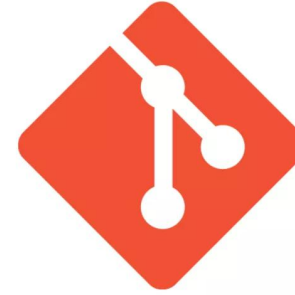
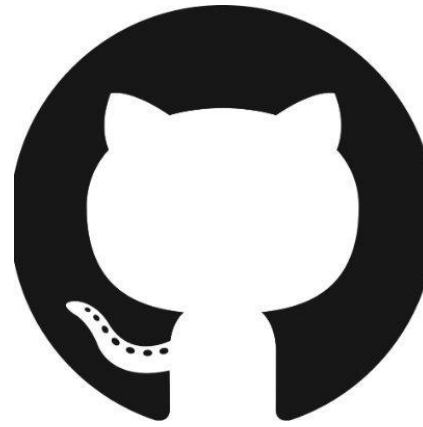


GIT AND GITHUB

VERSION CONTROL SYSTEM

Fatih Sevban UYANIK



git

What is VCS?

- Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.
- 'Version' means the saved status of a set of files in a certain time.



Project



Project Final



Project Last
Final




Project True Last
Final



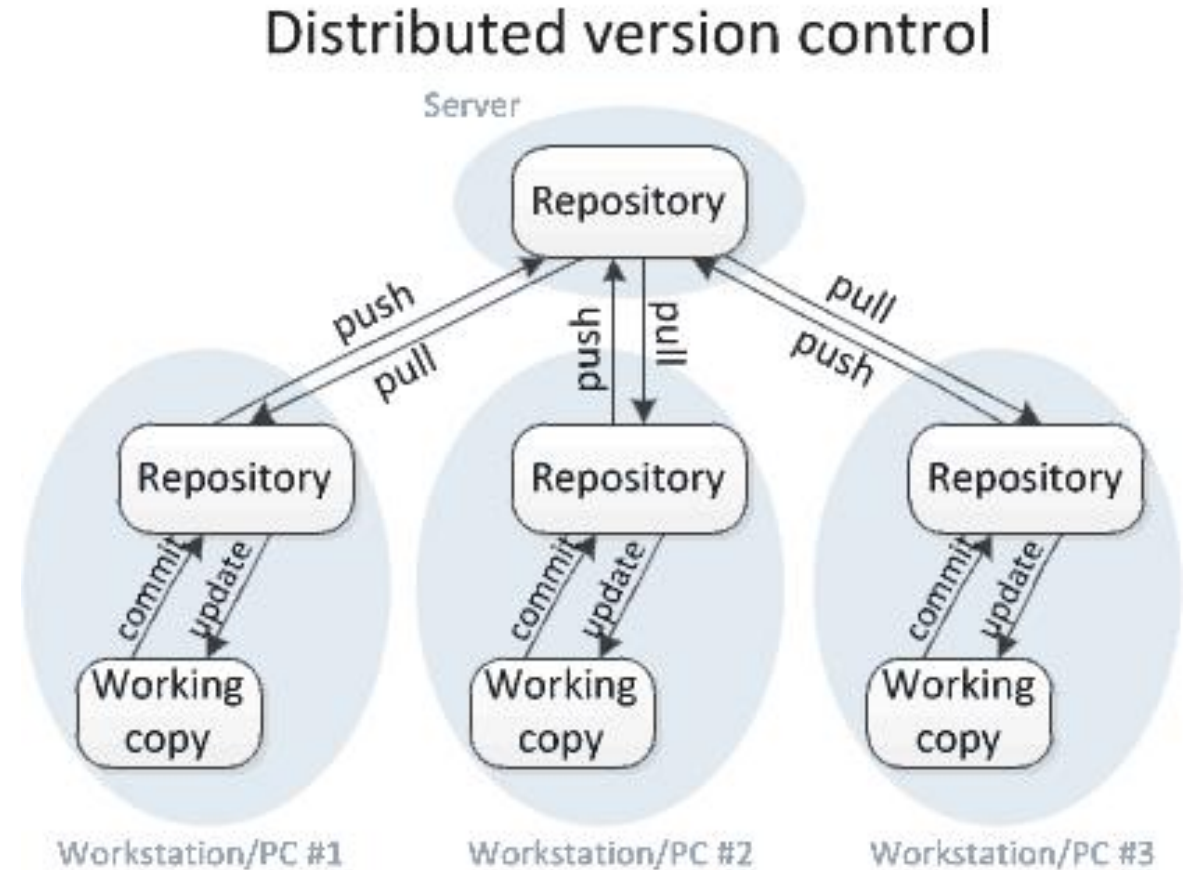
I swear, This is
Final Project

Why VCS?

- Enables to work as a team in a software project.
 - Enables to make changes easier and merge them.
 - Enables to resolve conflicts in the changed places.
 - Enables to return back to previous versions.
 - Enables to backup your software project.
- 

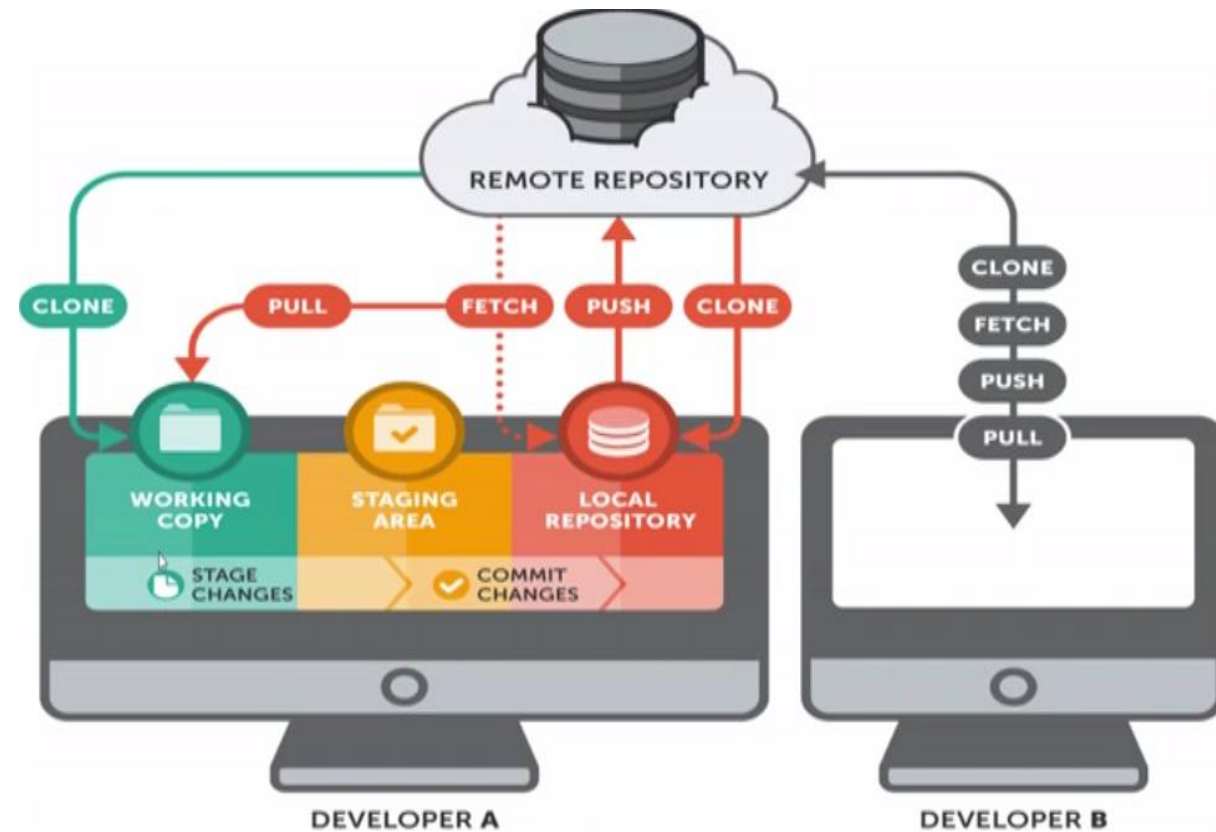
Git Concepts

- **Repository:** A Git repository is a folder containing multiple versions of a software project. This repository tracks all changes made to files and builds history over time.
- **Working Copy:** The directory of the project that you are working with. A specific version can be retrieved to working copy.



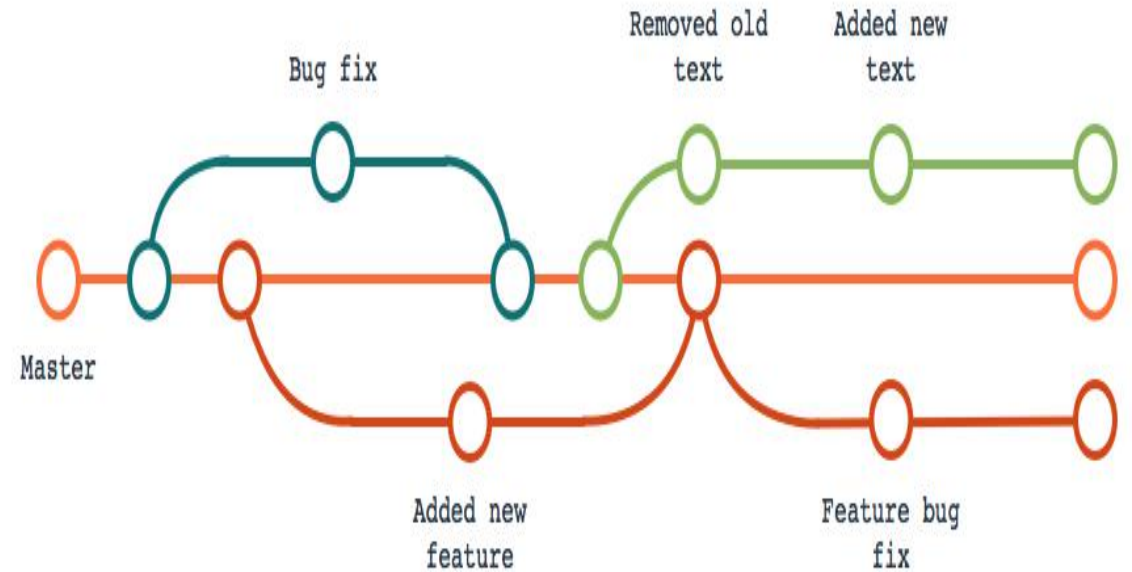
GIT CONCEPTS

- **Staging Area:** Imagine a box. You can put stuff into the box. You can take stuff out of the box.
- **Commit:** Adds the latest changes to the repository with a corresponding message.



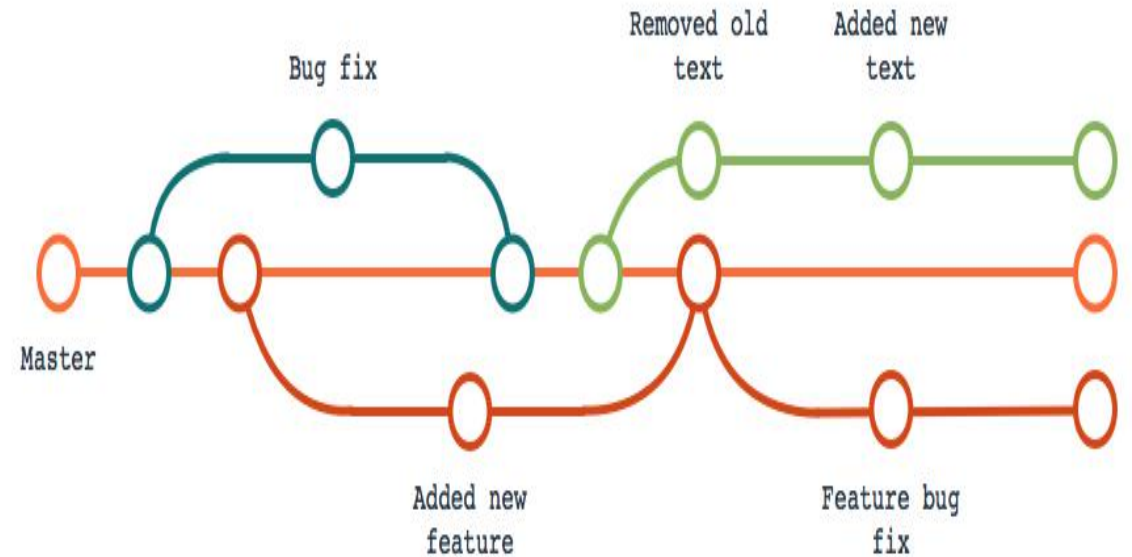
GIT CONCEPTS

- **Branching:** Branching is the practice of creating version copies of the software project in order to work in parallel versions.
- Branching means that you diverge from the main line of development and continue to do work without messing with the main line.

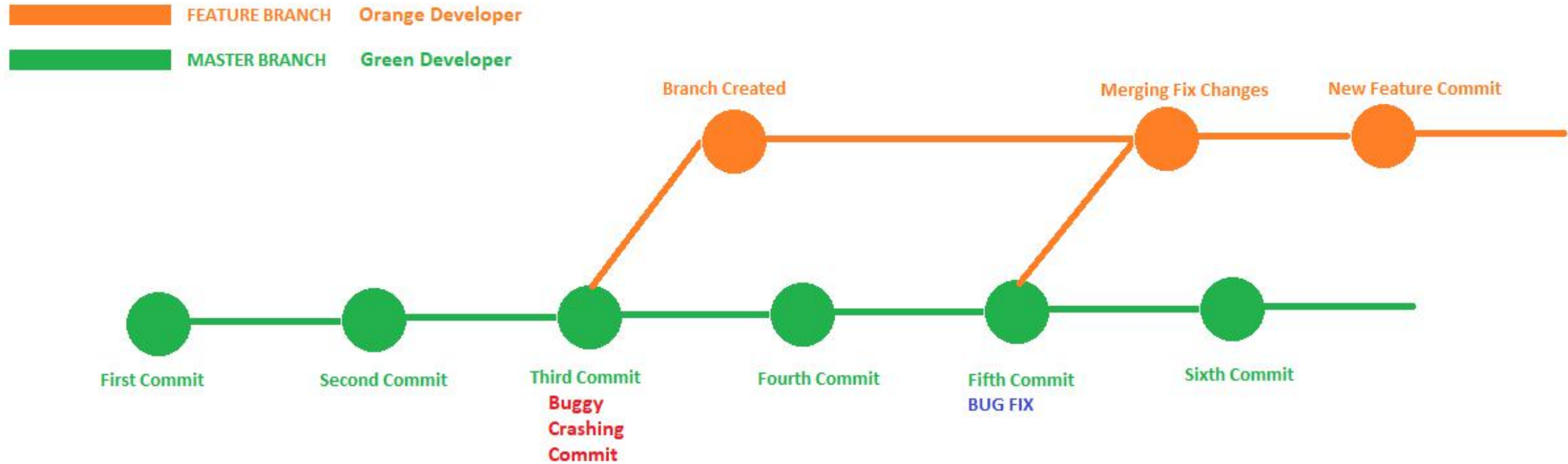


WHY BRANCHING?

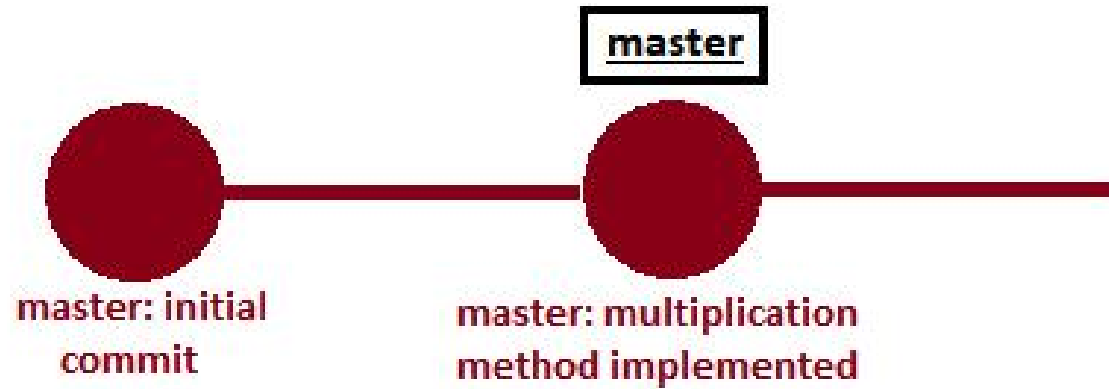
- Having a development effort such as refactoring, evolution or bug-fixes.
- Realizing that changes cannot be made safely make in the current development branch.
- Because of breaking the API, or introducing code that would break everything.
- Then, a new branch is needed.



UNDESIRED SCENARIO

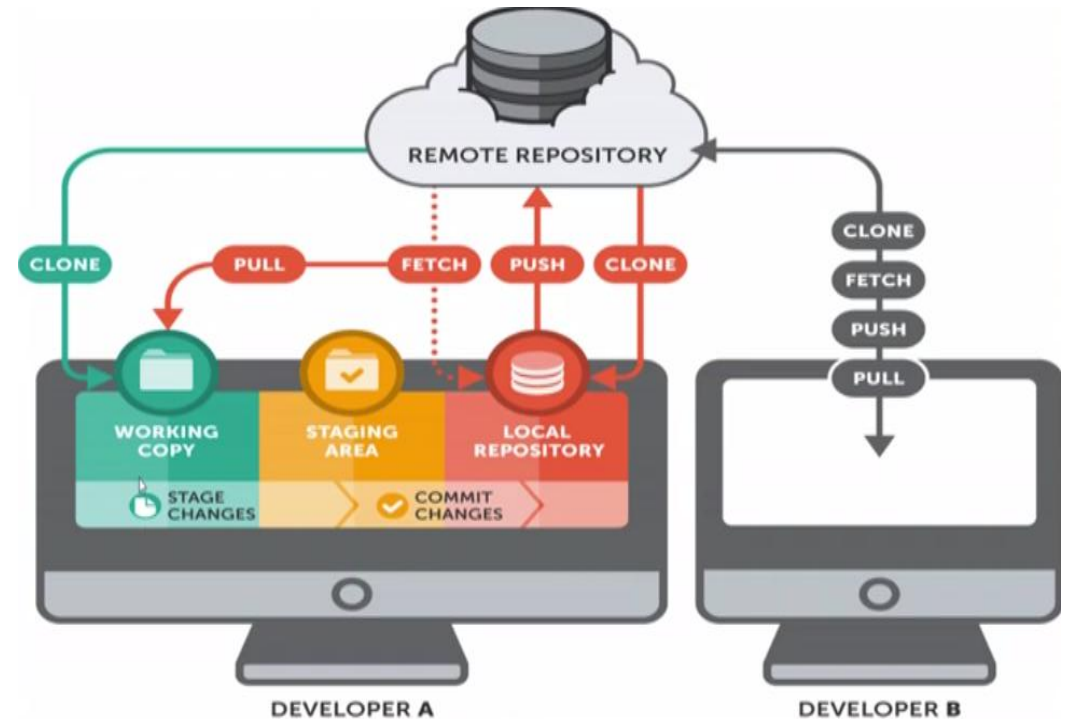


Local Repository Git Commands



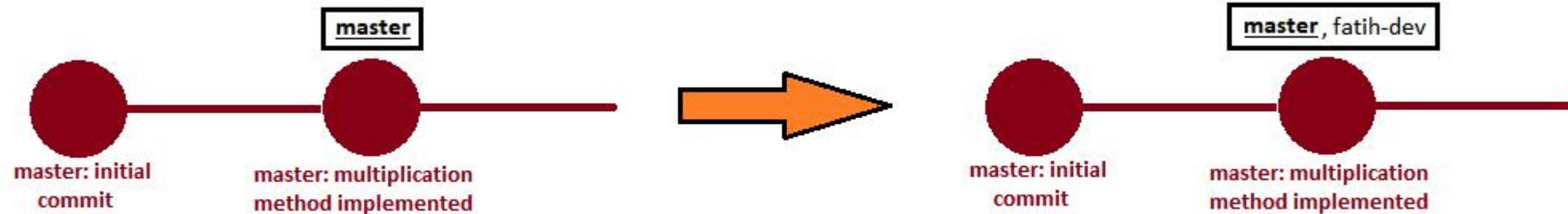
Local Repository Git Commands

- **git init:** initializes the git software in the project.
- **git status:** shows the status of the working copy and staging area.
- **git add <filename>:** adds the specified file or filesets to staging area.
- **git commit -m <message>:** commits the code with a message.
- **git log (--decorate --online --graph):** shows the current repository history.

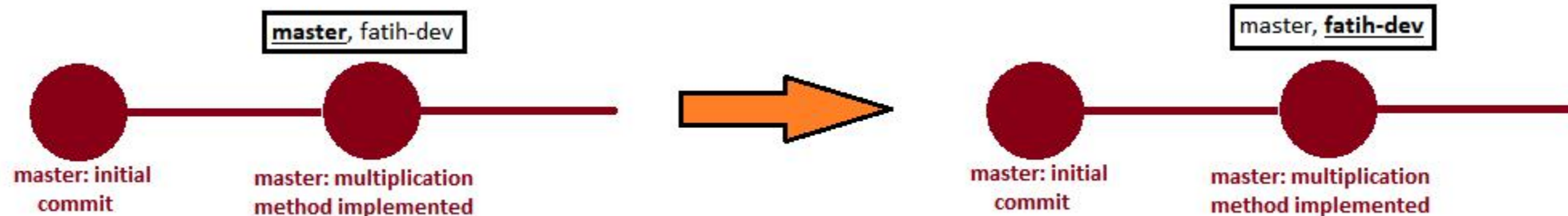


Local Repository Git Commands

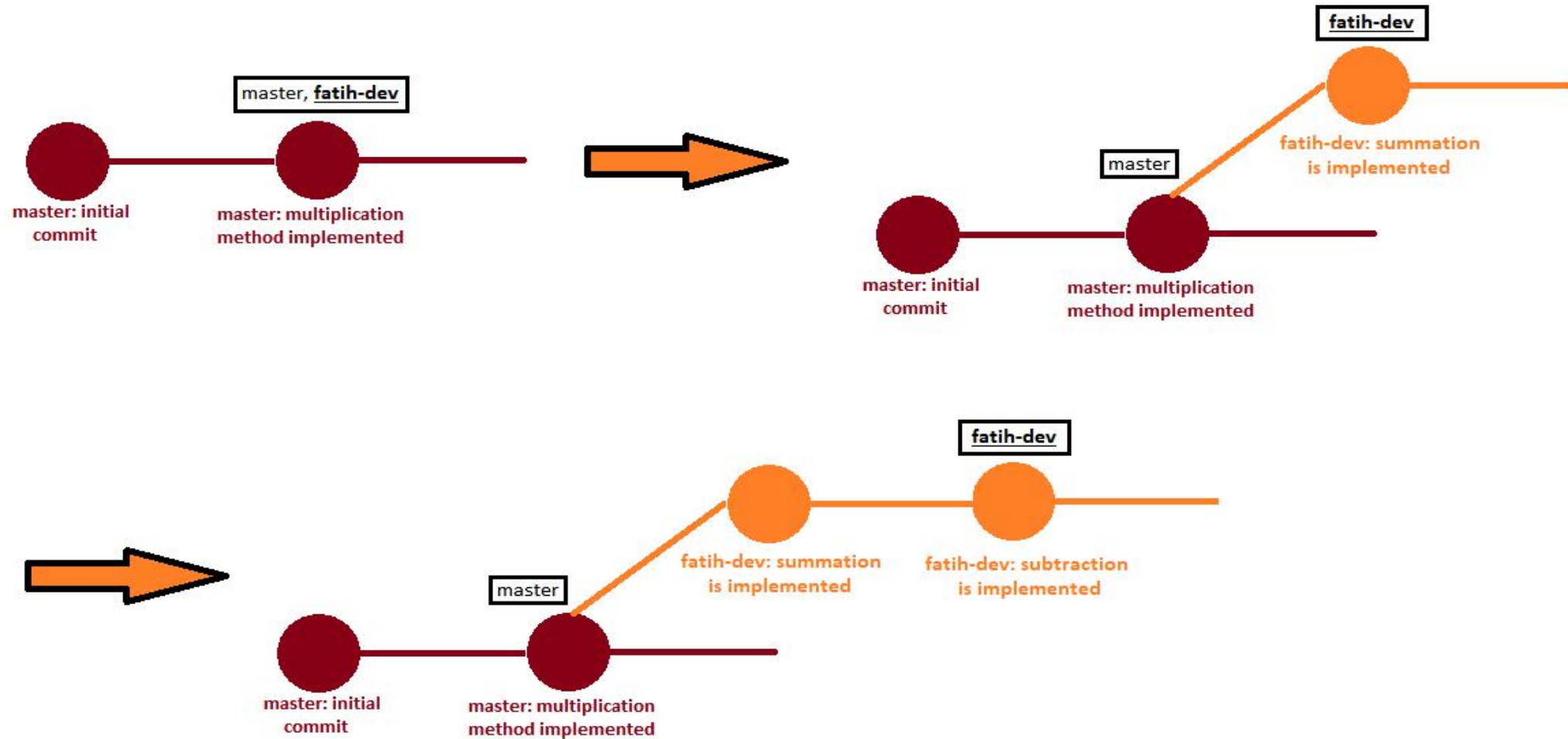
- **git branch**: shows all the branches and the active one.
- **git branch <branchname>**: creates a new branch.



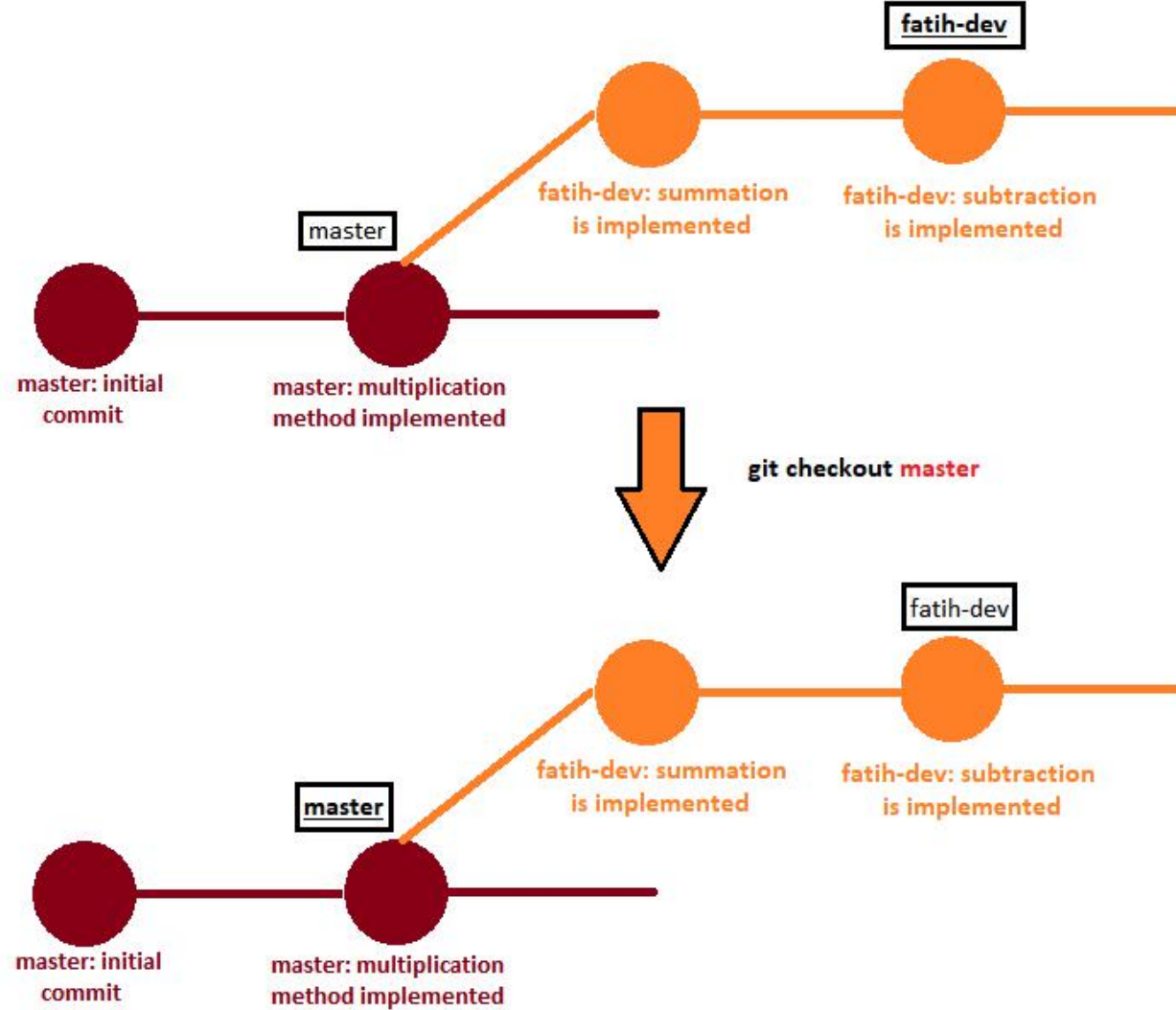
- **git checkout <branch-name>**: checkouts to another branch.



Local Repository Git Commands

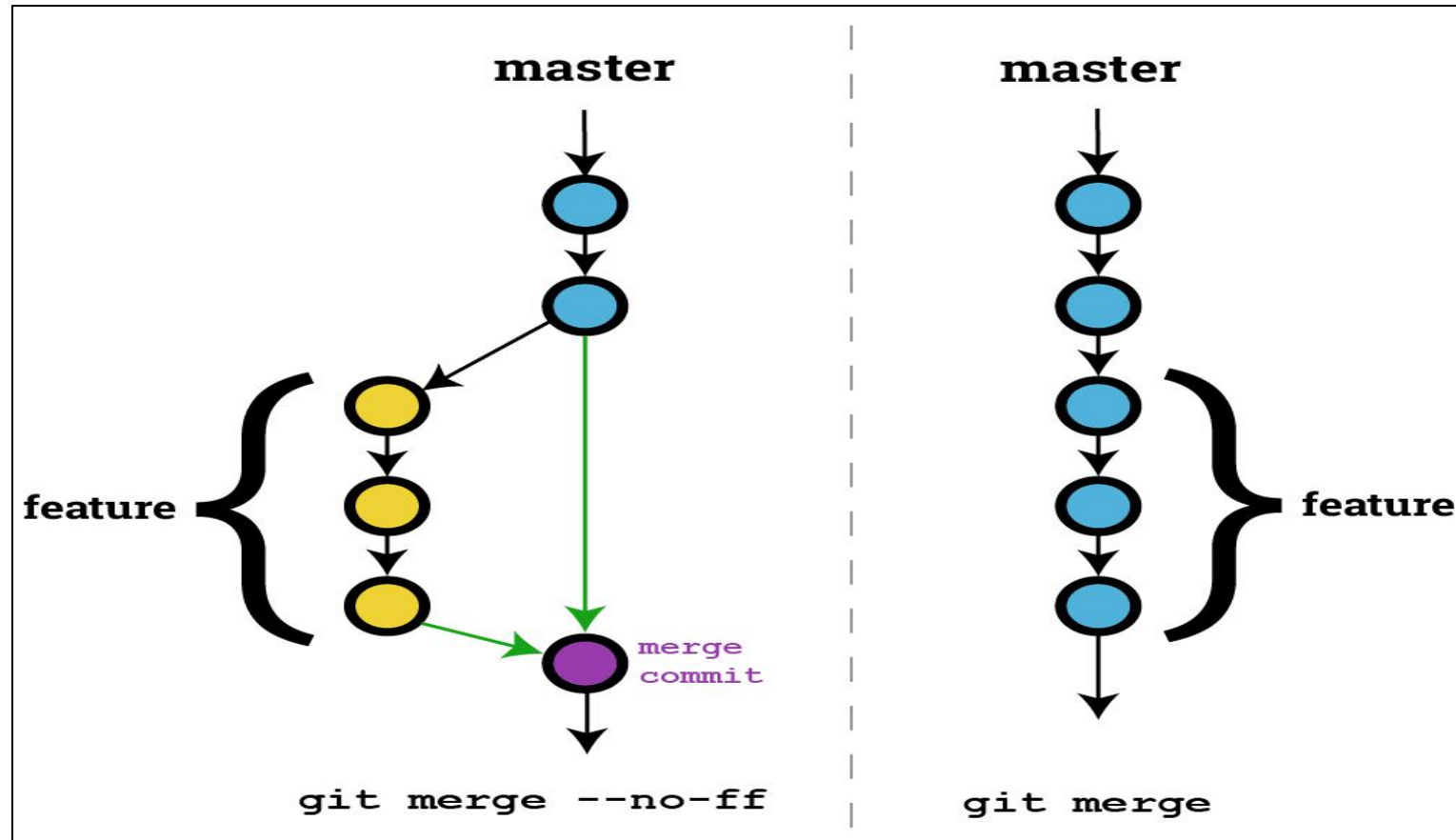


Local Repository Git Commands

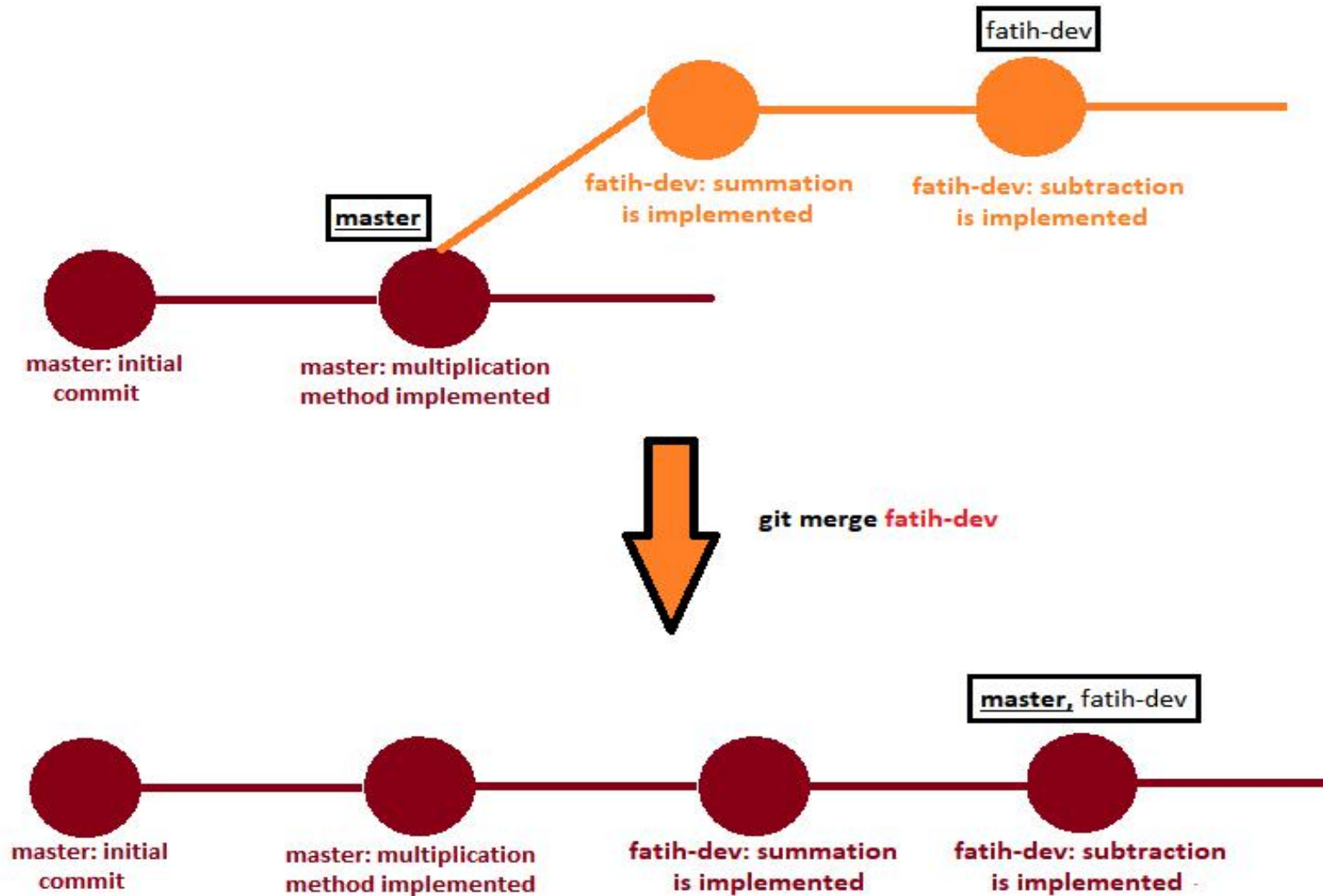


FAST FORWARD MERGE VS NO FAST FORWARD MERGE

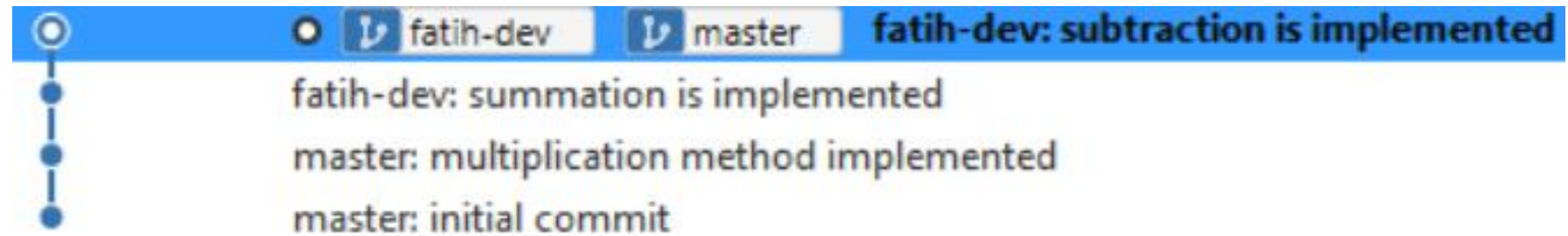
- **git merge --no-ff:** merges the branches with no fast forwarding.
- **git merge:** merges the branches with fast forwarding.



Local Repository Git Commands



SourceTree: viewing commits



SourceTree: checkout

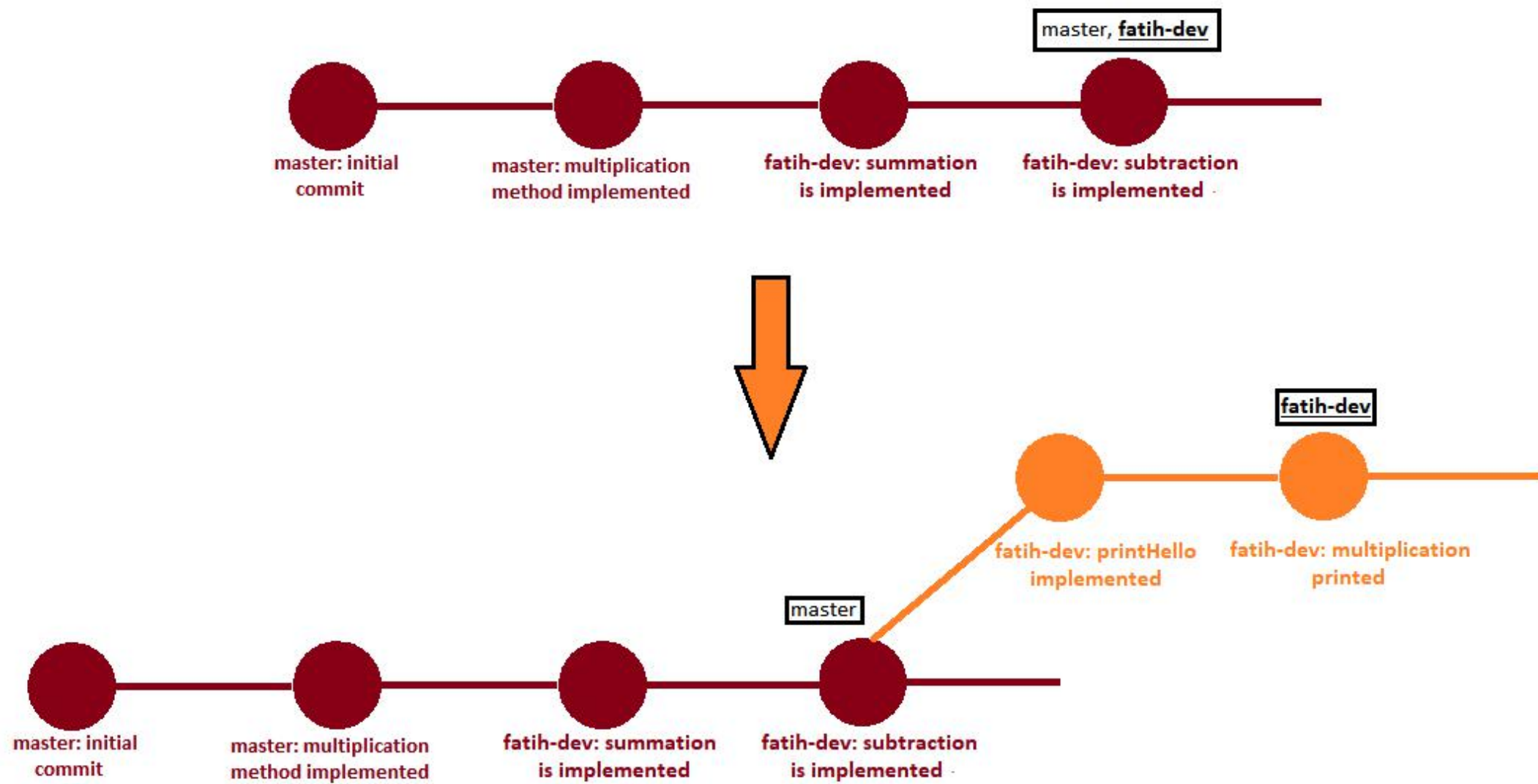


double click



checking out to **fatih-dev**





SourceTree: commit

Staged files

Unstage AllUnstage Selected

STAGING AREA

Unstaged files

Stage AllStage Selected

src/Main.java

+

CHANGED FILES IN
WORKING COPY

Select a file to view the diff

Ffatihsevban <fatihsevban15@gmail.com>

Commit options...

COMMIT MESSAGE

☐ Push changes immediately to -

Commit

SourceTree: commit

Staged files

Unstage All Unstage Selected

src/Main.java

STAGING AREA

Unstaged files

Stage All Stage Selected

CHANGED FILES IN
WORKING COPY

src/Main.java

Hunk 1 : Lines 20-26

Unstage hunk

20 20}
21 21 ...
22 22public static void printHello(int name) {
23// TODO
24System.out.println("Hello");
25 23+.....System.out.println("Hello" + name);
26 24}
27 25
28 26}
29 27No newline at end of file

fatihsevban <fatihsevban15@gmail.com>

Commit options...

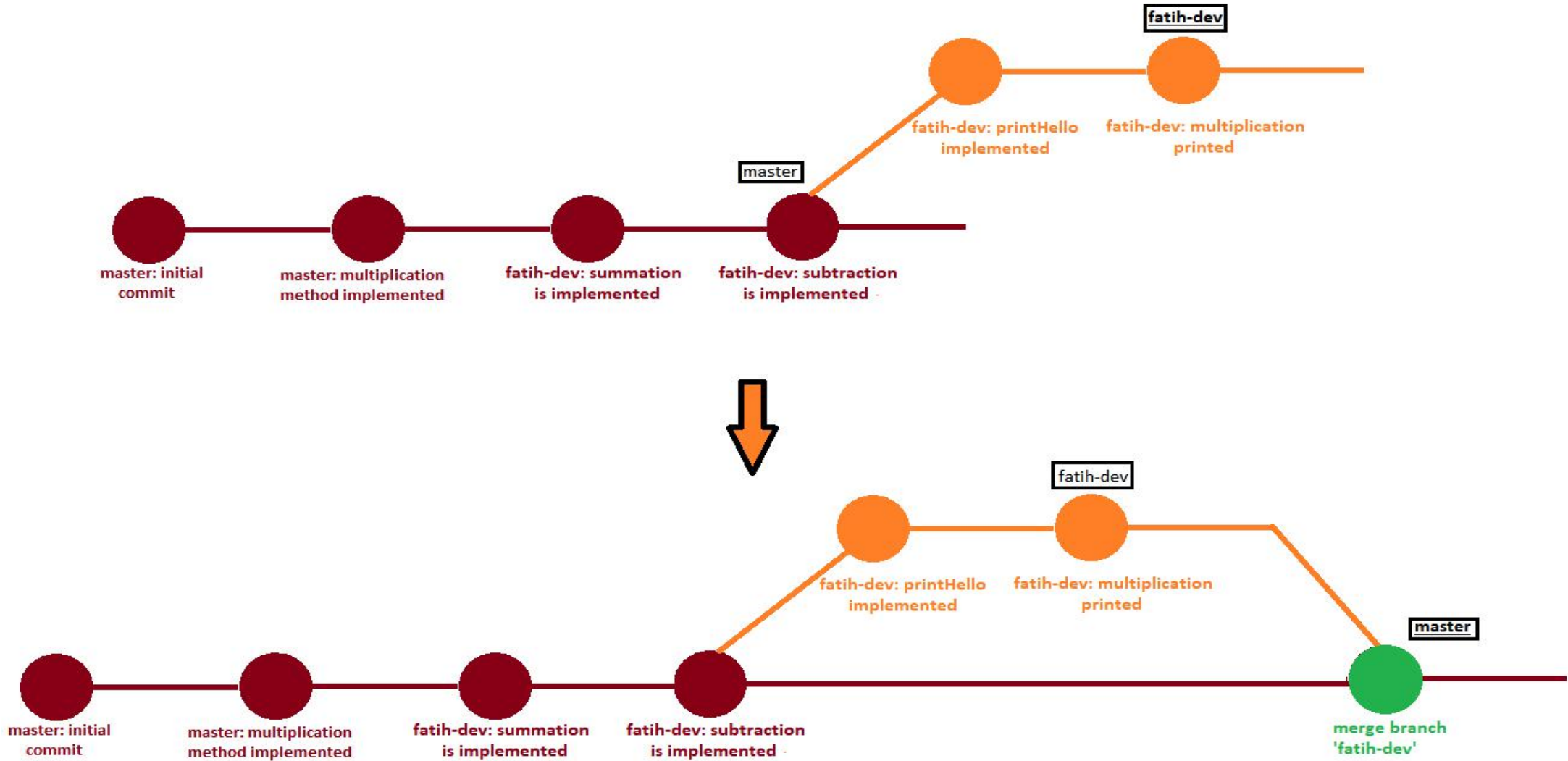
fatih-dev: printHello implemented

COMMIT MESSAGE

☐ Push changes immediately to -

Commit

SourceTree: no fast forward merging



SourceTree: no fast forward merging

Pick a commit to merge into your current branch

All Branches ☒ Show Remote Branches Date Order Jump to:

Graph	Description	Date	Author	Commit
	fatih-dev: multiplication printed	19 Feb 2020 23:53	fatihsevban <fatih:	64e27c5
	fatih-dev: printHello implemented	19 Feb 2020 23:48	fatihsevban <fatih:	27617c4
	fatih-dev: subtraction is implemented	19 Feb 2020 19:30	fatihsevban <fatih:	27f2fdf
	fatih-dev: summation is implemented	19 Feb 2020 19:30	fatihsevban <fatih:	0b41796
	master: multiplication method implemented	19 Feb 2020 19:28	fatihsevban <fatih:	bfc28fa
	master: initial commit	19 Feb 2020 1:11	fatihsevban <fatih:	6c14688

Sorted by file status

Commit: 64e27c544d1ce82c7da8d1f59663c07ee2bd5bd8 [64e27c5]
Parents: 27617c4d75
Author: fatihsevban <fatihsevban15@gmail.com>
Date: Wednesday, February 19, 2020 11:53:39 PM
Committer: fatihsevban

fatih-dev: multiplication printed

src/Main.java

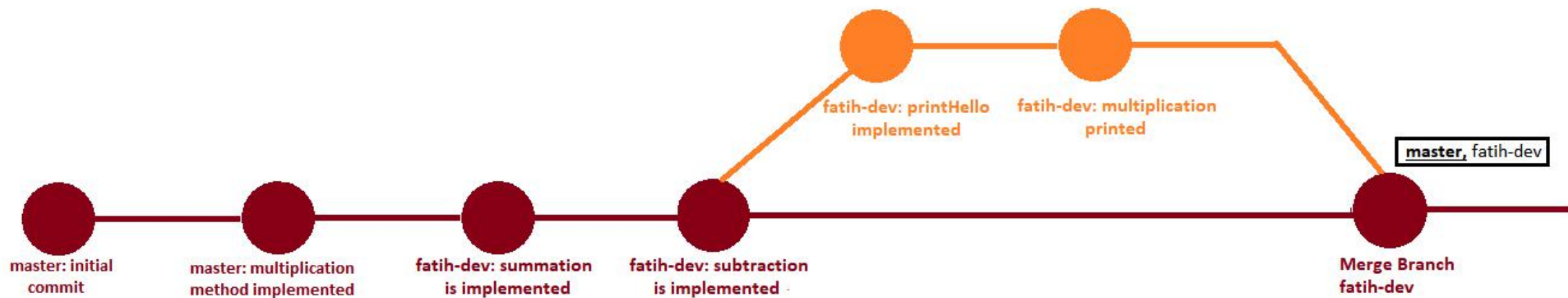
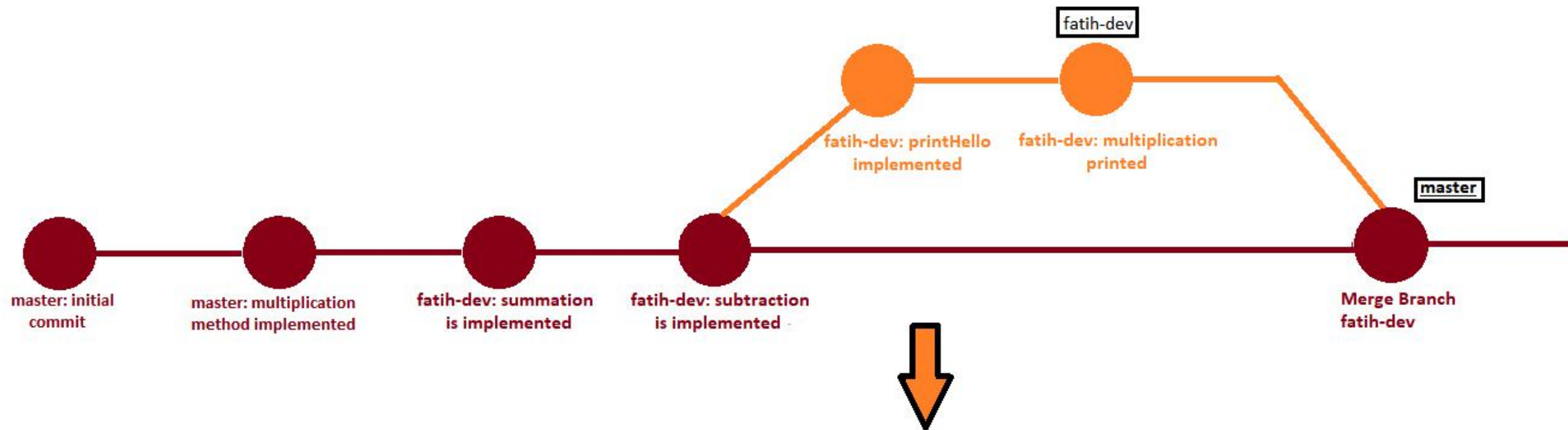
Hunk 1: Lines 5-11 Reverse hunk

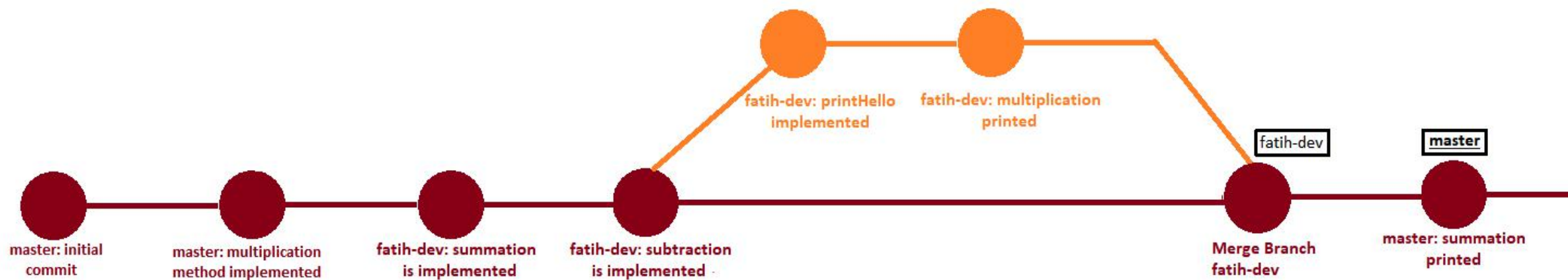
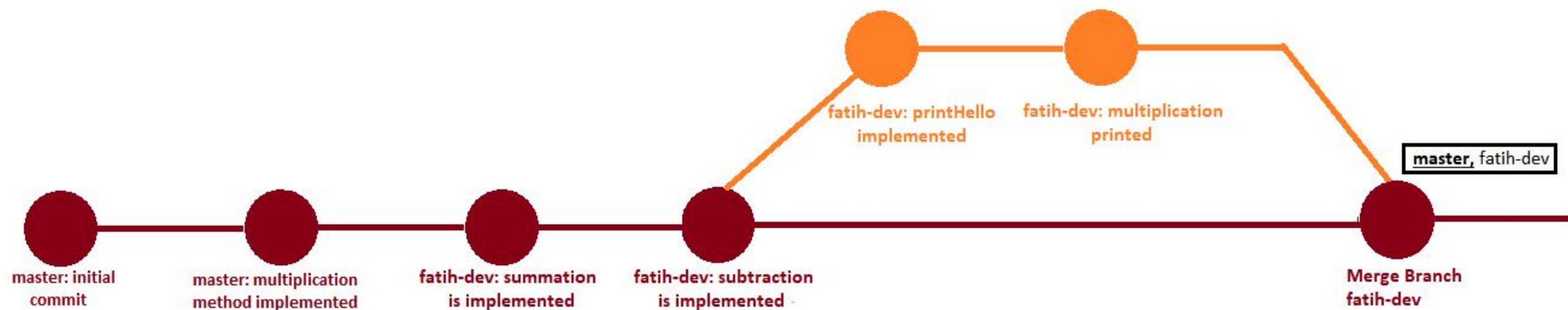
```
5 5 ..... int num2 = 5;  
6 6 ..... System.out.println("Number1: " + num1);  
7 7 ..... System.out.println("Number2: " + num2);  
8 8 ..... + ..... System.out.println("Multiplication: " + multiply(num1, num2));  
9 9 ..... }  
10 10 .....  
11 11 ..... public static int multiply(int num1, int num2) {
```

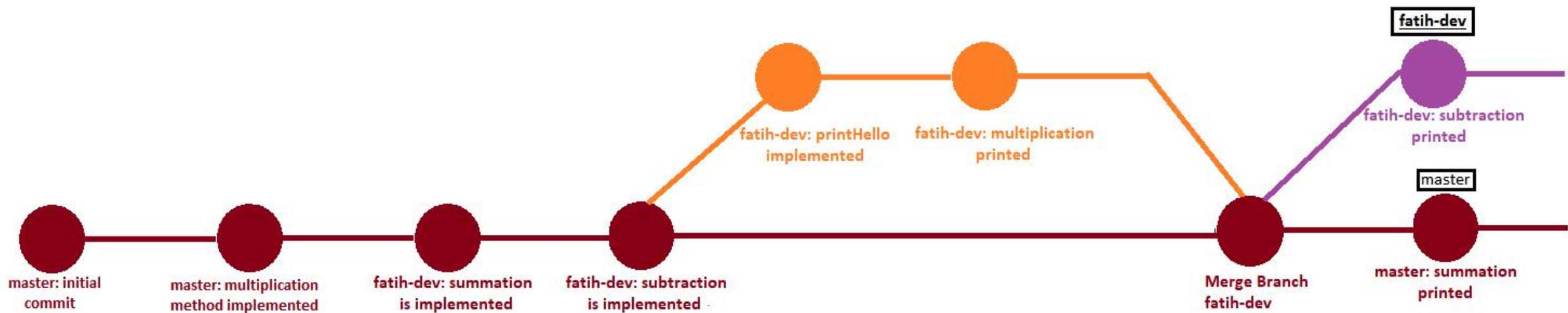
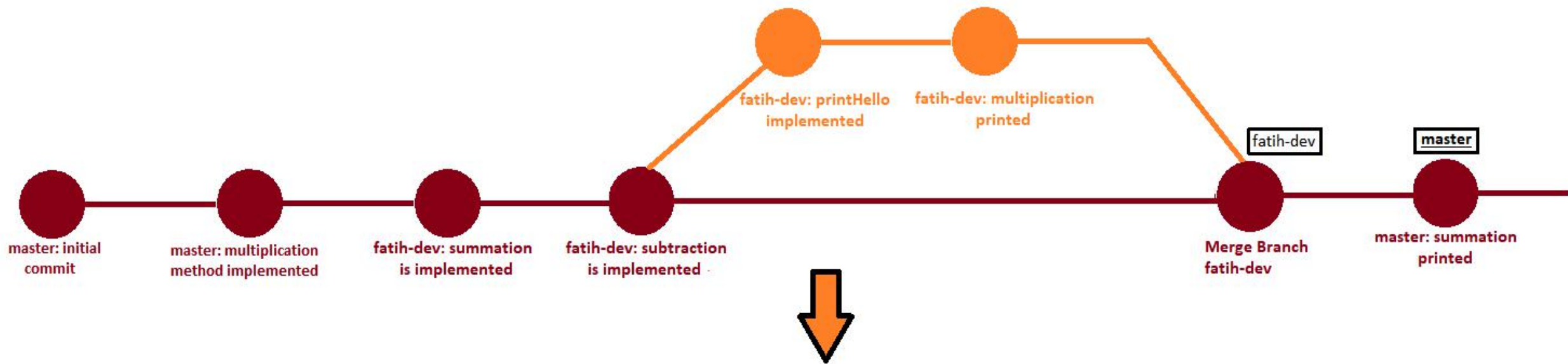
Options

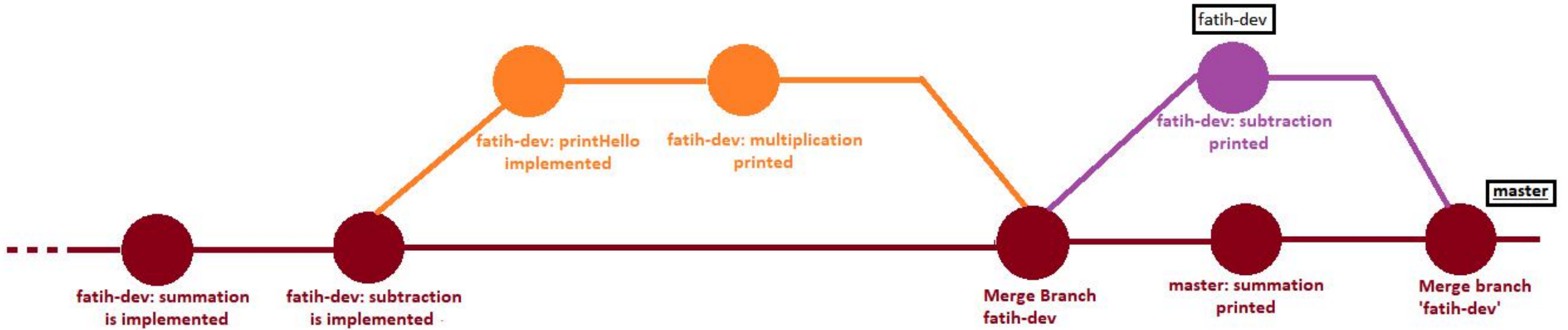
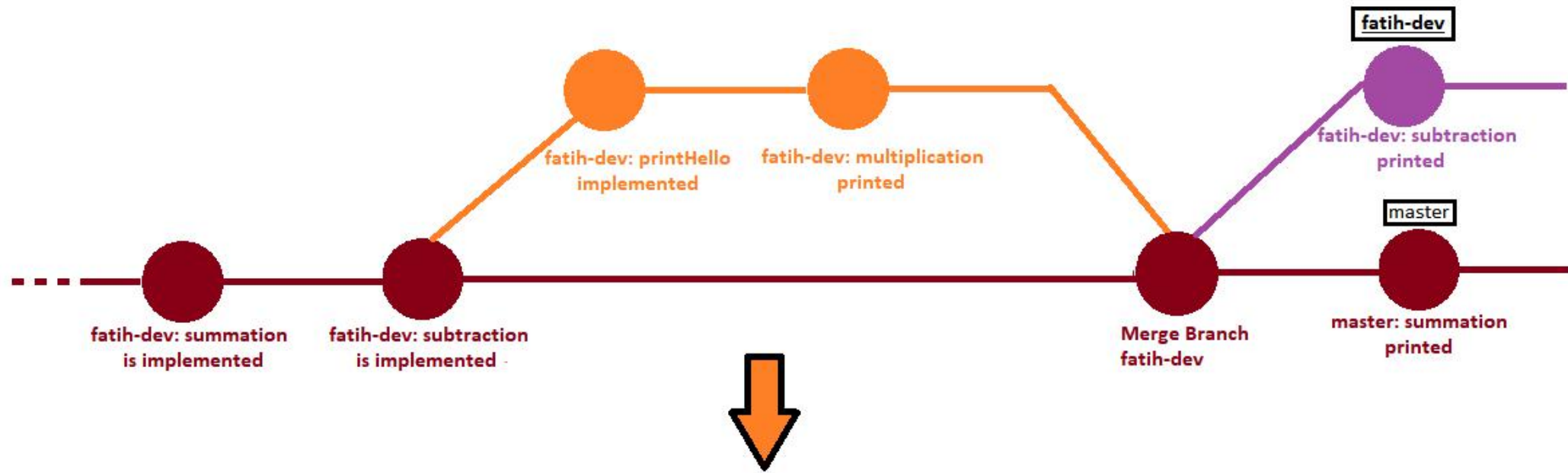
- ☒ Commit merge immediately (if no conflicts)
- ☐ Include messages from commits being merged in merge commit
- ☒ Create a new commit even if fast-forward is possible
- ☐ Rebase instead of merge (WARNING: make sure you haven't pushed your changes)
- ☐ Detect renames with similarity: 90 %

OK Cancel

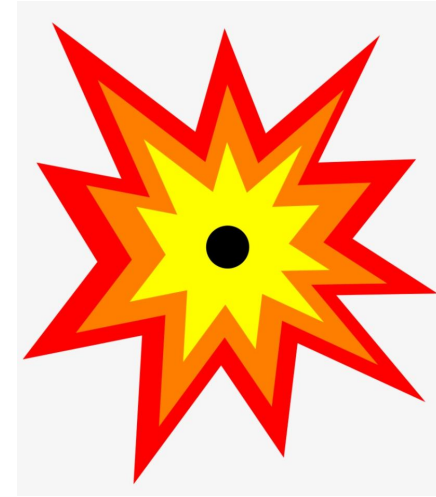
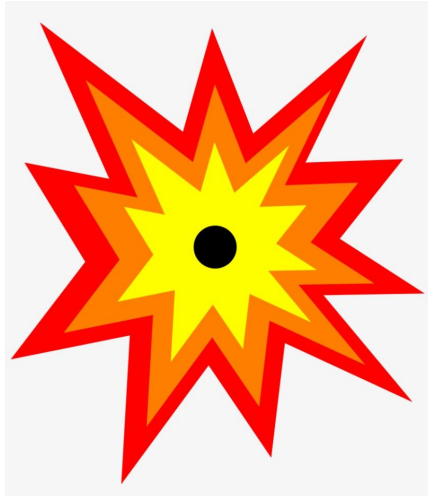








SourceTree: Conflict



```
<<<<<<< HEAD
      System.out.println("Summation: " + sum(num1, num2));
=====
      System.out.println("Subtraction: " + subtract(num1, num2));
>>>>>>> fatih-dev
```

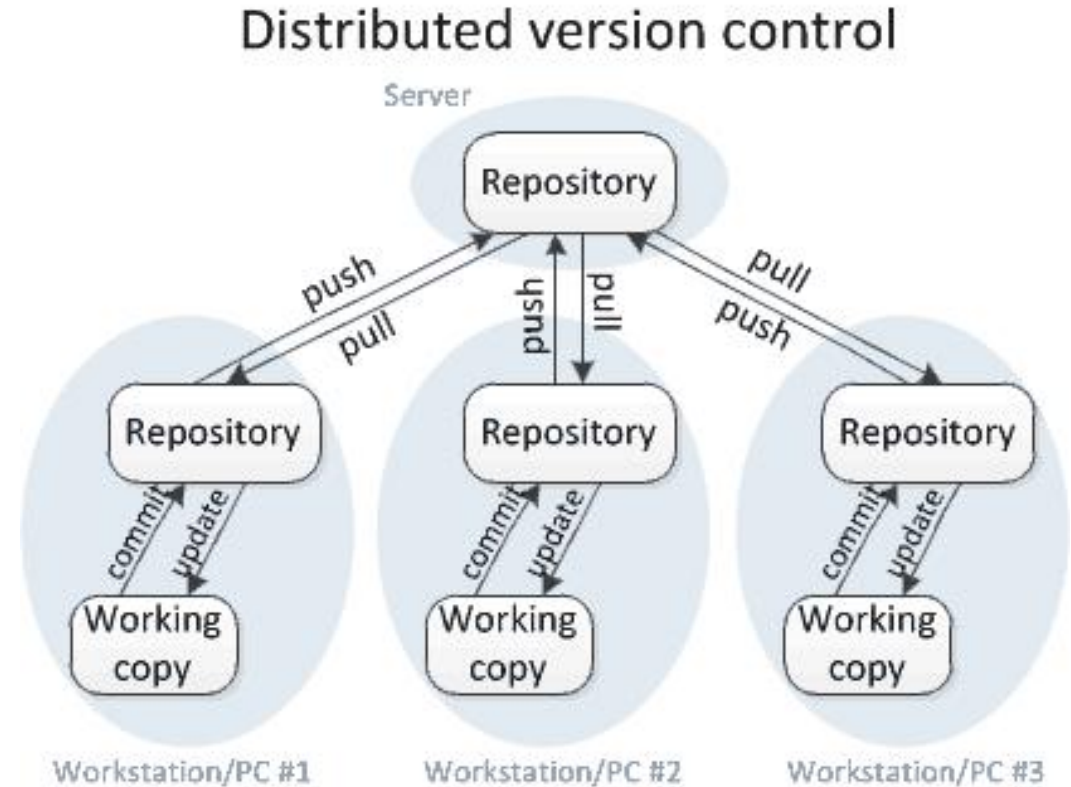


Current Branch Changes

Other Branch Changes

Remote Repository Git Commands

- **git push <branchname>**: command is used to upload local repository content to a remote repository.
- **git fetch**: Fetching imports commits to local branches
- **git pull <branchname>**: A combination of two other commands, git fetch followed by git merge.
- **git clone <repolink>**: clones a repository to your local.



Remote details


Required information

Remote name: ☐ Default remote

URL / Path:

Optional extended integration

Remote Account:

 Generic Account
Generic Host

▼

Legacy Account Settings:

Host Type:

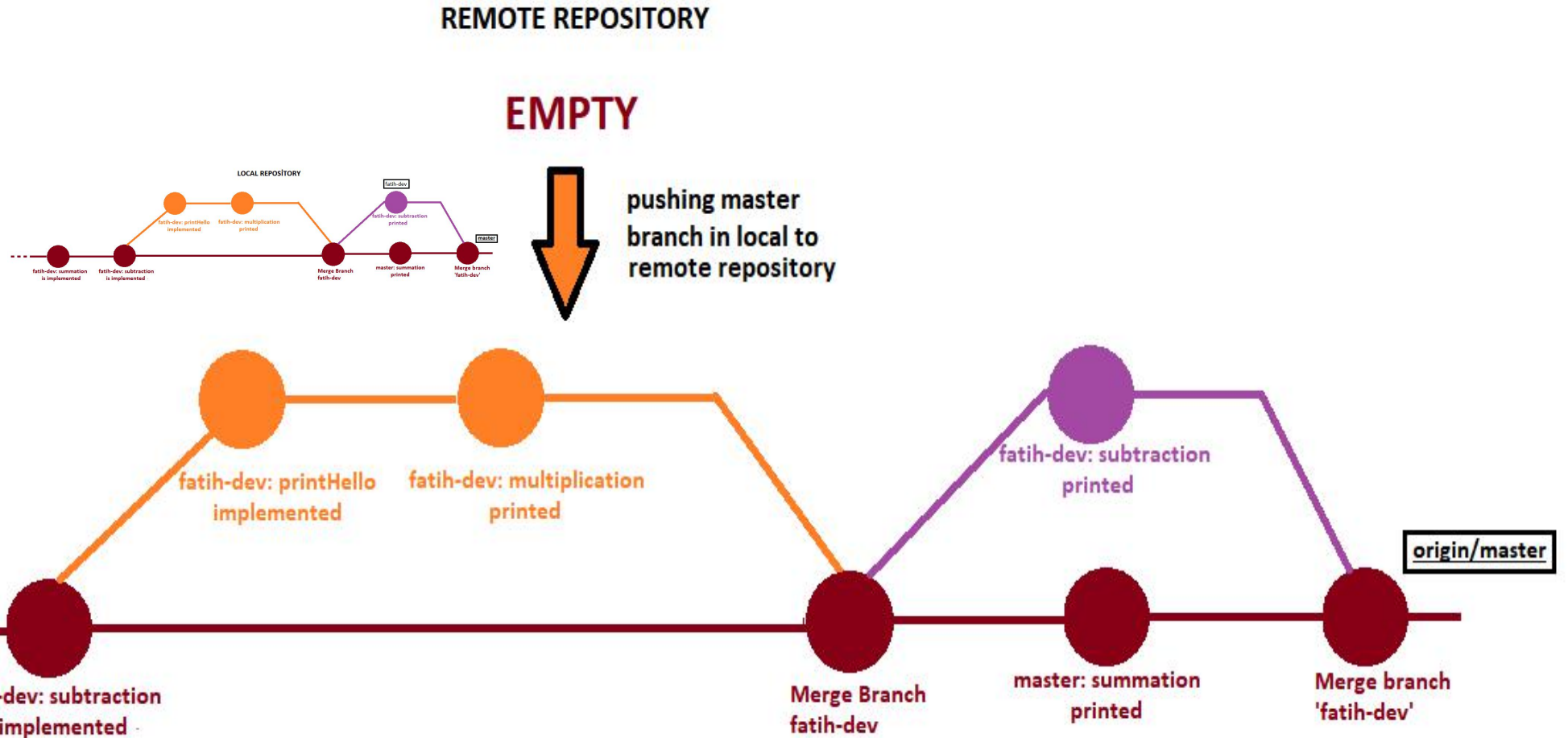
Host Root URL:

Username:

Extended integration is used to enable deeper integration with hosting providers such as Bitbucket, including locating existing clones when following links from sites and creating pull requests.

OK Cancel

Remote Repository Git Commands



Remote Repository: push

Push : GitTutorial

Push to repository: origin ▼ https://github.com/FatihSevbanUyanik/trial.git

Branches to push

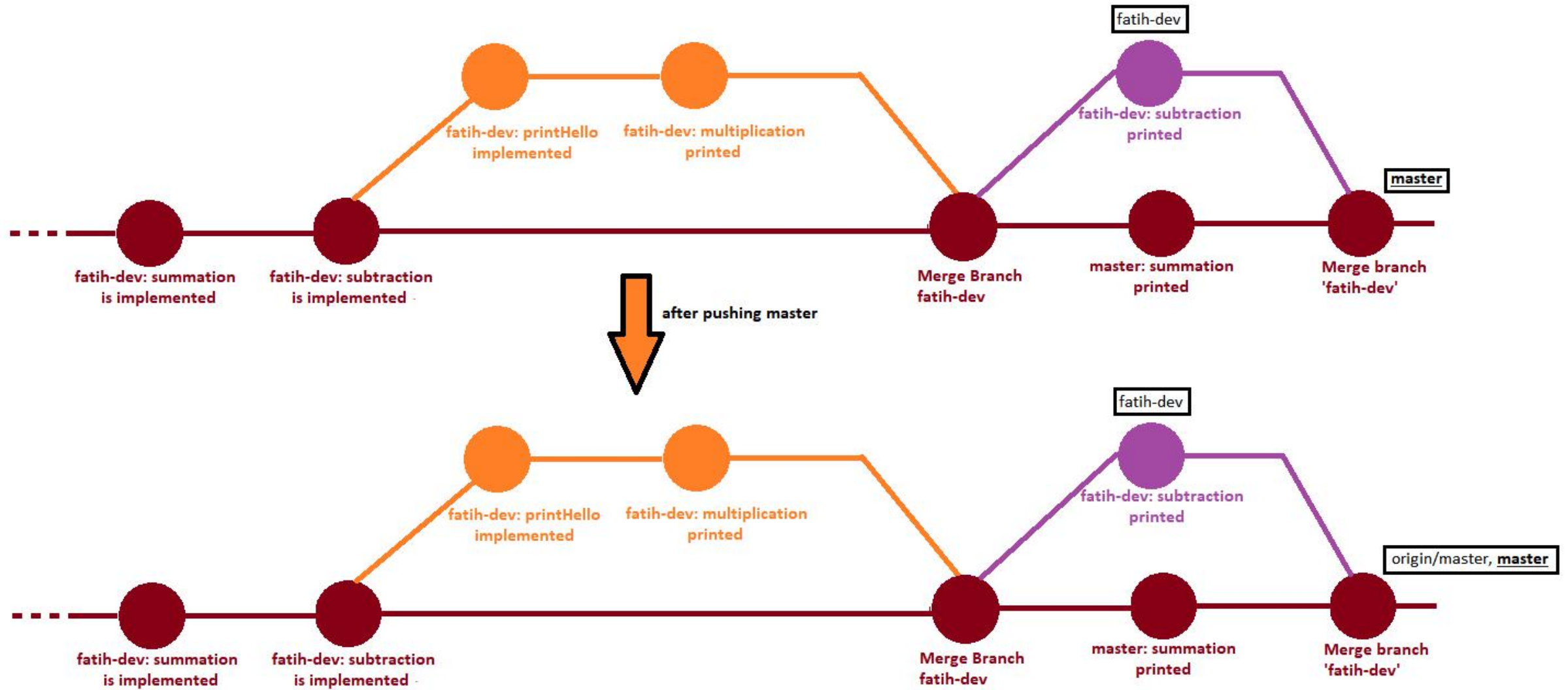
Push?	Local branch	Remote branch	Track?
<input type="checkbox"/>	fatih-dev	▼	<input type="checkbox"/>
<input checked="" type="checkbox"/>	master	master ▼	<input checked="" type="checkbox"/>

☒ Select All

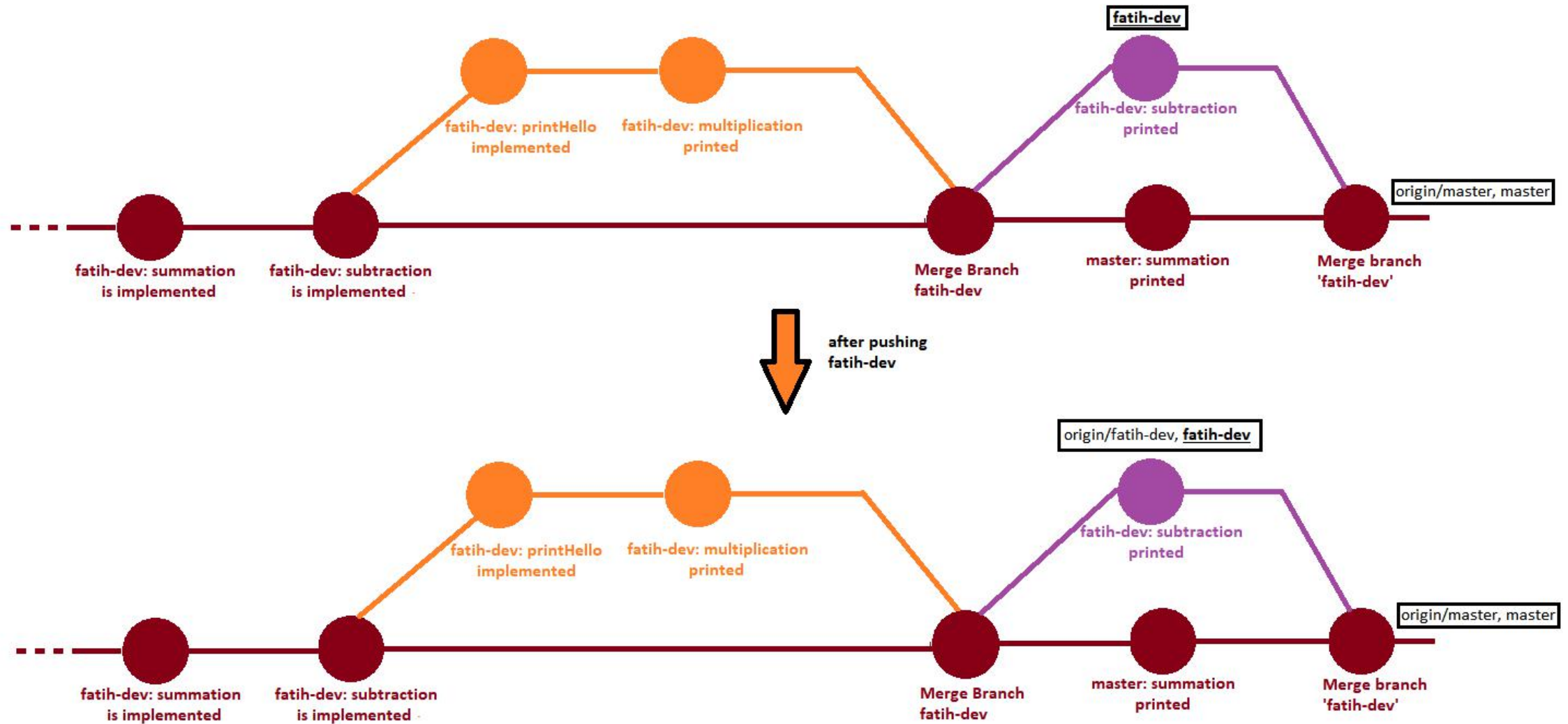
☒ Push all tags ☐ Force Push

Push Cancel

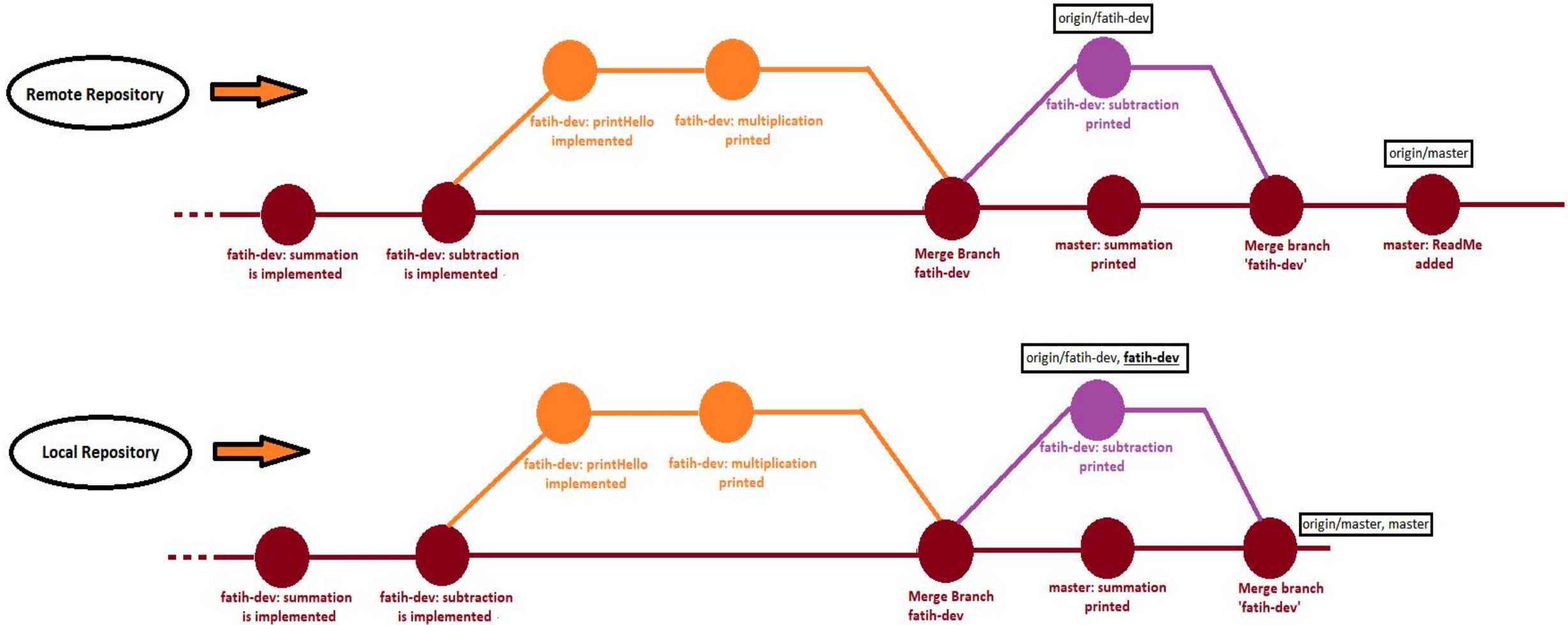
Remote Repository: push



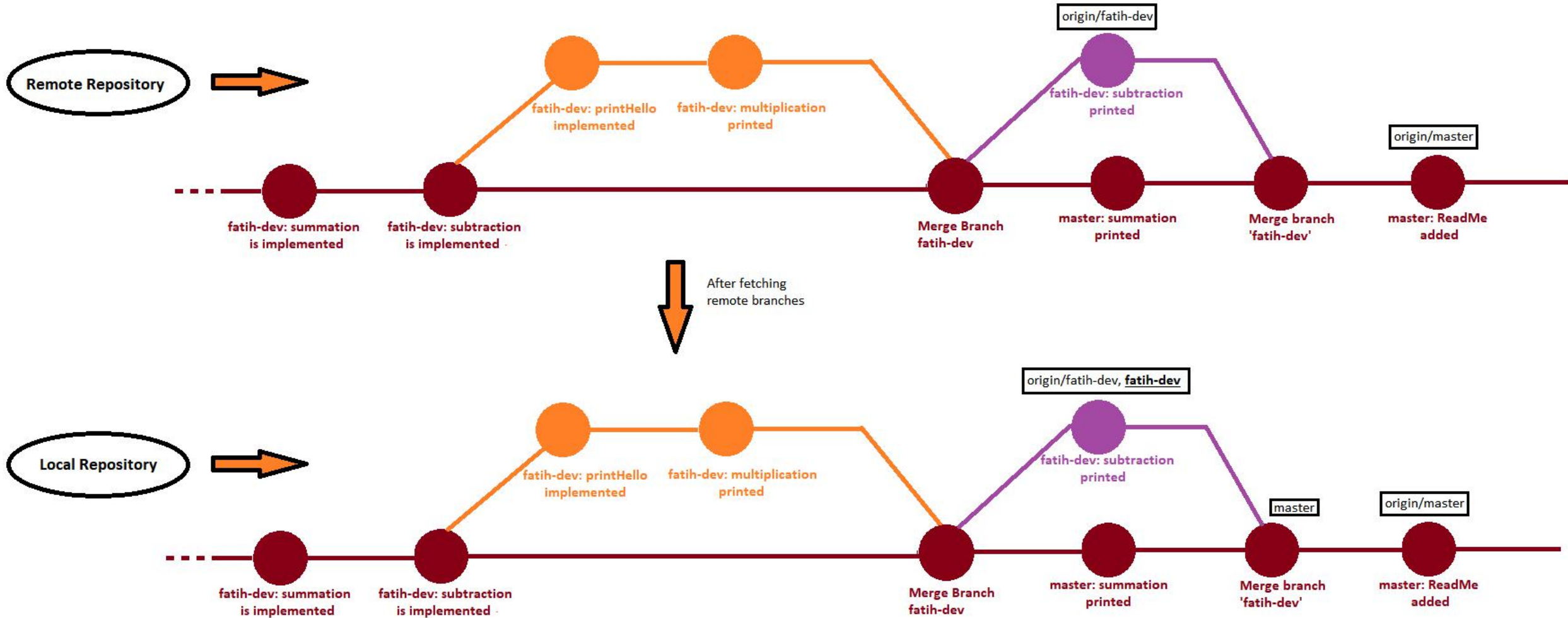
Remote Repository: push



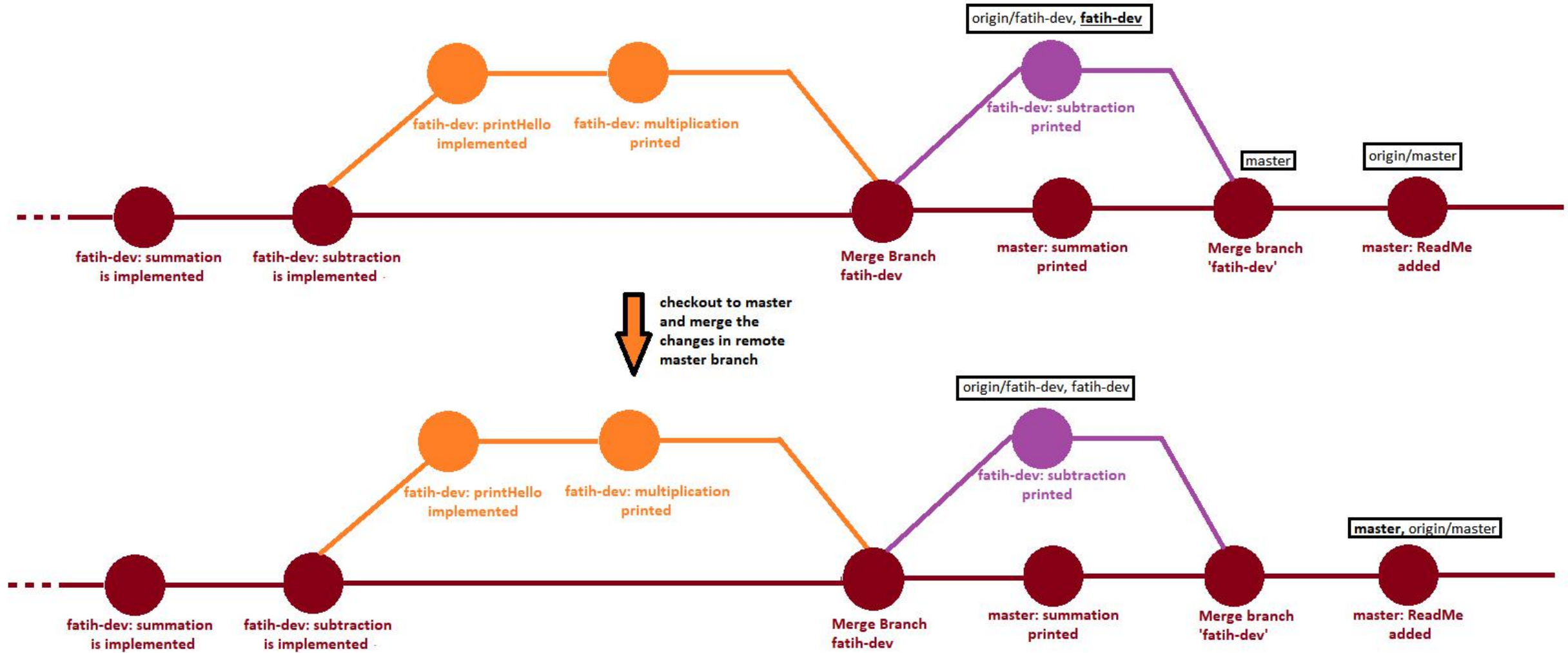
Remote Repository: fetch



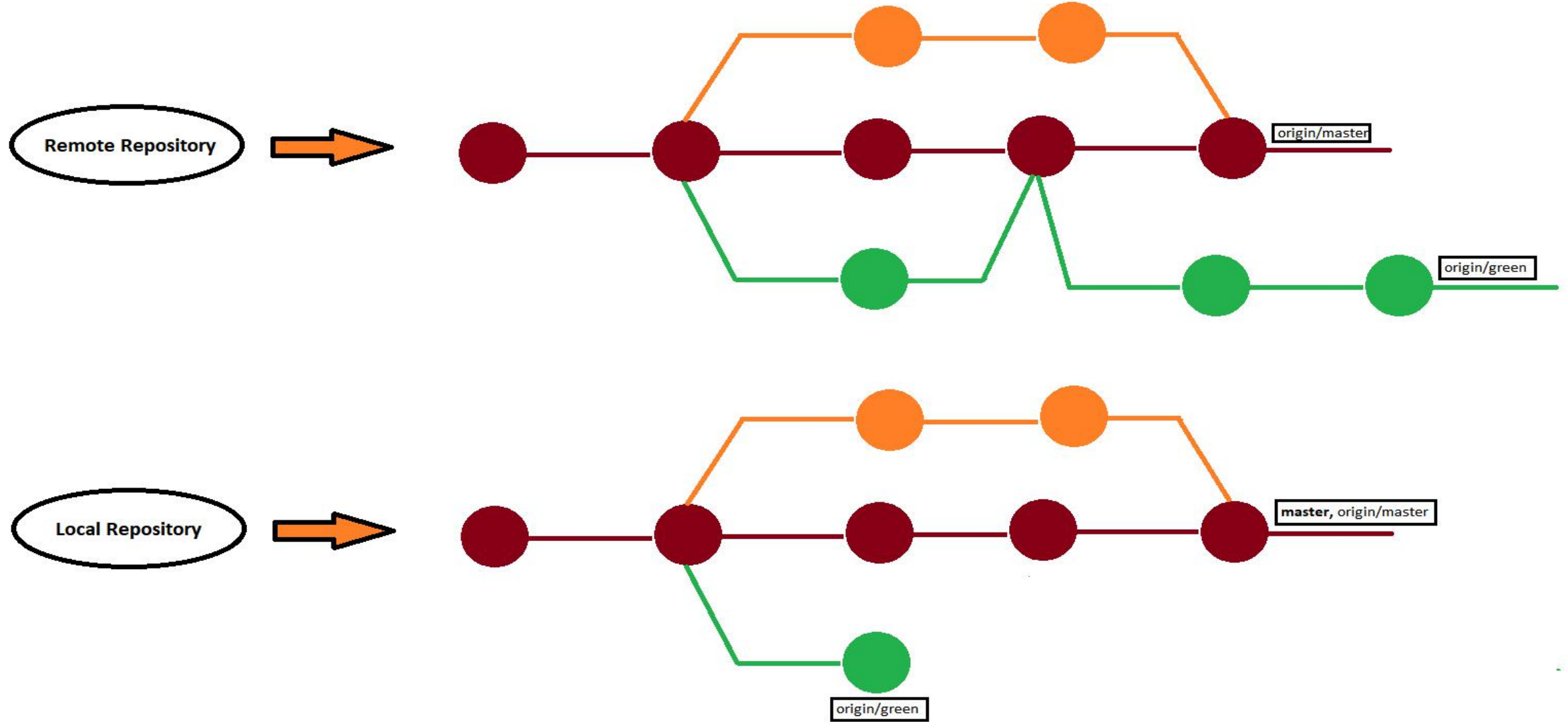
Remote Repository: fetch



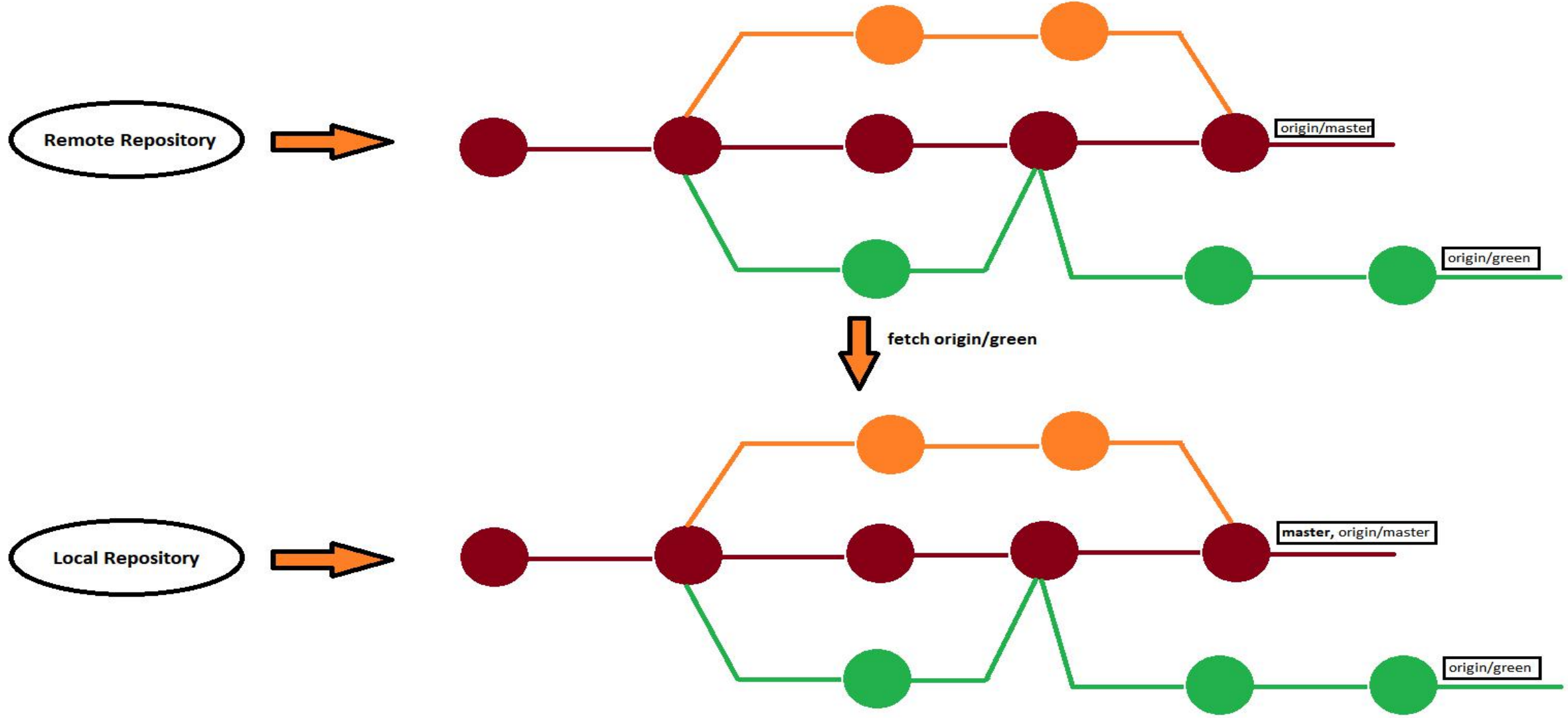
Remote Repository: fetch



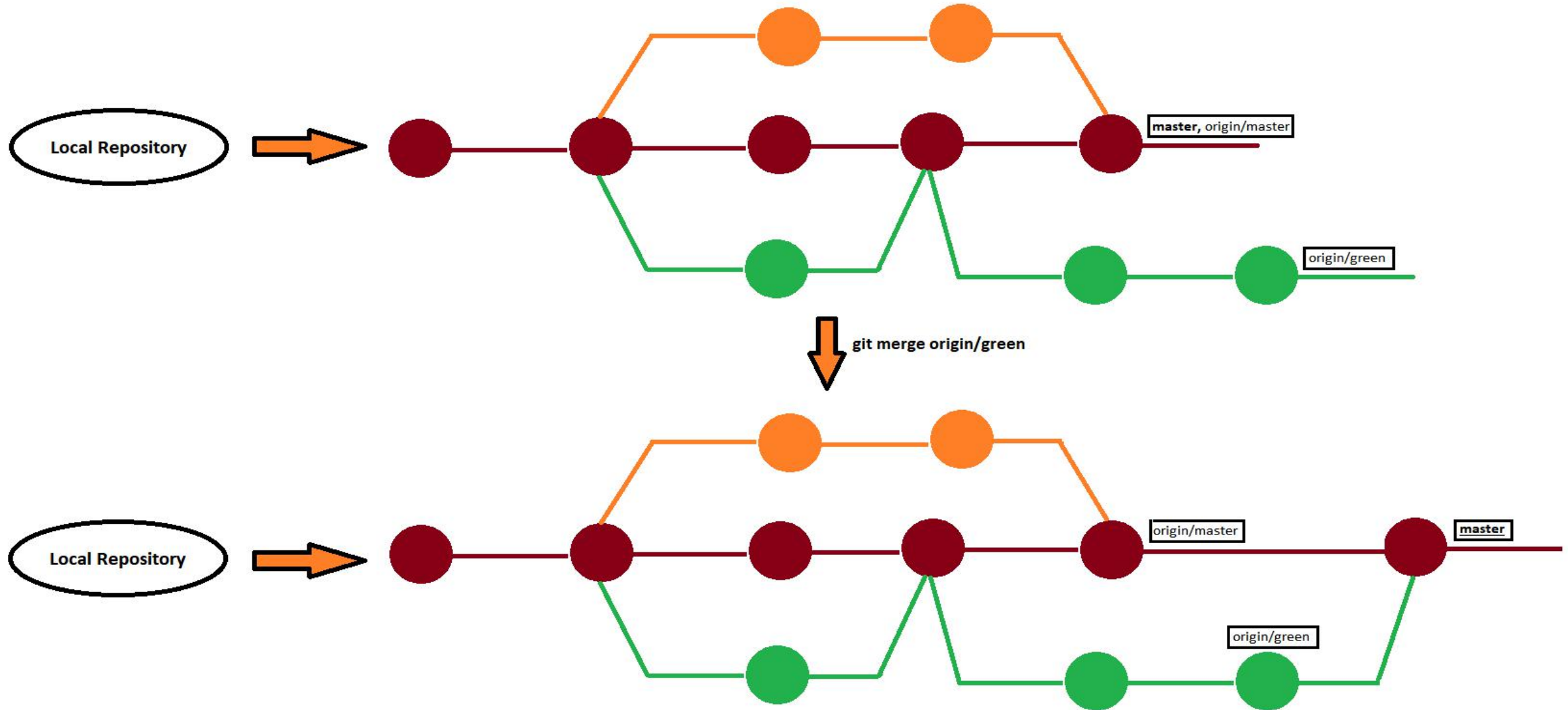
Remote Repository: pull = fetch + merge



Remote Repository: pull = fetch + merge



Remote Repository: pull = fetch + merge



Remote Repository: clone

Local

Remote

Clone


Add

Create

Clone

Cloning is even easier if you set up a [remote account](#)

Browse

Repository Type:  This is a Git repository

Browse

Local Folder:


[Root]

▼

> Advanced Options

Clone

References

- **Git Web Page:** <https://git-scm.com/>
 - **Git Documentation:** <https://git-scm.com/docs>
 - **SourceTree Web Page:** <https://www.sourcetreeapp.com/>
 - **SourceTree Documentation:** <https://confluence.atlassian.com/sourcetreekb/sourcetree-basics-780870007.html>
 - **How to install Git:** <https://www.youtube.com/watch?v=SQoPHgMDRYc>
 - **How to install SourceTree:** https://www.youtube.com/watch?v=9Oa_c8_Ewpc
 - **IntelliJ IDEA Web Page:** <https://www.jetbrains.com/idea/>
 - **Visual Studio Code:** <https://code.visualstudio.com/>
 - **Smart Code:** <https://www.youtube.com/channel/UC5XnpFJ1BEOywGZiby2mOuQ>
- 

Youtube Channel: Smart Code

