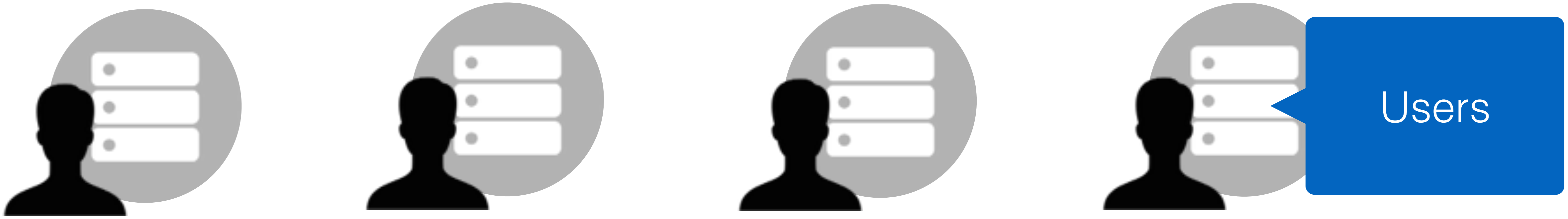
Everyone has an entire copy of the repository



Git

Introduction

Everyone has an entire copy of the repository

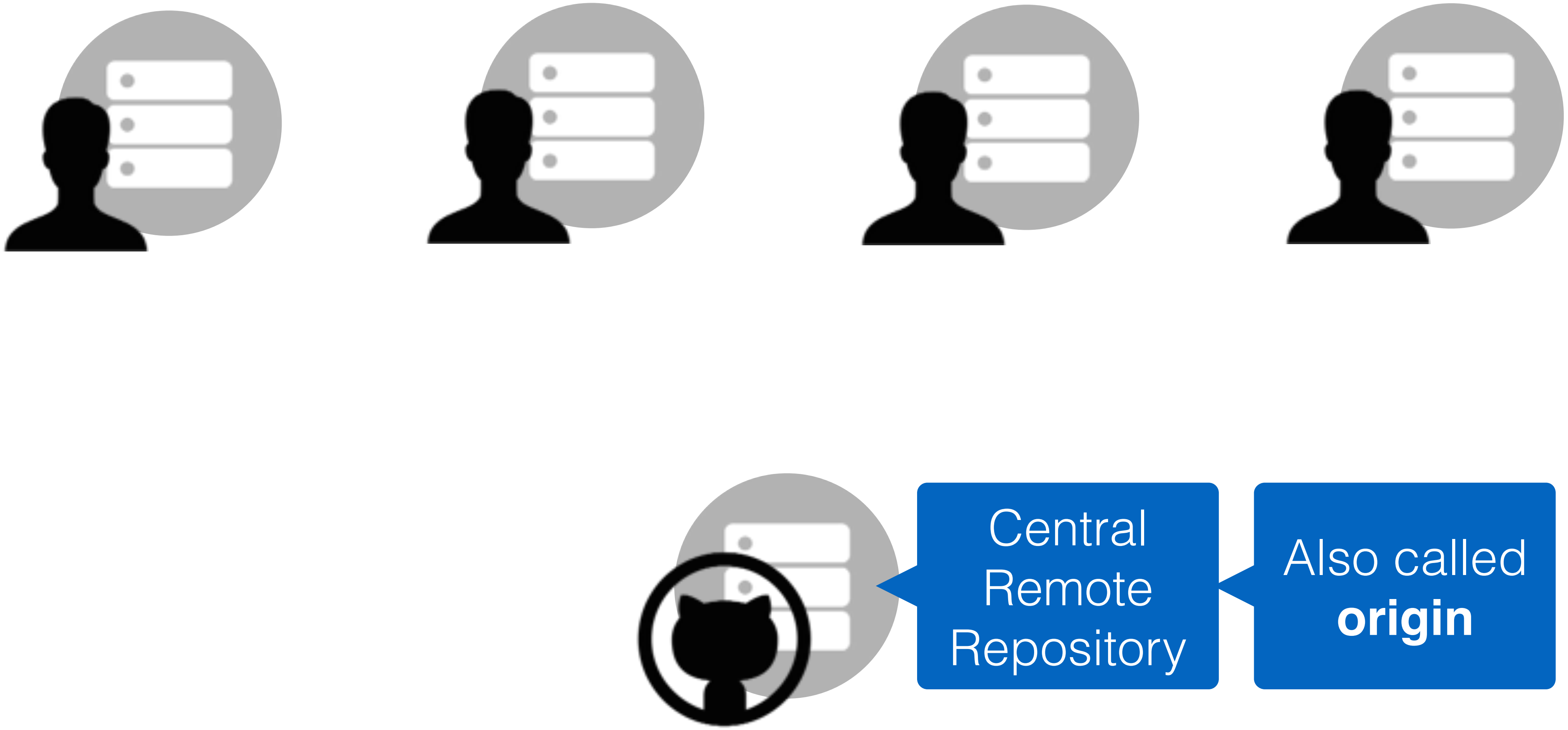


Users

Git

Introduction

A central remote repository is not different



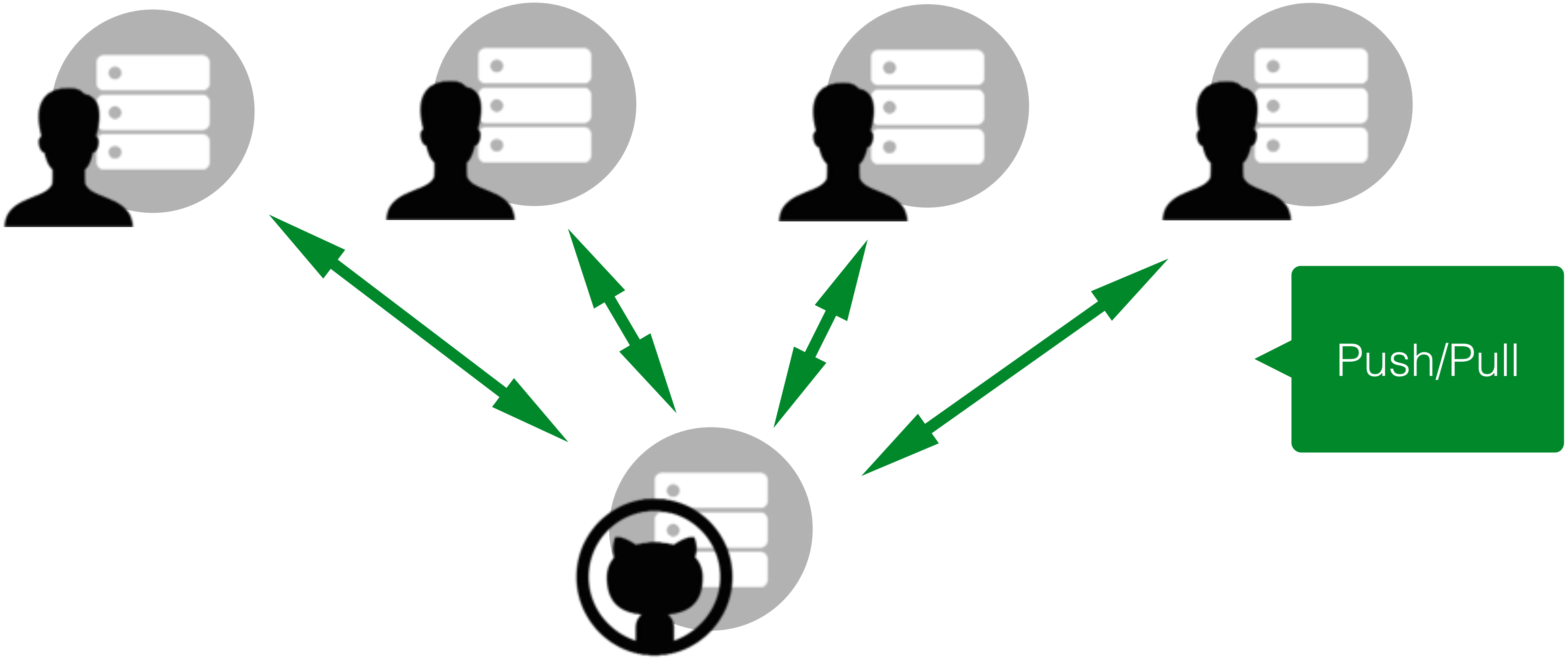
Central Remote Repository

Also called **origin**

Git

Introduction

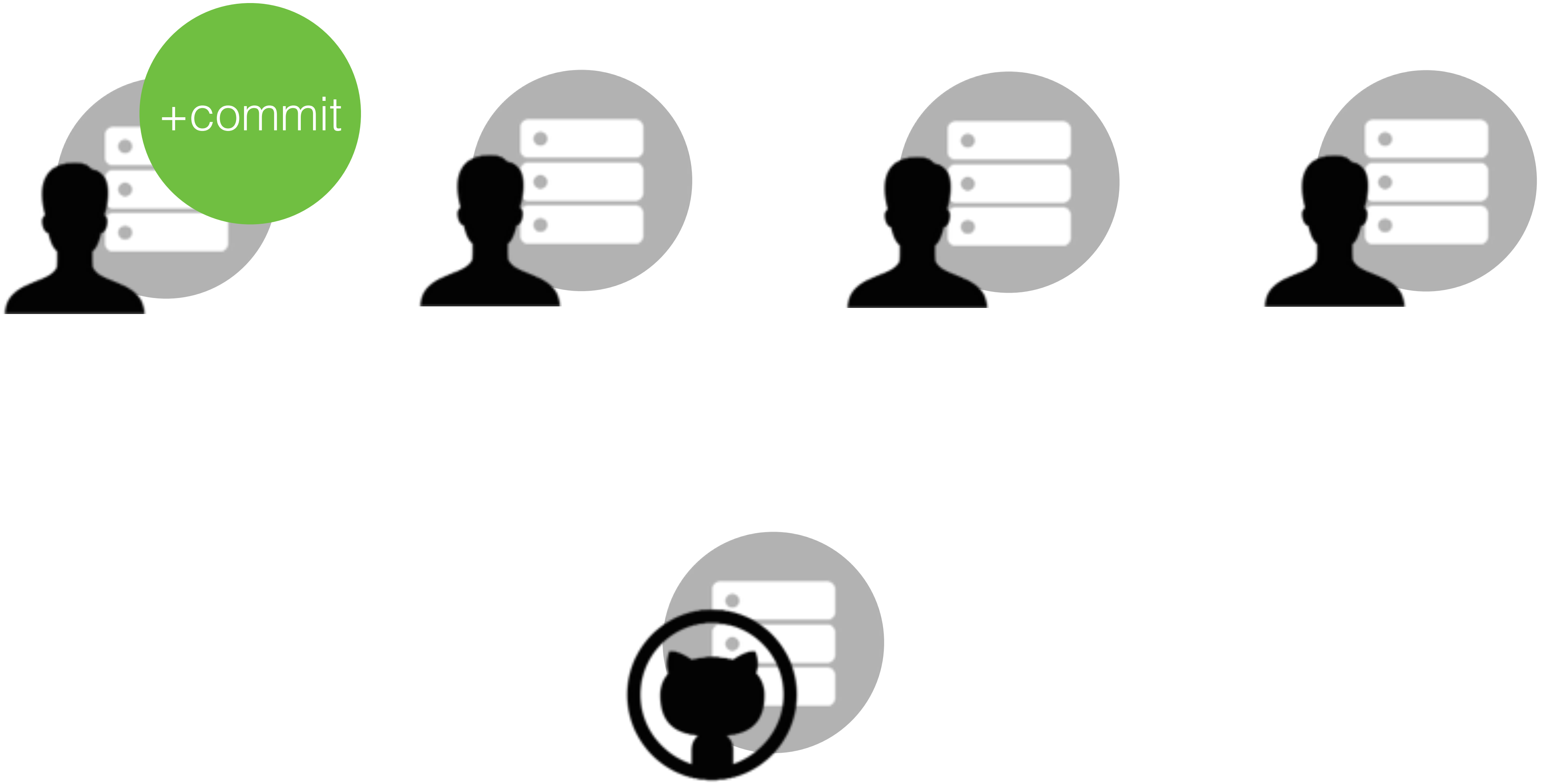
Synchronisation



Git

Introduction

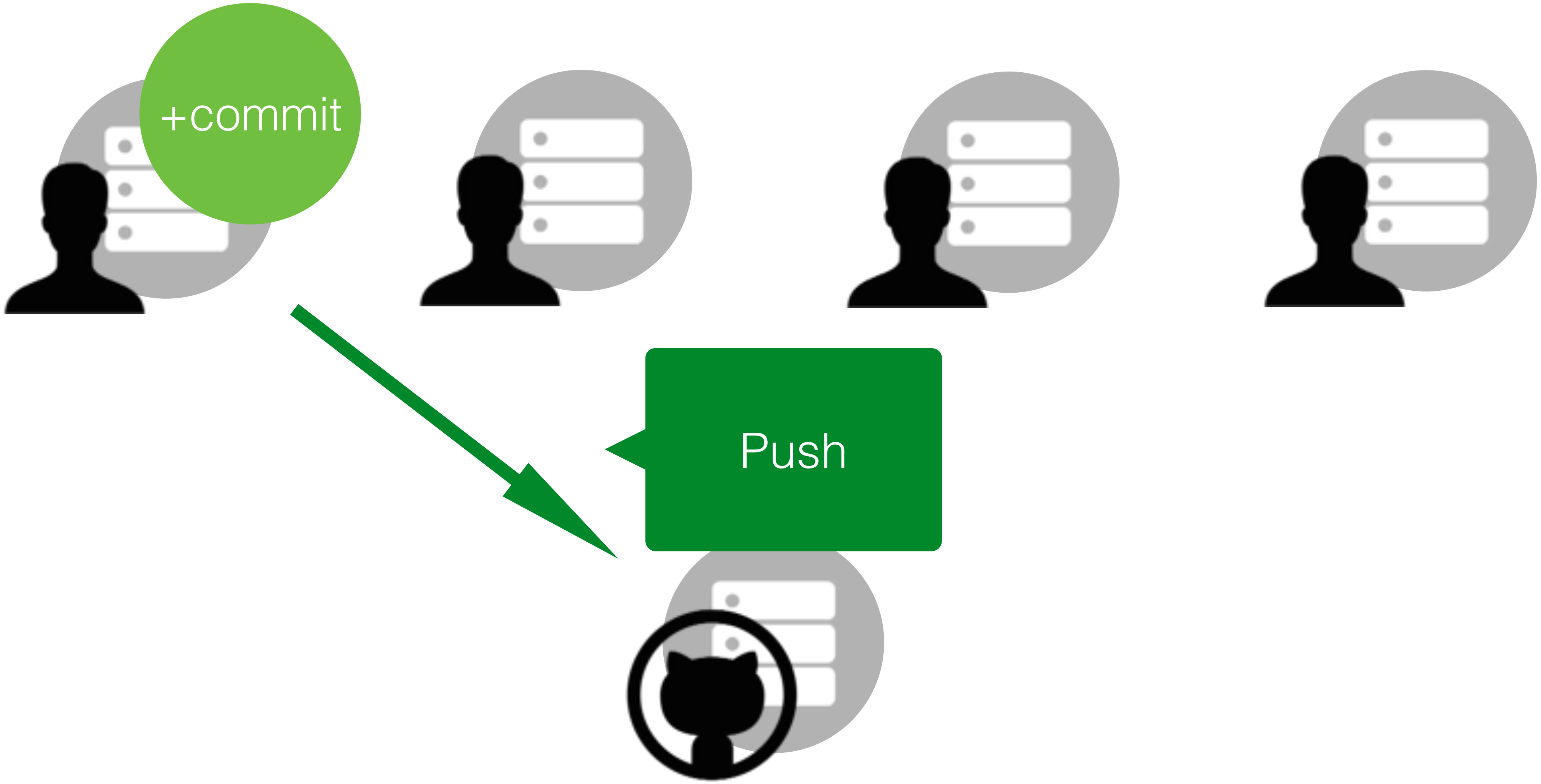
Commits are local



Git

Introduction

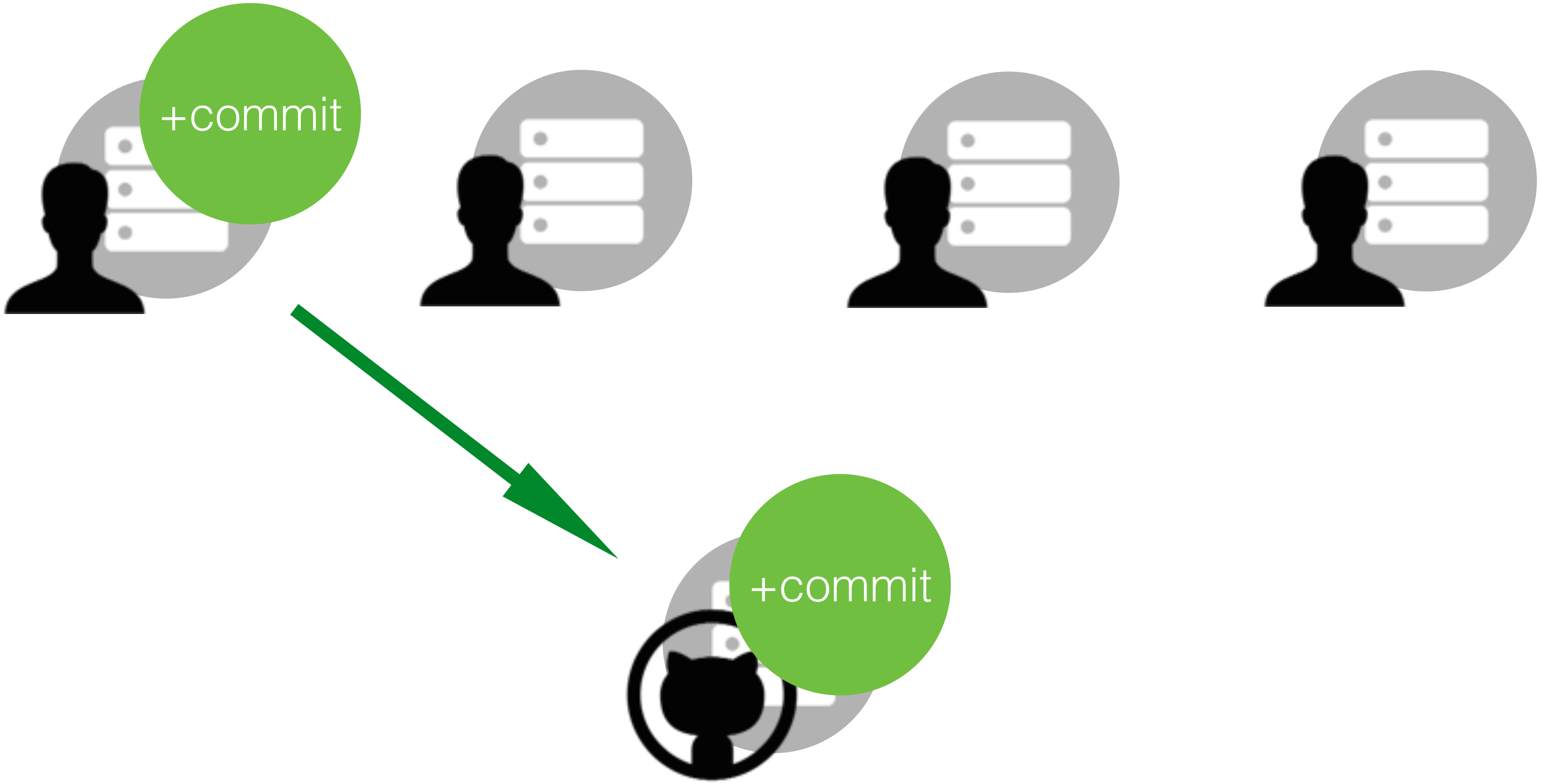
Commits are local



Git

Introduction

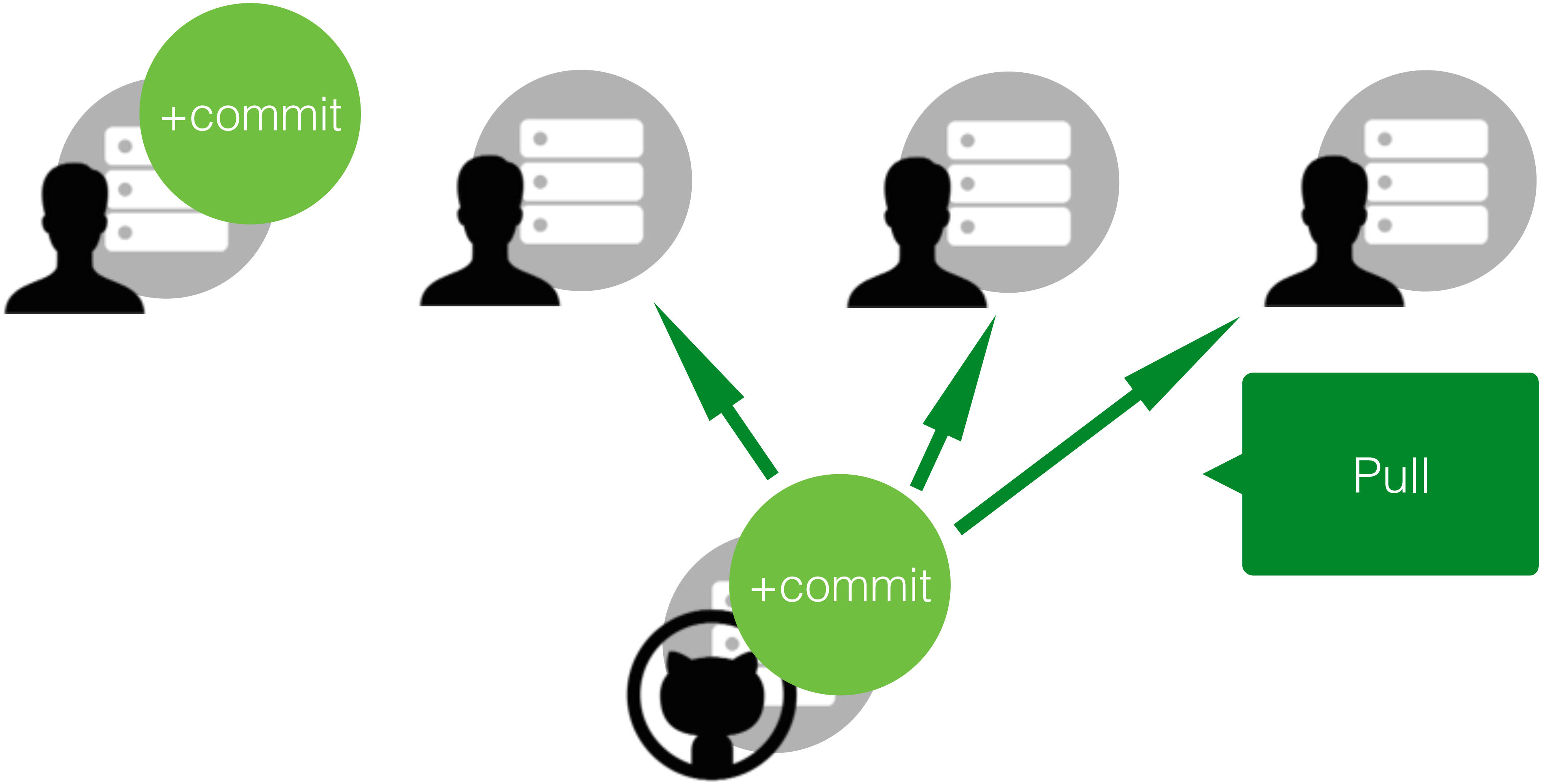
Commits are local



Git

Introduction

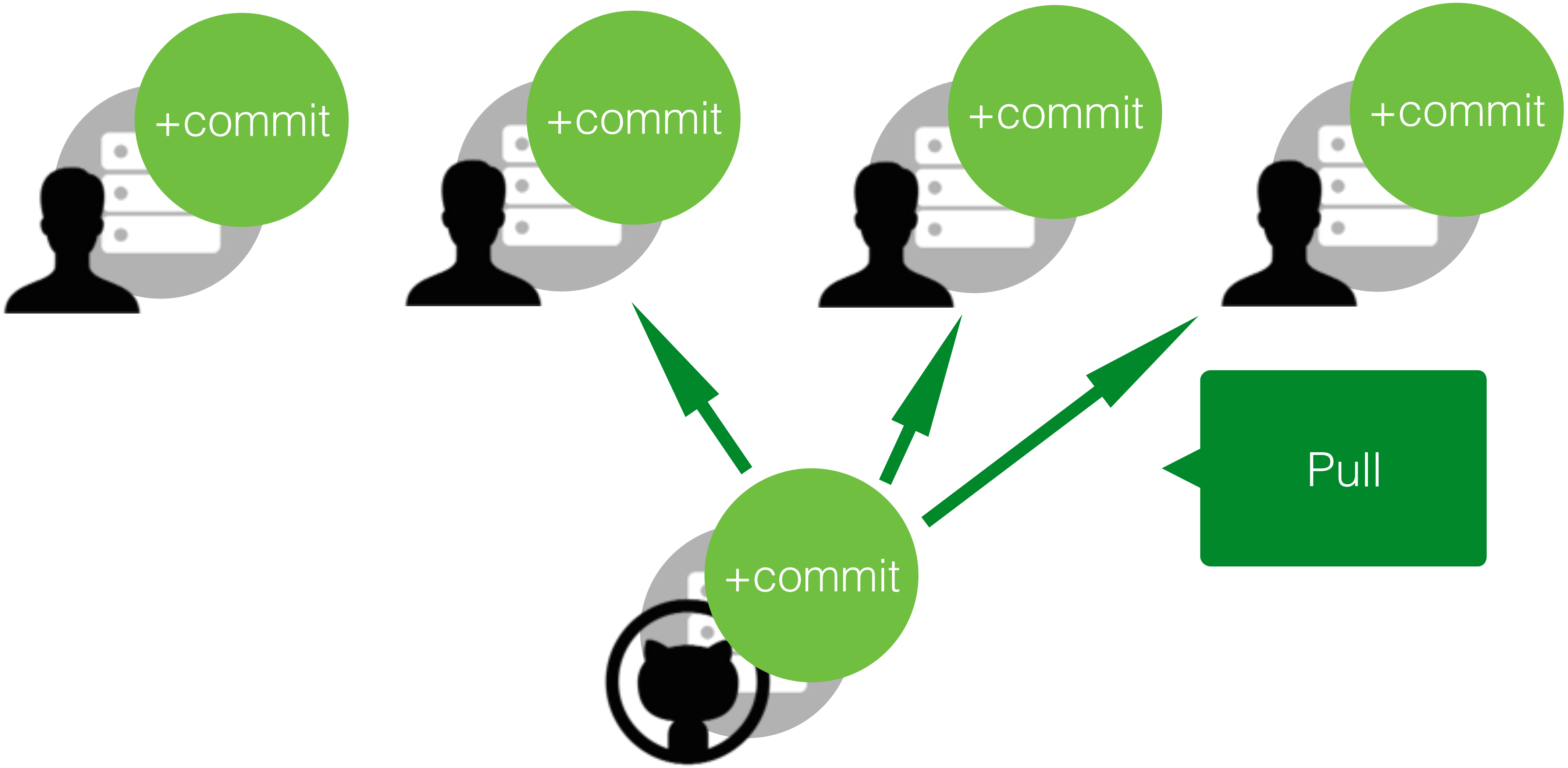
Commits are local



Git

Introduction

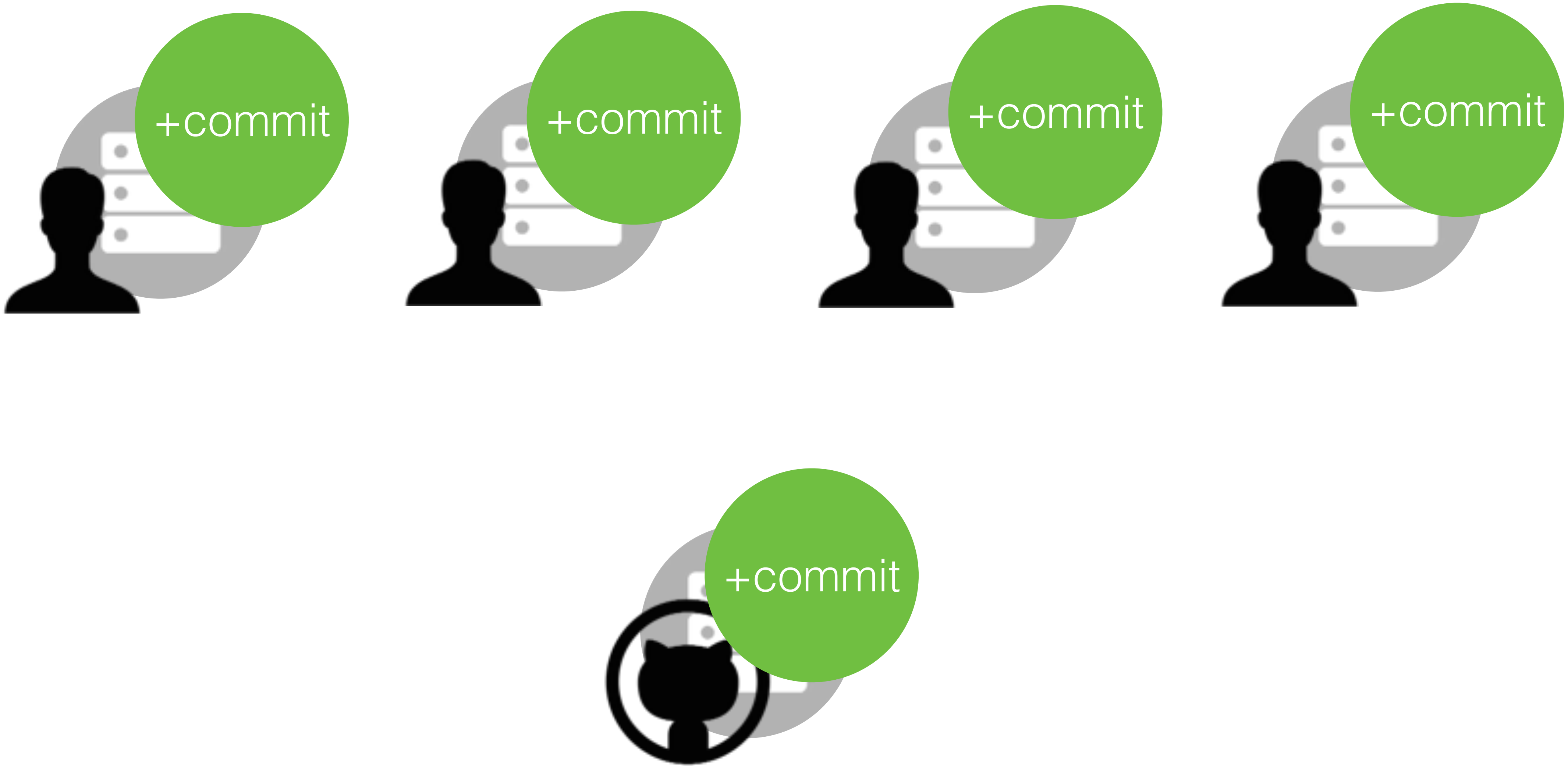
Commits are local



Git

Introduction

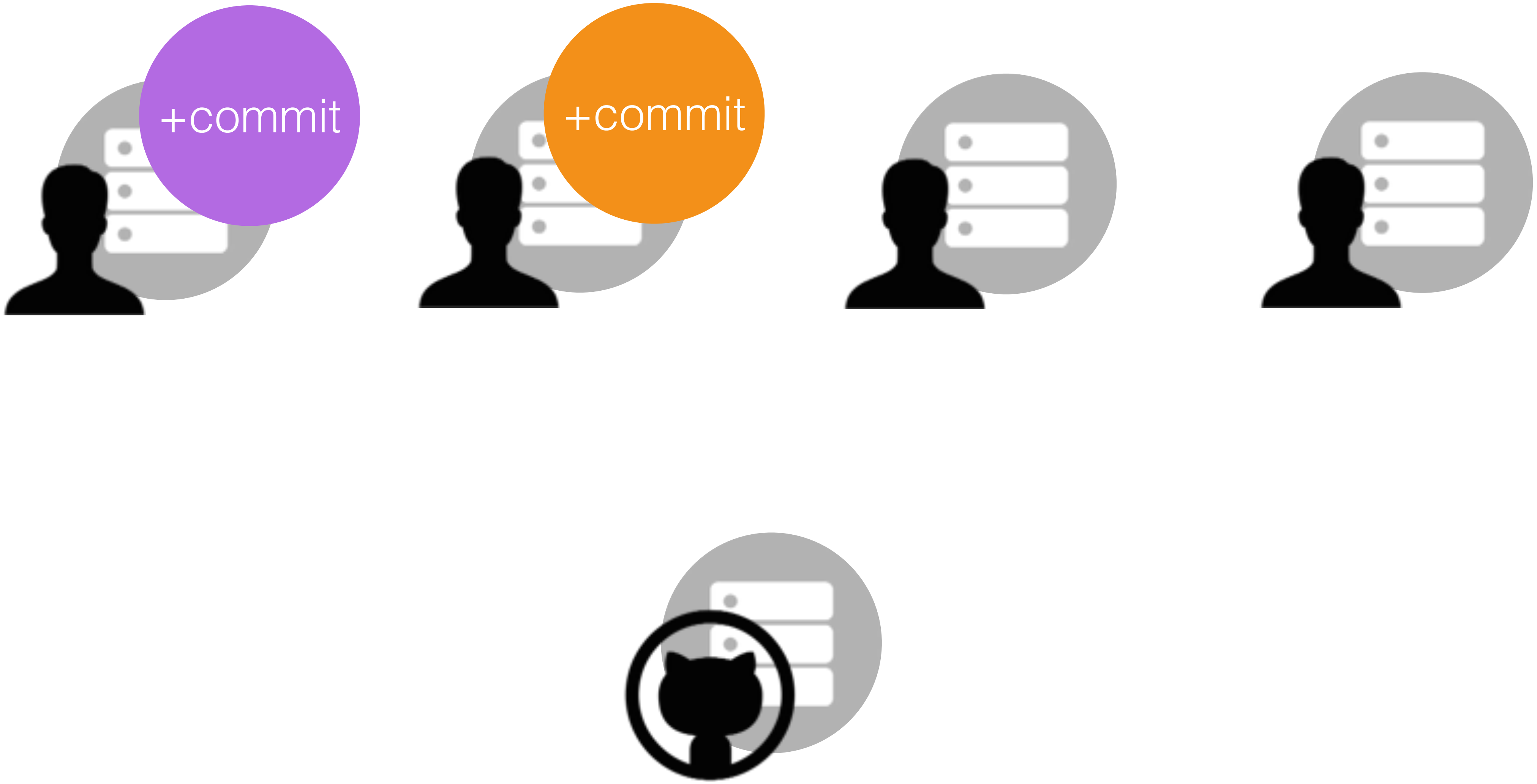
Commits are local



Git

Introduction

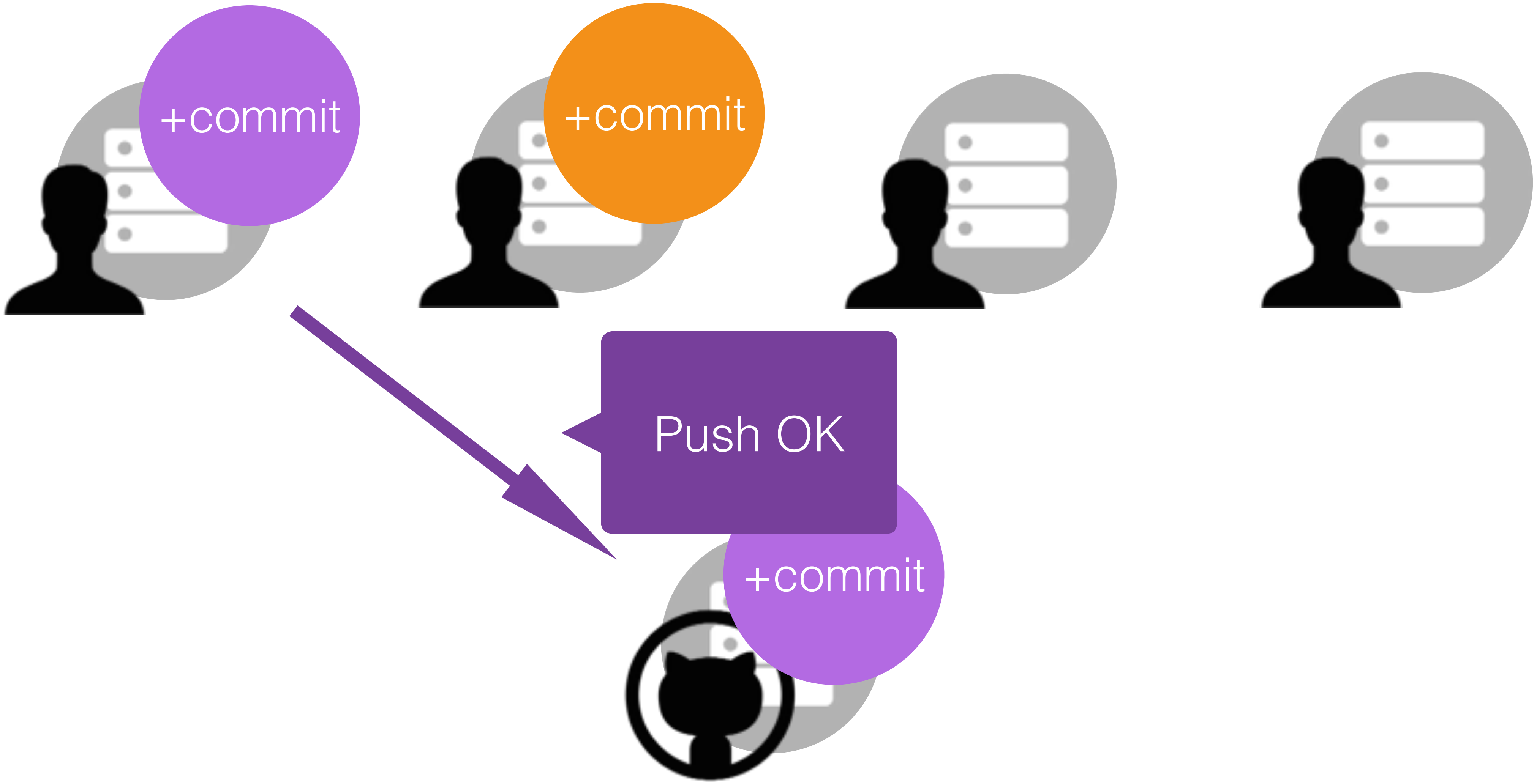
Conflicts



Git

Introduction

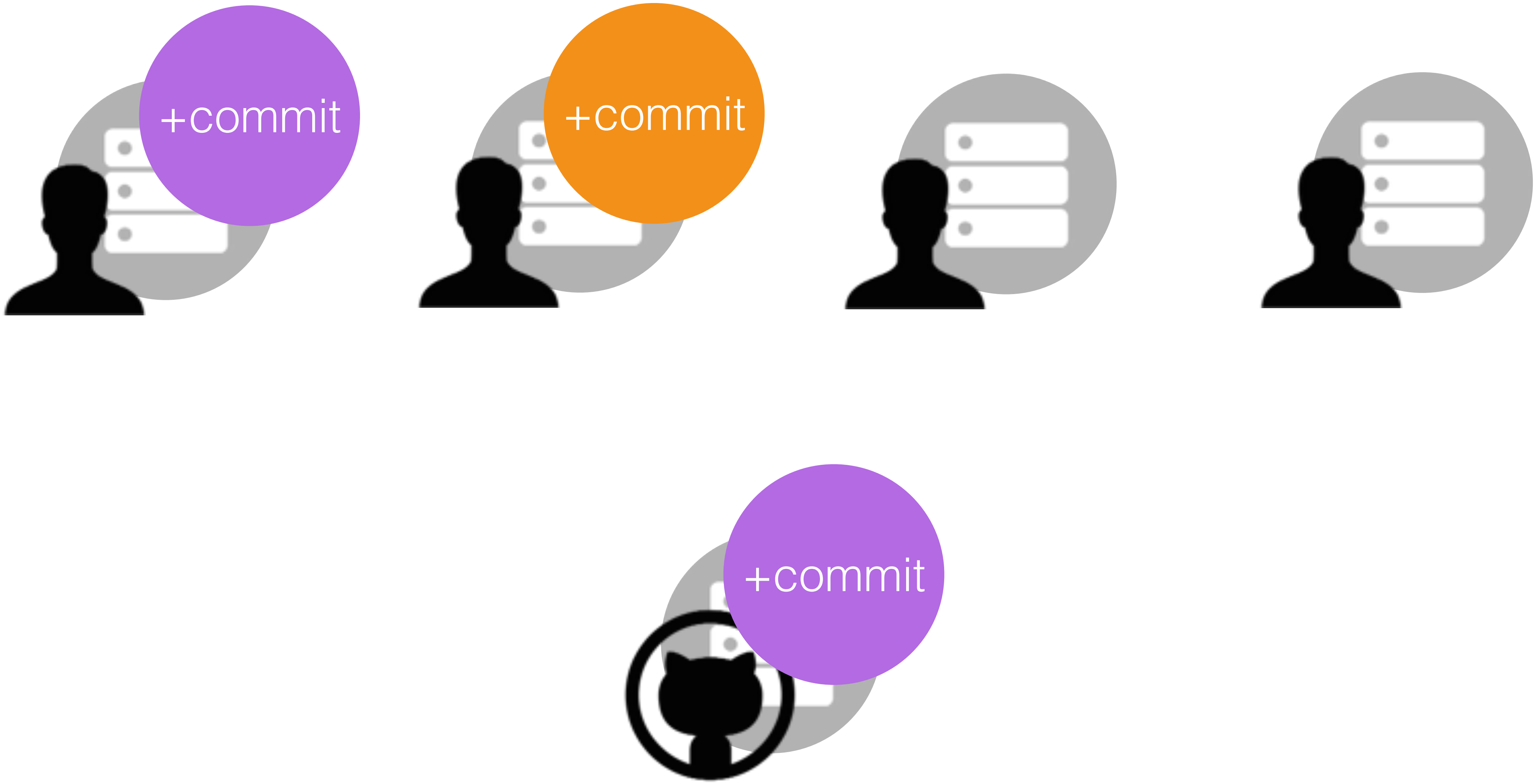
Conflicts



Git

Introduction

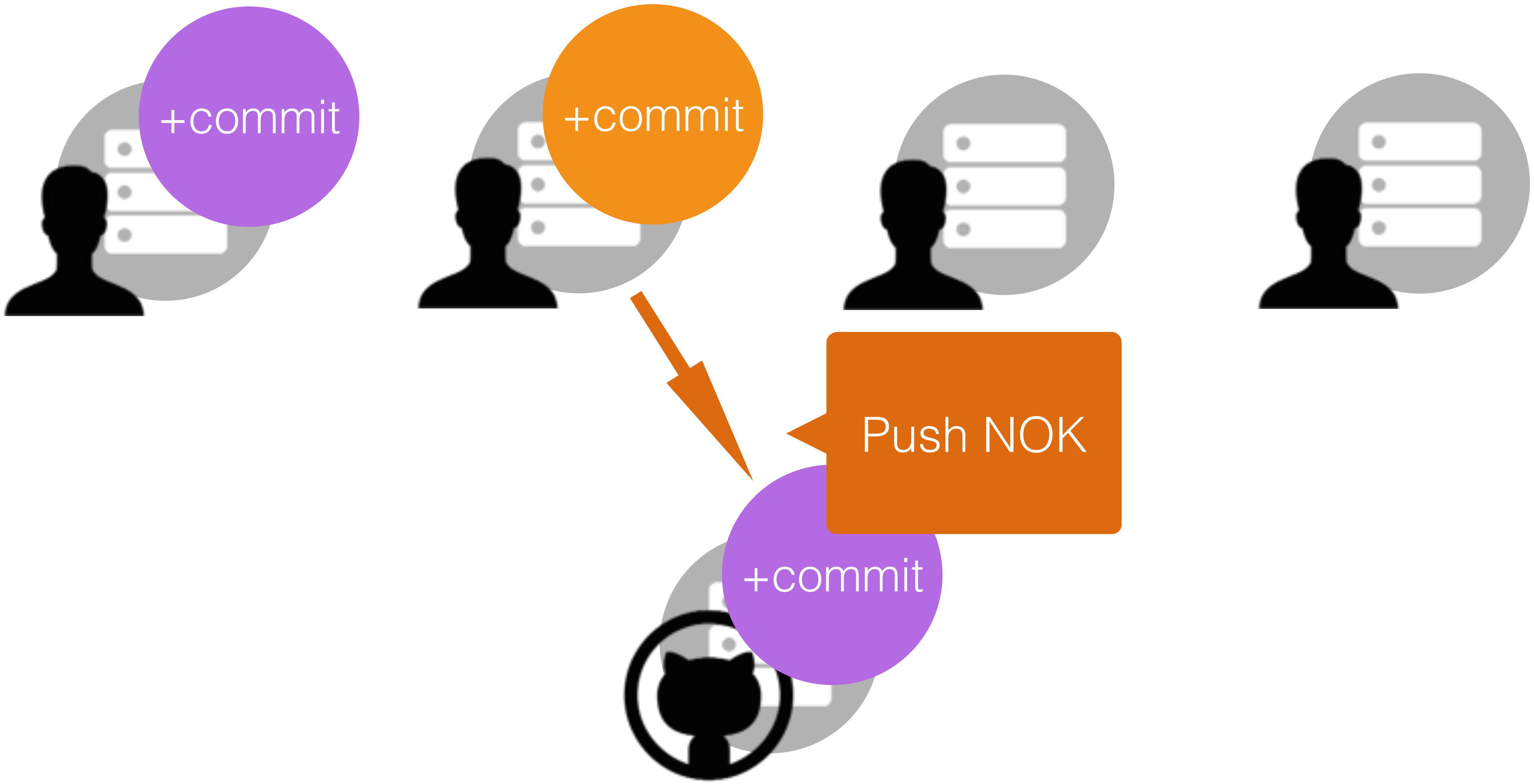
Conflicts



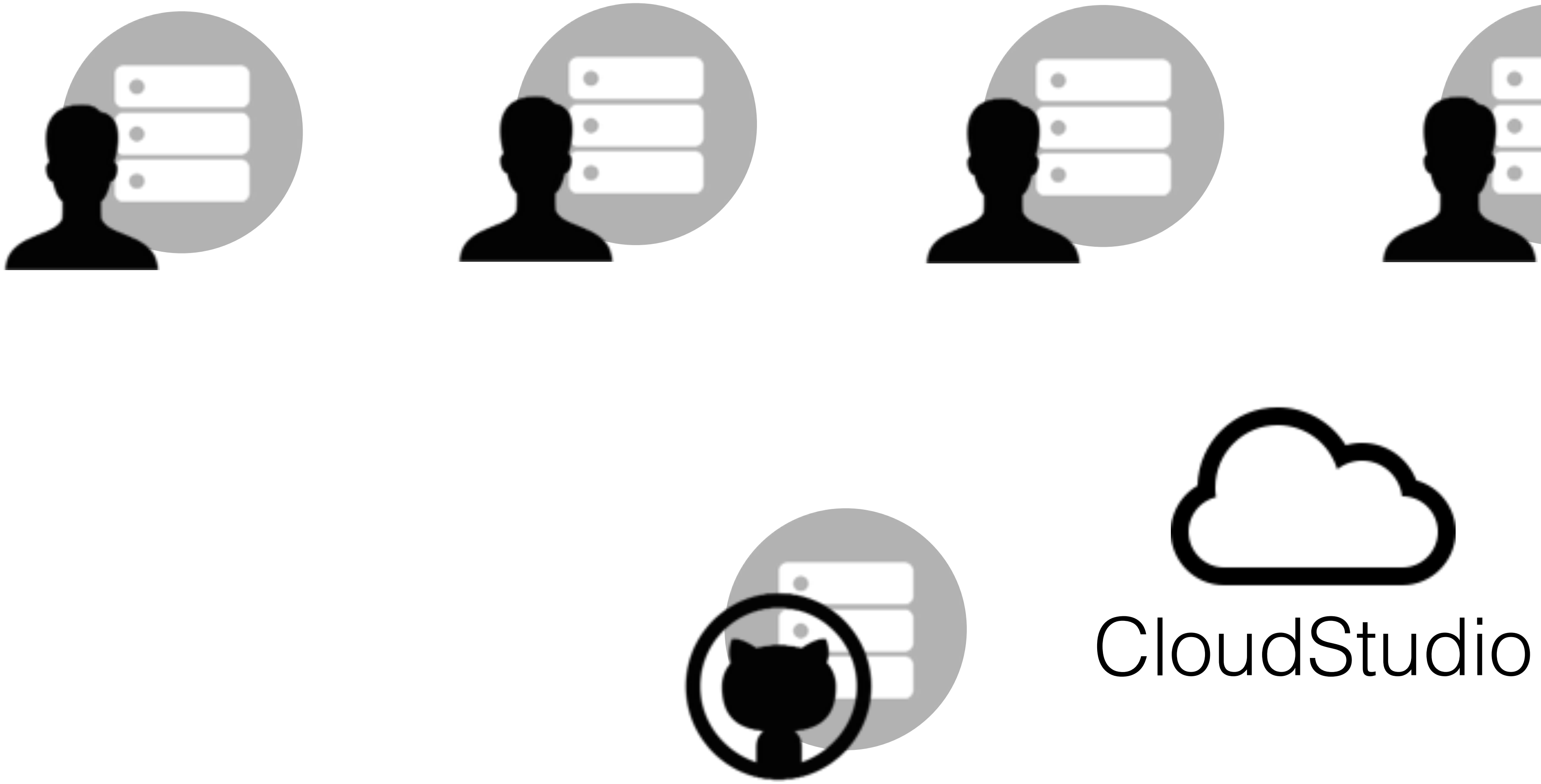
Git

Introduction

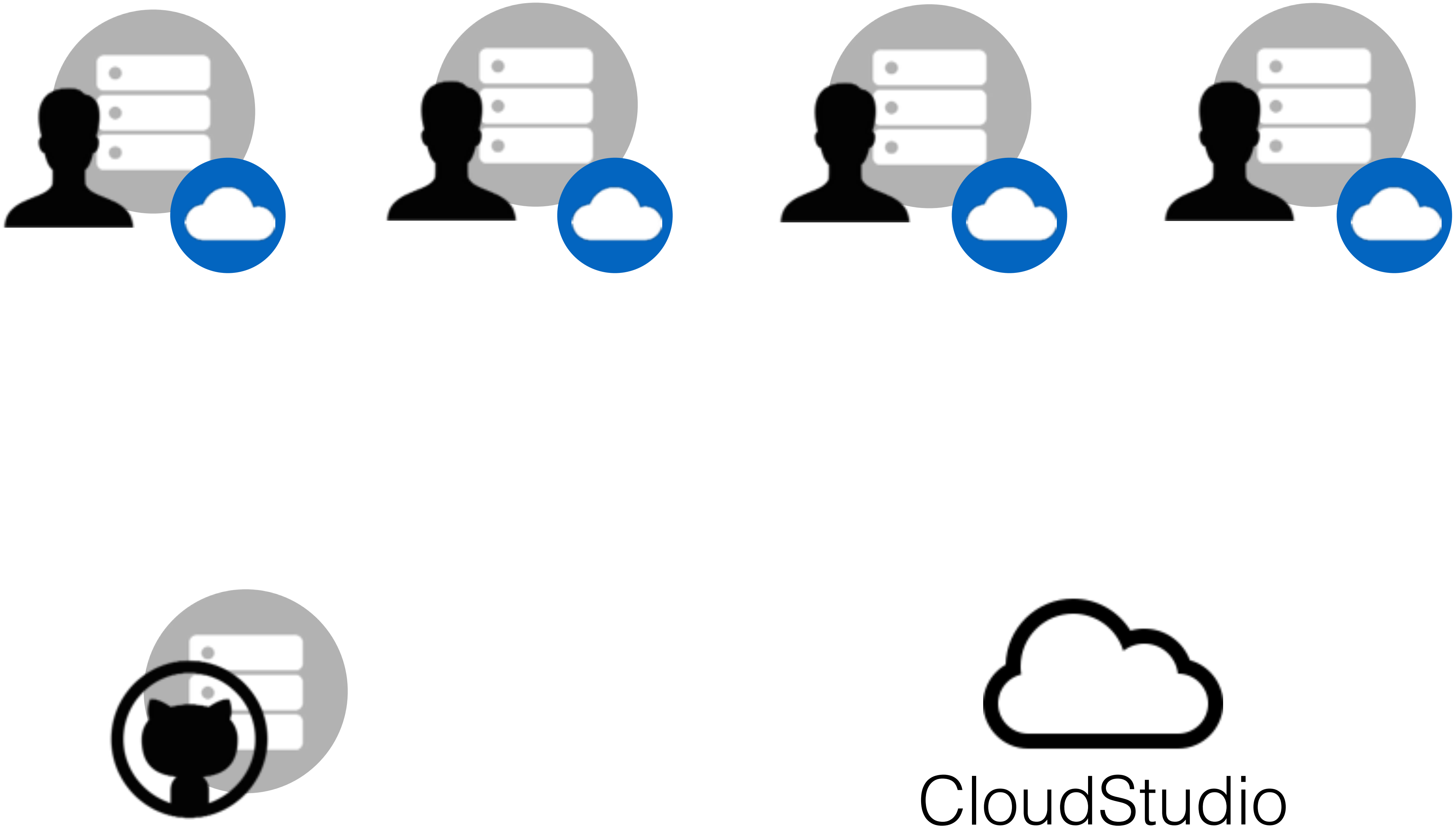
Conflicts



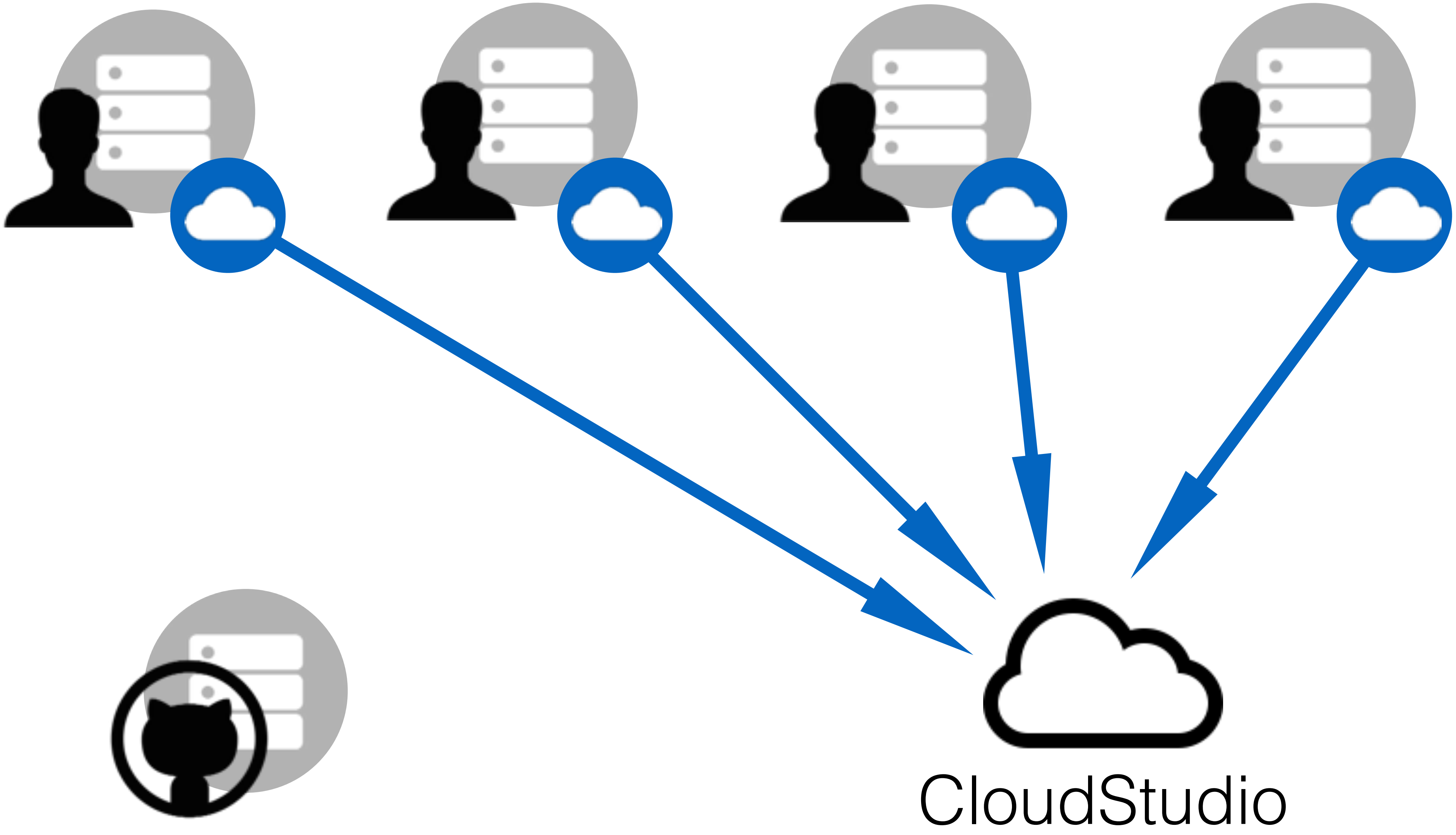
Setup



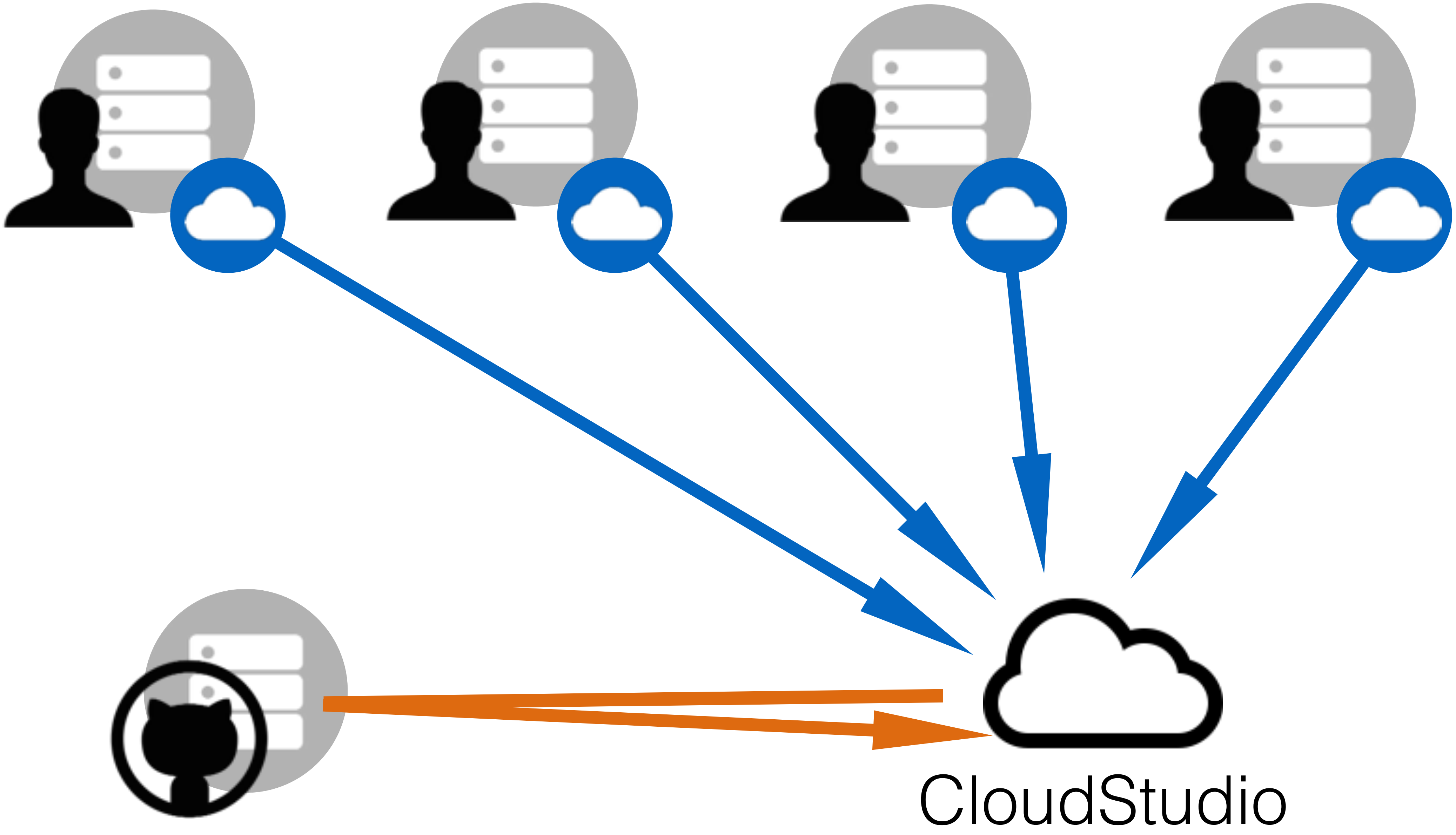
Setup



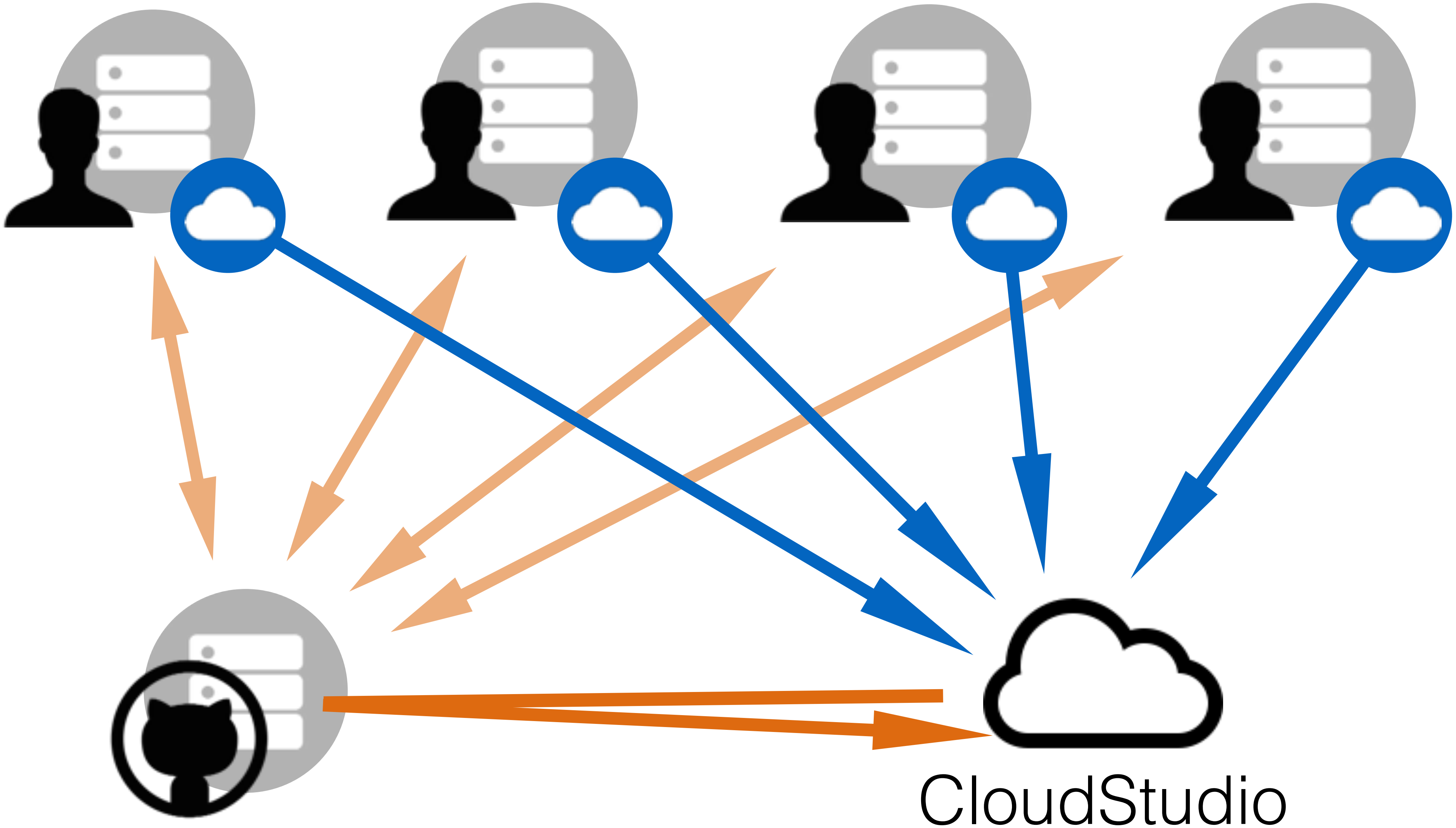
Setup



Setup



Setup



Architecture

Architecture
CloudStudio Client

Features

- Periodically read local Git repositories (JGit)
- Send updates to CloudStudio server
- Allows multiple repositories
- Configuration
- GUI
- Error handling

Architecture
CloudStudio Server

I/O

- Accepts data from clients
- Requests data from *central remote repositories*
- Provides API
- Provides web interface

Architecture
CloudStudio Server

Logic

- User, session and repository management
- Stores information from all users in SQL database
  - Can be used to calculate entire commit graph
  - File comparison via checksum
  - Content comparison via diff/diff3 algorithms
  - Find merge base for conflict detection (JGit)

Architecture
CloudStudio Server

Additional features

- Configuration
- Robustness
- Error handling

Architecture
CloudStudio Web Interface

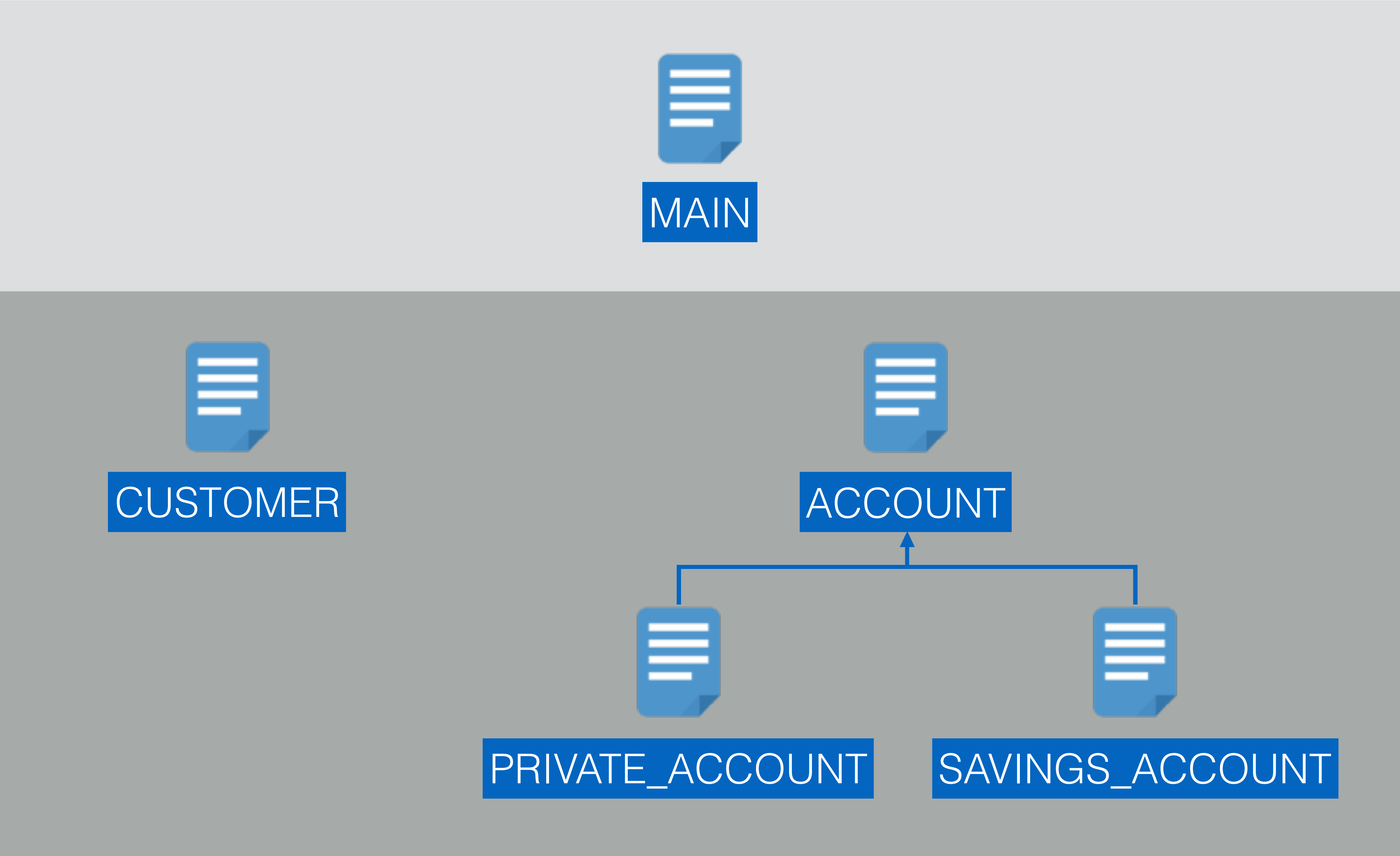
Features

- Directly uses the API
- Provides access to awareness information
- User and repository management
- Views
  - Branch level awareness
  - File level awareness
  - Content level awareness

Demo

Demo project

BankAccountDemo



MAIN

CUSTOMER

ACCOUNT

PRIVATE_ACCOUNT

SAVINGS_ACCOUNT

Part 1
Setup

Part 2

Change Awareness

6

Martin

EDIT

CUSTOMER

+ middle name setter

```
set_middle_name(a_middle_name: STRING)
        -- sets the middle name
    do
        middle_name := a_middle_name
    end
```

Christian

EDIT

CUSTOMER

+ age assignment

```
age := a_age
```

Martin and Christian **make changes** to CUSTOMER

7

Martin

COMMIT
CUSTOMER

+ middle name setter

```
set_middle_name(a_middle_name: STRING)
        -- sets the middle name
    do
        middle_name := a_middle_name
    end
```

Christian

COMMIT
CUSTOMER

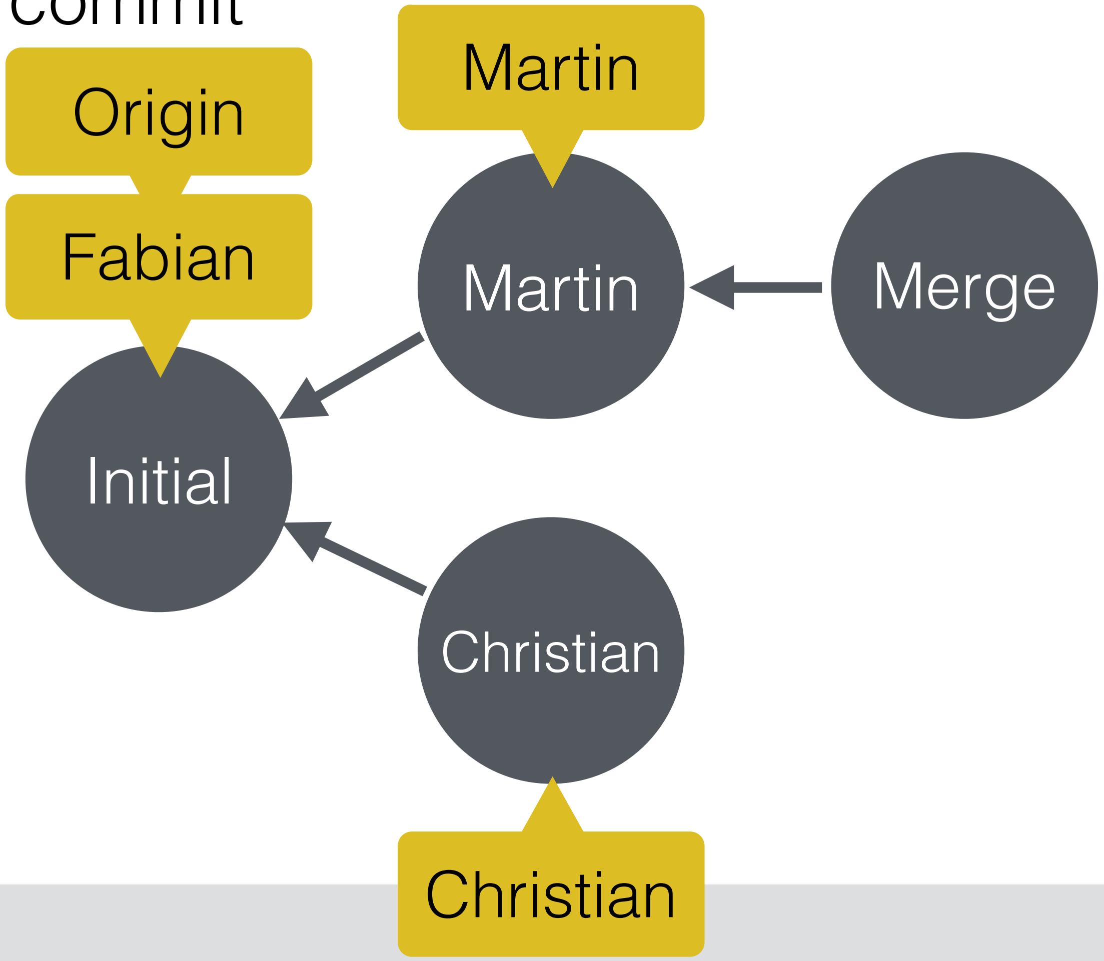
+ age assignment

```
age := a_age
```

Martin and Christian **commit their changes locally**

8

Martin	PUSH	
Christian	PULL	→ Creates merge commit
Christian	PUSH	
Martin	PULL	



Martin and Christian **synchronise their changes**

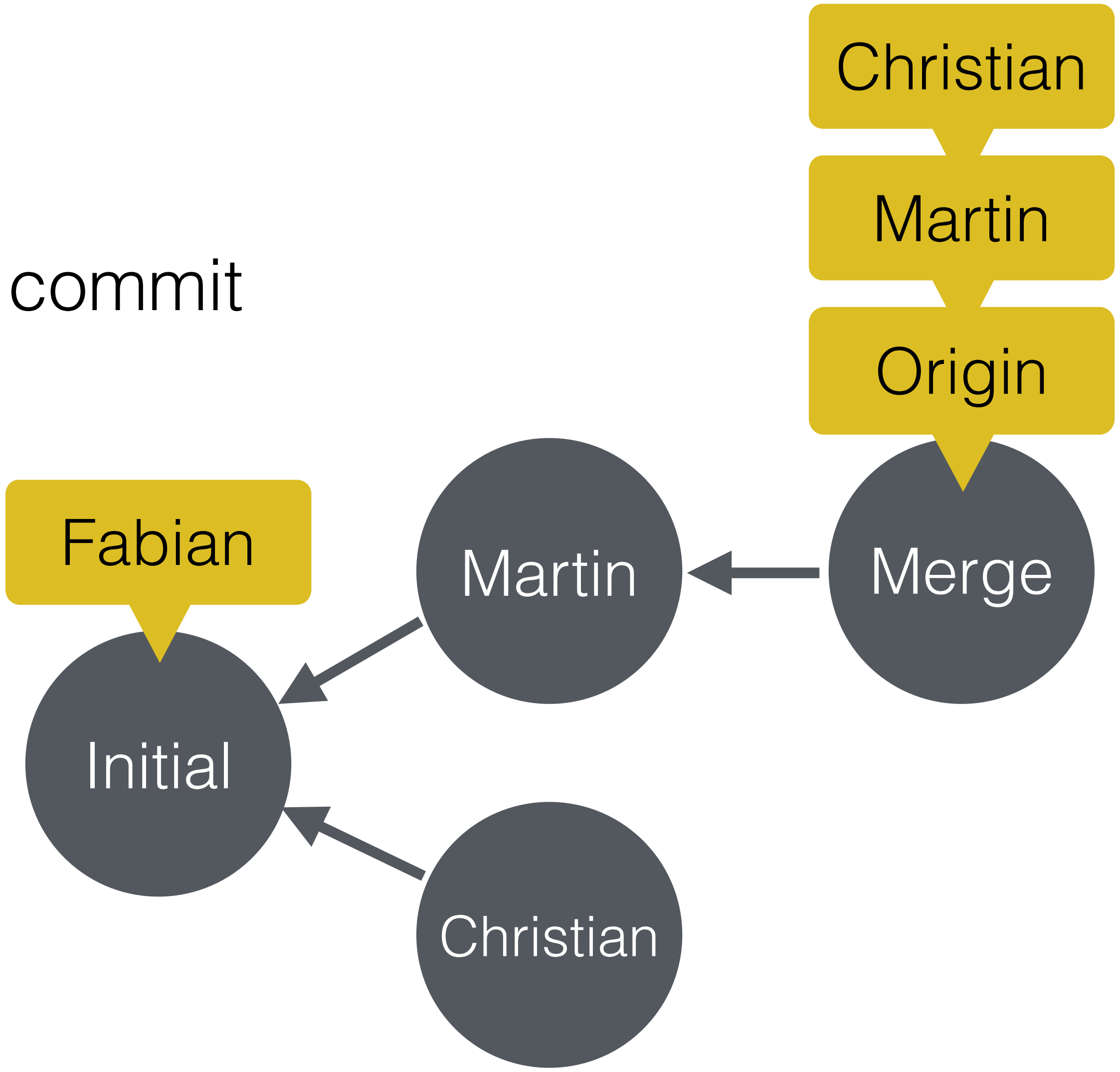
8

Martin PUSH

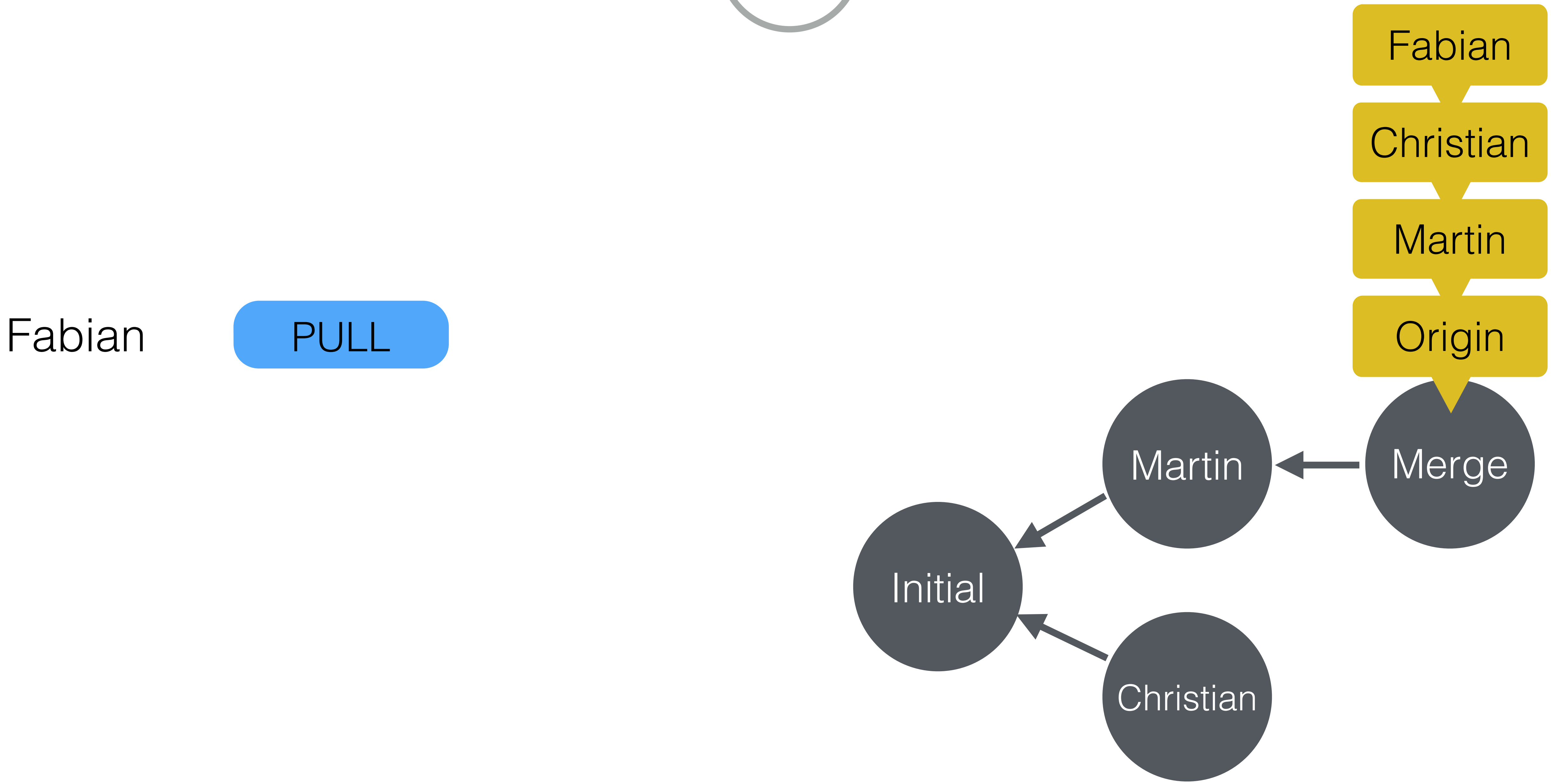
Christian PULL → Creates merge commit

Christian PUSH

Martin PULL



Martin and Christian **synchronise their changes**



Fabian **updates** to the latest commit

Part 3

Conflict Detection

10

Fabian

EDIT

SAVINGS_ACCOUNT

change interest rate

```
balance := balance * 1.04
```

Martin

EDIT

SAVINGS_ACCOUNT

change interest rate

```
balance := balance * 1.03
```

Fabian and Martin **make changes** to SAVINGS_ACCOUNT

11

Fabian

COMMIT

SAVINGS_ACCOUNT

change interest rate

```
balance := balance * 1.04
```

Martin

COMMIT

SAVINGS_ACCOUNT

change interest rate

```
balance := balance * 1.03
```

Fabian and Martin **commit their changes locally**

12

Martin PUSH

Fabian PULL → Auto merge failed

→ **Manually resolve merge** (and then commit and push)

Fabian and Martin **try to synchronise**

13

Martin PULL

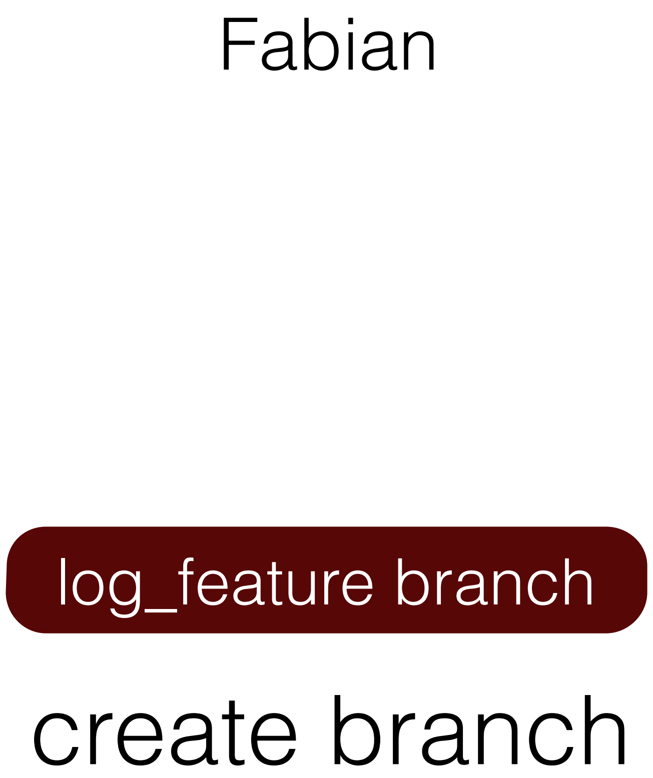
Christian PULL

Martin and Christian **update** to the latest commit

Part 4

Branching

14



Fabian

log_feature branch

create branch

Fabian **creates and checks out a new branch**

15

Fabian

EDIT

ACCOUNT

log_feature branch

implement *print_log*

```
print("Printing log ...%N”)
from
    log.start
until
    log.after
loop
    print(log.item)
    log.forth
end
```

Fabian **implements print_log** in log_feature branch

15

Fabian

EDIT

ACCOUNT

log_feature branch

implement *print_log*

```
print("Printing log ...%N”)
from
    log.start
until
    log.after
loop
    print(log.item)
    log.forth
end
```

Martin

EDIT

ACCOUNT

master branch

implement *print_log*

```
print("Start of log")
-- working on it...
print("End of log")
```

Fabian **implements print_log** in master branch

Try it out for yourself!

http://cloudstudio.ethz.ch:7330/

Documentation

https://github.com/fgremper/CloudStudio

Future Work

- Improve performance
- Write plugins and tools that use the API
- Incremental updates
- Heuristics, e.g. for file renames
- DOSE

Conclusions

- Distributed engineering: awareness is needed
- CloudStudio: new software solution proposal for awareness information and conflict detection
- Requirements set from the start and tightly followed
- Interesting, but challenging
- DOSE

Questions?