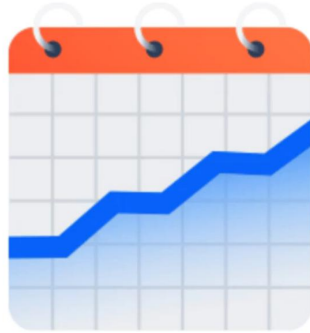


CONTINUOUS IMPROVEMENT

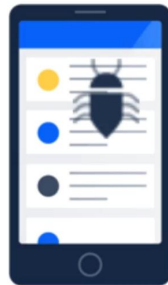
A key principle of DevOps



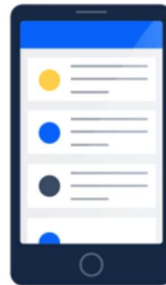
3

SMALL BATCH SIZE

Small batch size leads to continuous improvement



product with a bug

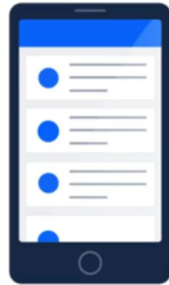


product without a bug

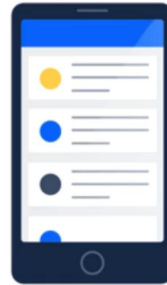
4

ADDING A FEATURE

Small batch size also applies to features



product with all blue icons



product with icon color

5

Topics

Continuous improvement

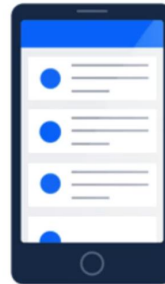
Managing project versions

Branches

Pull requests

OUR PROJECT

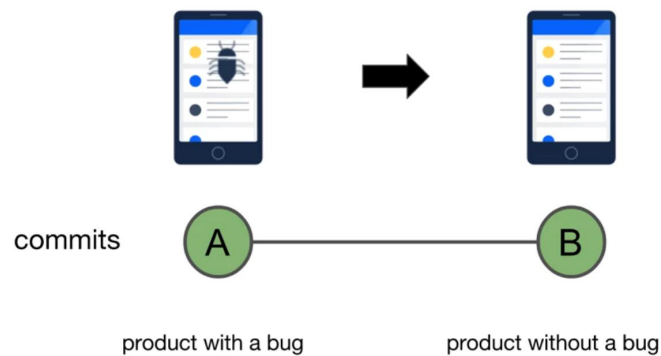
- Product contains 50 files of code
 - This is our "project"
- We want to continuously improve it
- How does Git help?



8

GIT COMMITS

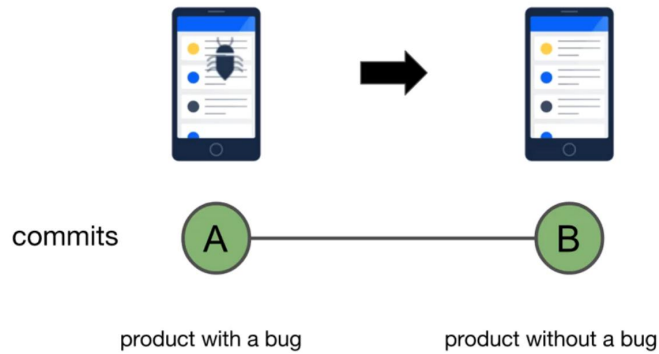
- Git manages versions of projects
- Each version of a project is called a *commit*



9

GIT COMMITS

- Git manages versions of projects
- Each version of a project is called a *commit*

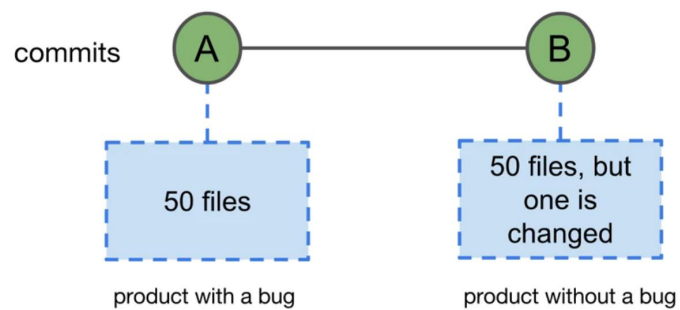


9

GIT COMMITS

Commits A and B only differ by one file

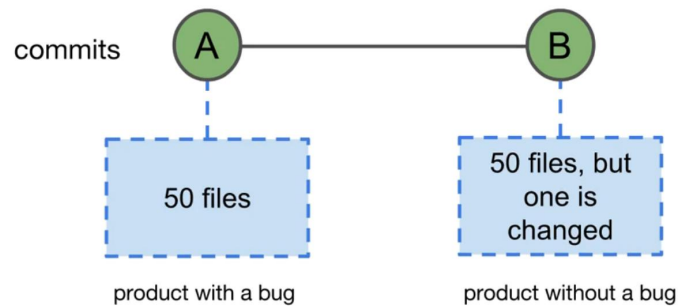
- A small improvement in the product



10

IS GIT EFFICIENT?

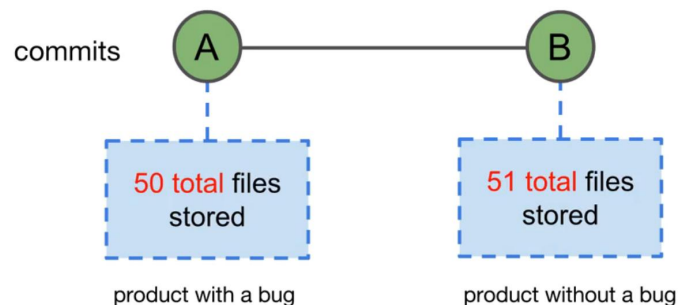
- Each commit is a snapshot of the entire project



11

IS GIT EFFICIENT?

- Each commit is a snapshot of the entire project
- Behind the scenes, Git is very efficient at storing commits
 - Each unique file is stored only once

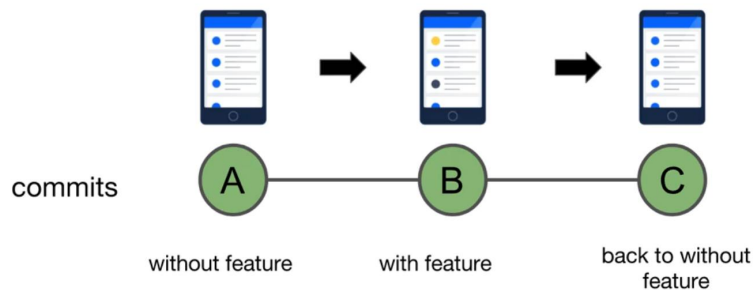


12

PROJECT HISTORY

The collection of commits contain the history of the project

- You can review the history
- You can "undo" a change



13

Topics

Continuous improvement

Managing project versions

Branches

Pull requests

BRANCHES

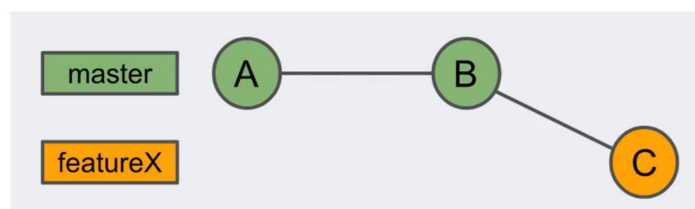
- All commits belong to a *branch*
 - An independent line of development of the project
- By default, there is a single branch and it is called *master*



16

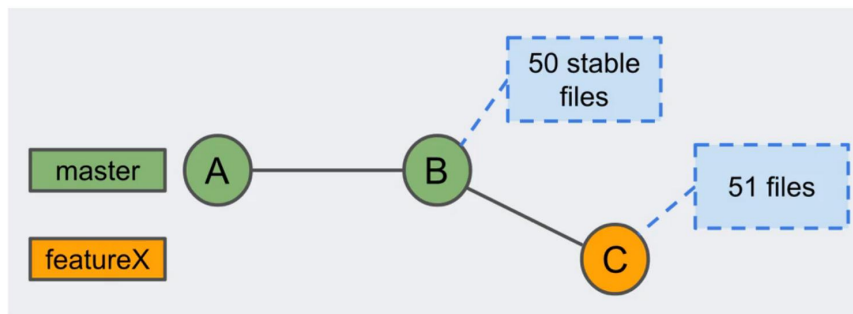
CREATING BRANCHES

- How do you maintain a stable project at the same time that you are working on it?
 - Create a separate branch



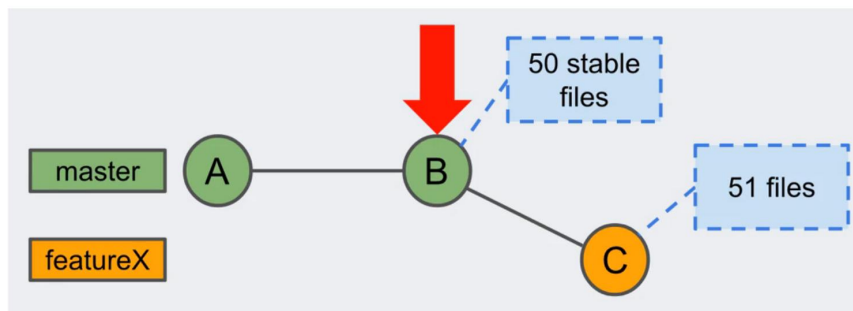
17

BRANCH INDEPENDENCE



19

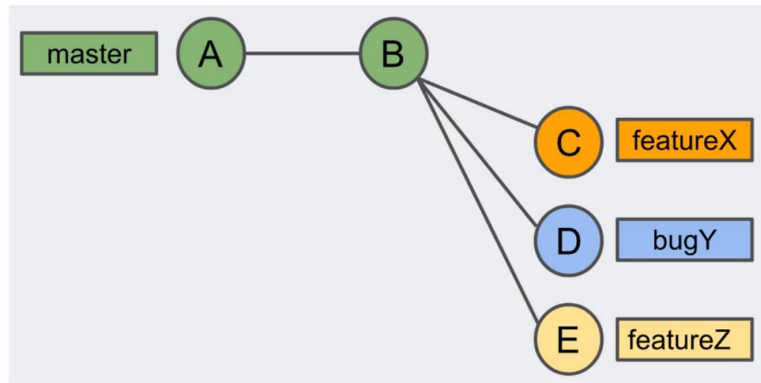
BRANCH INDEPENDENCE



19

TEAMS

The independence of branches allows teams to scale their work



20

Topics

Continuous improvement

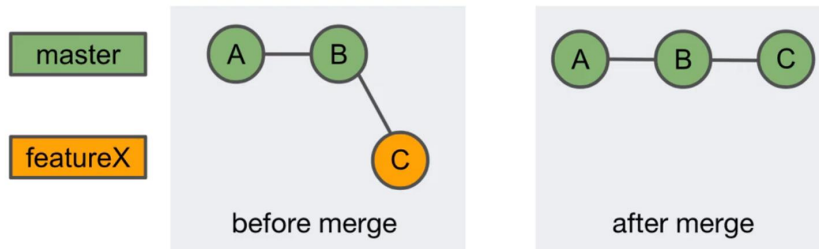
Managing project versions

Branches

Pull requests

MERGING

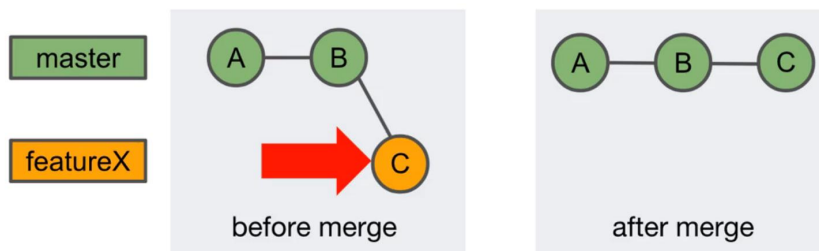
After a *merge*, the *master* branch contains the new feature



23

MERGING

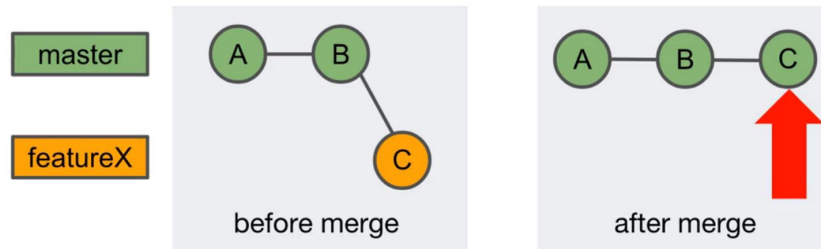
After a *merge*, the *master* branch contains the new feature



23

MERGING

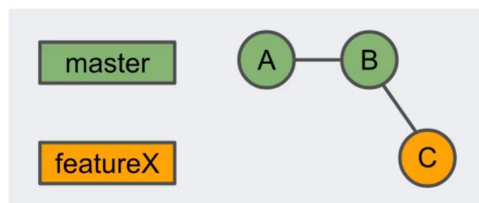
After a *merge*, the *master* branch contains the new feature



23

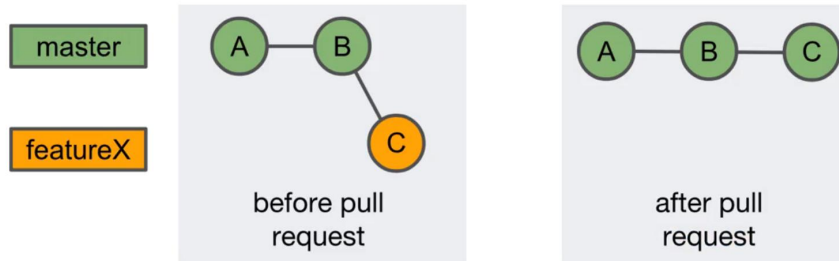
PULL REQUEST

Request to merge your branch into another branch



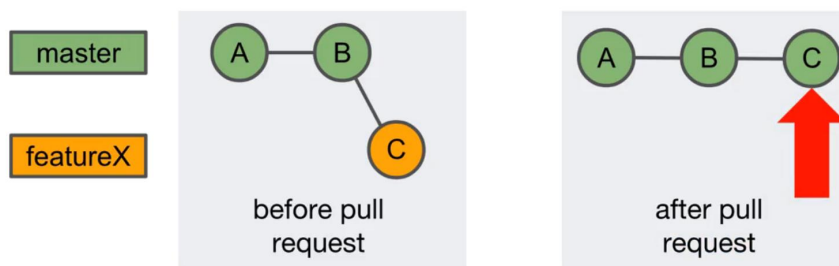
24

PULL REQUEST ACCEPTED



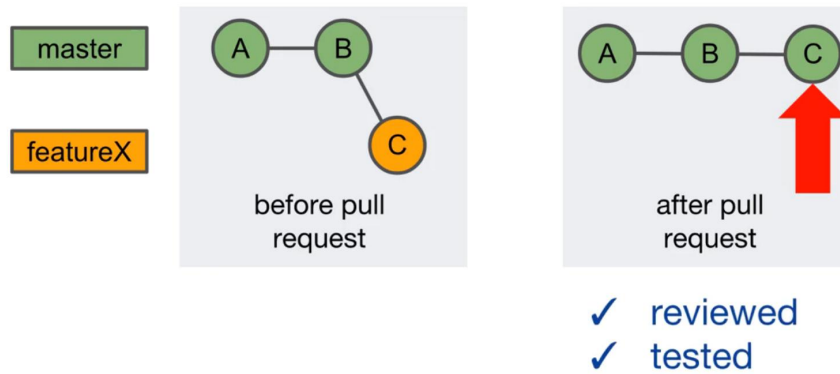
25

PULL REQUEST ACCEPTED



25

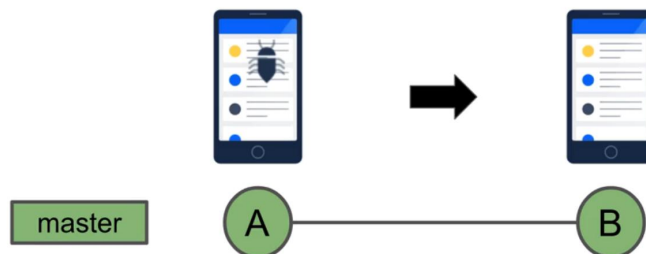
PULL REQUEST ACCEPTED



25

REVIEW

- Continuous improvement- **commits**
- Simultaneous stability and development- **branches**
- Improved quality- **pull requests**



27