Wprowadzenie do git

Konferencja Inżynierii Oprogramowania beIT'17

Kto ja?

Łukasz Rybka

Team Leader / Senior Software Developer w Solwit S.A.
Trener / szkoleniowiec w infoShare Academy
Wykładowca na Politechnice Gdańskiej

dragonia.org.pl

Podstawowe ustalenia

- Nie jestem alfą ani omegą
- Podstawowa znajomość git - „sufficient to be efficient”
- Pytania mile widziane, szczególnie w trakcie!
- Slajdy to tylko notatki, roadmapa

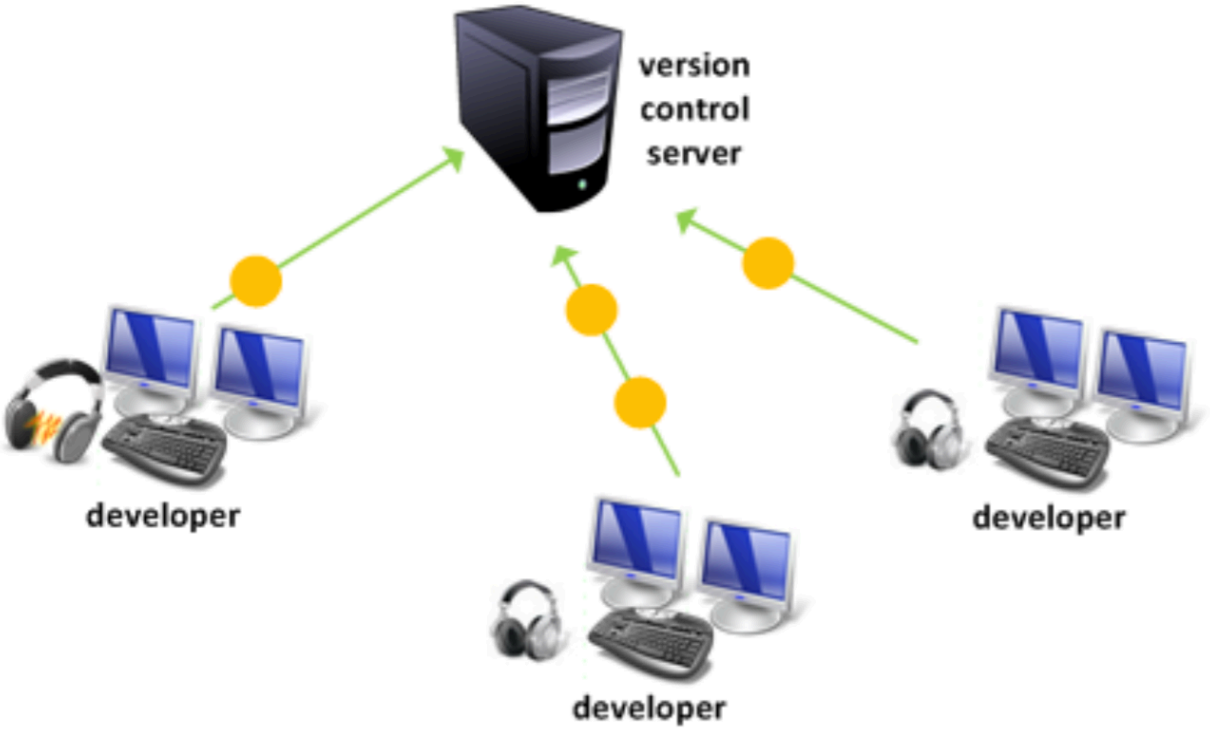
Cele warsztatu

- Wprowadzenie teoretyczne do rozproszonych systemów kontroli wersji na przykładzie git
- Stworzenie lokalnego repozytorium i wykonanie podstawowych operacji na nim
- Podłączenie zdalnego repozytorium (Bitbucket) i synchronizacja
- Praca z branchami i zapoznanie się z kilkoma jej strategiami
- ….

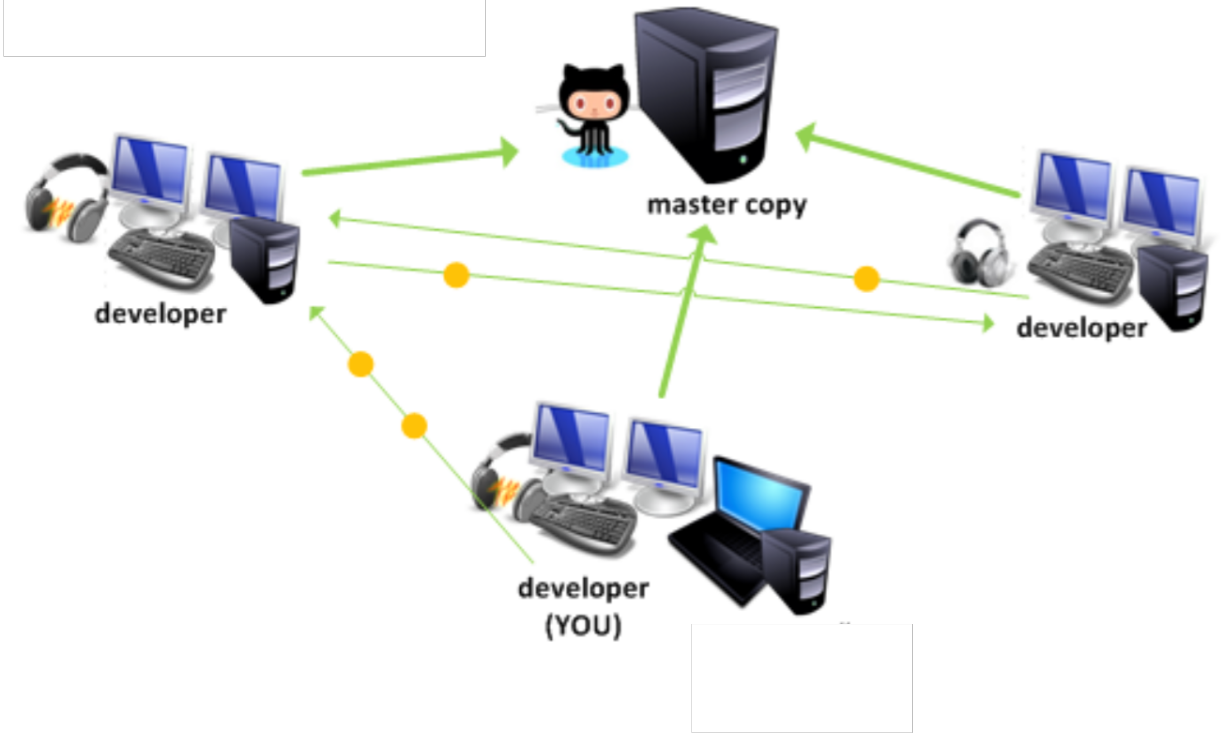
Pytanie

Co to jest system kontroli wersji?

Scentralizowany system kontroli wersji



Rozproszony system kontroli wersji

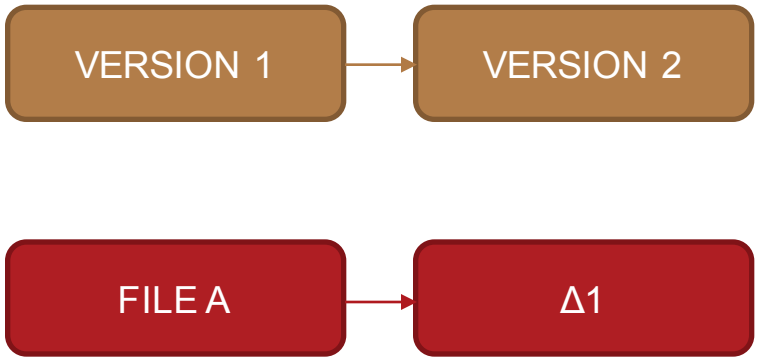


Pytanie

Co to jest commit?

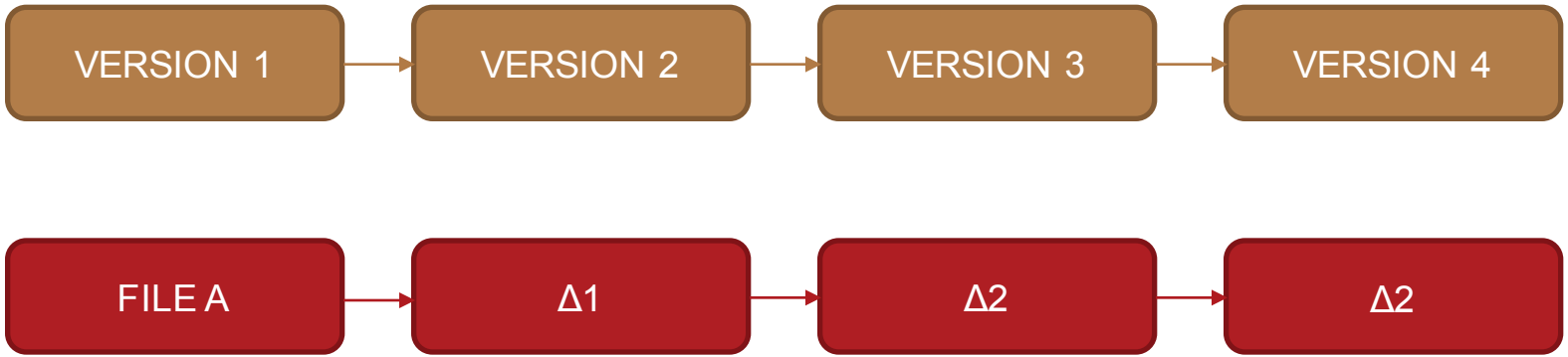
Czym jest commit?

Czym jest commit?



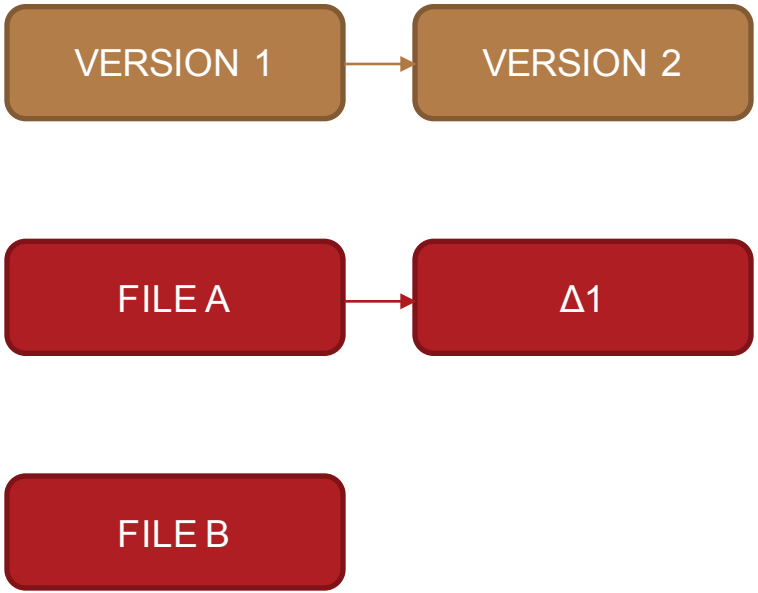
Czym jest commit?

Czym jest commit?

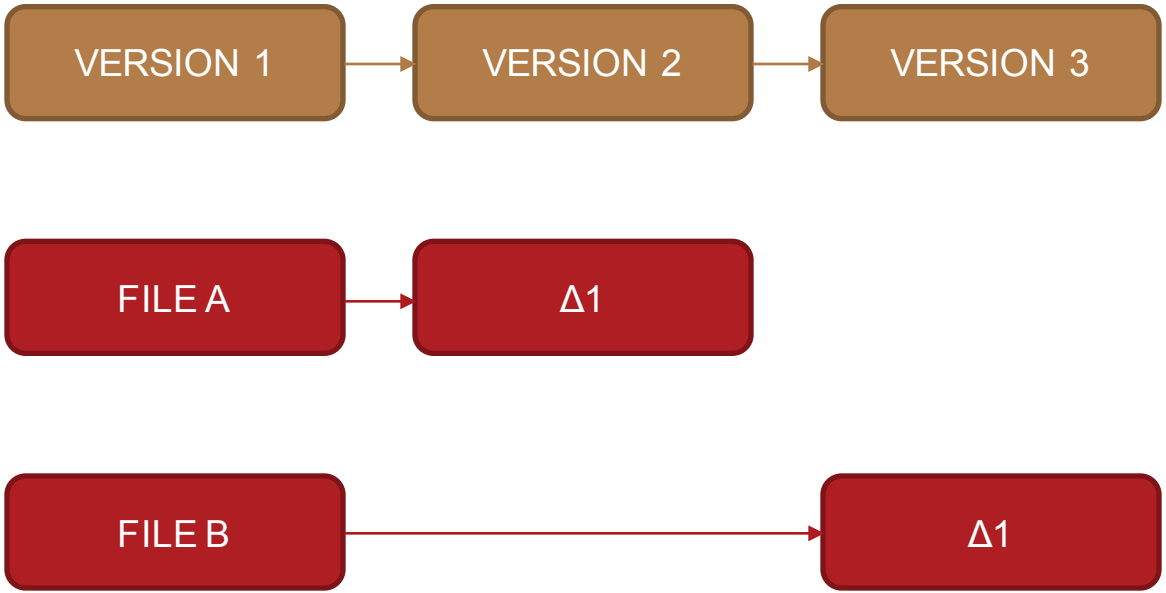


Czym jest commit?
Model SVN-like

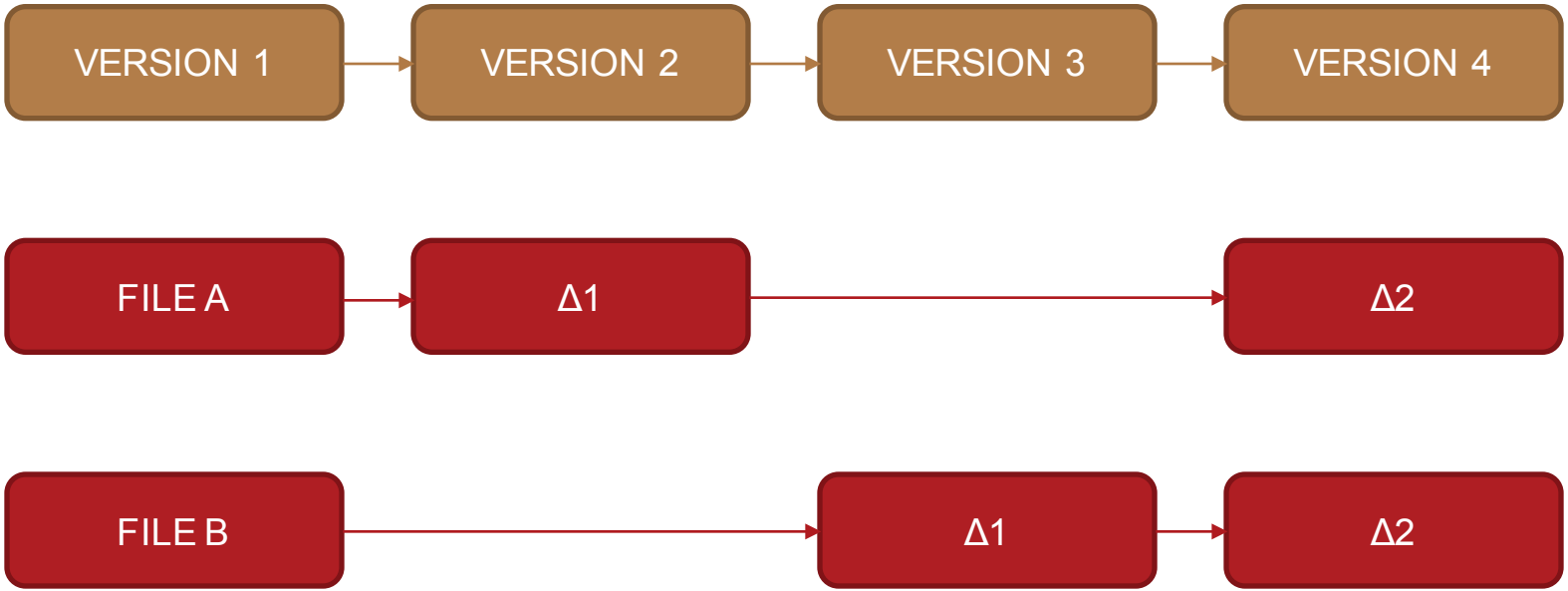
Czym jest commit?
Model SVN-like



Czym jest commit?
Model SVN-like

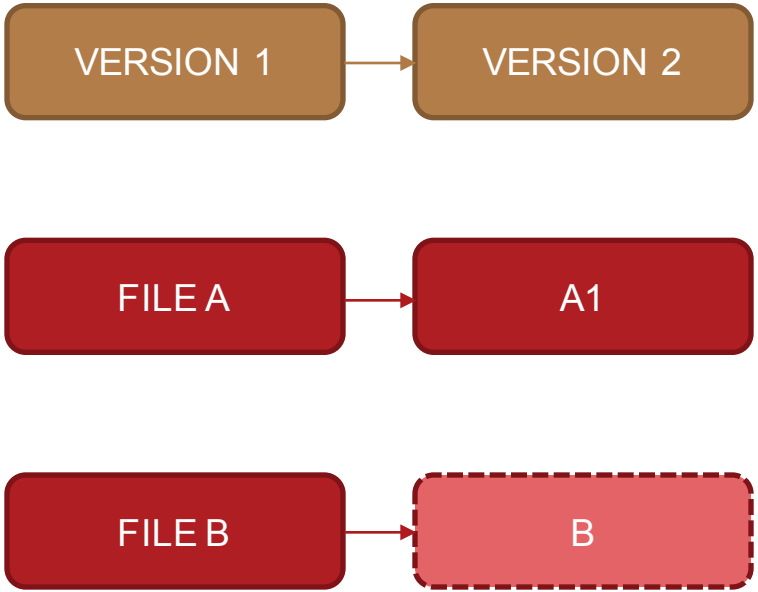


Czym jest commit?
Model SVN-like

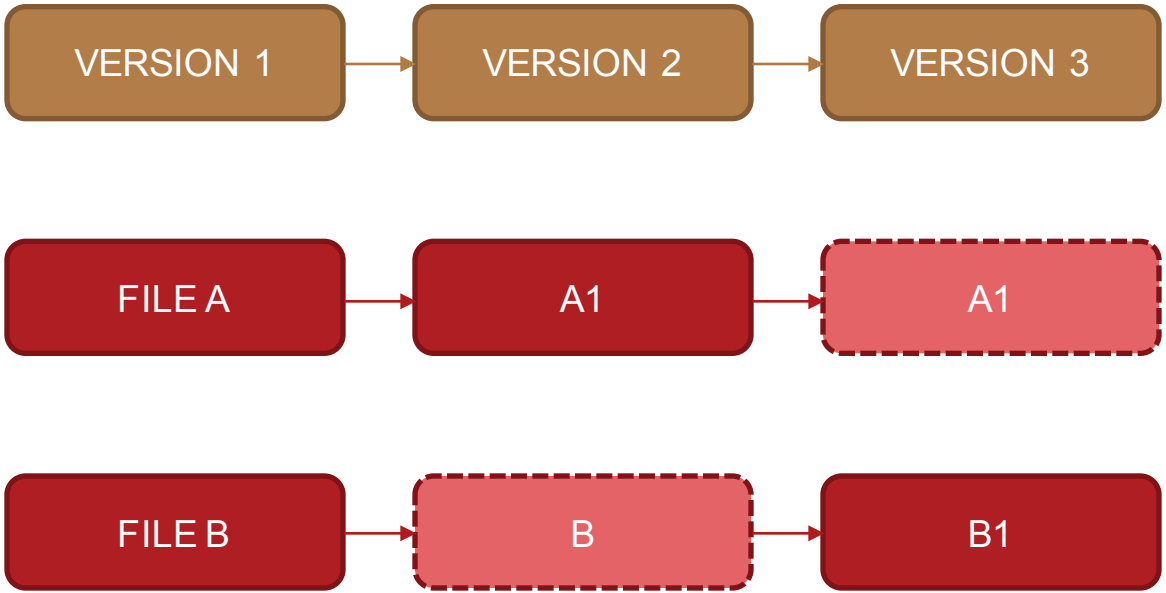


Czym jest commit?
Model GIT-like

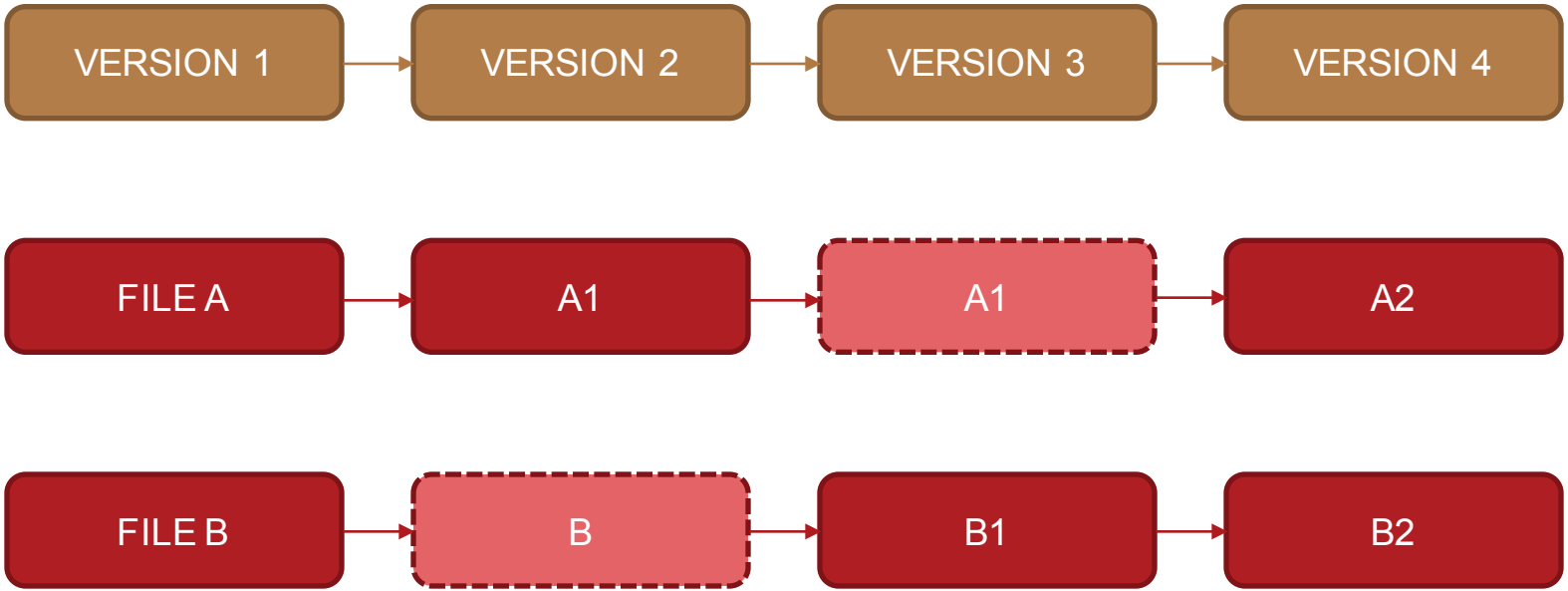
Czym jest commit?
Model GIT-like



Czym jest commit?
Model GIT-like



Czym jest commit?
Model GIT-like



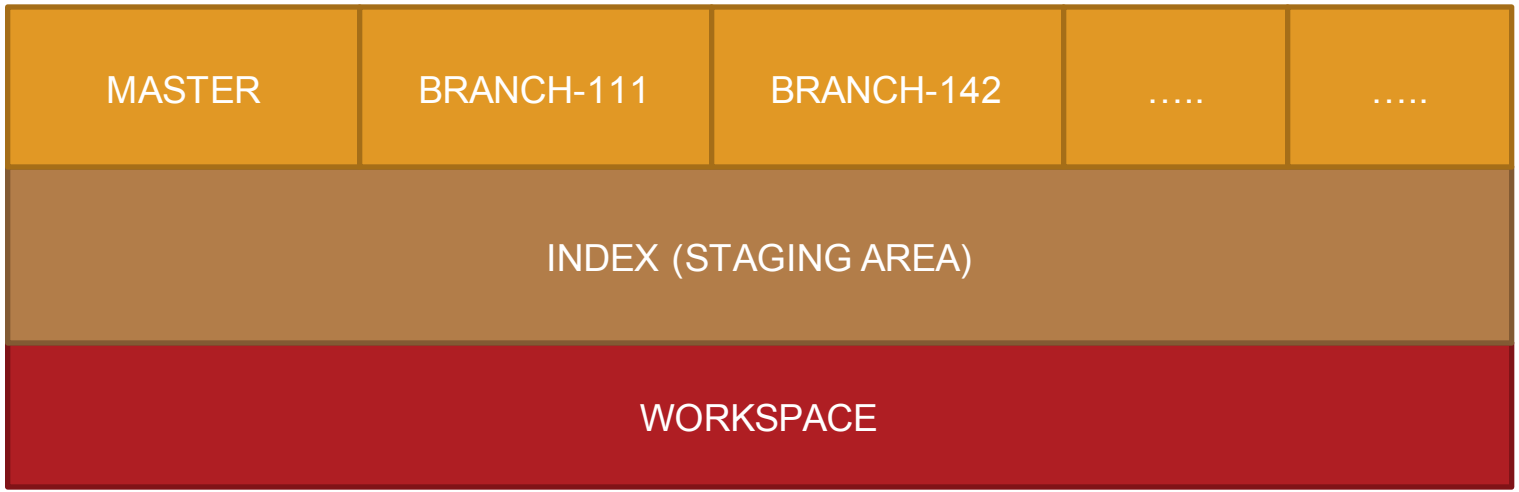
Pytanie

Co to jest repozytorium?

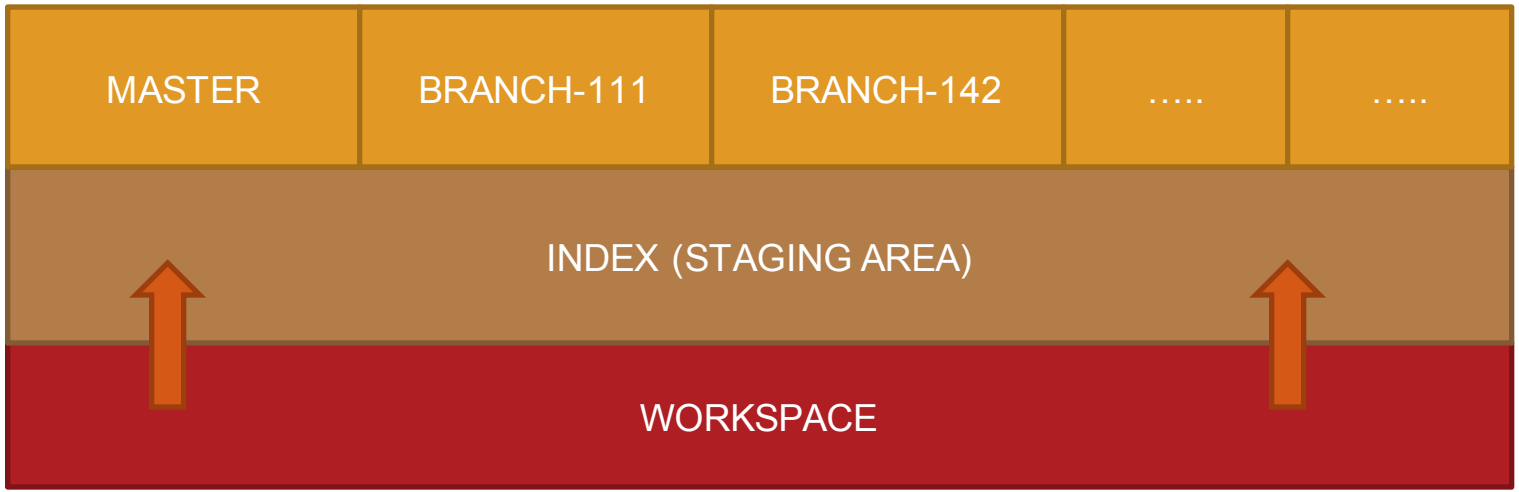
Lokalne repozytorium

Lokalne repozytorium

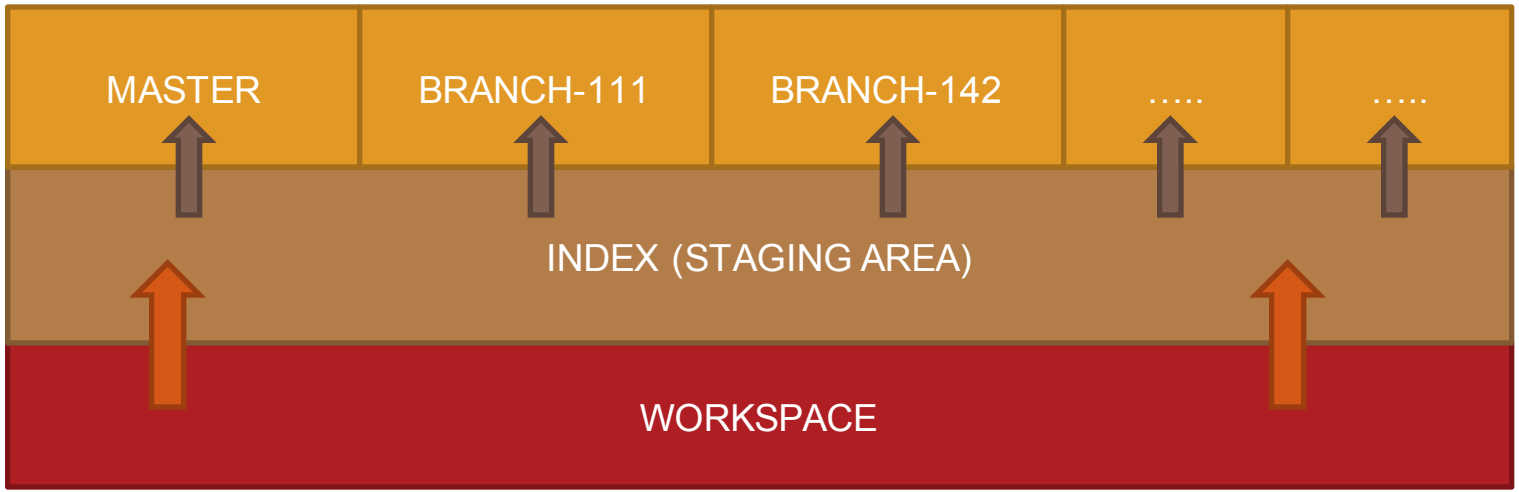
Lokalne repozytorium



Lokalne repozytorium



Lokalne repozytorium



Krok #1
Inicjalizacja repozytorium

Krok #2
Konfiguracja danych

Krok #2
Konfiguracja danych

git config user.name
git config user.name “Billy Everyteen”

git config user.email
git config user.email “your_email@example.com”

Krok #2
Konfiguracja danych globalnych

```
git config --global user.name “Billy Everyteen”
git config --global user.email “your_email@example.com”
```

Krok #3
Podstawowe operacje

Krok #3
Podstawowe operacje

- **git status** - informacje o stanie lokalnego repozytorium
- **git add** - przenosi zmianę z przestrzeni roboczej do indeksu
- **git commit** - tworzy commit
- **git log** - pokazywanie historii lokalnego repozytorium

Krok #4
Operacje na branchach

Krok #4
Operacje na branchach

- **git branch** - listuje, tworzy i usuwa branche
- **git checkout** - przełącza branche

- **git checkout -b <branch_name>** - tworzy nowy branch i przełącza na niego

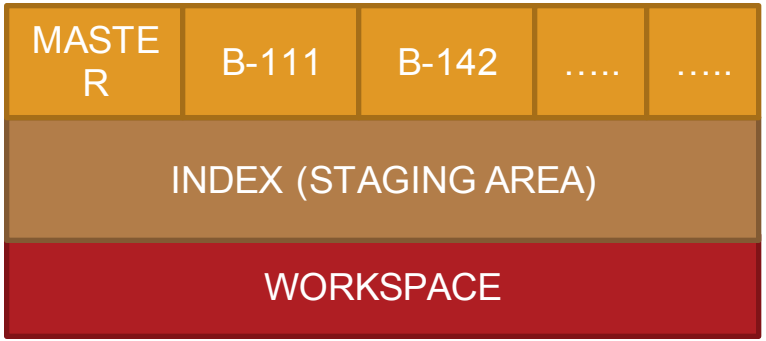
Krok #4
Merge'owanie zmian

- **git merge**- łączy dwie lub więcej historie prac w jedną

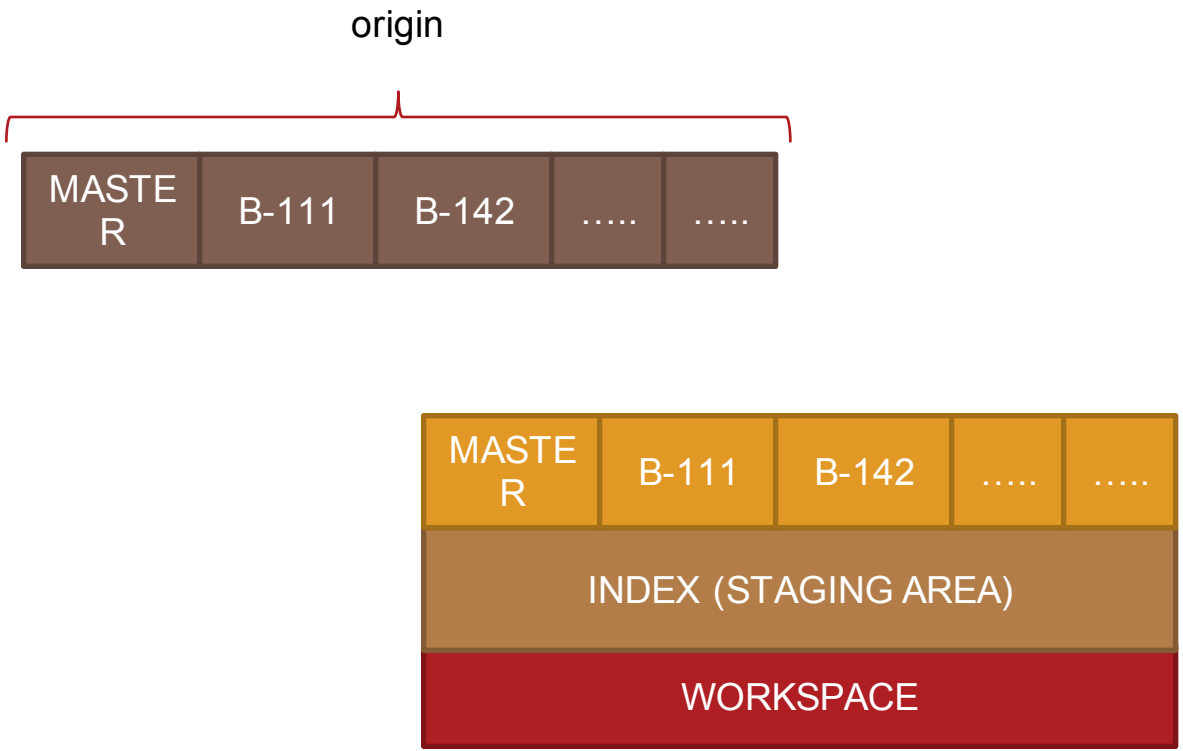
- **git add** - wskazanie, że konflikt w danym pliku został rozwiązany
- **git commit** - specjalny commit do oznaczenia rezultatu łączenia historii (branchy)

Krok #5
Zdalne repozytoria

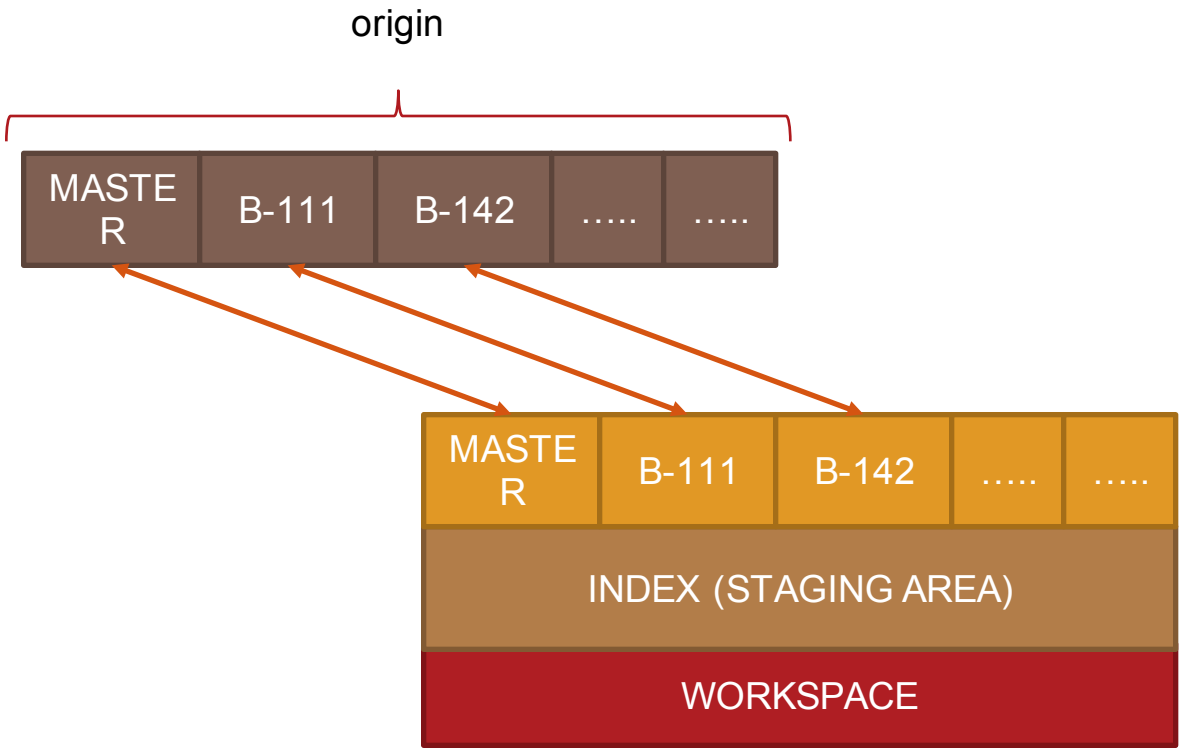
Krok #5
Zdalne repozytoria



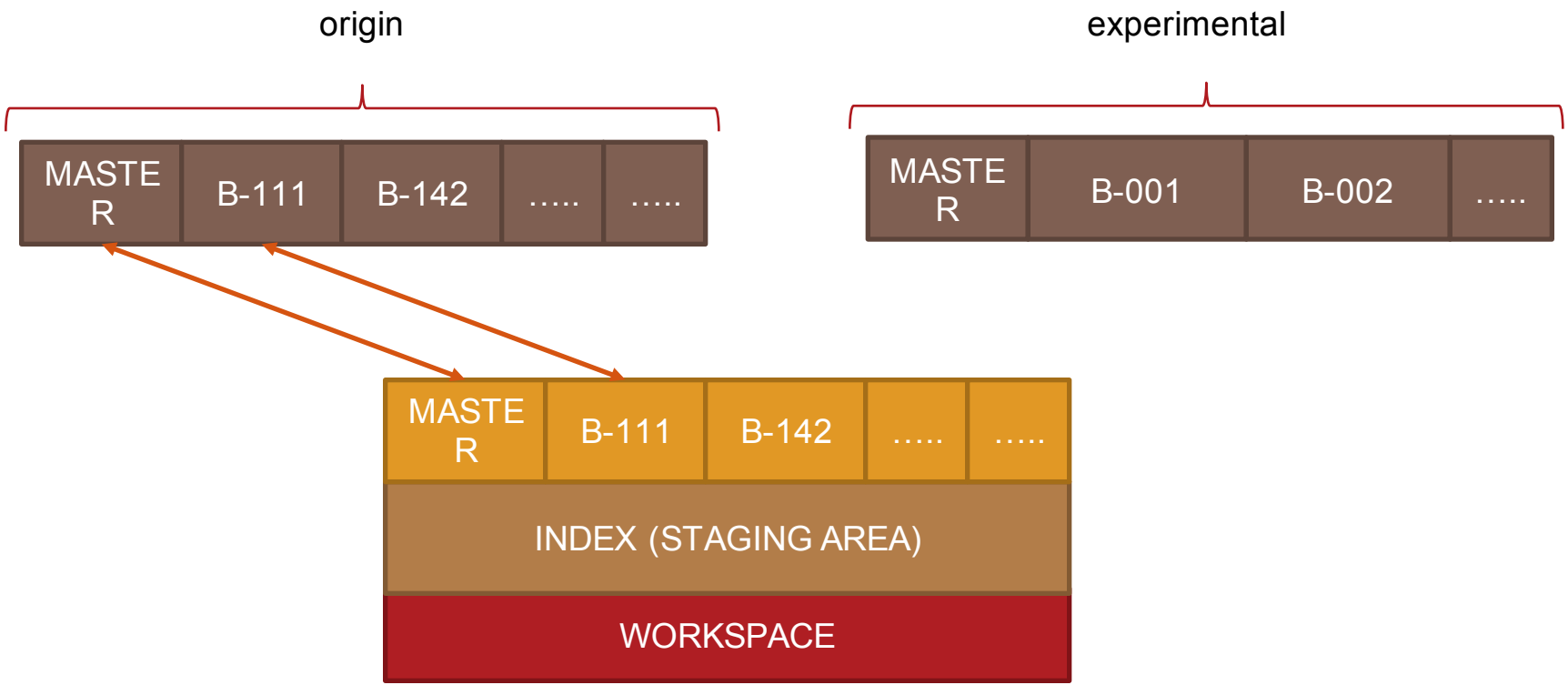
Krok #5
Zdalne repozytoria



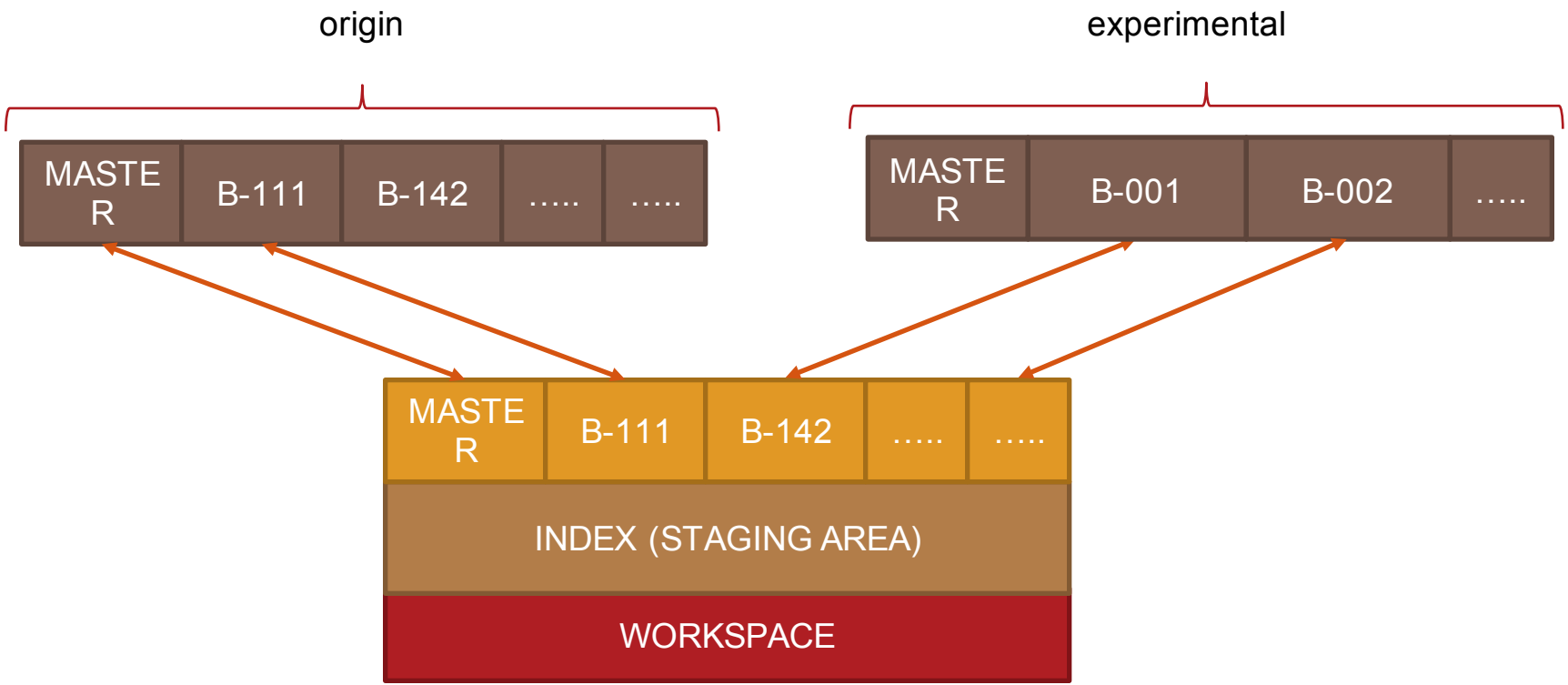
Krok #5
Zdalne repozytoria



Krok #5
Zdalne repozytoria



Krok #5
Zdalne repozytoria



Krok #5
Zdalne repozytorium - podłączenie

- **git remote** - wypisuje wszystkie zdalne repozytoria
- **git remote -v** - jak wyżej ale z dodatkową informacją
- **git remote add <name> <url>**
- **git remote rm <name>**
- **git remote rename <old_name> <new_name>**

- **git push <name> <branch>**

Krok #5
Zdalne repozytorium - fetch

- **git fetch <remote>** - pobiera wszystkie branche z repozytorium
- **git fetch <remote> <branch>**

Krok #5
Zdalne repozytorium - remote branch

- git branch -r

- git branch --set-upstream-to=<remote>/<branch>
- git branch –u <remote>/<branch>

- git merge <remote>/<branch>
- git pull <remote>

Krok #6
Cofanie zmian

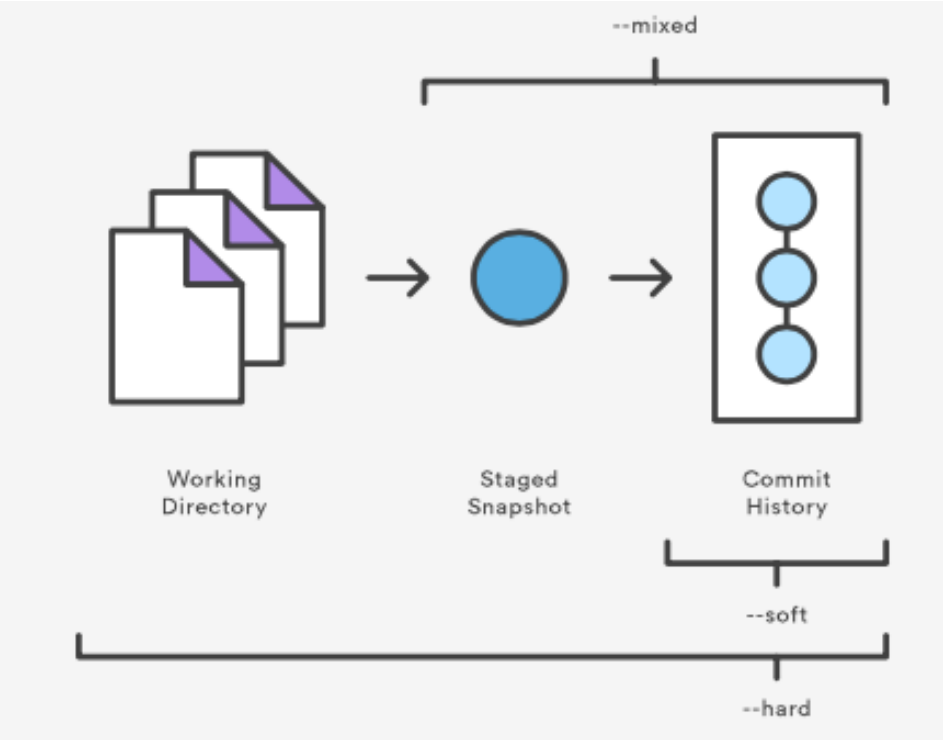
Krok #6
Cofanie zmian

- git checkout -- <file>
- git checkout <branch>
- git checkout <commit>

- git revert <commit>

- git clean -d
- git clean -f

Krok #6
Cofanie zmian - reset



Krok #6
Cofanie zmian - reset

- git reset --soft HEAD~<n>
- git reset --soft <sha>

- git reset --hard HEAD~<n>
- git reset --hard <sha>

- git reset --mixed HEAD~<n>
- git reset --mixed <sha>

Krok #7
Operacje na plikach

Krok #7
Operacje na plikach

- git rm <file>
- git rm --cached <file>

- git mv <file> <new_file_name>

Krok #8
Logowanie

Krok #8
Logowanie

- git log -<n>
- git log -p
- git log --stat
- git log --graph
- git log --oneline
- git log --pretty=oneline|short|full|fuller|format=“”

Krok #9
Squash

Krok #9
Squash

- **git merge --squash <branch>**

Inna strategia łączenia historii developmentu.

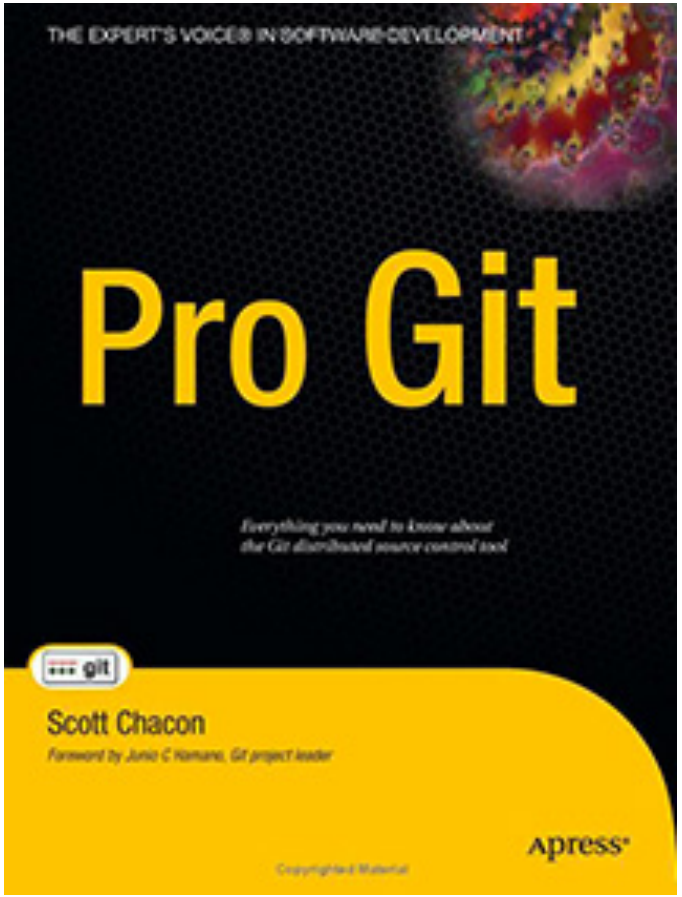
Zasada „single work = one commit” i utrzymywanie porządku w repozytorium.

Bitbucket Cloud

Idealne miejsce do nauki podstawowych zagadnień związanych z git'em.

Pro Git
Scott Chacon

Idealna książka dla osób, które chcą dokładnie wiedzieć co robią!



Thanks!