



Mark
Bynens

Sumesh
Kalappurakal



TT09: All about versioning – an intro into Distributed Version Control

Name: Mark Bynens

Title: Senior Manager Statistical Programming and Analysis

Company: Janssen R&D

Mark has 15 years experience in the industry

Mark's interests are reading, sports and cooking

Name: Sumesh Kalappurakal

Title: Head Analytics Innovation, Statistical Programming and Analysis

Company: Janssen R&D

Sumesh has 18 years experience in the industry

Sumesh's interests are photography, listening to music and cooking



The opinions expressed are those of the authors and do not necessarily represent the opinions of PhUSE, members' respective companies or organizations, or the products' respective companies mentioned in this paper.

Although GIT and GIT based remote repositories have been used to explain more about distributed version control systems and remote repositories, the authors believe that other distributed version control systems or other remote repositories can equally be used.

It is the reader's or audience's responsibility to determine which distributed version control system or remote repository would best suit their need.

Additionally, the authors do not endorse any specific commercial products or services.

INTRODUCTION

Have you ever added versions to your files, perhaps with suffixes like "final" or "latest" and then had to later deal with a new final version?

Have you ever commented out code blocks because you wanted to disable certain functionality without deleting the code, fearing that there may be a use for it later?



new.r



newfinal.r



newfinalfinal.r



newfinalestfinal.r



newfinalestfinalforsure.r



new_v6.r

Name	Date modified	Type	Size
 Archive	10/11/2019 12:35	File folder	
 demo_bitbucket1.mp4	24/10/2019 13:32	MP4 Video	4,673 KB
 demo_bitbucket2.mp4	4/11/2019 14:58	MP4 Video	3,979 KB
 PHUSE_EU_Connect_TT02_Final.pptx	10/11/2019 11:56	Microsoft PowerP...	4,480 KB
 PHUSE_EU_Connect_TT02_Final_Final.pptx	10/11/2019 12:38	Microsoft PowerP...	4,467 KB

Introduction

THE SOURCE

The life-blood of any software development process

SOURCE CONTROL

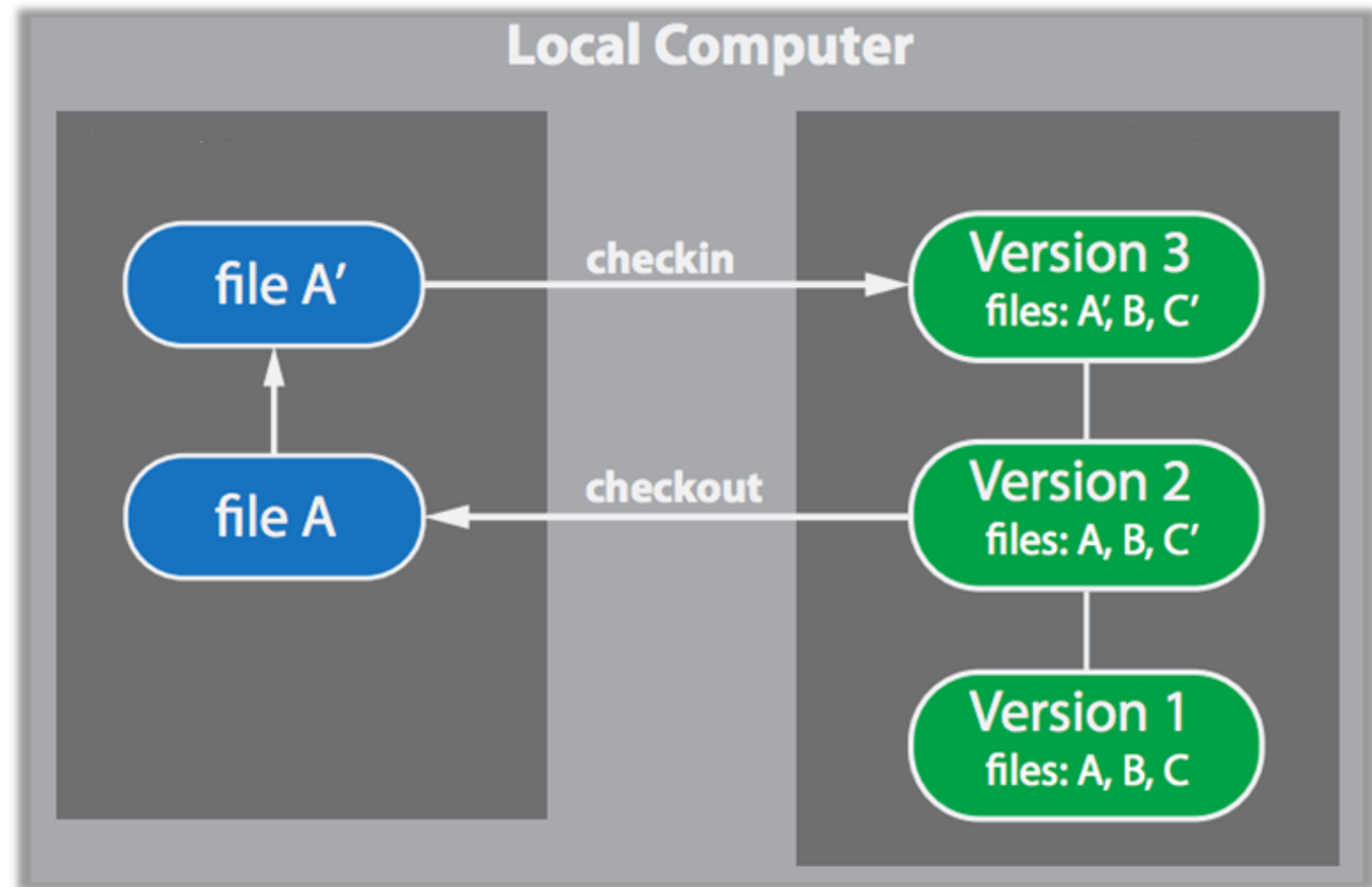
Source control systems are a category of software tools that help a software team manage changes to source code over time. Version control software keeps track of every modification to the code.



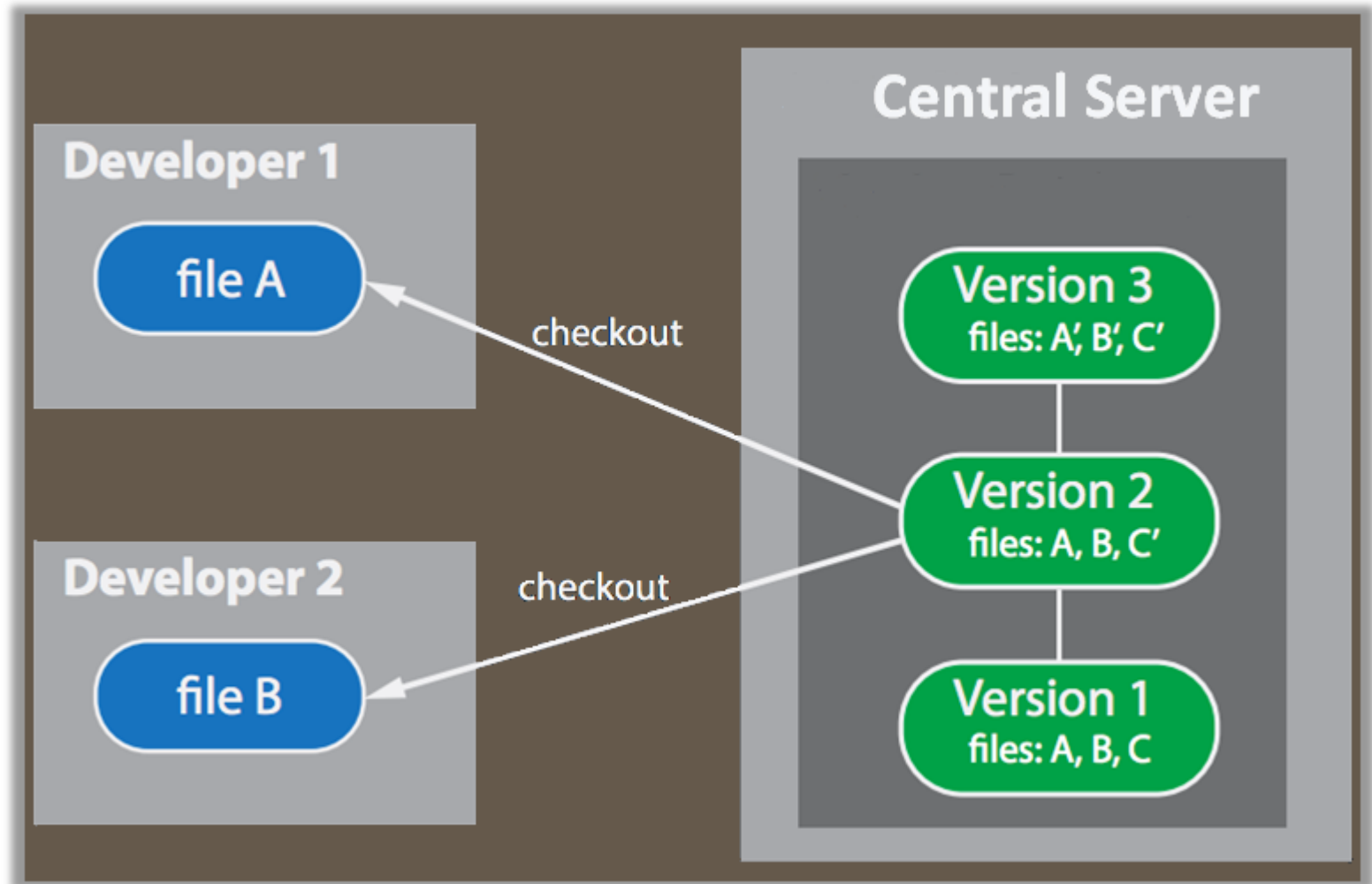


- Collaboration
- Storing versions (properly)
- Restoring previous versions
- Understanding what happened
- Backup

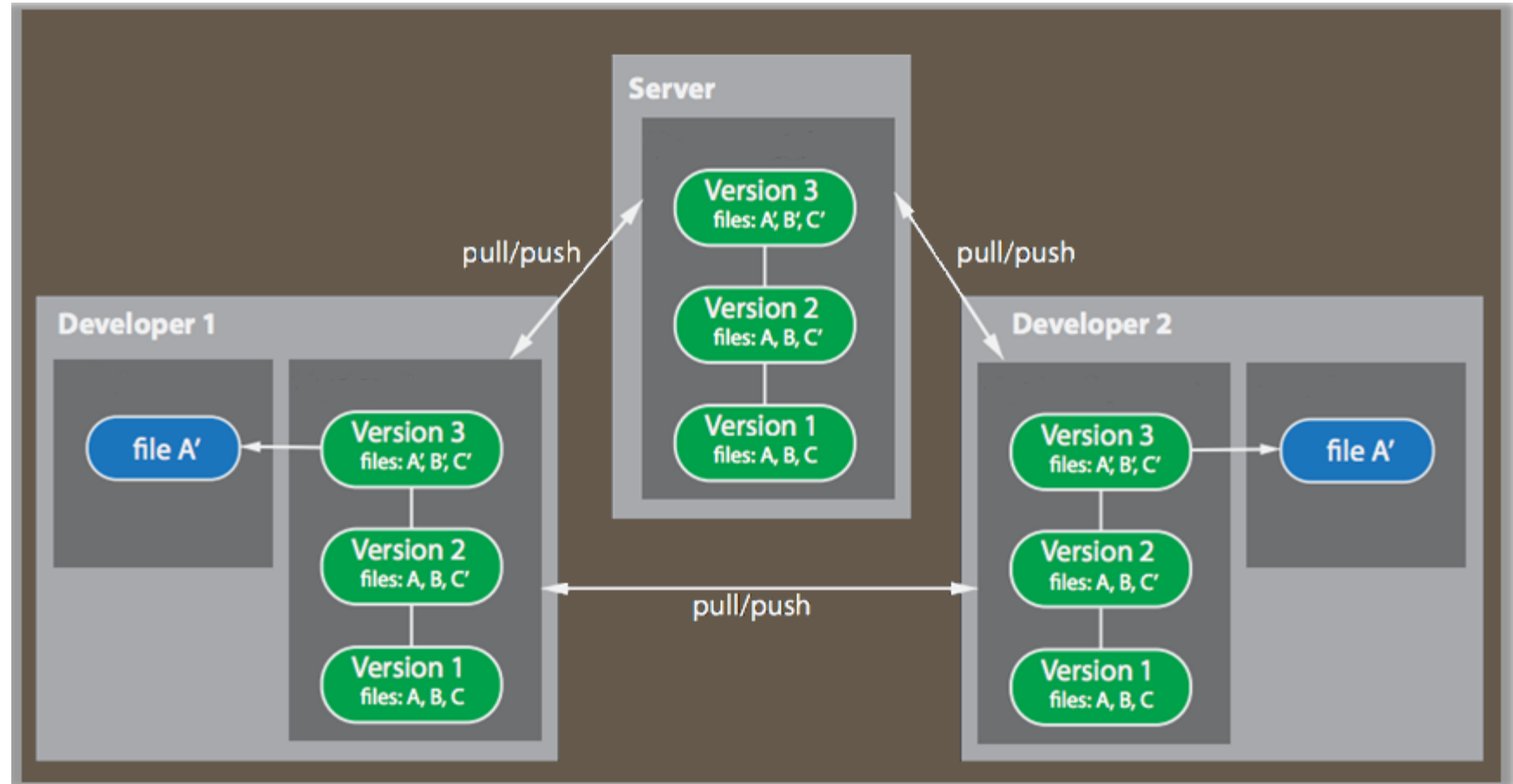
LOCAL
VERSION
CONTROL
SYSTEM



CENTRALIZED
VERSION
CONTROL
SYSTEM



DISTRIBUTED VERSION CONTROL SYSTEM

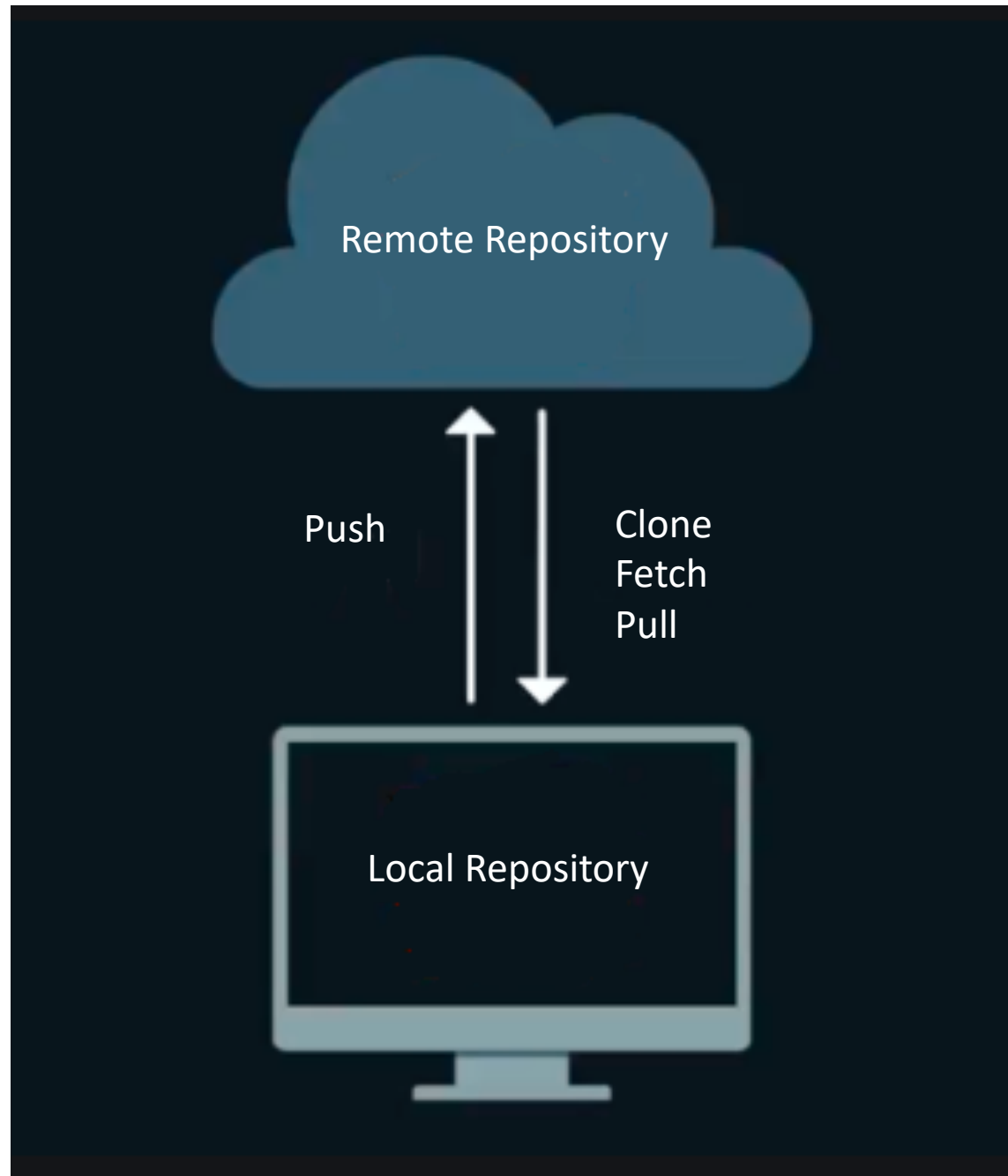




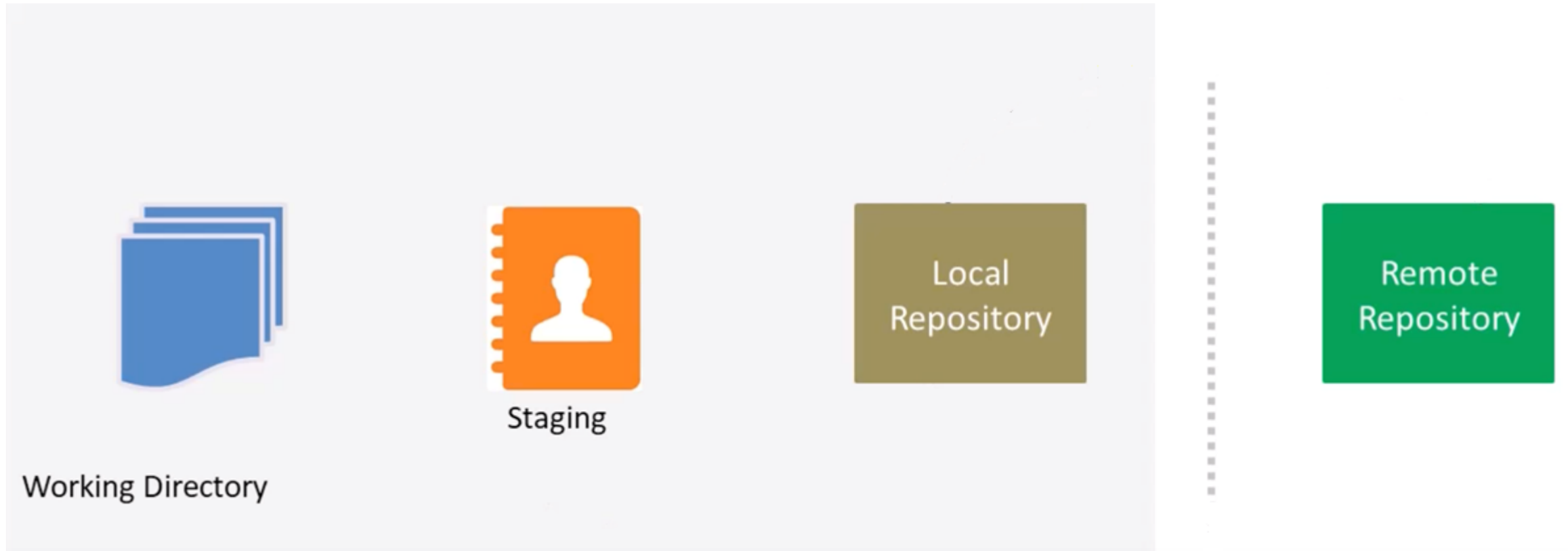
Started in 2005

Created by Linus
Torvalds to aid in
Linux kernel
development

REMOTE vs LOCAL REPOSITORY



GIT FRAMEWORK



Git Process (Local Repo)

Step 1

INSTALL GIT

- GUI (EG. SOURCETREE) or directly following the instruction from <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

Step 2

CREATE REPO

- Existing folder can be converted to a git repo – `git init`
- Clone a repo – `git clone`

Step 3

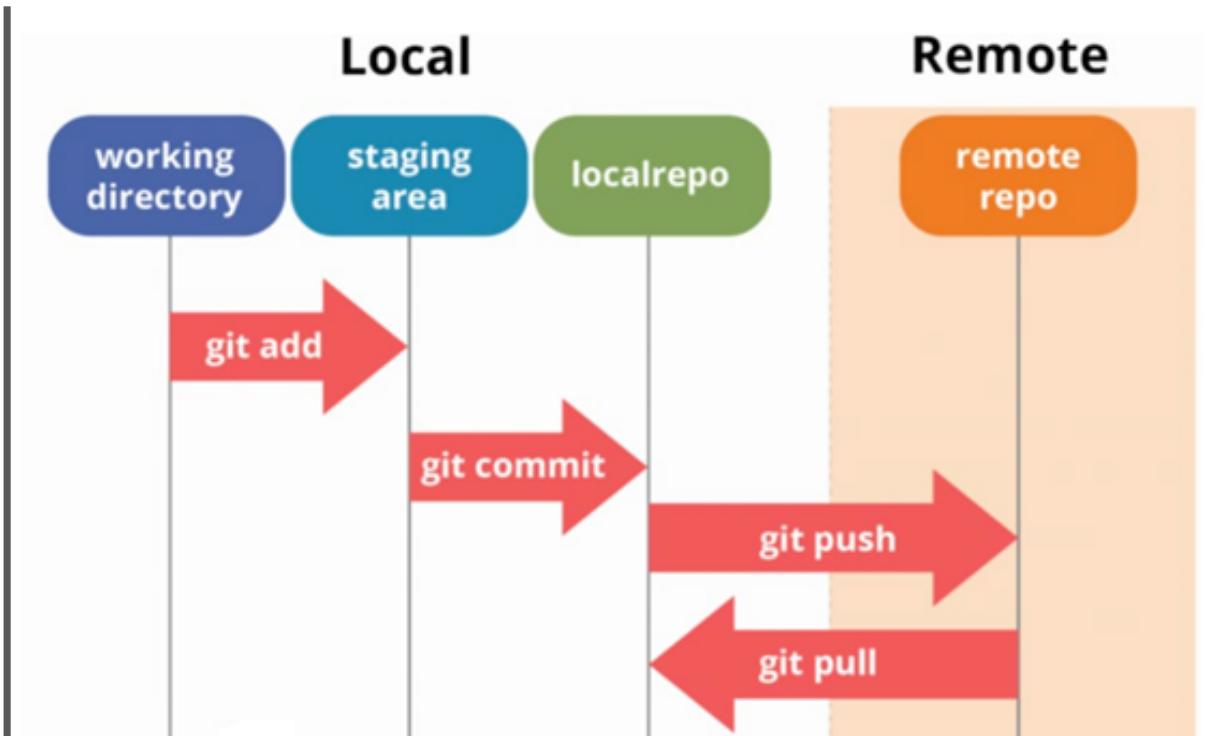
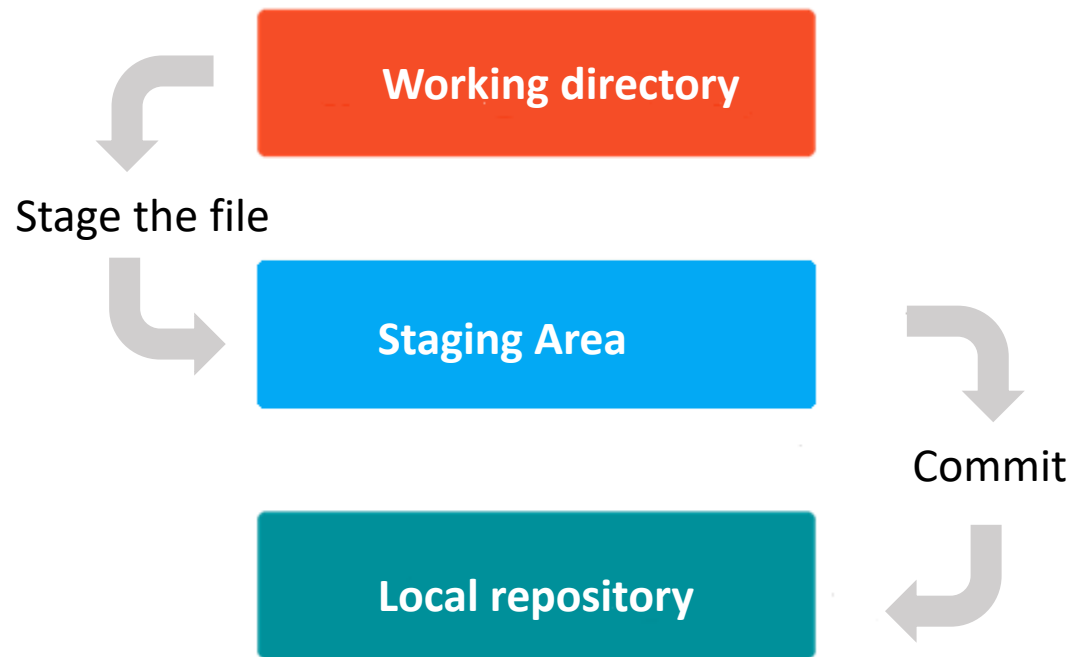
ADDING FILE TO GIT REPO

- Need to add file to the git to be tracked – `git add`

Step 4

COMMIT

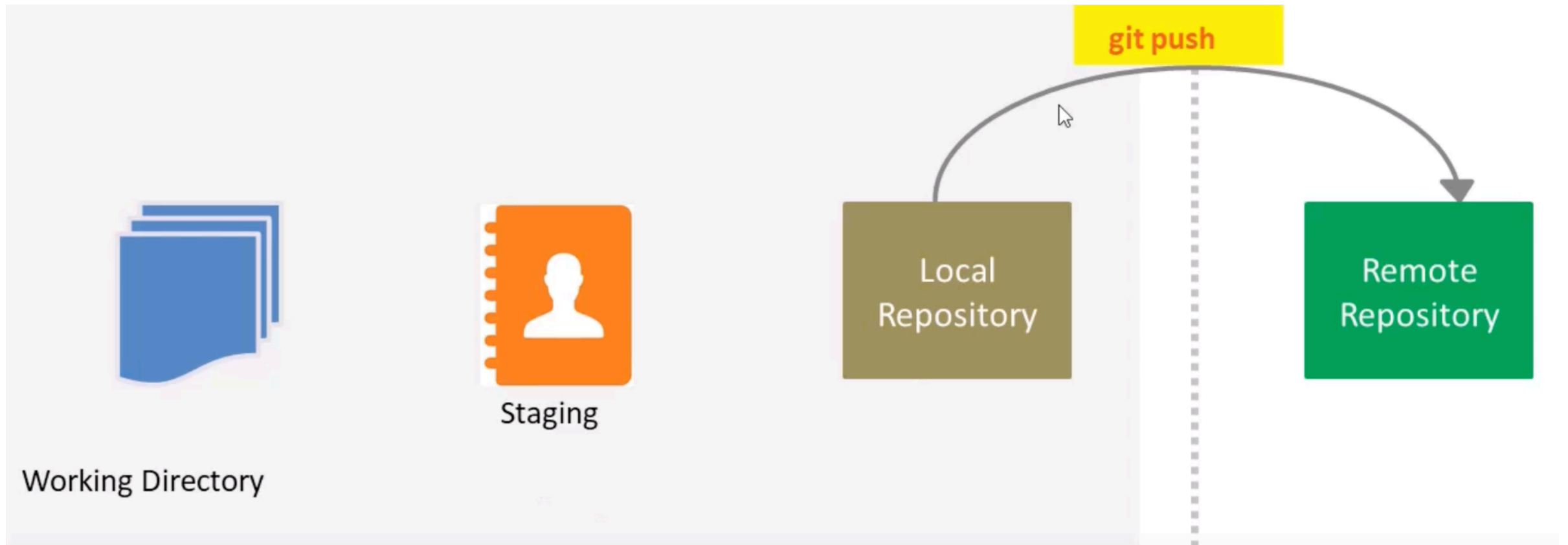
- Commit is to moving the file to local repo – `git commit`



GIT WORKFLOW

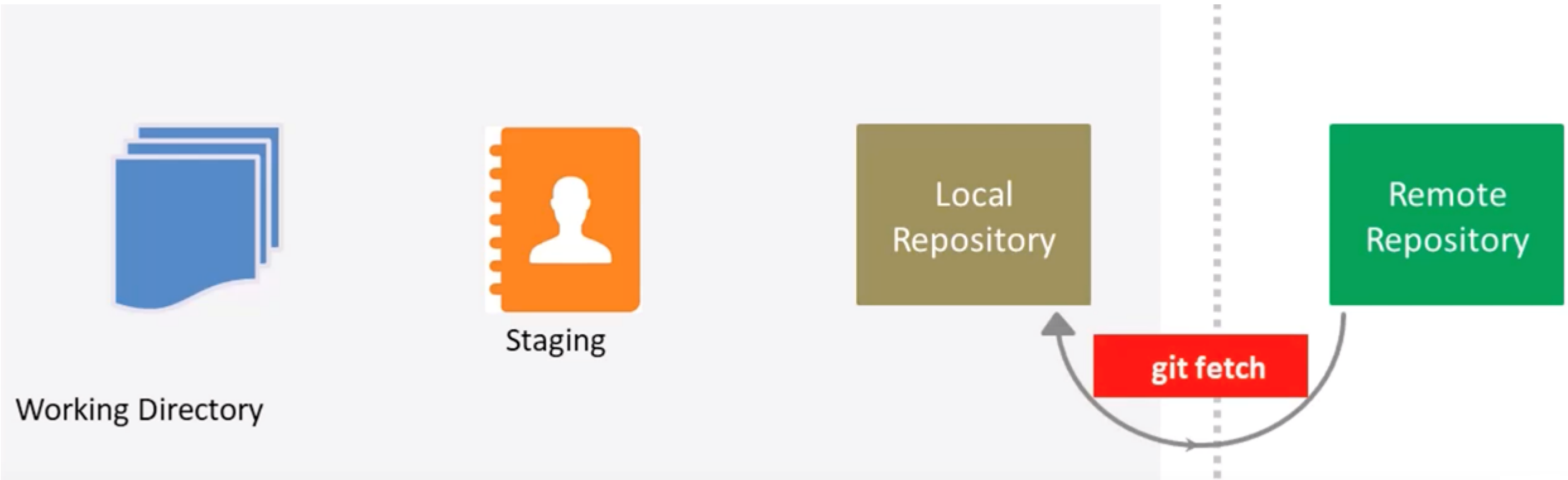
Remote repository

Step 5– git push



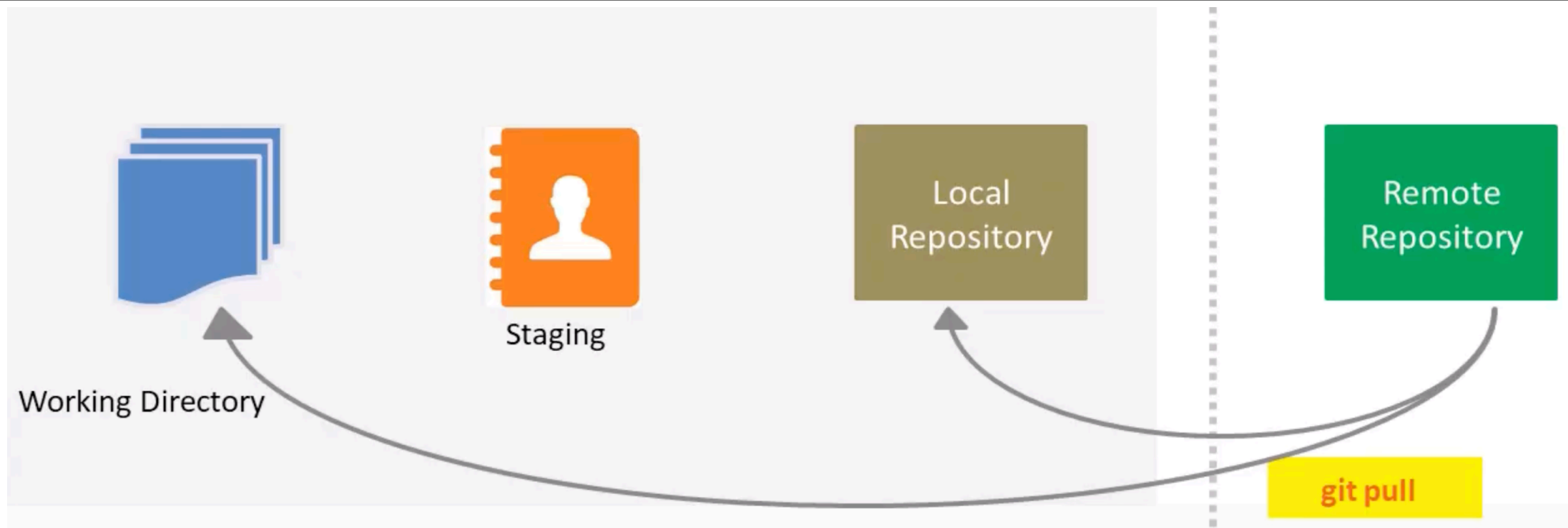
git fetch

Step 6a – git fetch



git pull

Step 6b – git pull



`git fetch`

+

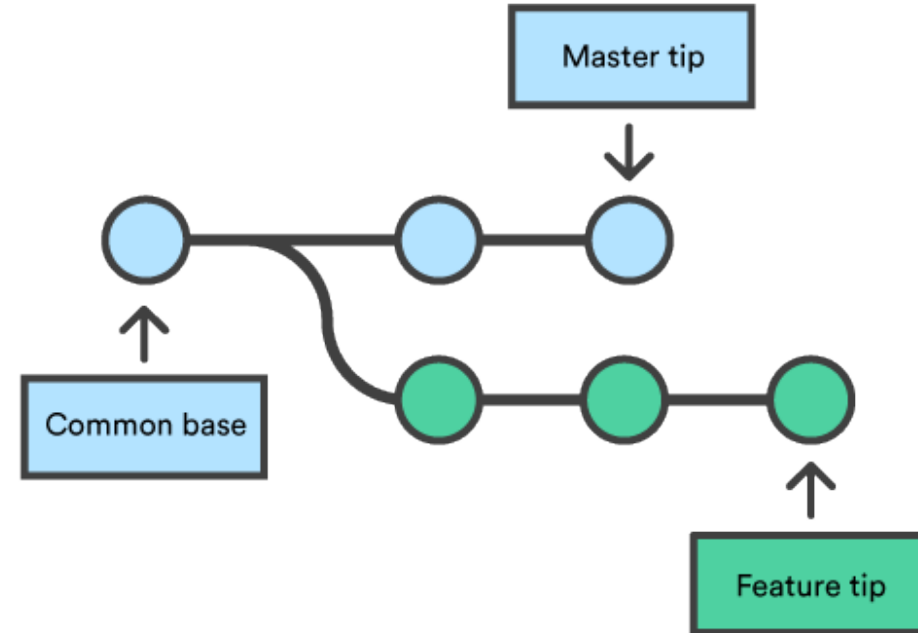
`git merge`

=

`git pull`

Branching

- To add a new feature
- To fix a bug
- Harder for unstable code to get merged into the main code base
- Make it easy for multiple people to work on a feature set
- Chance to clean up your feature's history before merging it into the main branch.



REPOSITORY MANAGEMENT SERVICES



Repository management services are third-party web applications that wrap and enhance a version control system.

A good repository management service will provide tools for discussing, managing, measuring and monitoring software development in an efficient and organized way.

Launched in 2008

Acquired by Microsoft in 2018

Early adoption of social features:
social networking site for software
developers, ability to easily follow
other users.

Public repository discovery
functions.

Intuitive, simple and sane

Not open source

Support open source projects

Offers free public and private
repositories



GitHub



GitLab

Launched in 2011

Open-source alternative to GitHub

Supports entire DevOps lifecycle

Used by IBM, Sony, NASA, and Alibaba

Public repository discovery functions.

Open core

Support open source projects

Offers free public and private repositories

Launched in 2008

Acquired by Atlassian 2011

Integrates smoothly with other services from Atlassian

Designed for professional teams

Ability to easily follow other users.

Public repository discovery functions.

Supports Mercurial repositories.

Not open source

Support open source projects

Offers free public and private repositories





Similarities in basic features

- Pull request
- Code review
- Integrated editing
- Bug tracking
- Markdown support
- Two-factor authentication
- API with extended features
- Forks/clones of repositories
- Snippets
- 3rd party integrations
- Advanced permission management



Free cloud-hosted plan

GitHub:

- Unlimited number of public and private repositories. Limit of 3 collaborators on private repositories.

Bitbuckets:

- Unlimited number of public and private repositories. Limit of 5 collaborators on private repositories.

GitLab:

- Unlimited number of public and private repositories. Unlimited number of collaborators on private repositories.



Paid cloud-hosted plan

GitHub:

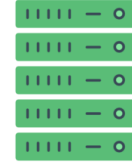
- Individual pro account: \$7 per user/per month.
- Team plan: \$25 per month for 5 users. Additional users: \$9 per user/per month.

Bitbucket:

- Standard plan: \$15 per month for 5 users. Additional users: \$3 per user/per month.
- Premium plan: \$30 per month for 5 users. Additional users: \$6 per user/per month.

Gitlab:

- Bronze (\$4 per user/per month)
- Silver (\$19 per user/per month)
- Gold (\$99 per user/per month).



Paid self-hosted plan

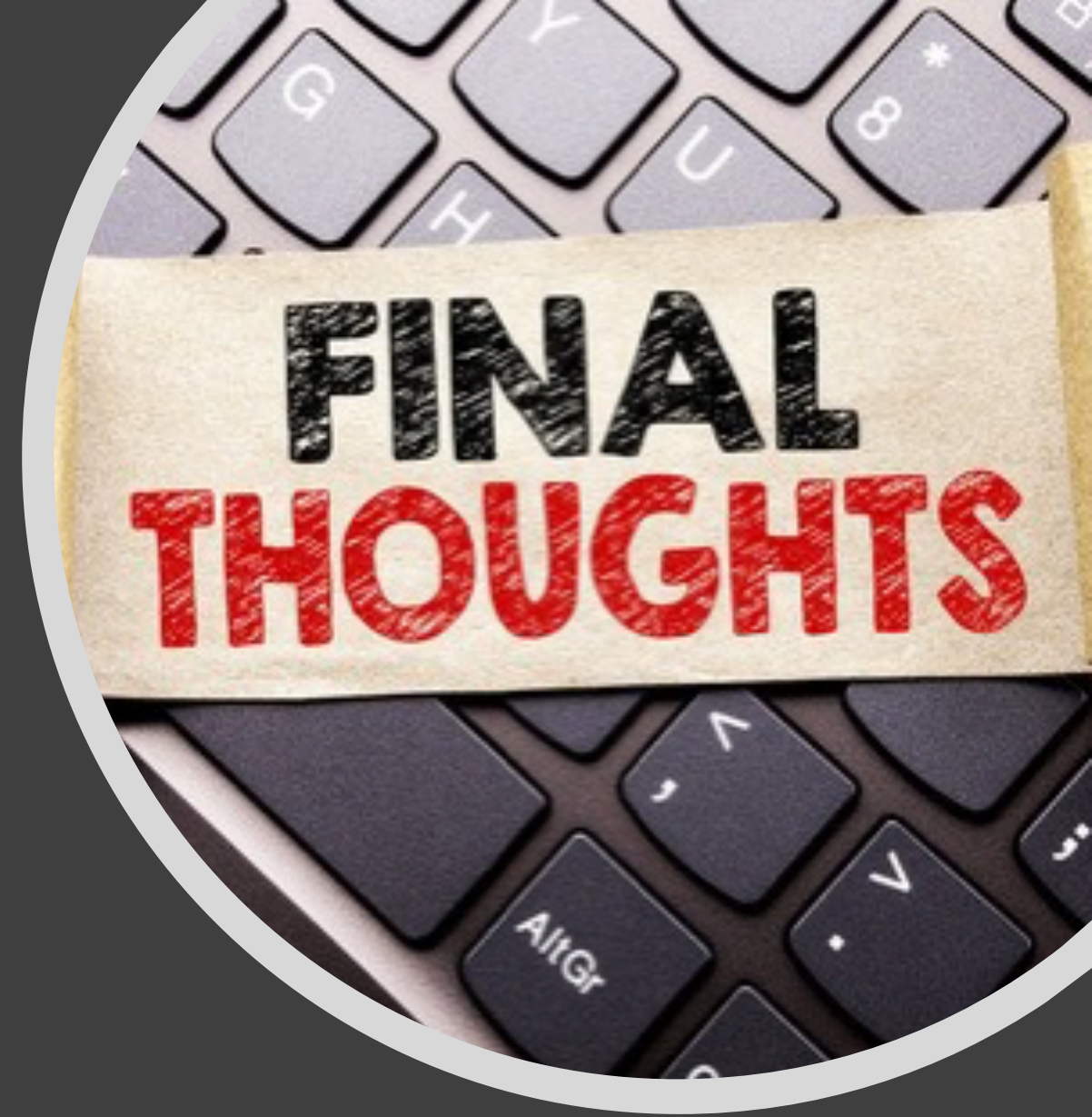
GitHub, GitLab, and Bitbucket self-hosted versions come with several enhanced features compared to their cloud-hosted counterparts.

CONCLUSION

We cannot announce one service to be ultimately superior to the others. All of them are powerful and feature rich services.

In certain scenarios we can nevertheless recommend a certain service:

- when working on open source projects, GitHub makes sense
- when you also want to use other products from Atlassian like Confluence or Jira, Bitbucket seems like a good choice
- when you want an open source solution, you should go with GitLab.



Any
1.1
Questions