Google Developer Student Clubs

Podstawy Git’a

WARSZTAT

Krzysztof Kaczyński
GDSC Lead

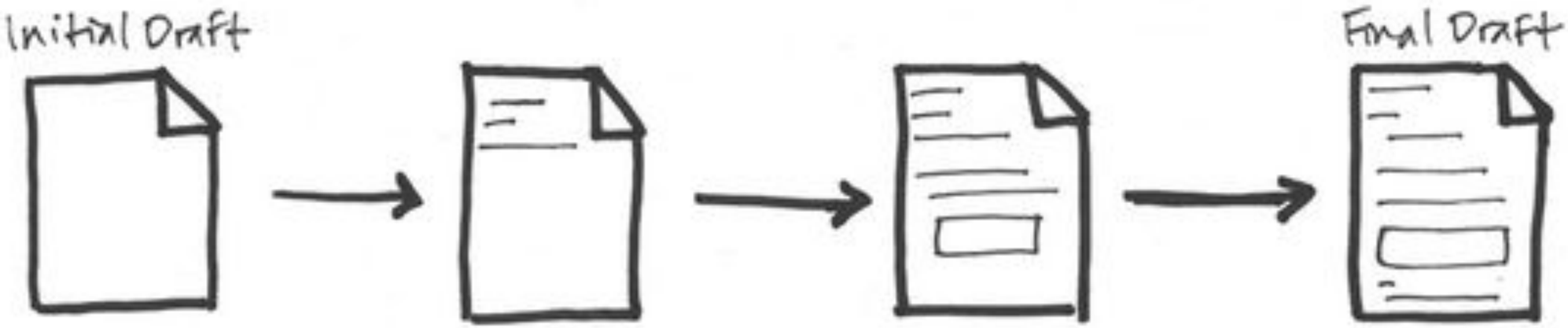
Agenda

- Czym jest Git i co to jest repozytorium,
- Czym jest GitHub,
- Kontrolowanie stanu,
- Przeglądanie historii
- Gałęzie,
- Przywracanie i przenoszenie zmian,
- Łączenie zmian i rozwiązywanie konfliktów,
- Co to jest PullRequest i Fork

Czym jest Git i co to jest repozytorium

`Rozproszony system kontroli wersji`

- **Historii zmian** dokonanych w projekcie (kto i kiedy wprowadził zmianę),
- **Rozgałęzianie projektu**, tak żeby wiele osób mogło w tym samym czasie pracować nad różnymi zmianami,
- Doskonałe narzędzie do **pracy zespołowej**,
- Możliwość pracy **offline**

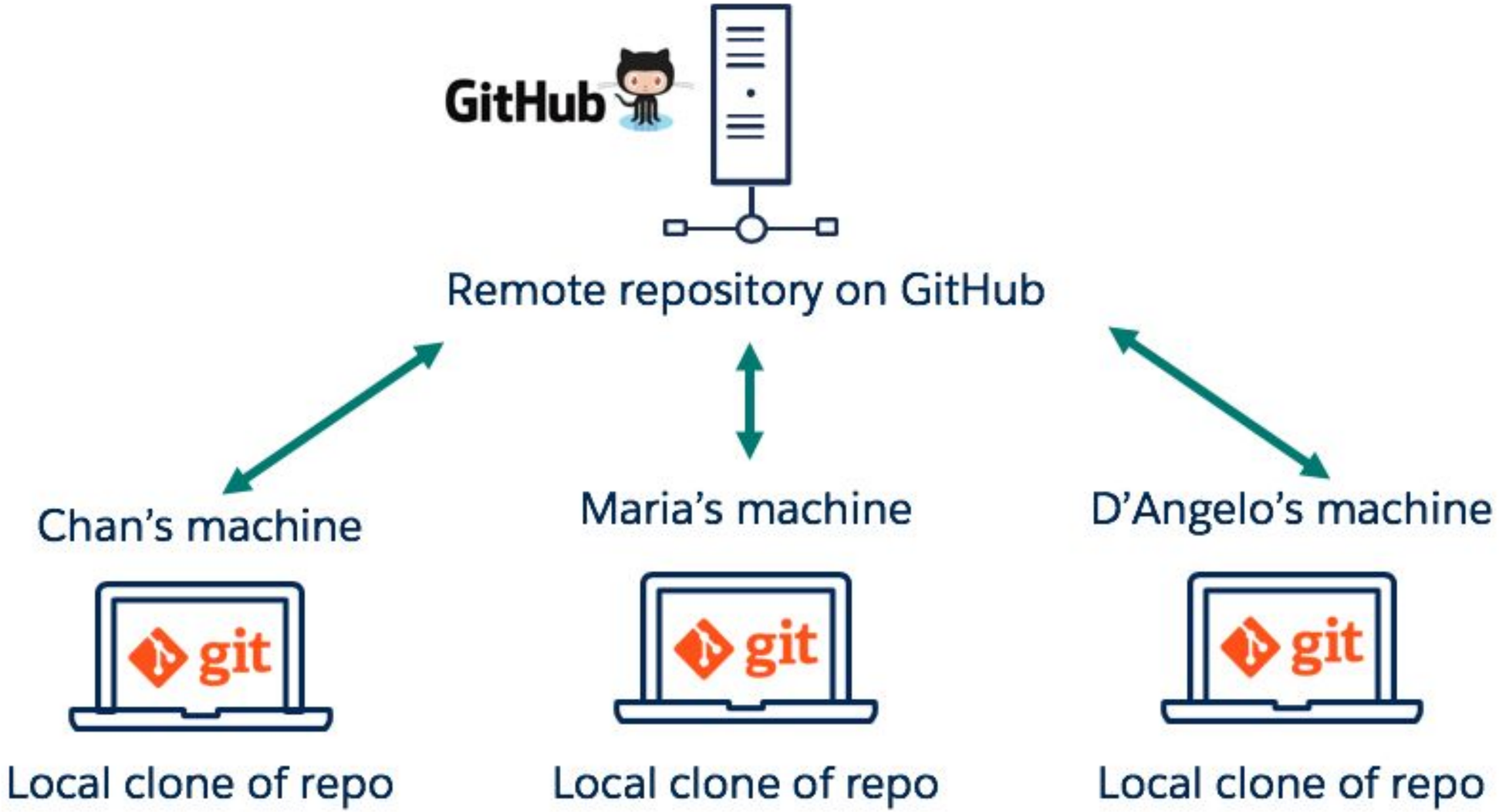

Initial Draft
Final Draft

Czym jest GitHub

Usługa chmurowa do zarządzania repozytoriami git

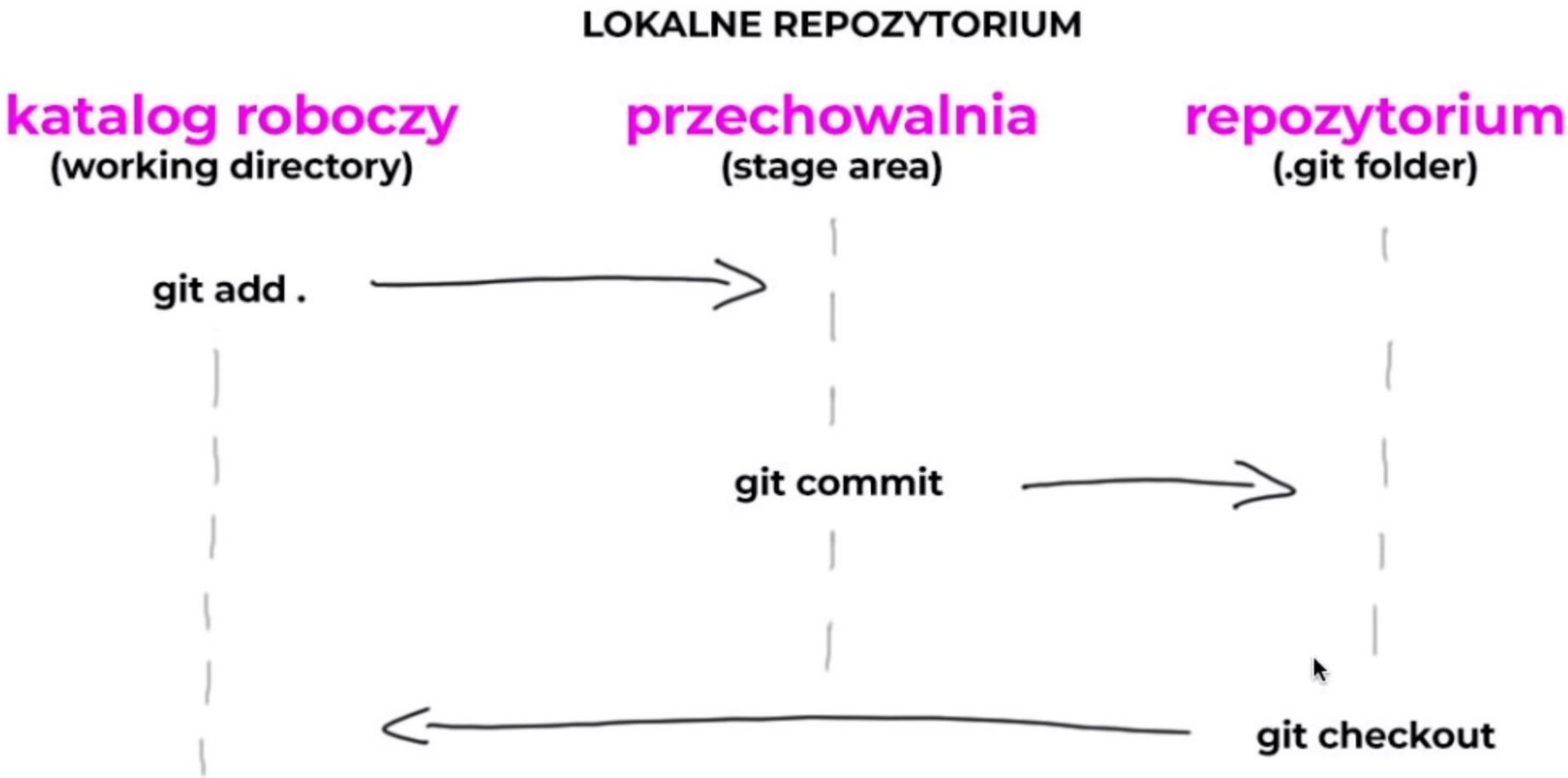
Platforma umożliwiające zdalny dostęp do repozytoriów git. GitHub udostępnia także funkcje planowania projektów oraz tworzenia procesów CI/CD

Google Developer Student Clubs


GitHub
Remote repository on GitHub
Chan's machine
Maria's machine
D'Angelo's machine
git
git
git
Local clone of repo
Local clone of repo
Local clone of repo

Kontrolowanie stanu

`git add, git commit, git status`


LOKALNE REPOZYTORIUM
katalog roboczy
(working directory)
przechowalnia
(stage area)
repozytorium
(.git folder)
git add .
git commit
git checkout

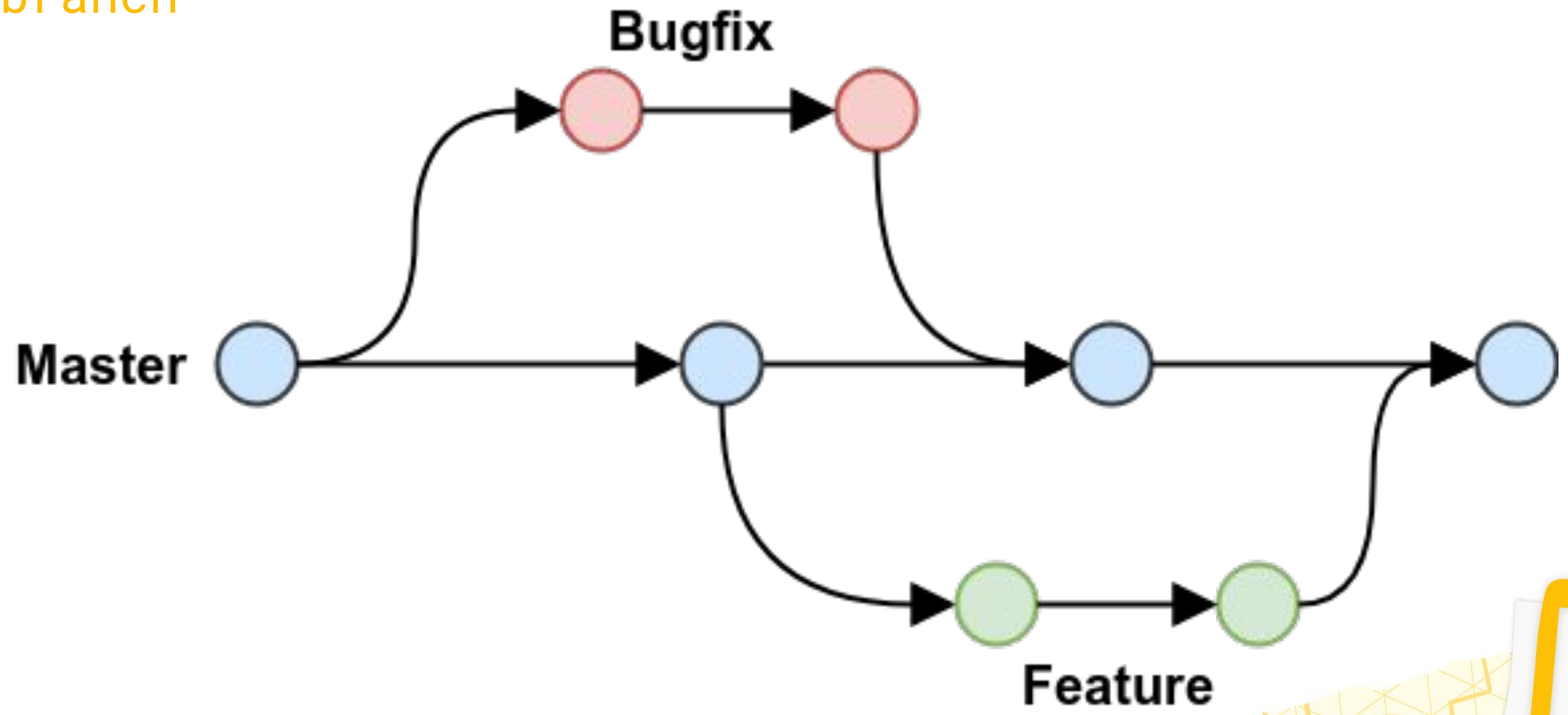
Przeglądanie historii

git log, git shortlog

```
* 99e05a5 - (HEAD -> master) feat: add second paragraph (10 minutes ago) <Krzysztof>
|
* 06b6c75 - feat: add first paragraph (16 minutes ago) <Krzysztof>
|
* 6c9ec42 - feat: initialize file with examples (21 hours ago) <Krzysztof>
|
* 5b45bf0 - doc: add basic README documentation (21 hours ago) <Krzysztof>
|
* 74dddb2 - initialize repository (21 hours ago) <Krzysztof>
```


Gałęzie

`git branch`



Przywracanie i przenoszenie zmian

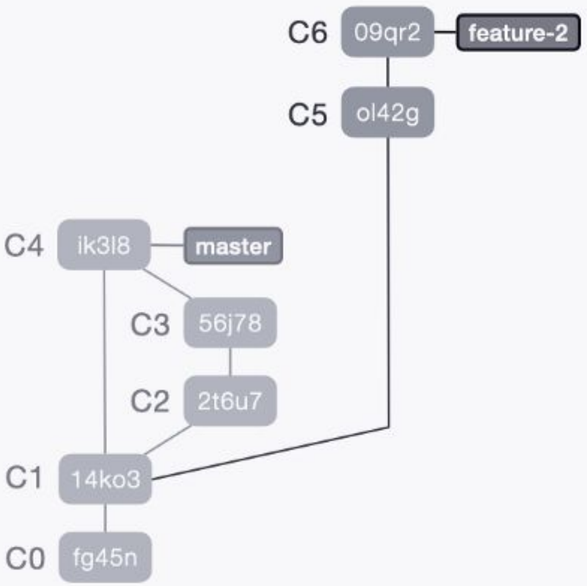
`git checkout, git revert, git reset, git clean, git stash`

Git **merge** i **rebase**

Start case

There are two main ways to integrate changes between branches, **merge** or **rebase**.

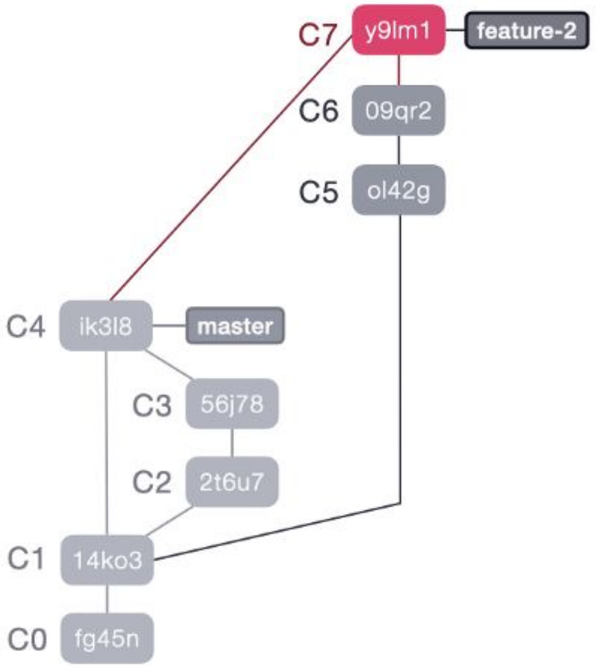
Here, *feature-2* is to be updated with changes from *master*.



Post merge

Merge preserves history as it happened, creating only one new merge commit.

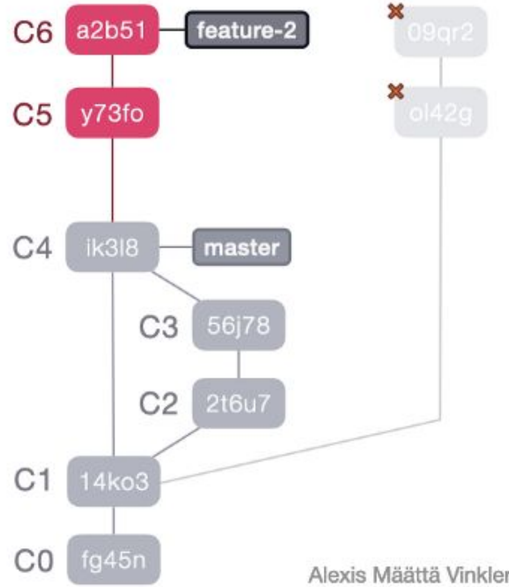
Here, commit **C7** intertwines the two branches – creating a non-linear diamond shaped history.



Post rebase

Rebase rewrites history, reapplying commits on top of another base branch.

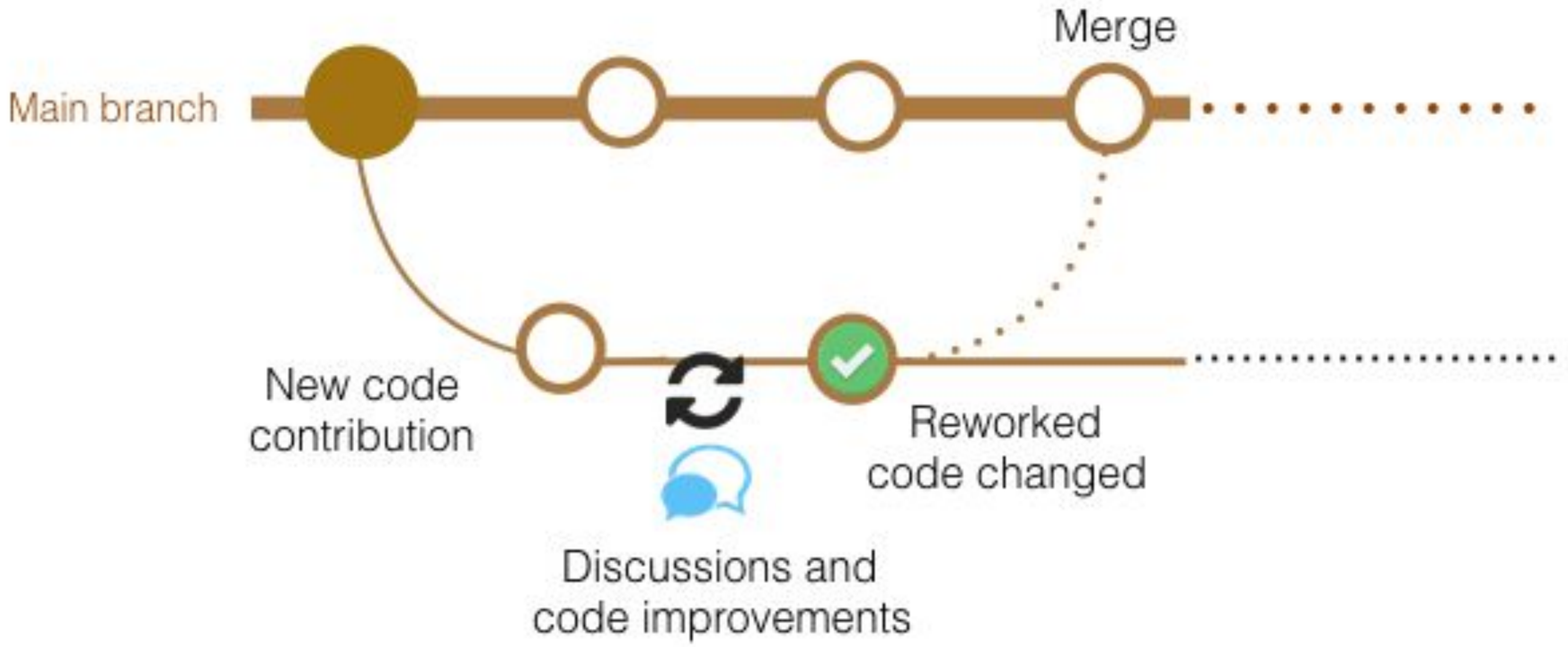
Here, commits **C5** and **C6** have been reapplied on top of **C4** – creating a linear history.



Alexis Määttä Vinkler

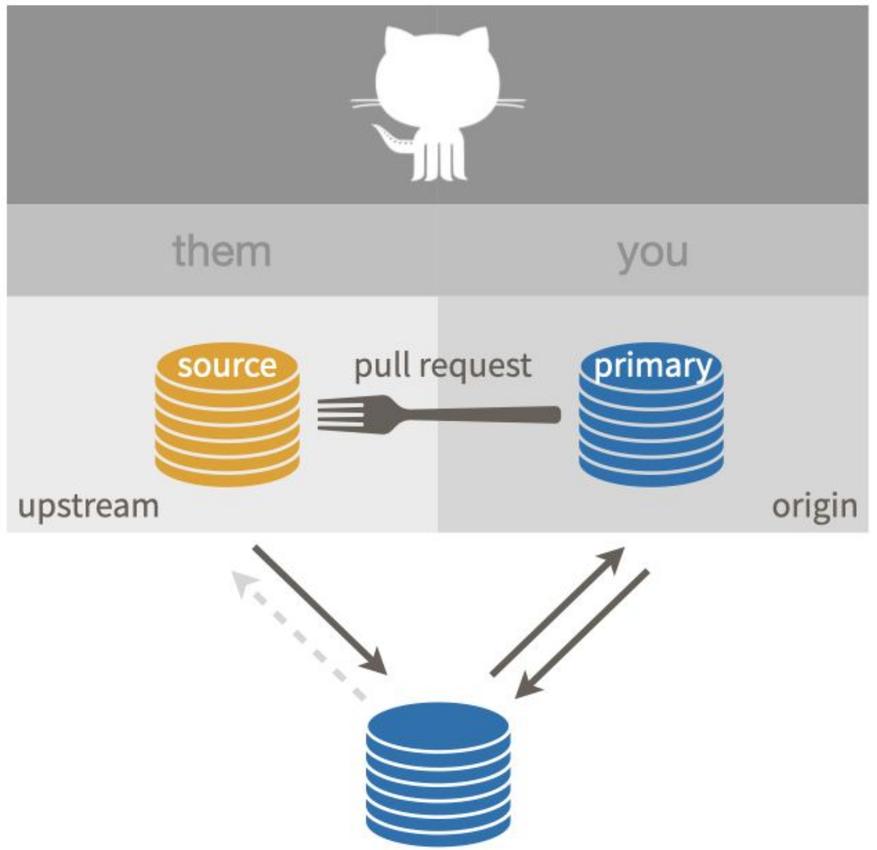
Pull request

Zgłaszanie zmiany gotowej do połączenia z głównym branch’em



Fork

Kopia repozytorium, która może być równolegle rozwijana



Grafiki użyte w prezentacji

Rebase vs Merge
https://betterprogramming.pub/differences-between-git-merge-and-rebase-and-why-you-should-care-ae41d96237b6
Fork
https://happygitwithr.com/upstream-changes.html
Gałąź
https://harness.io/blog/git-branching
Repozytorium zdalne:
https://sfdctechie.wordpress.com/2019/12/27/how-to-add-a-salesforce-dx-project-to-source-control-step-by-step-guide/
Zmiany w historii
https://jmcglone.com/guides/github-pages/
Stan repozytorium
https://www.youtube.com/watch?v=4bXuEv2R3W4&list=PLjHmWifVUNMKIGHmaGPVqSD-L6i1Zw-MH&index=2