

# Git 的使用姿势

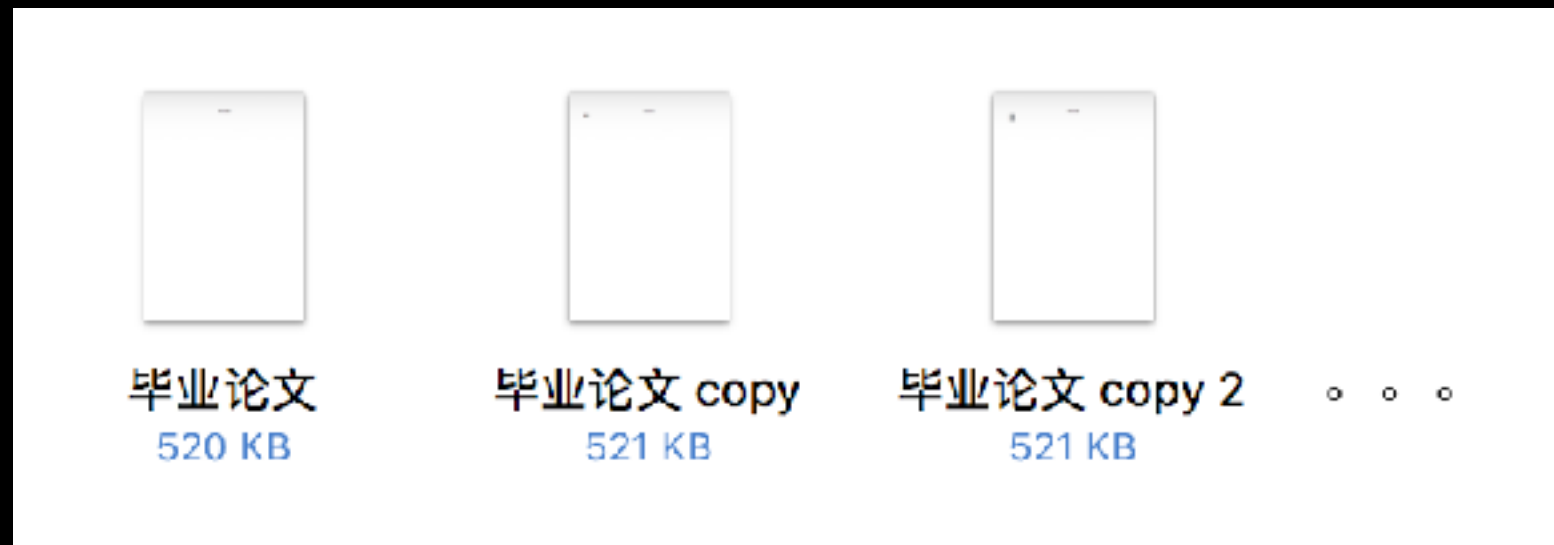
# 目录

- Git 是什么
- Git 基础
- Git 分支
- 远程仓库
- MR 工作流
- 小技巧

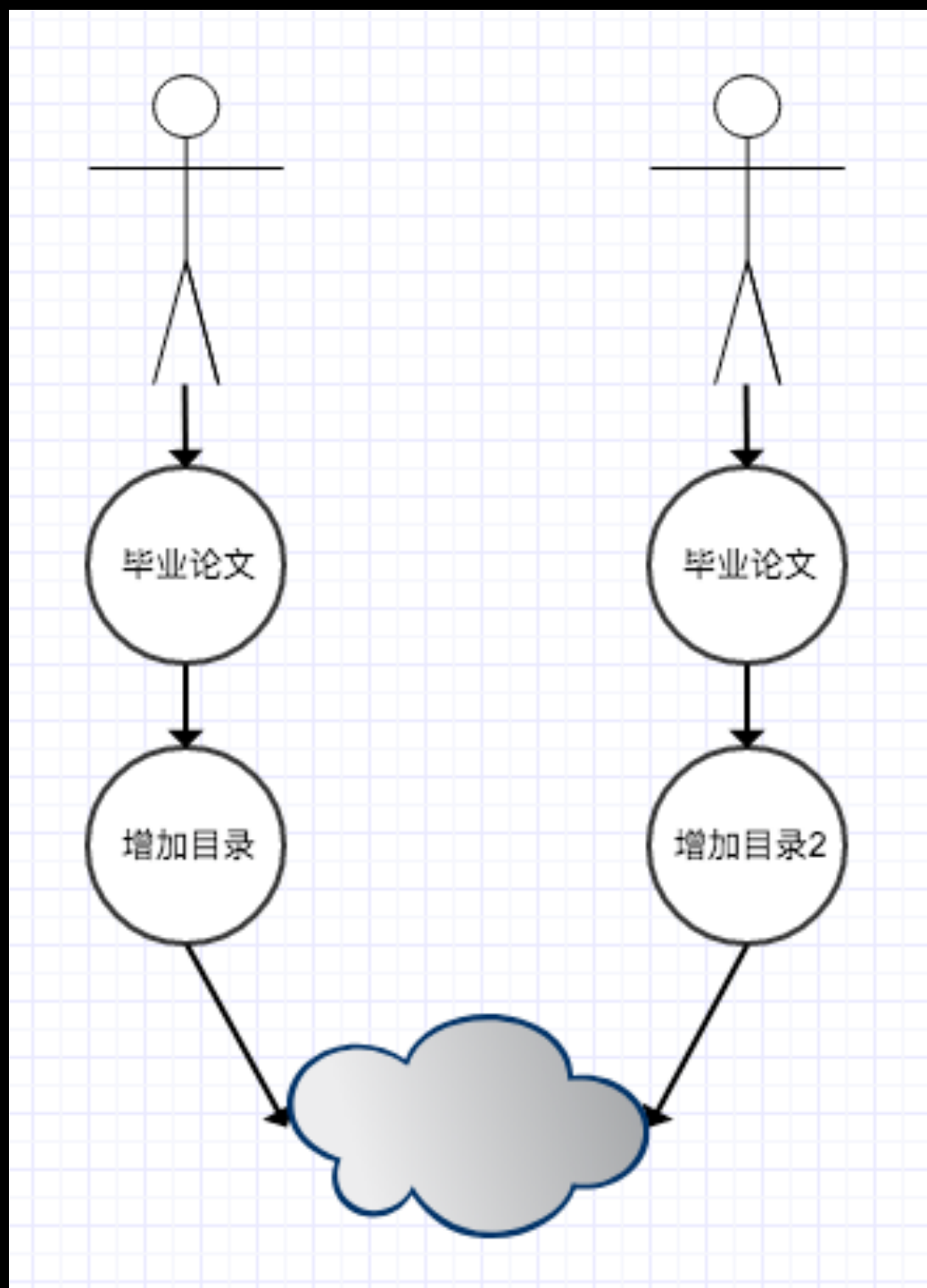
# Git 是什么

- 版本控制系统
- 代码托管系统

# 版本控制



# 代码托管



# github.com

The screenshot shows the GitHub repository page for `nodejs/node`. The browser address bar displays `https://github.com/nodejs/node`. The repository name `nodejs / node` is shown at the top, along with statistics: 2,339 watchers, 38,001 stars, and 7,533 forks. Below this, navigation tabs include `Code`, `Issues` (546), `Pull requests` (291), `Projects` (5), `Wiki`, and `Insights`. The repository description is "Node.js JavaScript runtime" with a link to `https://nodejs.org`. A list of tags includes `nodejs`, `javascript`, `node`, `js`, `runtime`, `mit`, `linux`, `macos`, and `windows`. A progress bar indicates 18,473 commits, 125 branches, 440 releases, and 1,475 contributors. Below the progress bar, there are buttons for `Branch: master`, `New pull request`, `Create new file`, `Upload files`, `Find file`, and `Clone or download`. The commit history table shows the latest commit by `okyantero` with the message "cover all HTTP methods that parser supports" 3 days ago. The table lists recent changes across various files and folders.

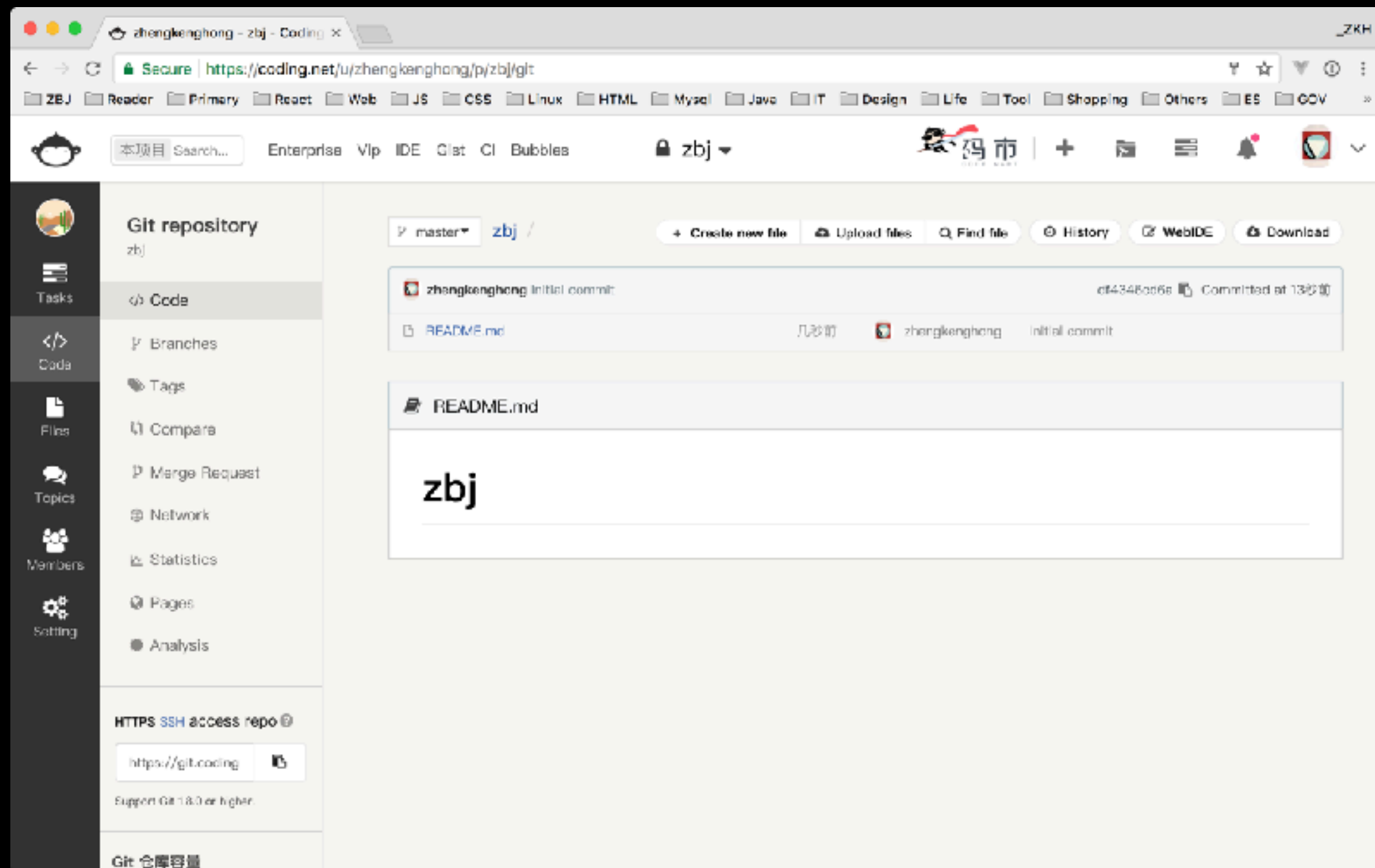
File/Folder	Commit Message	Time Ago
<code>.github</code>	doc: fix commit guideline url	3 months ago
<code>benchmark</code>	benchmark: fix and extend assert benchmarks	12 hours ago
<code>deps</code>	deps: cherry-pick f19b889 from V8 upstream	3 days ago
<code>doc</code>	doc: add missing 'changes:' metadata for util	11 hours ago
<code>lib</code>	lib: move deprecationWarning var	an hour ago
<code>src</code>	src: mention that node options are space-separated	13 hours ago
<code>test</code>	test: cover all HTTP methods that parser supports	12 minutes ago
<code>tools</code>	tools: fix tools/addon-verify.js	4 days ago

# git.zhubajie.la

The screenshot shows a web browser window displaying a GitHub repository page. The address bar shows the URL <https://git.zhubajie.la/ads-nodejs/zzgg-zhuma-manager-web/tree/master>. The repository name is 'ads-nodejs / zzgg-zhuma-manager-web'. The page shows the 'Repository' tab selected, with a list of files and folders. The files listed include 'app', '.babelrc', '.exclude\_list', '.gitignore', '.gitlab-ci.yml', '.package\_nodejs.sh', '.project', '.project.properties', 'Dockerfile', and 'README.md'. The 'Last Update' and 'Last Commit' columns are visible, along with a 'History' link.

Name	Last Update	Last Commit
app	about 5 hours ago	merge master
.babelrc	4 months ago	add ueditor
.exclude_list	3 months ago	solve conflict
.gitignore	4 months ago	add ueditor
.gitlab-ci.yml	8 months ago	add pinpoint_sampling
.package_nodejs.sh	4 months ago	行业和类目的校验
.project	2 months ago	页面提交
.project.properties	about a month ago	version
Dockerfile	about a month ago	Add todo
README.md	about a month ago	Add todo

# coding.net





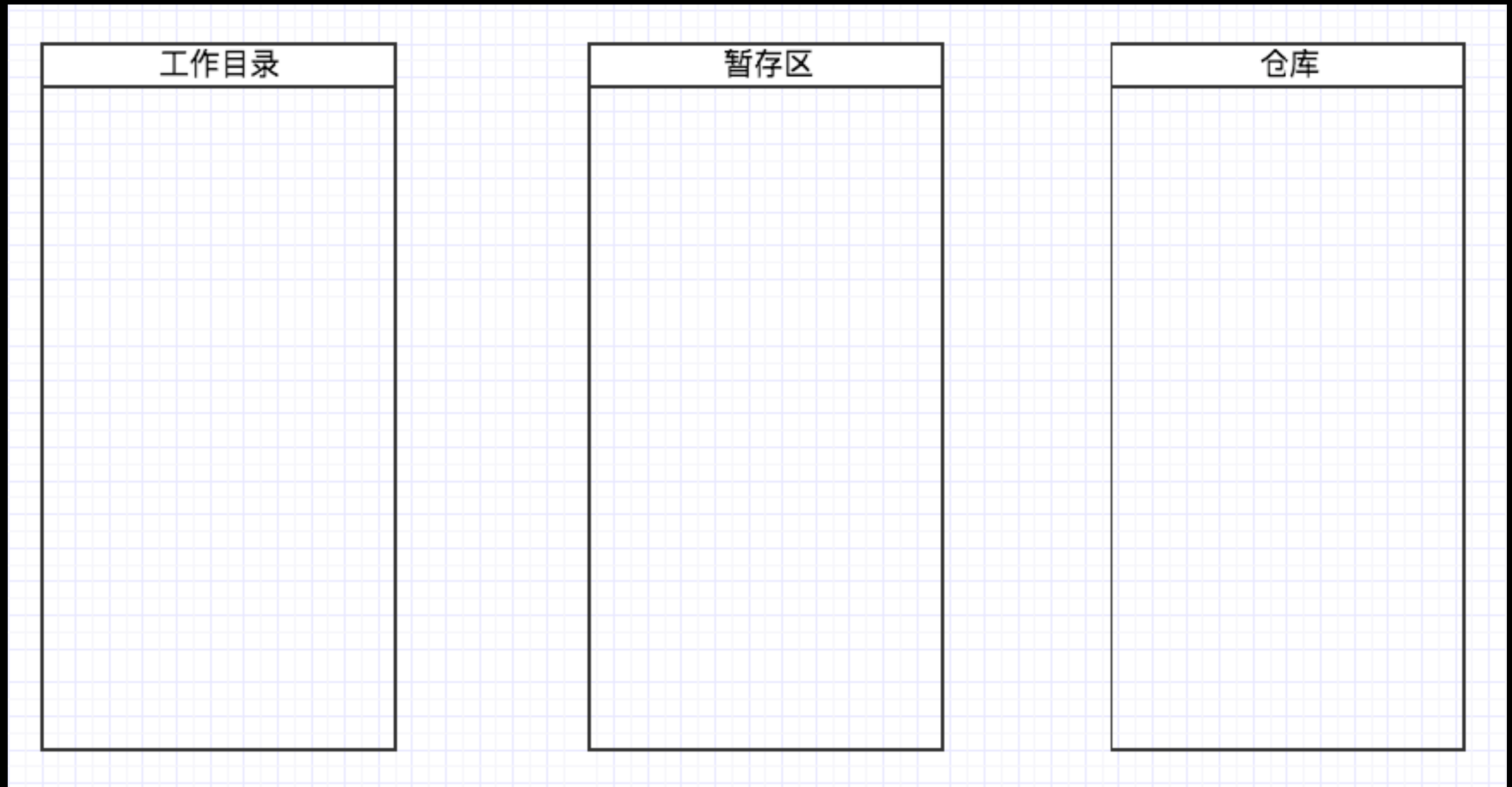
# 回顾

- 版本控制系统
- 代码托管系统

# 创建 Git 仓库

- `mkdir learn_git`
- `cd learn_git`
- `git init`

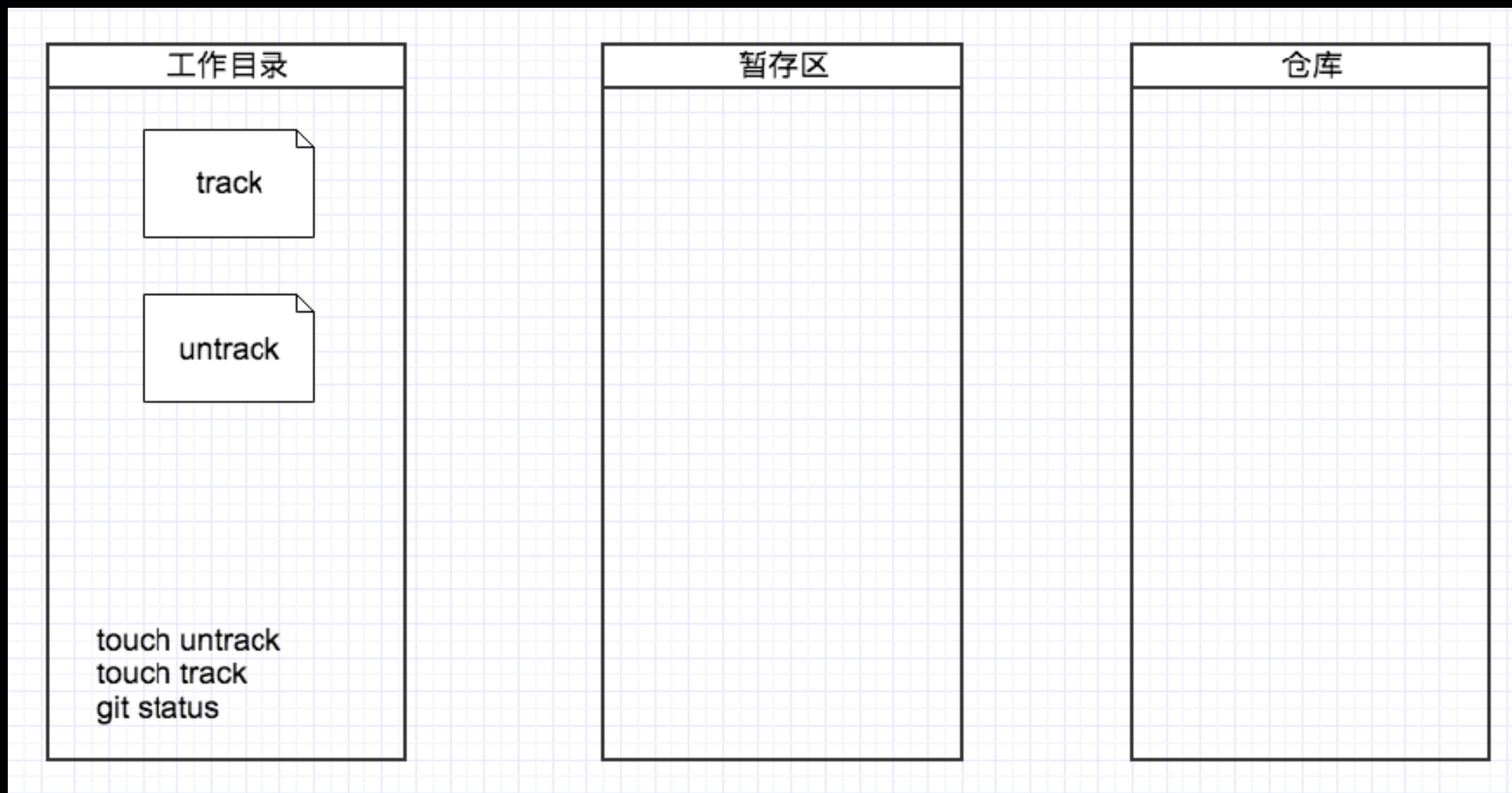
# 创建 Git 仓库



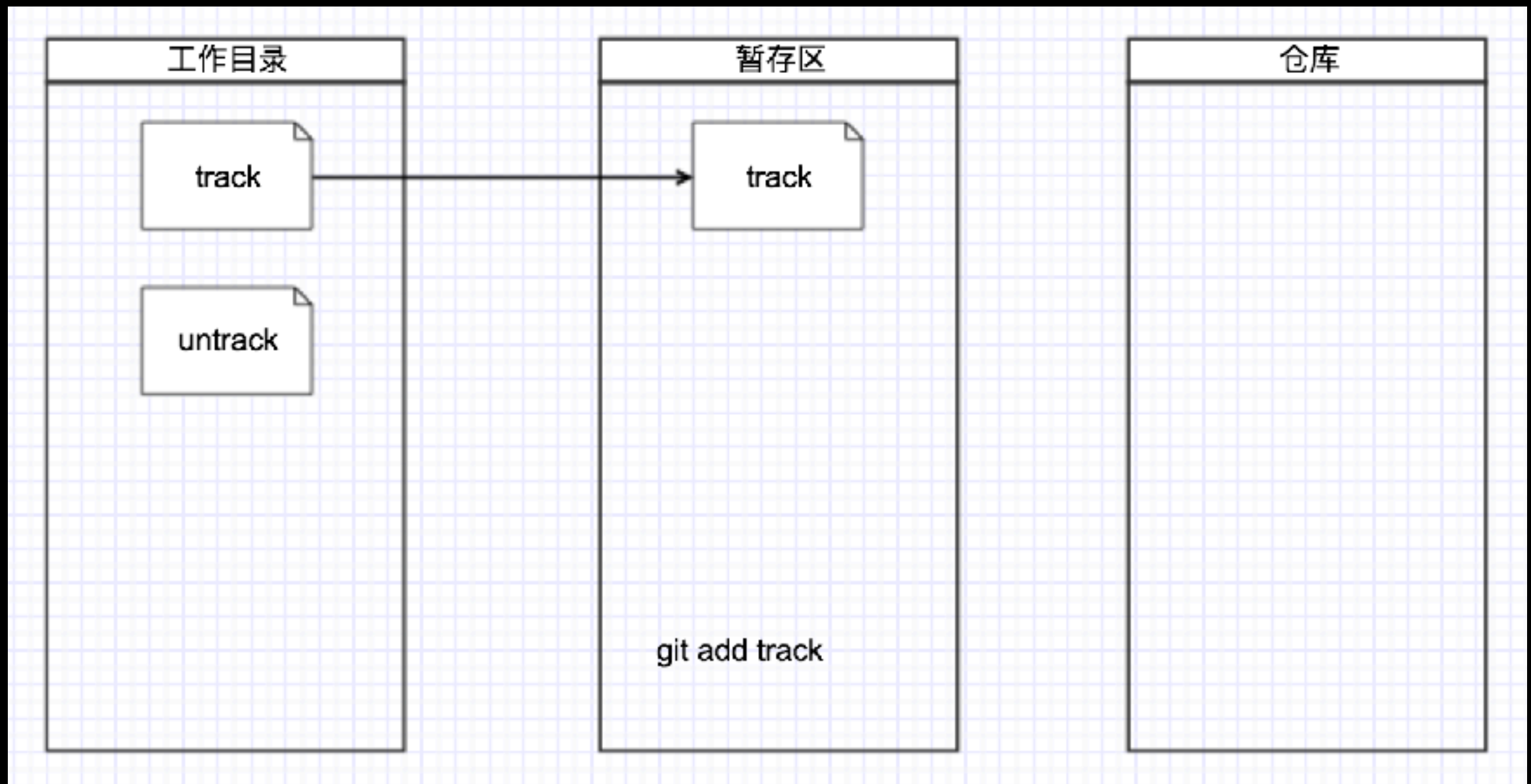
# 文件状态

- 未跟踪
- 已跟踪
  - 已暂存
  - 已提交
  - 已修改

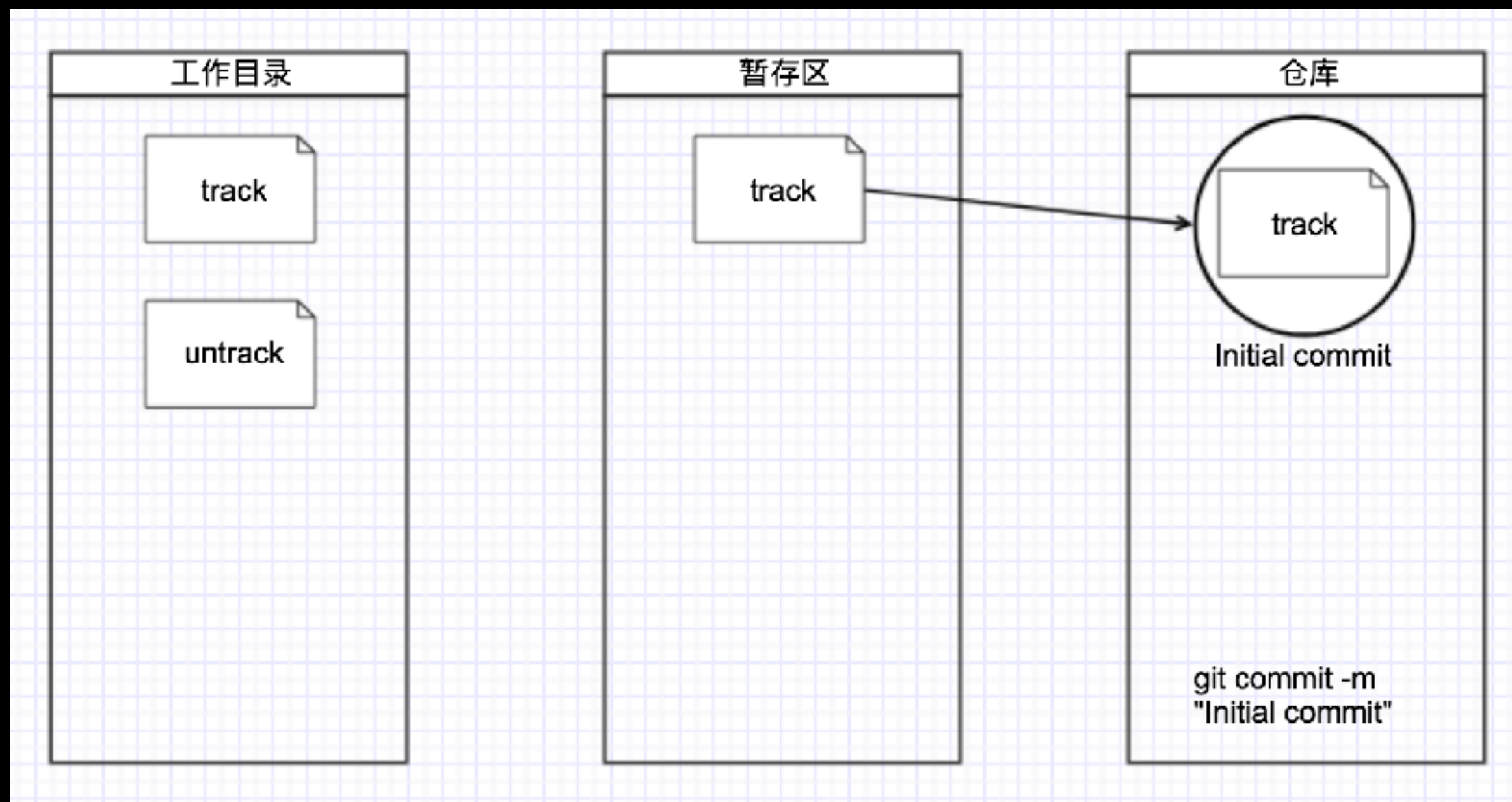
# 未跟踪



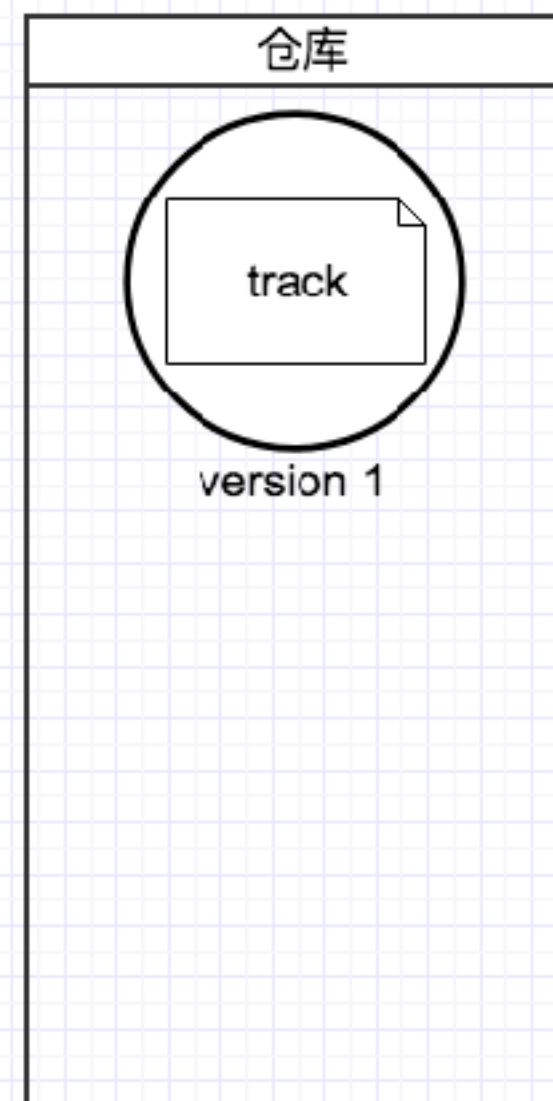
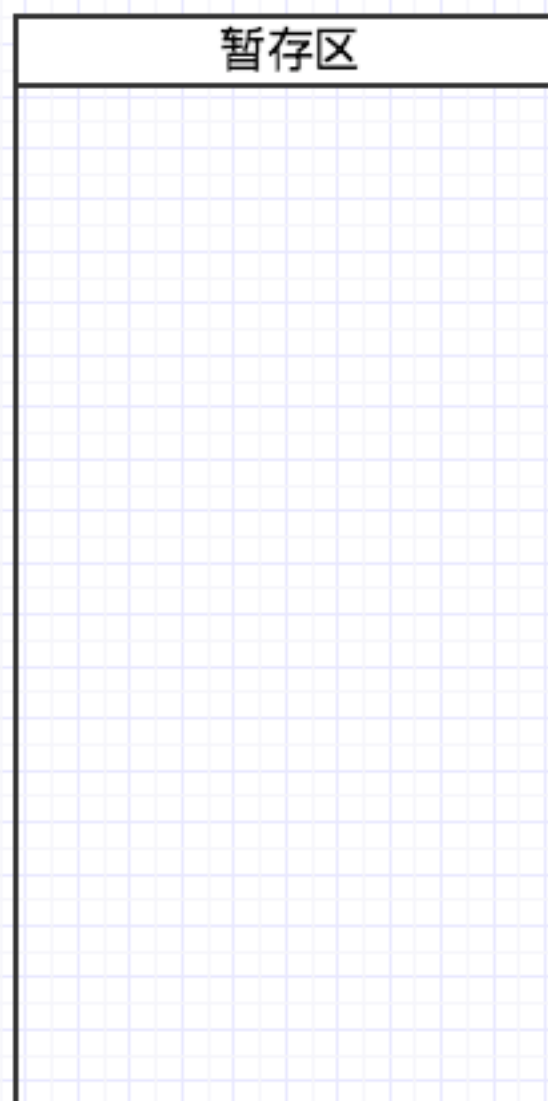
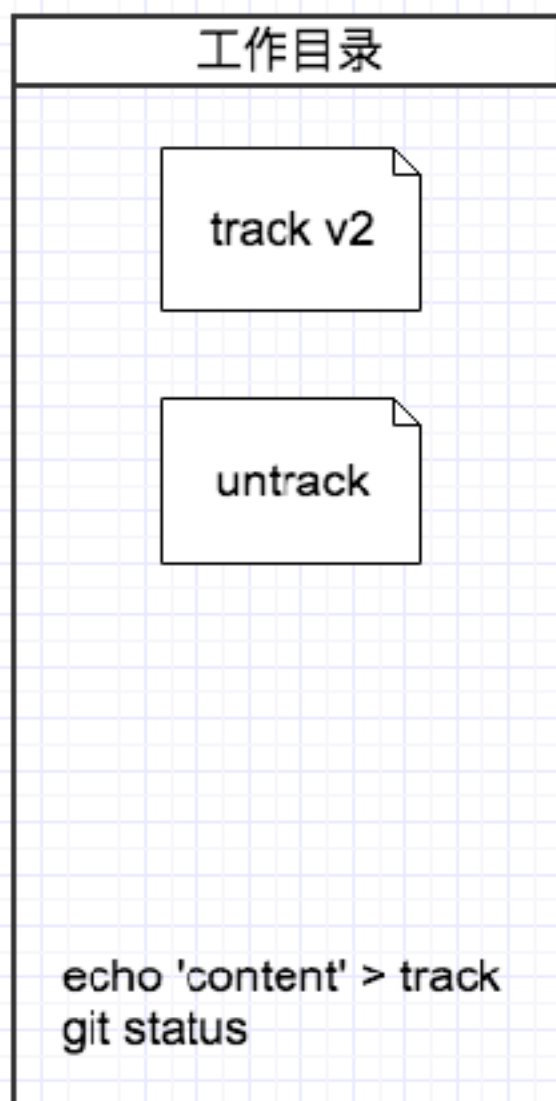
# 已暂存



# 已提交

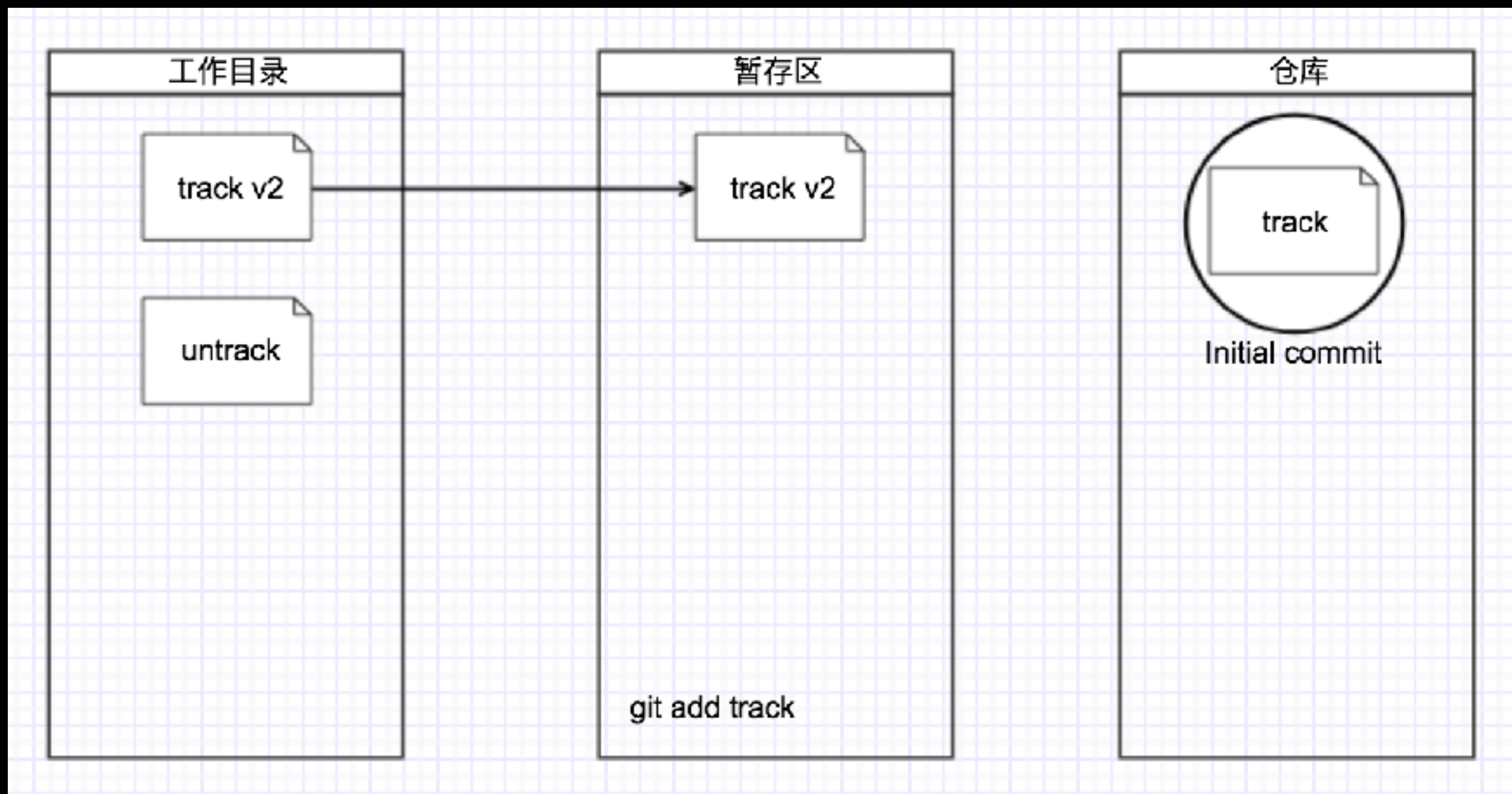


# 已修改

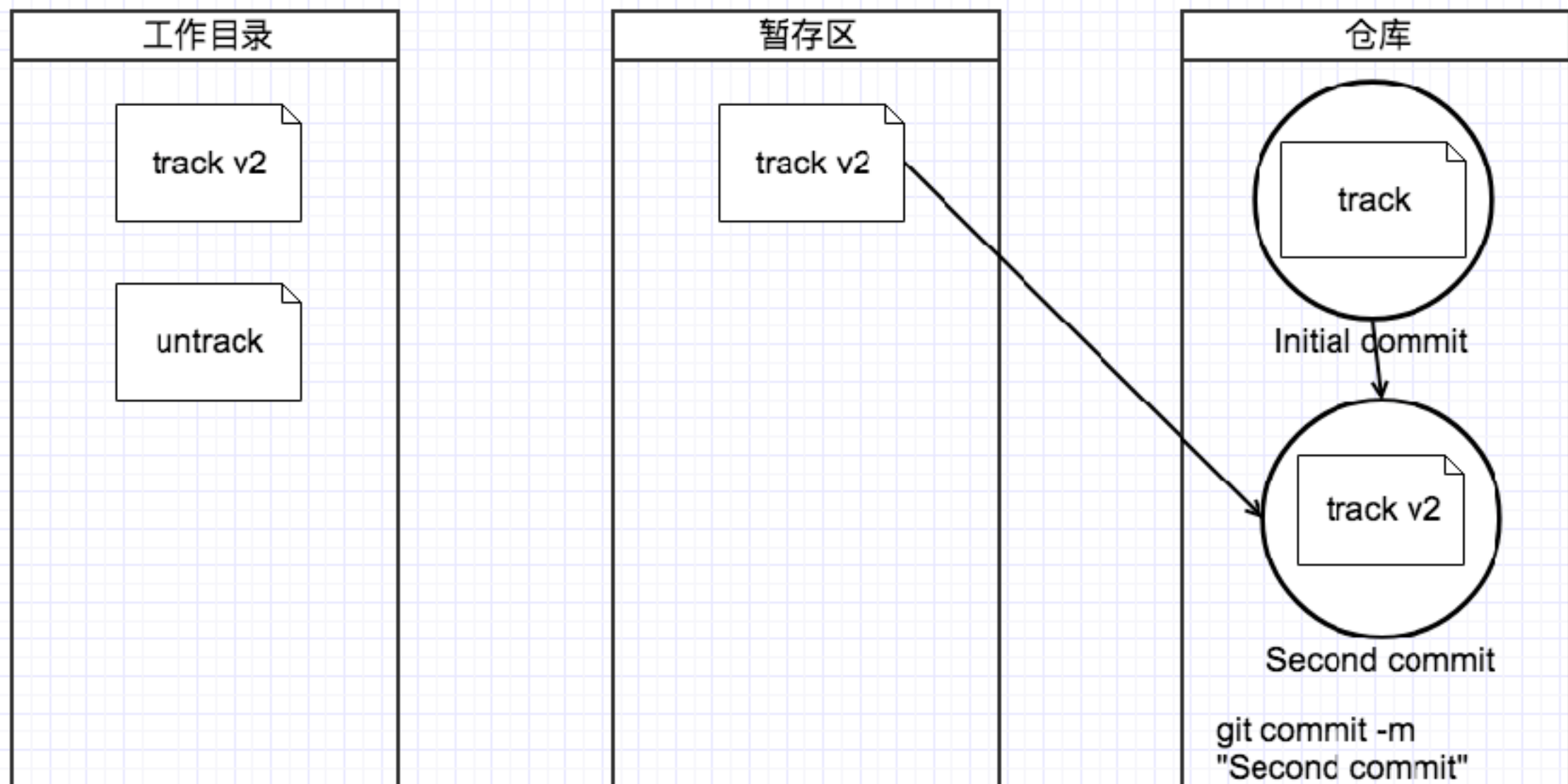




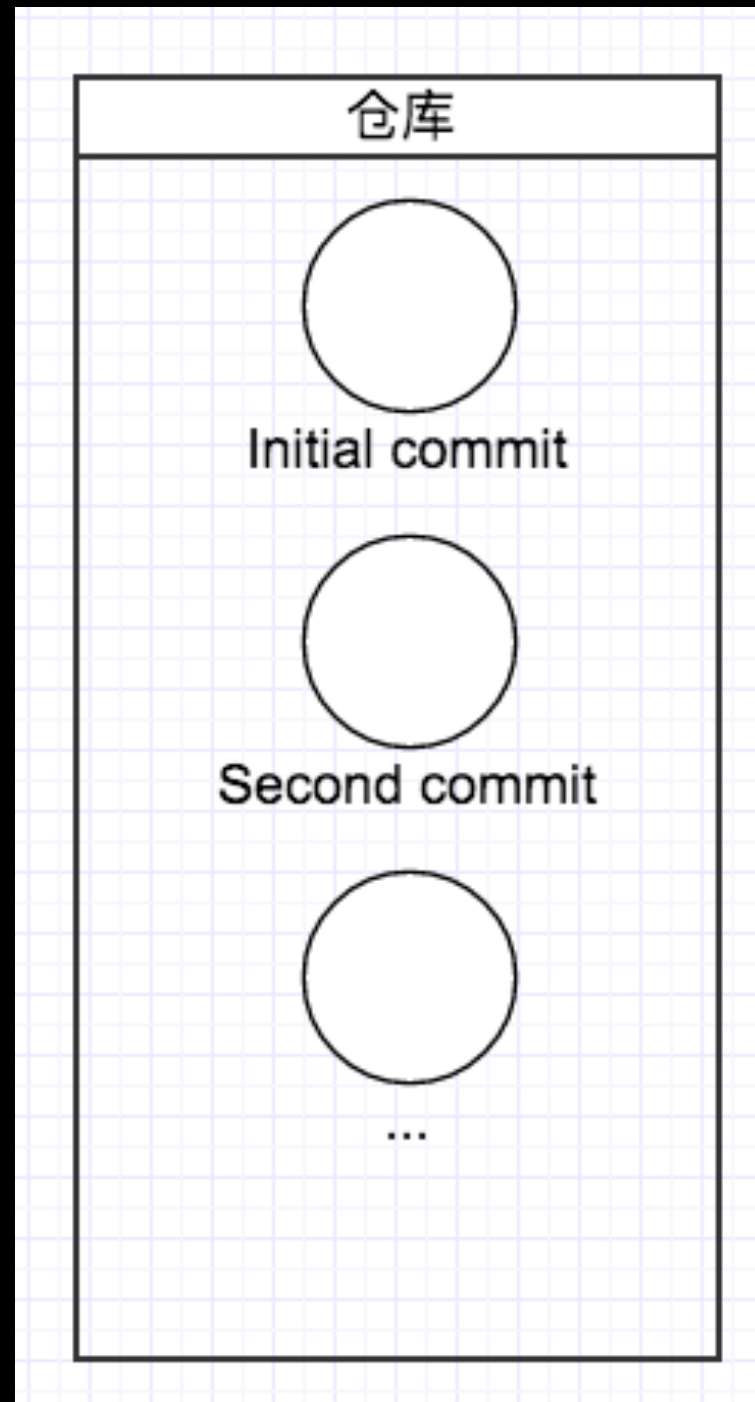
# 已暂存



# 已提交



# git log

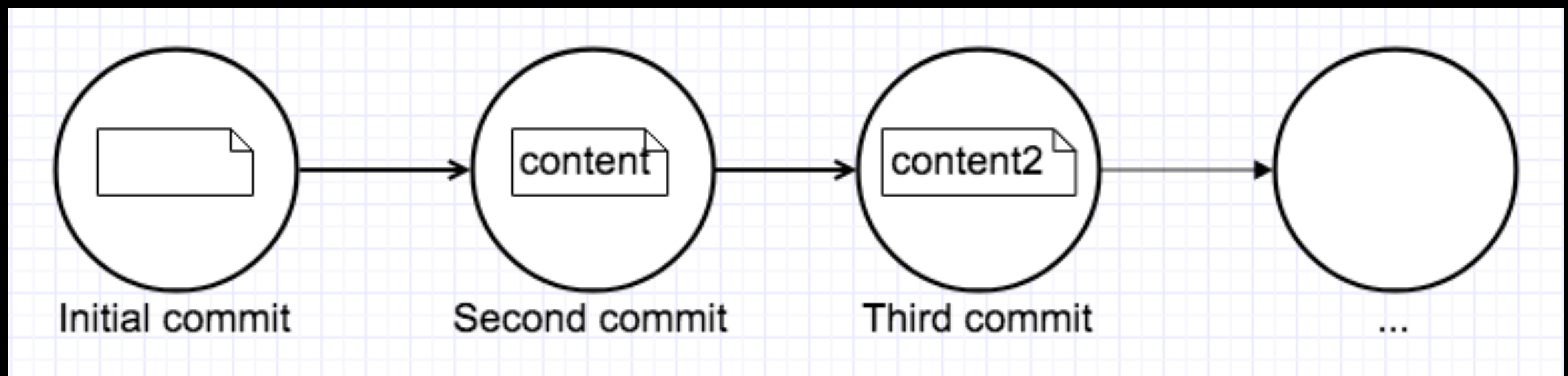


# 回顾

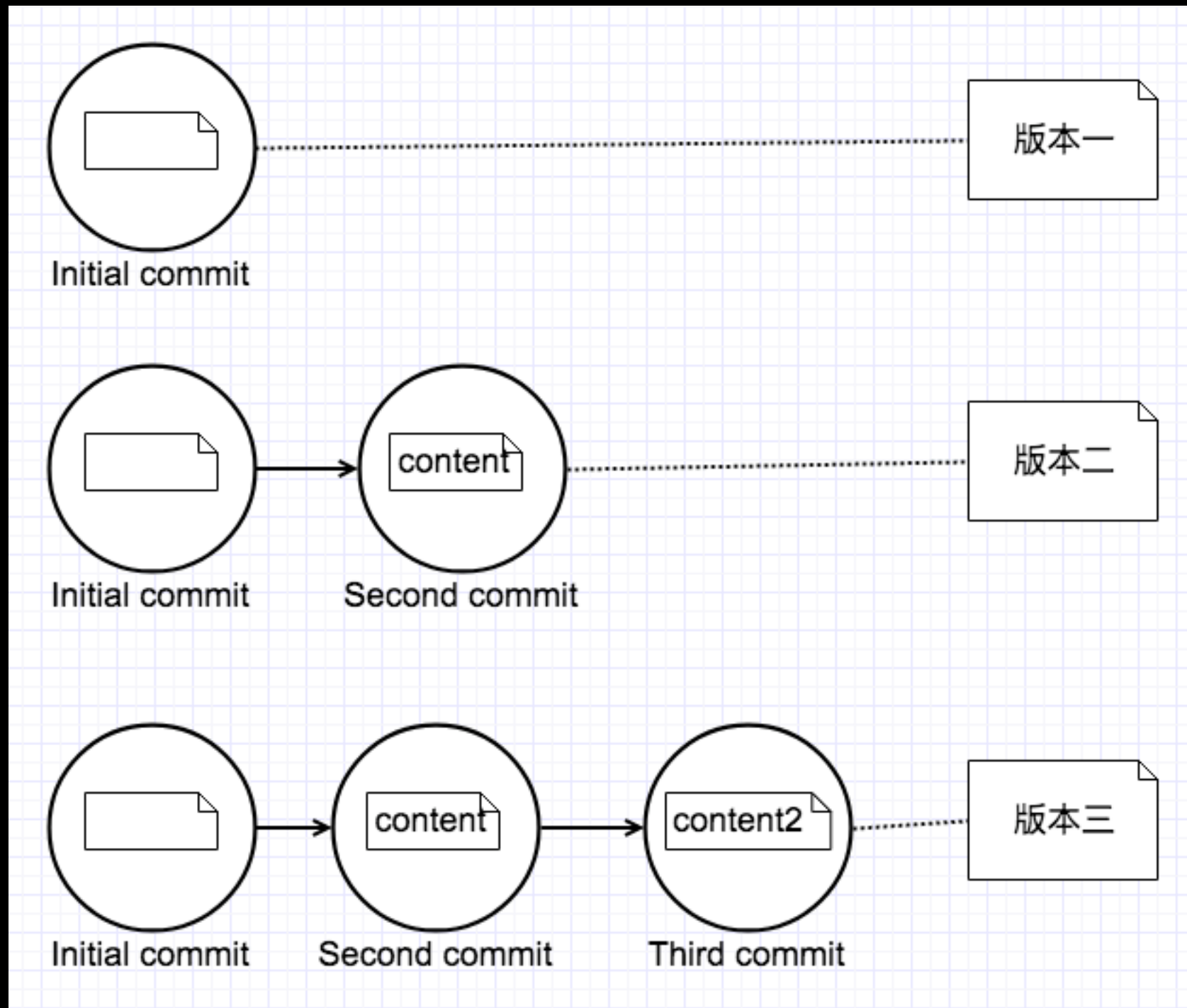
- git init
- git status
- git add
- git commit
- git log

# commit

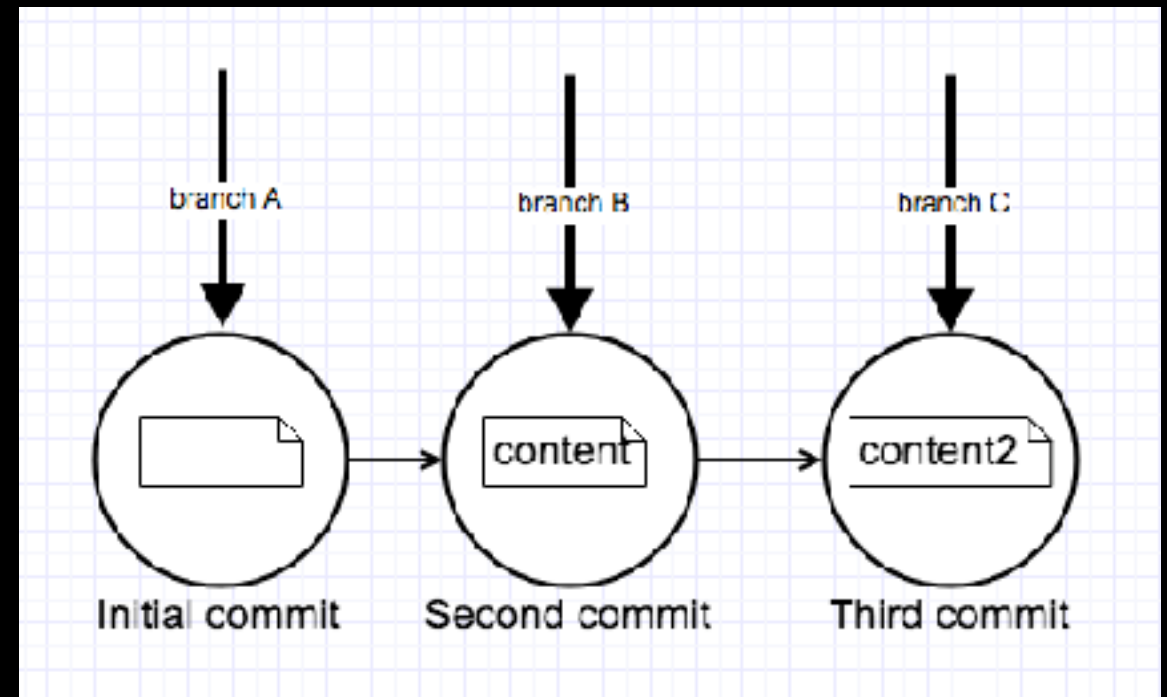
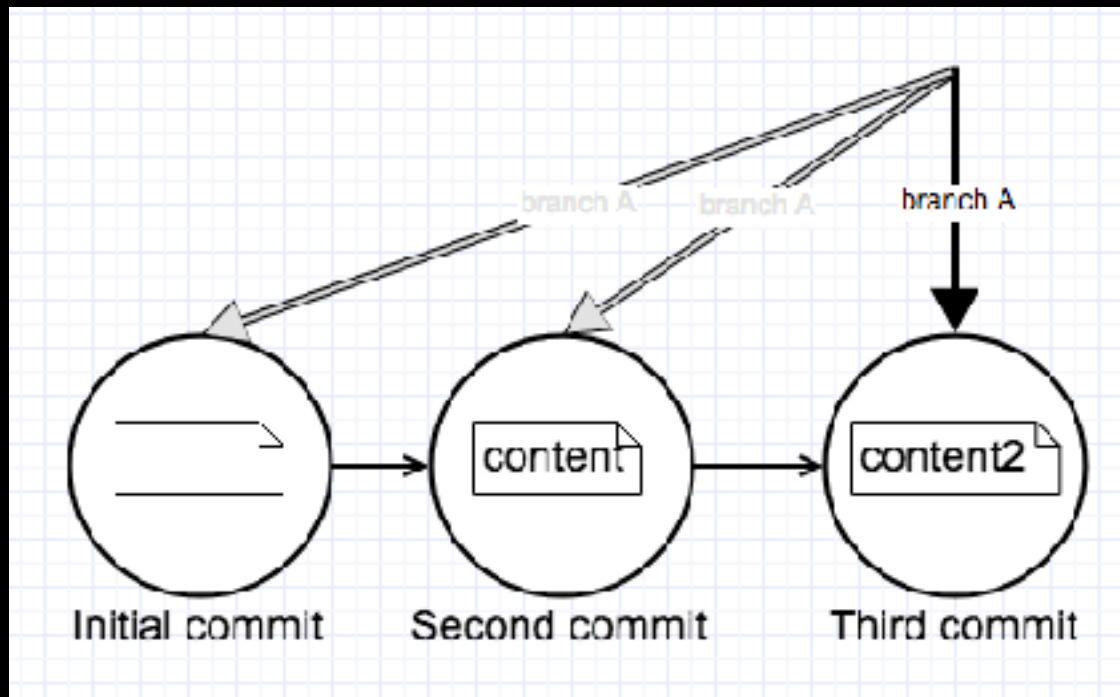
- 修改的集合



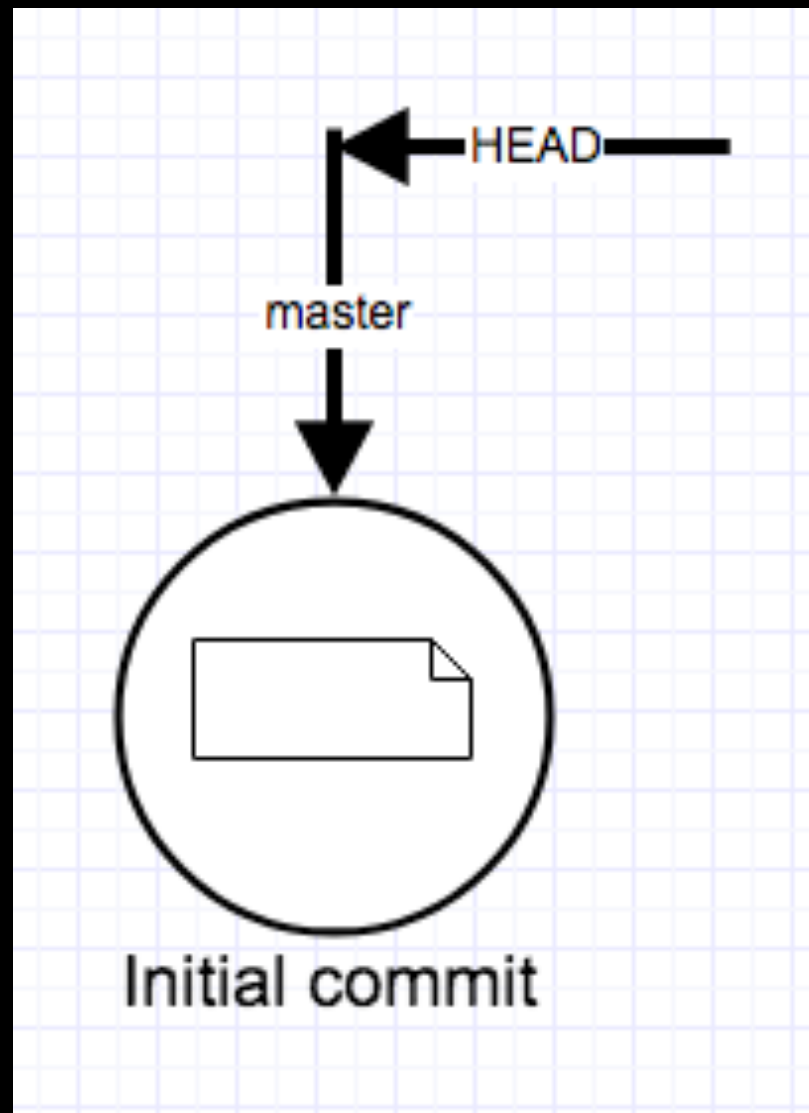
# commit



# branch

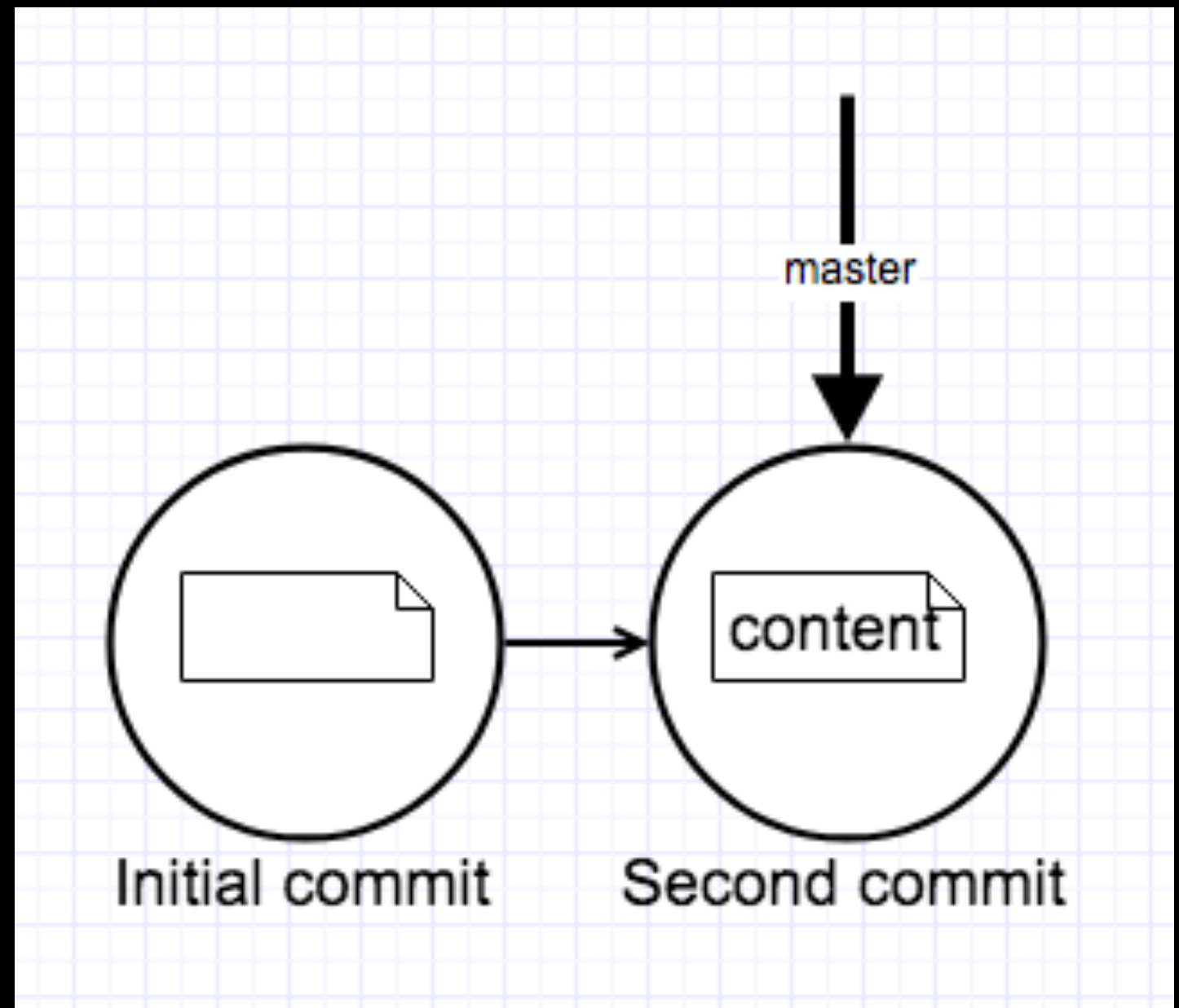
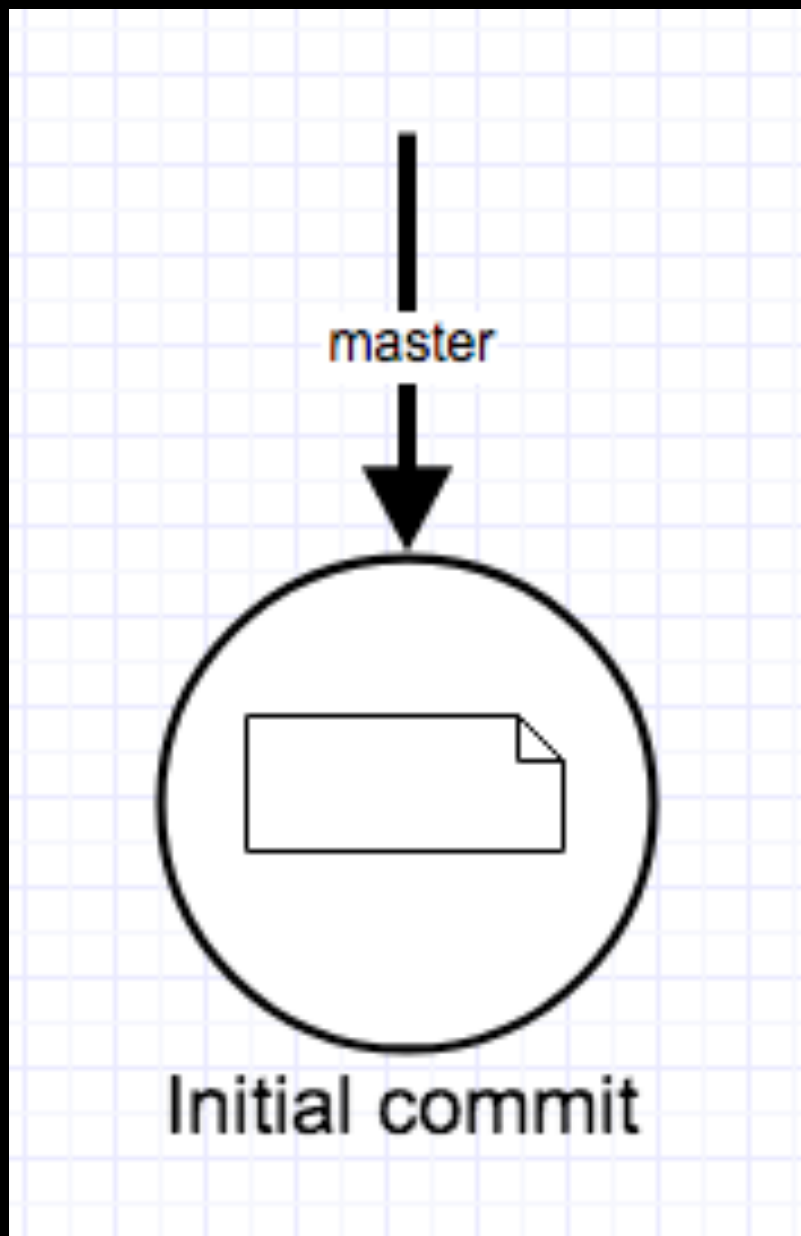


# branch

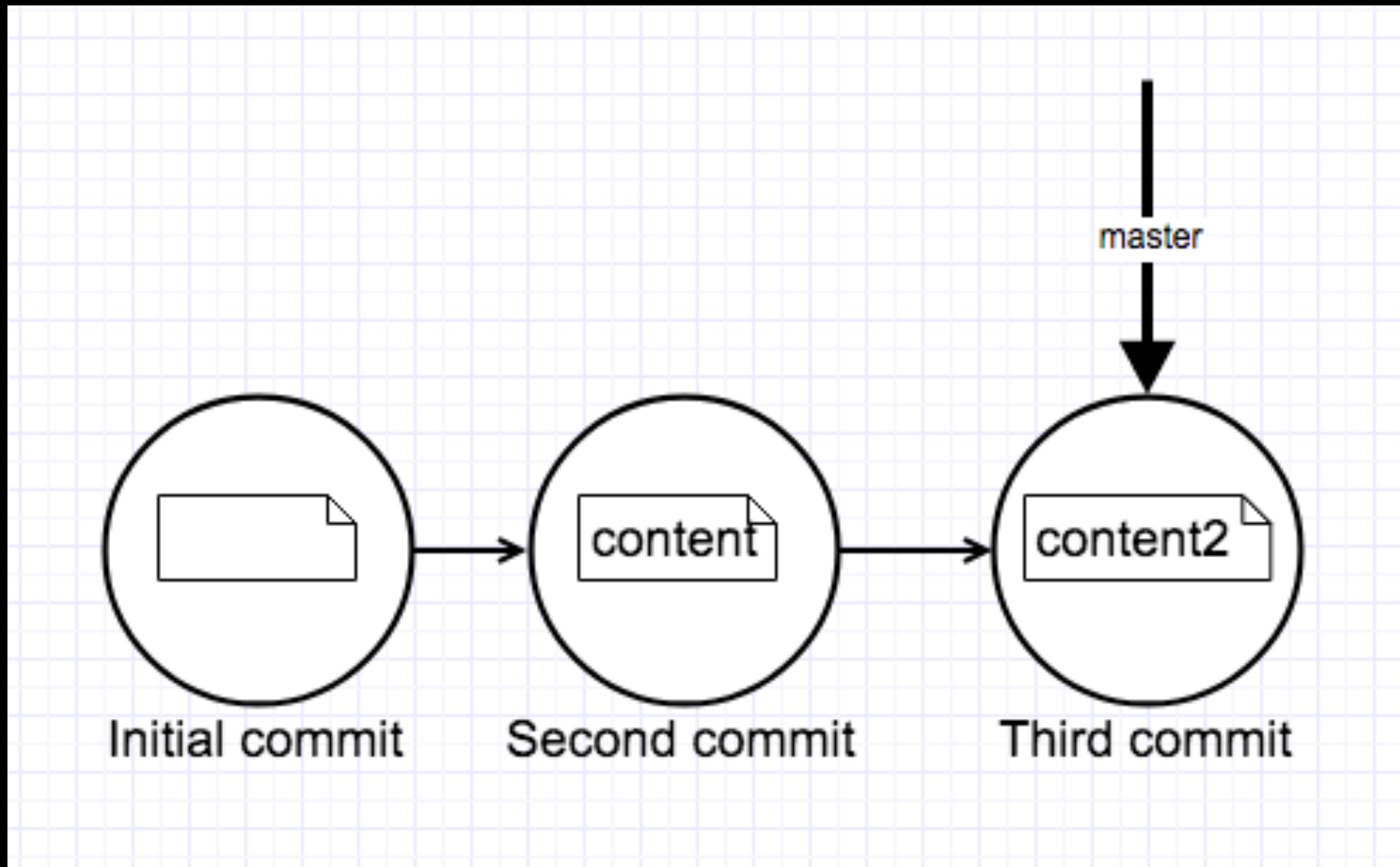




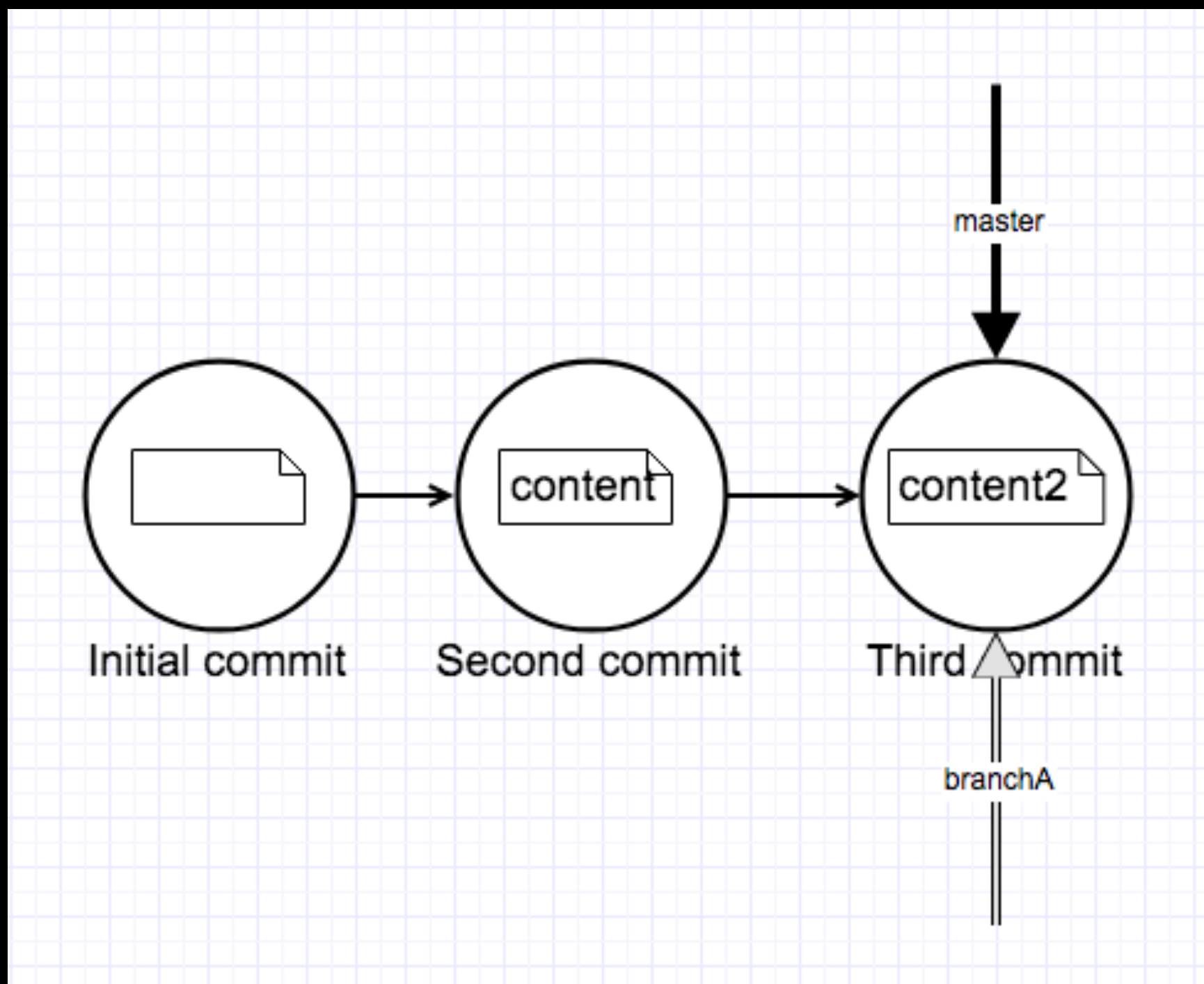
# branch



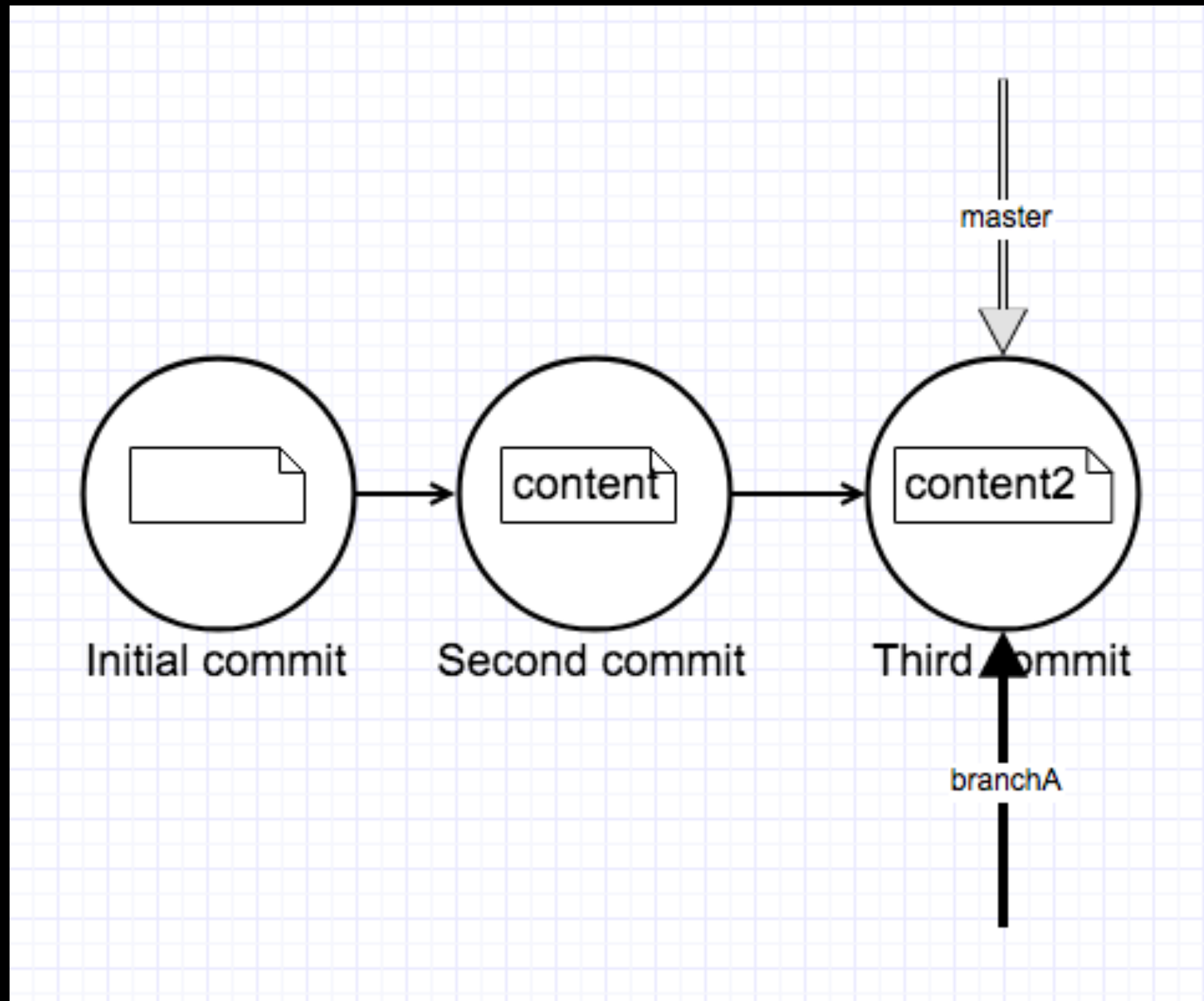
# branch



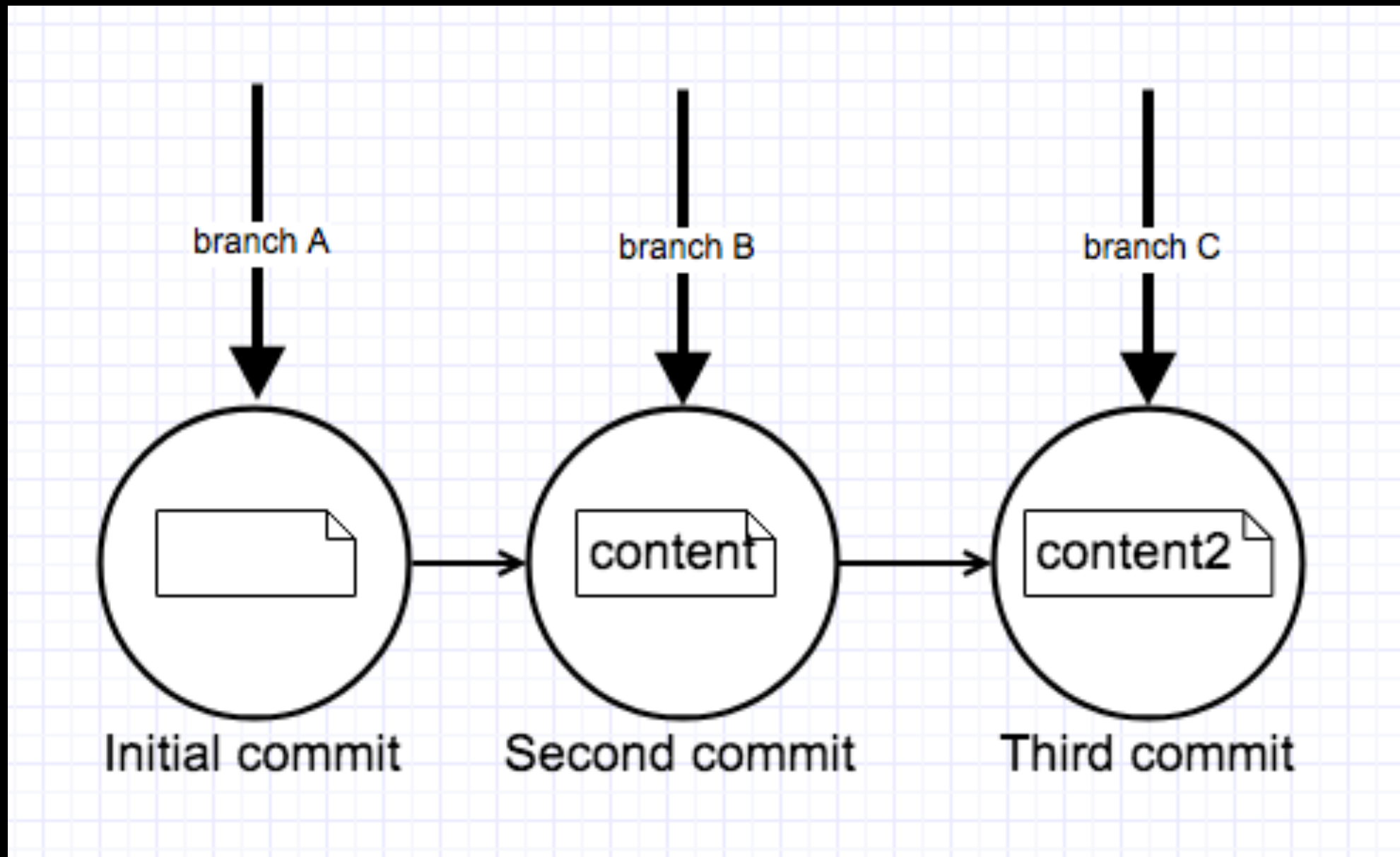
# 创建 branch



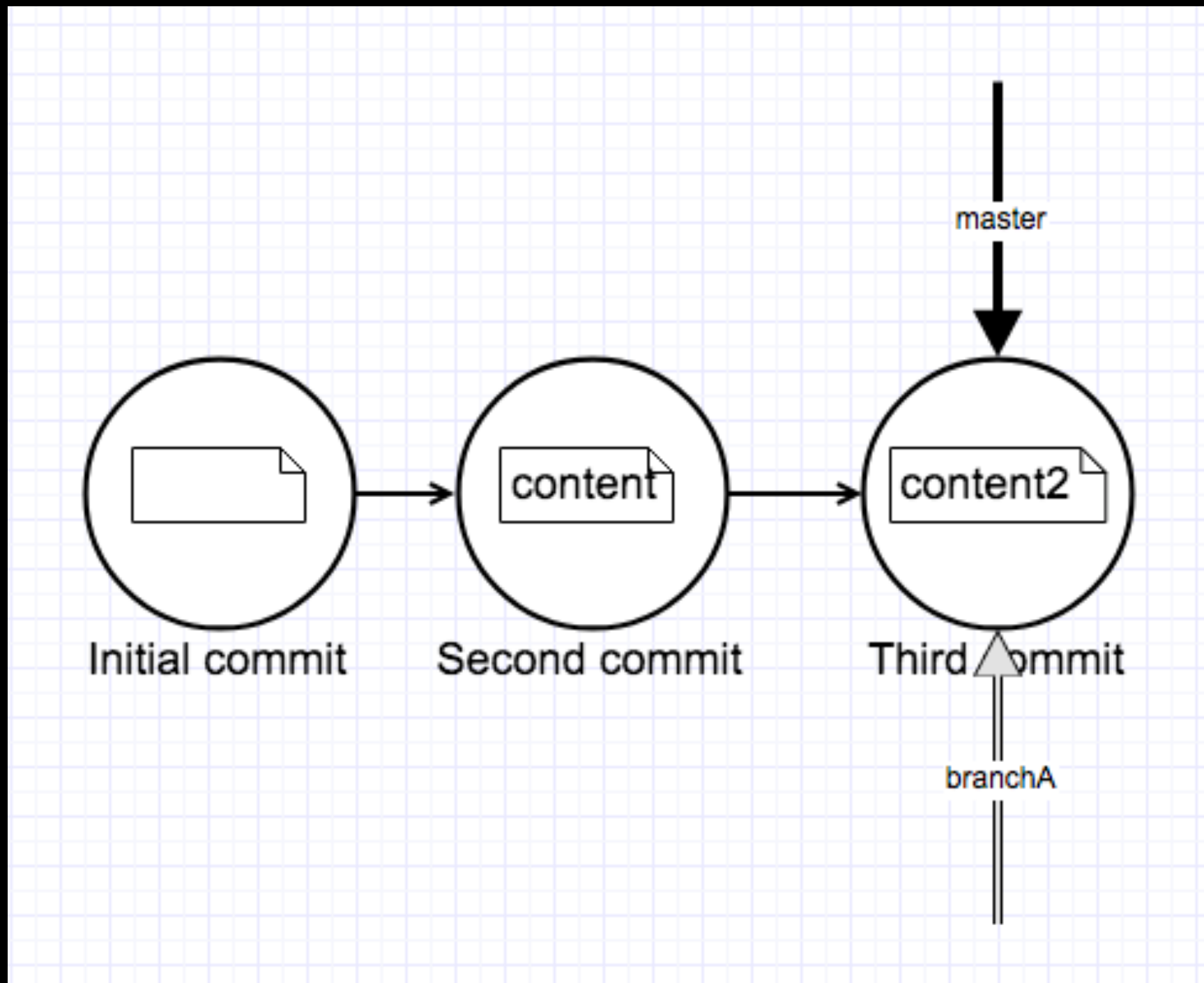
# checkout 分支



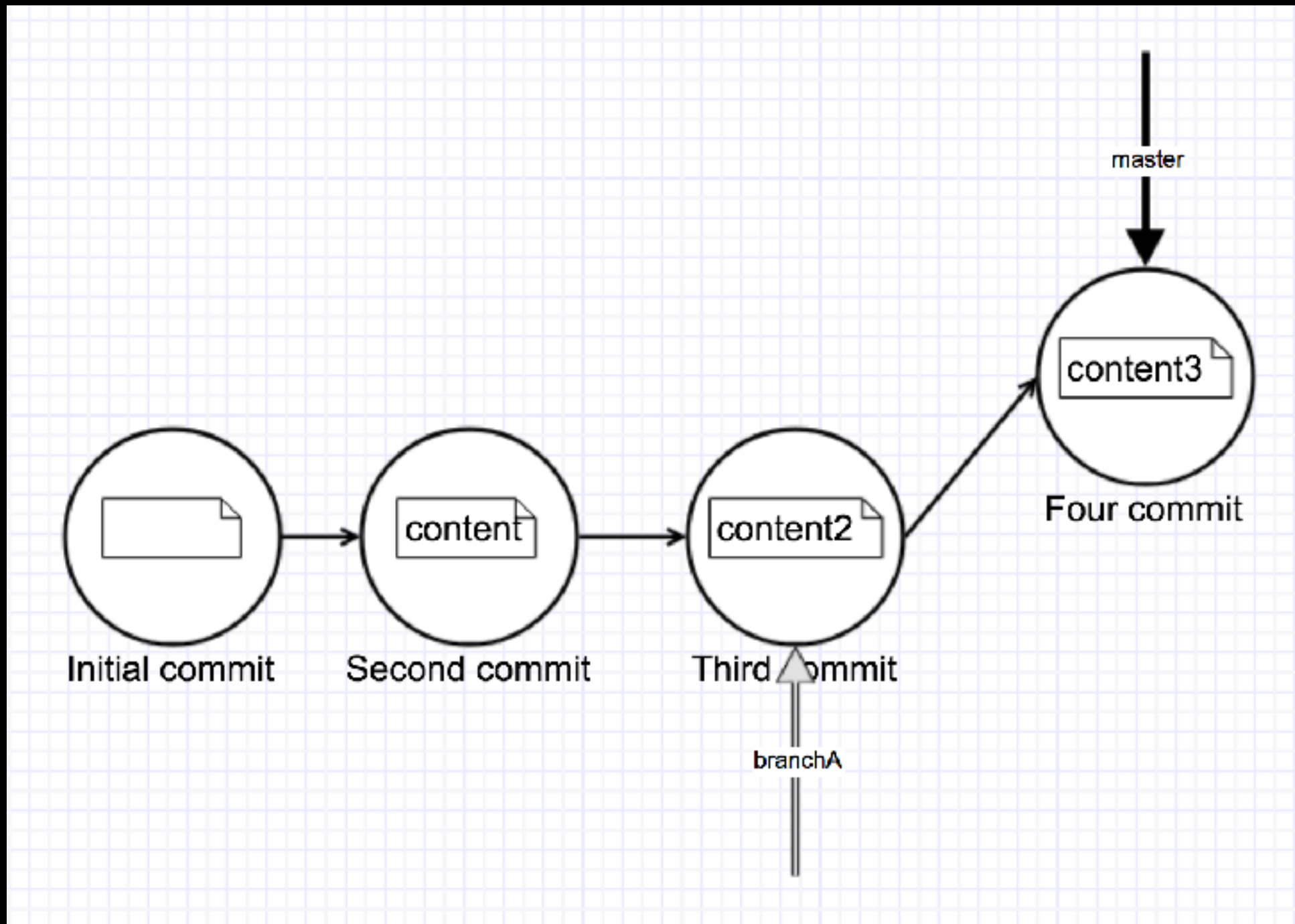
# checkout 分支



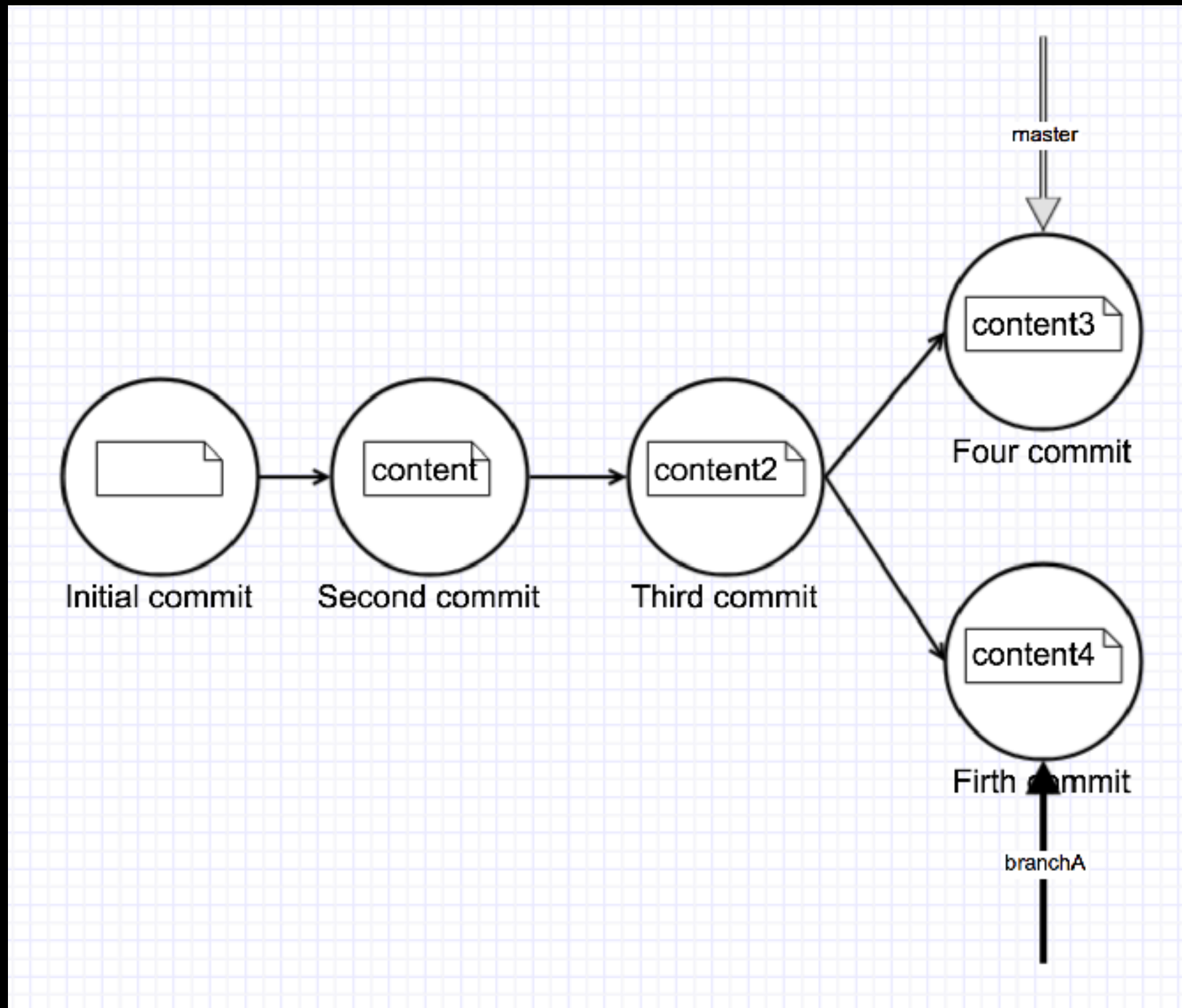
# branch



# branch



# branch

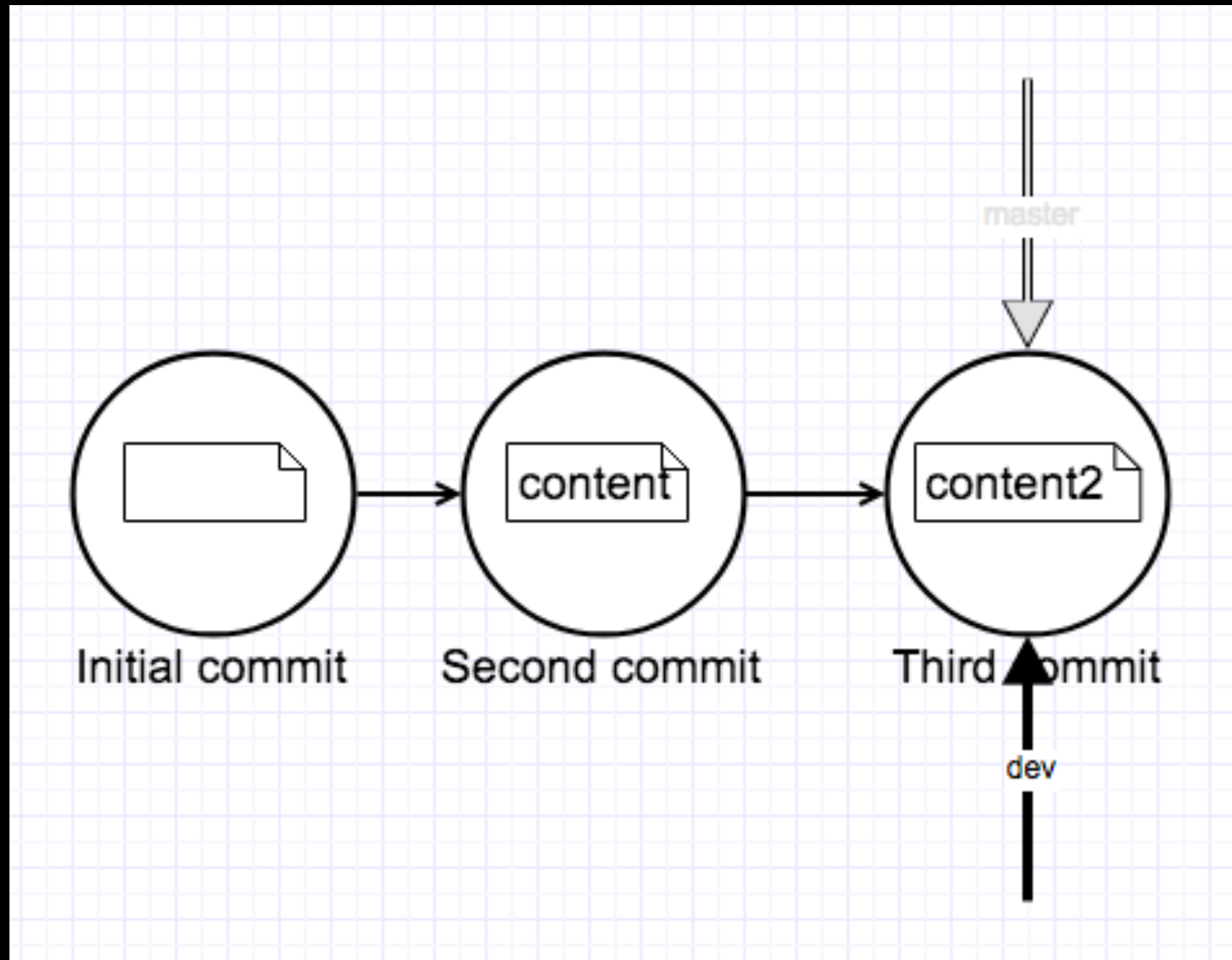




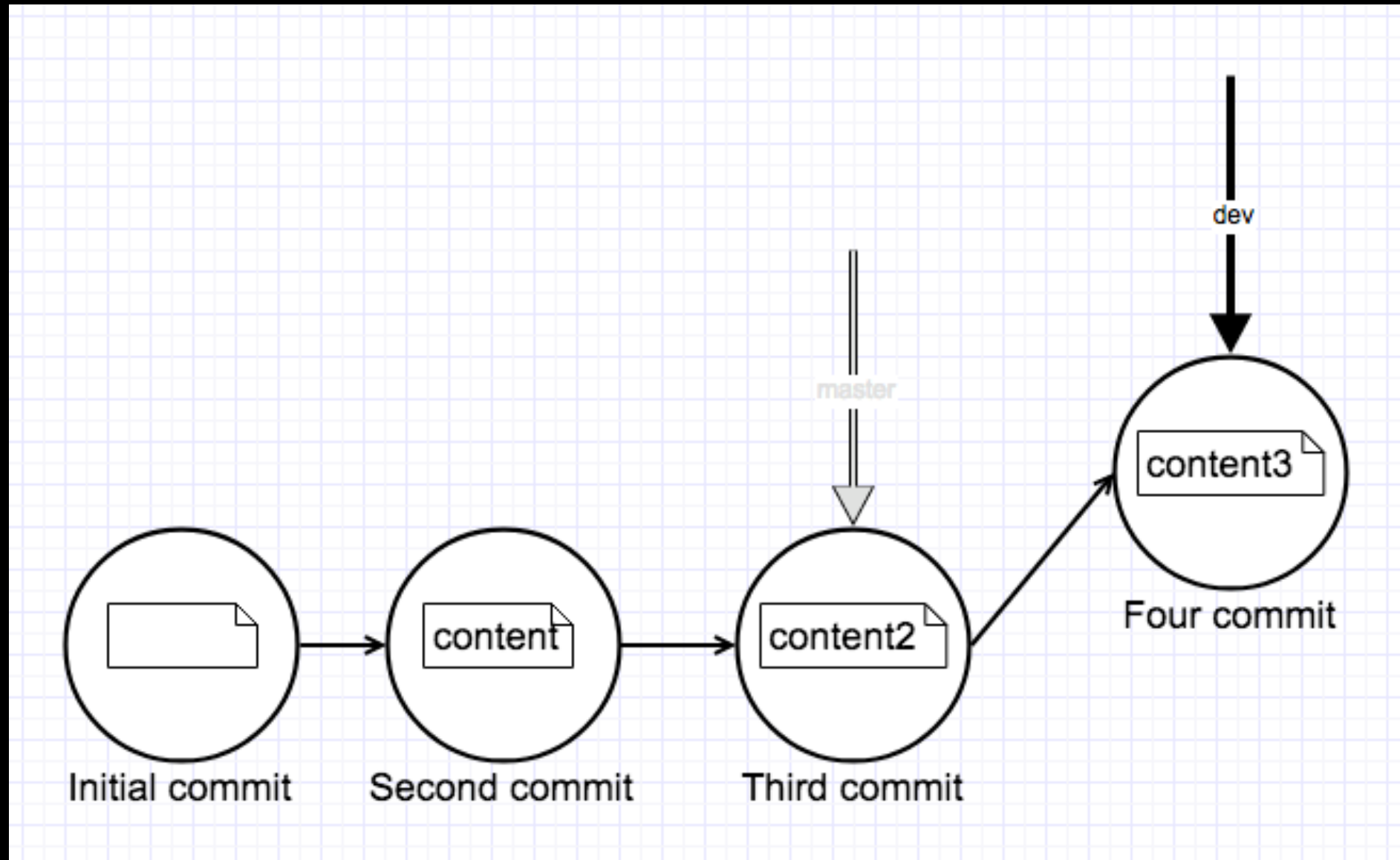
# 回顾

- git branch
- git checkout

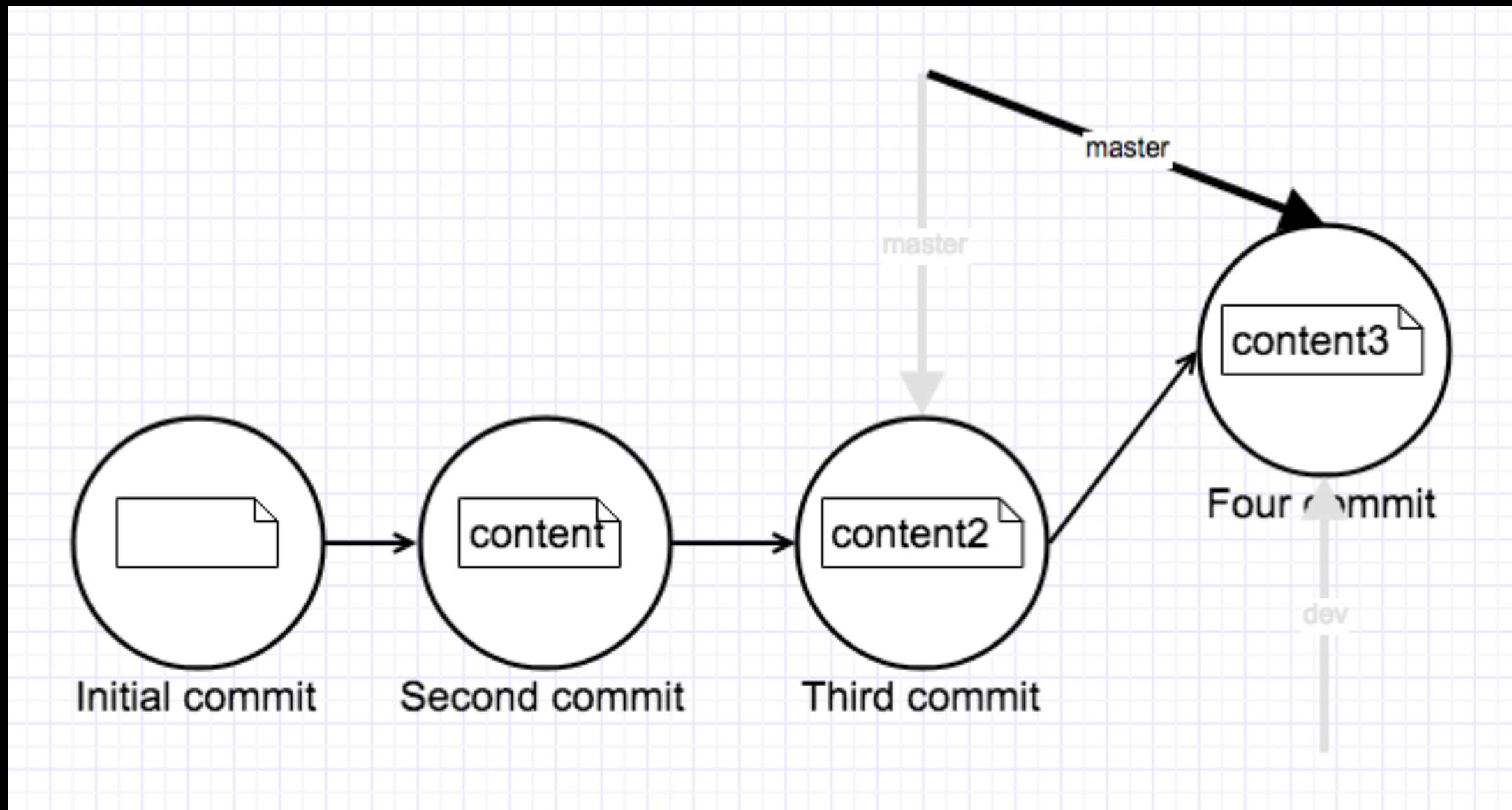
# merge



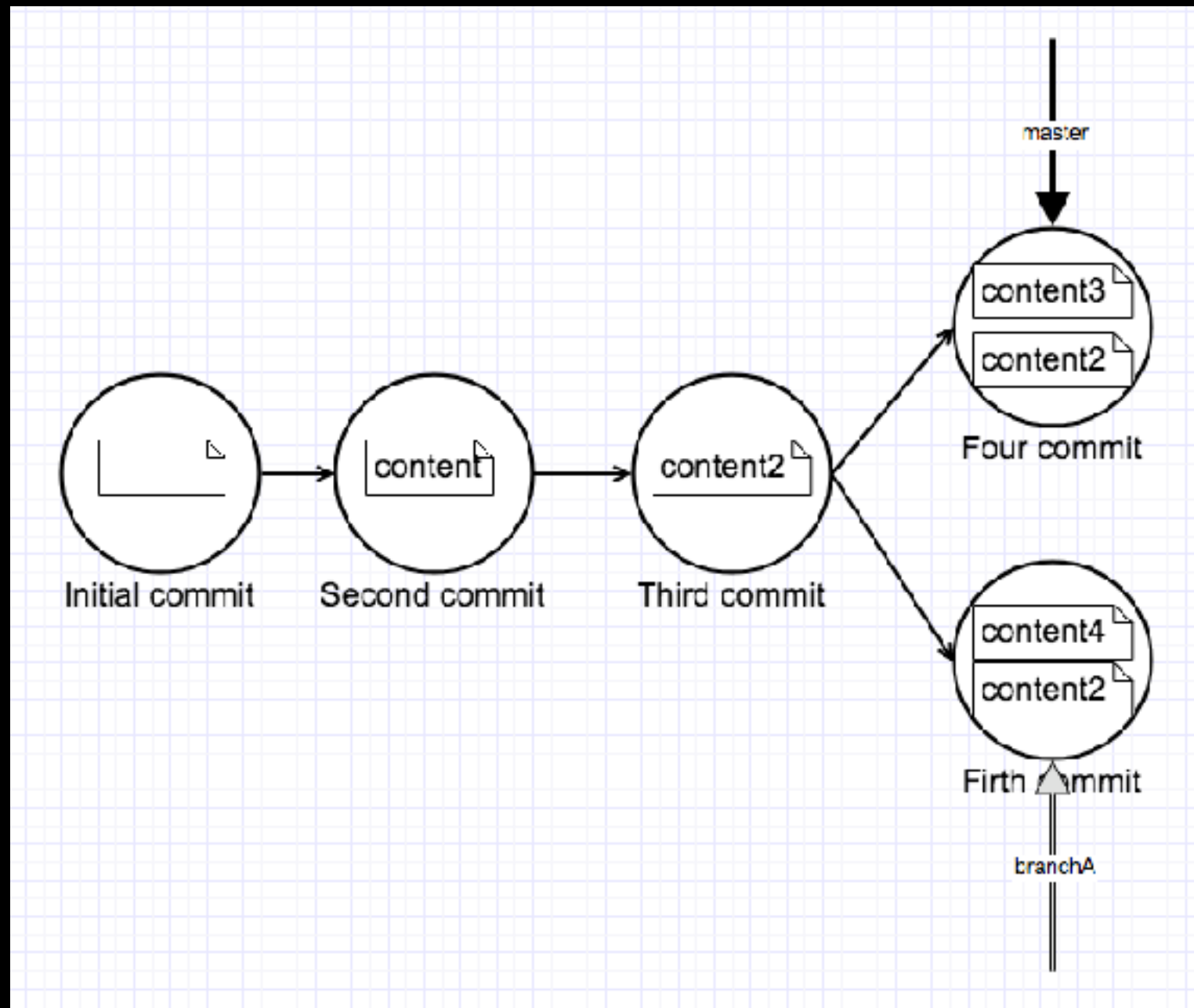
# merge



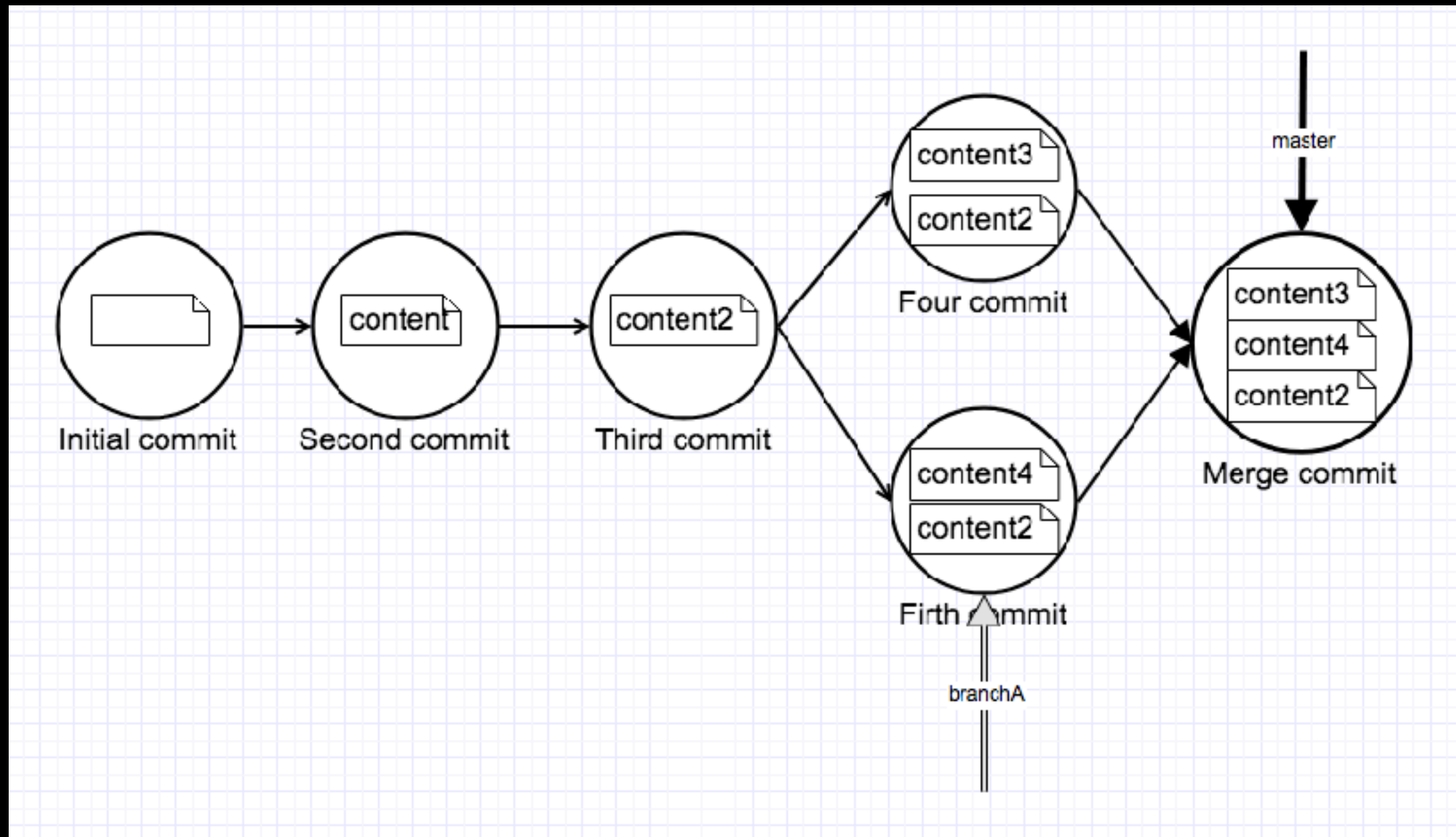
# merge (fast-forward)



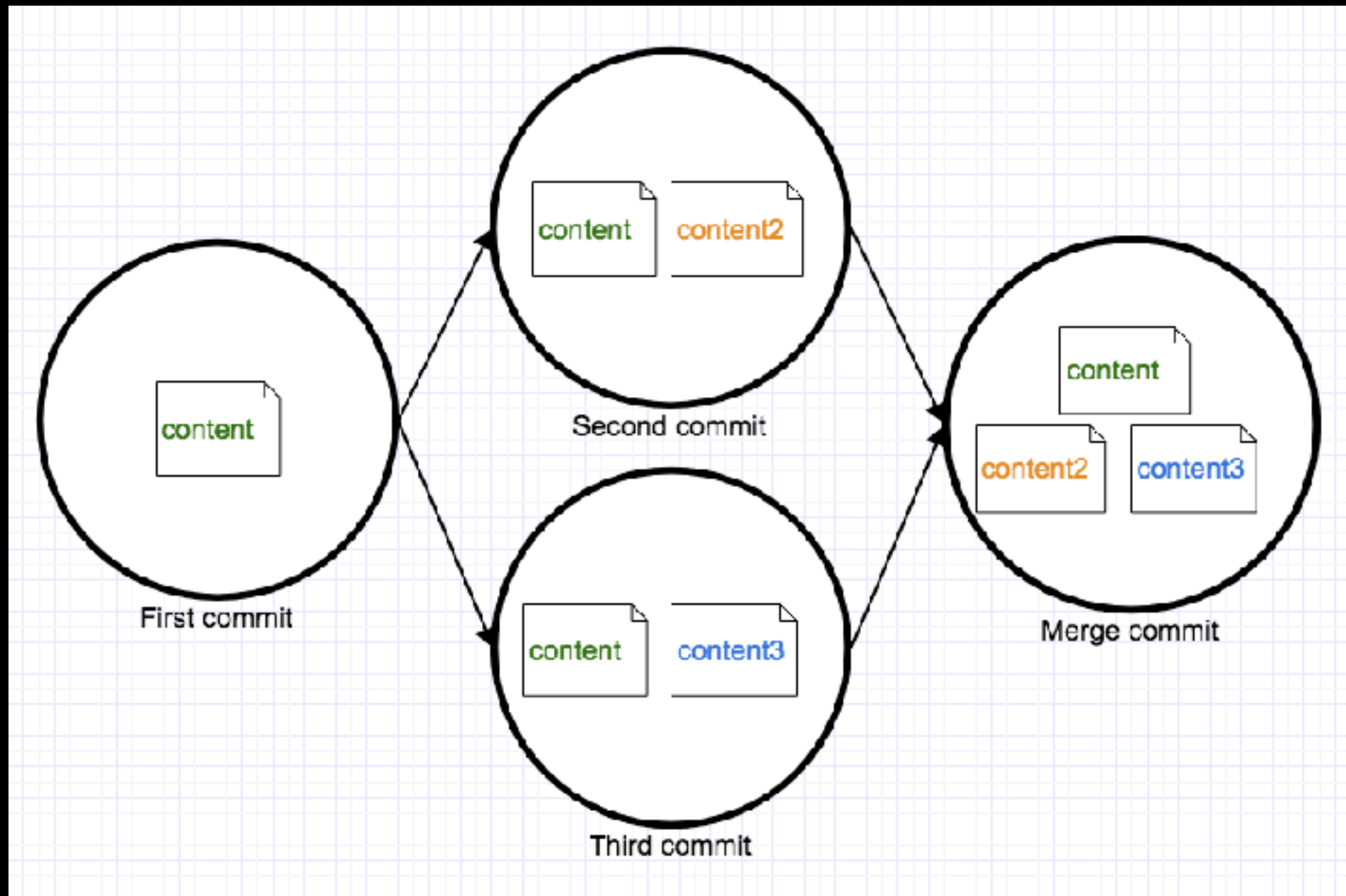
# merge (not-ff)



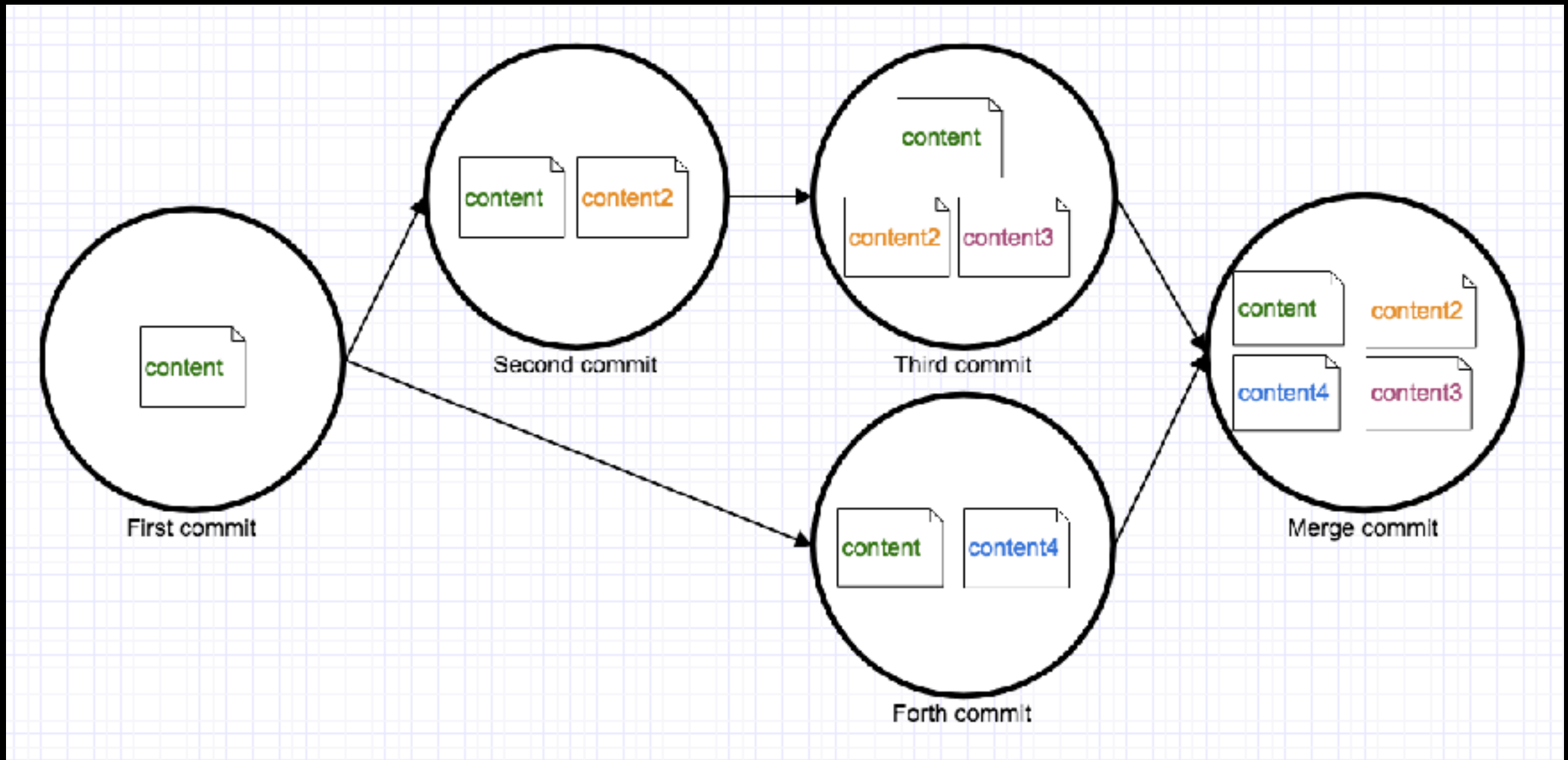
# merge (not-ff)



# merge (not-ff)

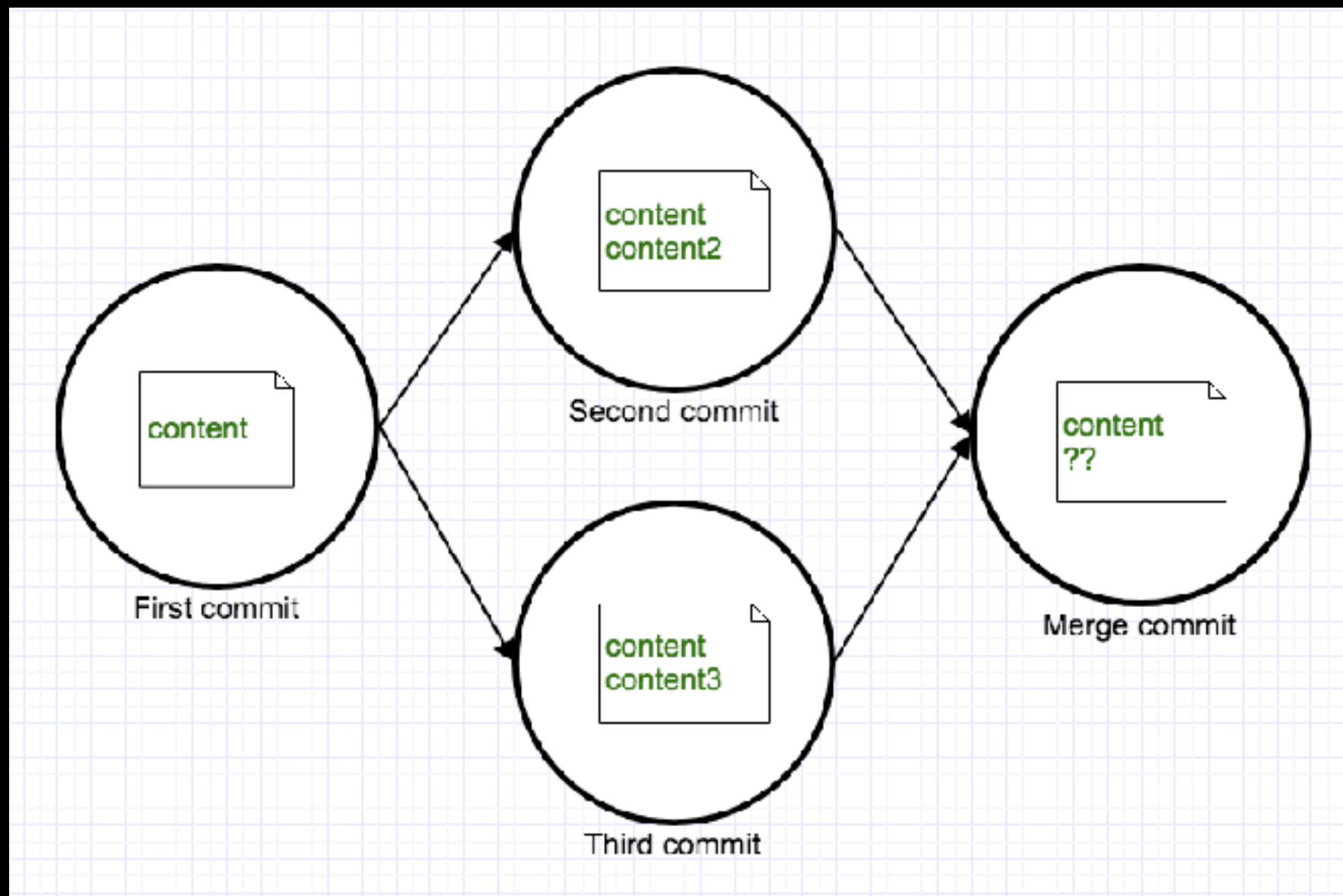


# merge (not-ff)





# merge 冲突

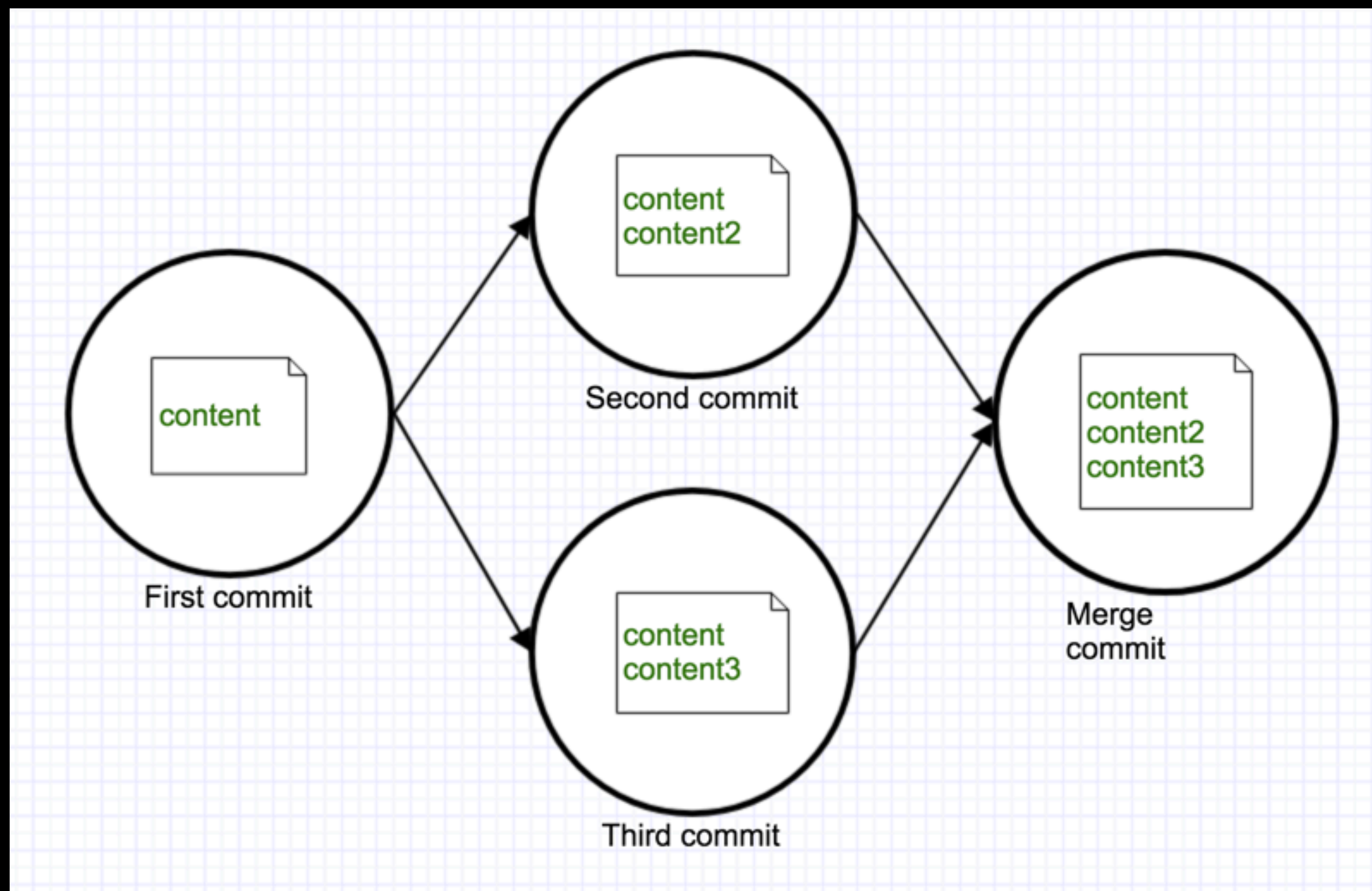


# merge 冲突

```
1 content  
2 <<<<<< HEAD  
3 content2  
4 =====  
5 content3  
6 >>>>>> dev
```

```
1 content  
2 <<<<<< HEAD  
3 content3  
4 =====  
5 content2  
6 >>>>>> master
```

# merge 冲突



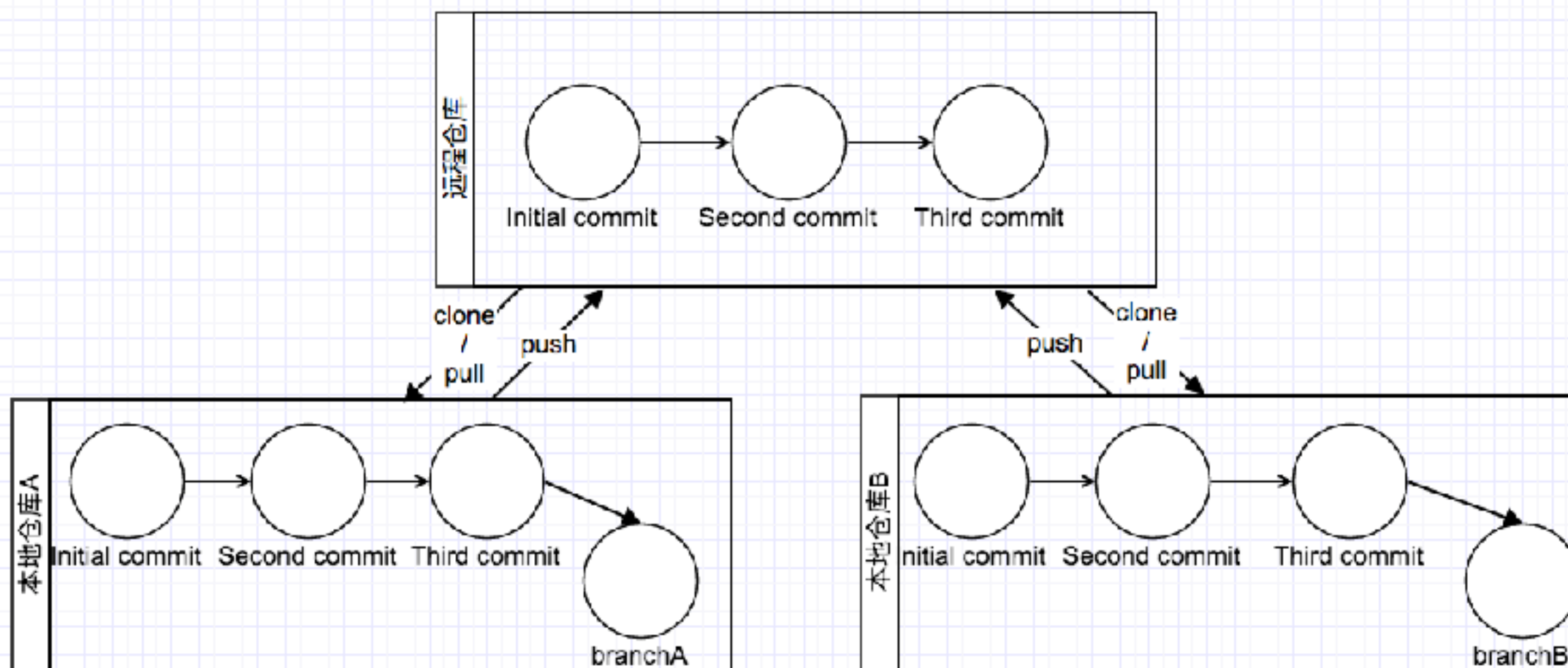
# 回顾

- git merge

# 回顾

- `git merge xxx`
- 提示冲突
- 修改冲突部分，去除冲突标记（<<<< ===== >>>>）
- `git add xxx`
- `git commit`

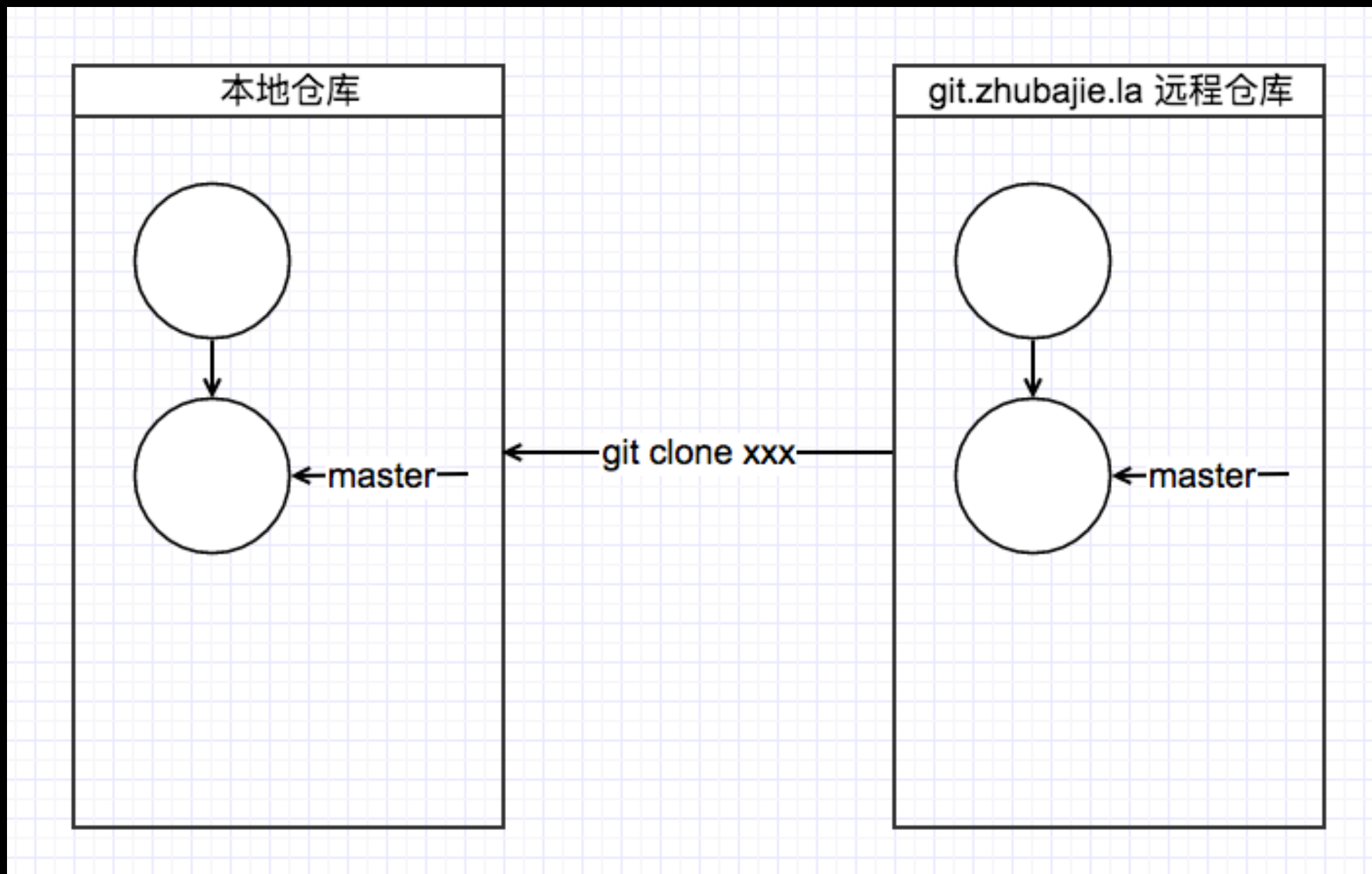
# 远程仓库



# git.zhubajie.la

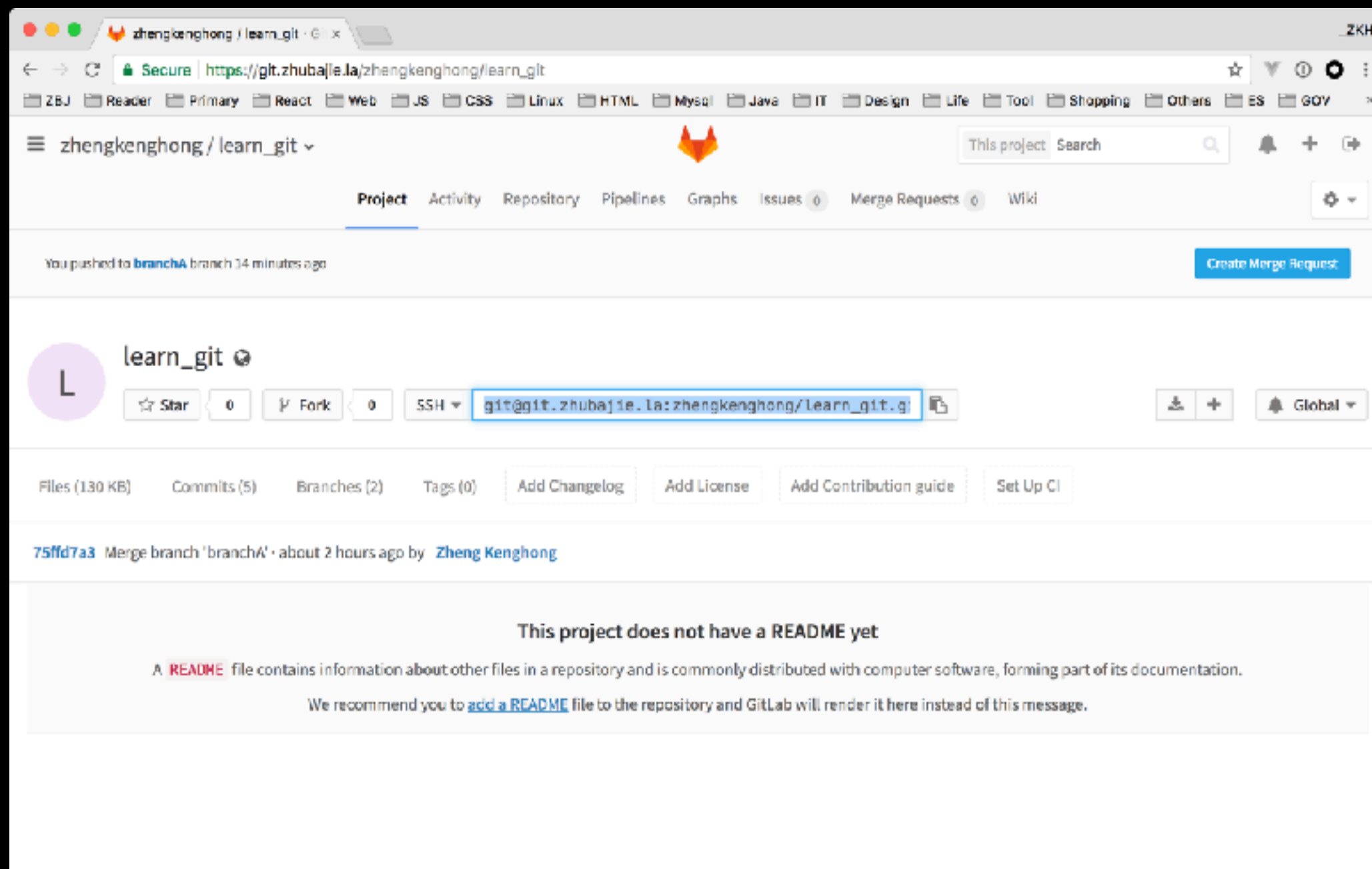
- GitLab
- 代码托管系统
- git 远程仓库

# 远程仓库





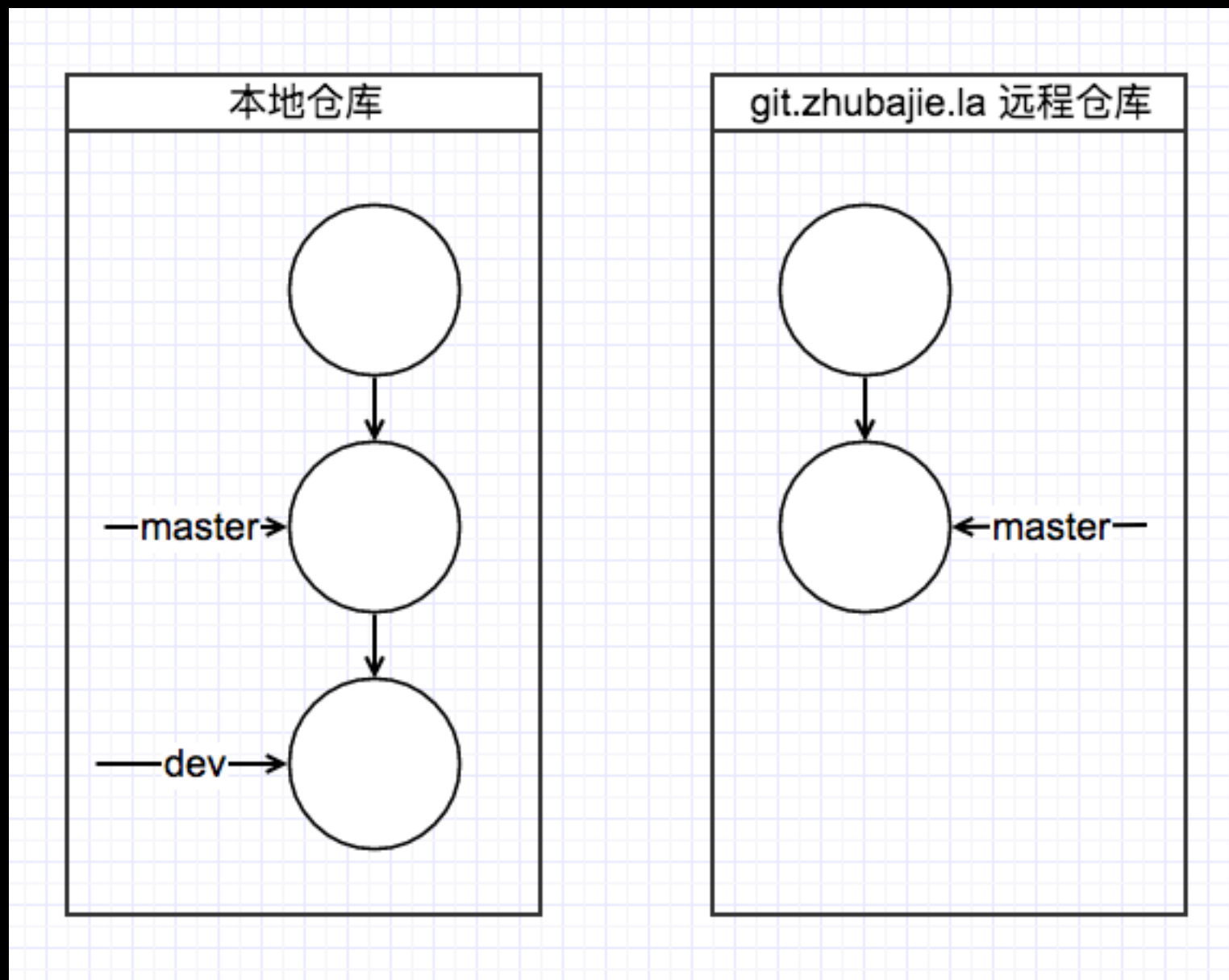
# 远程仓库



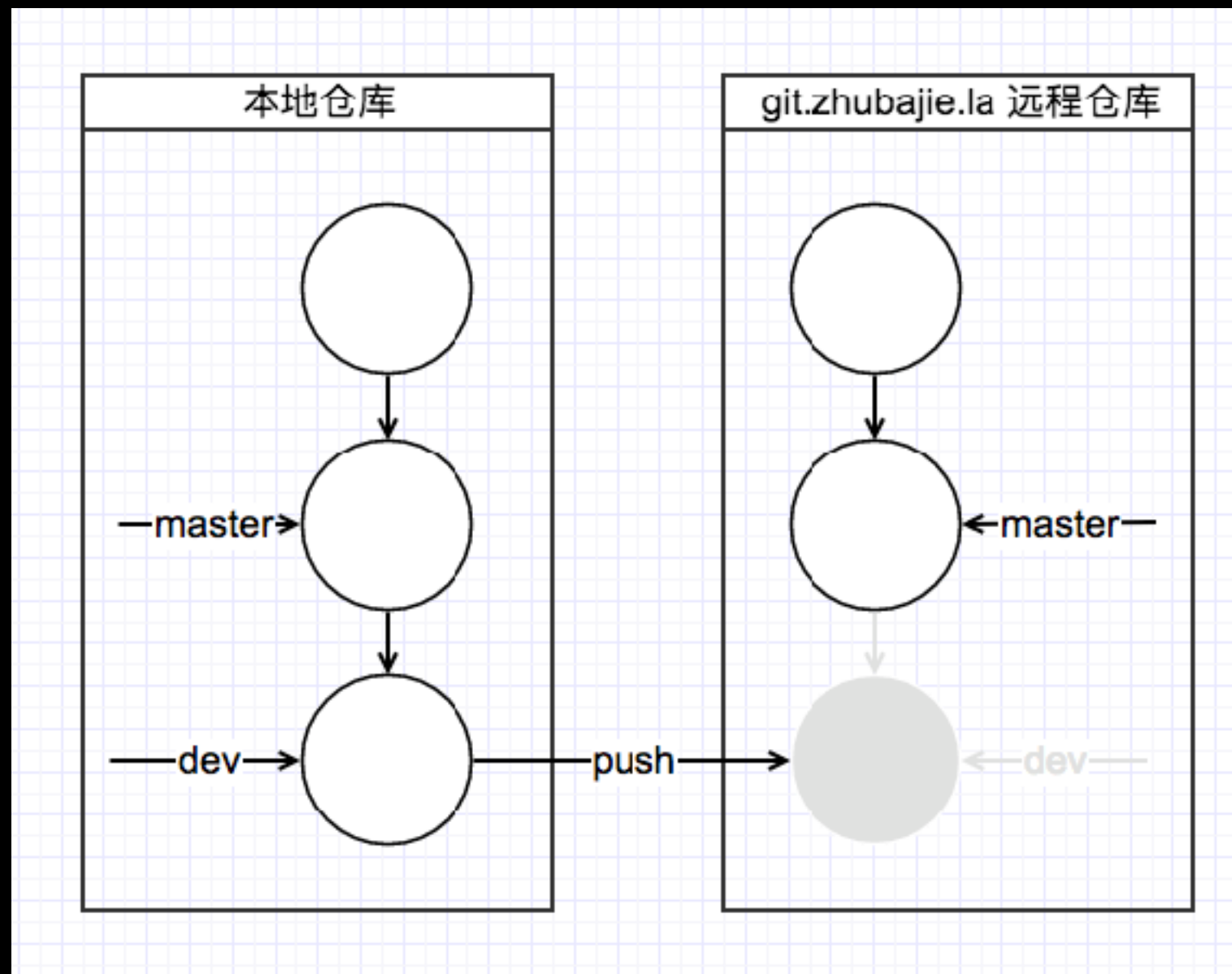
# clone

- 公有仓库 / 私有仓库
- HTTP vs SSH
- 生成 SSH: <https://git.zhubajie.la/help/ssh/README>
- 配置 SSH: <https://git.zhubajie.la/profile/keys>

# 远程仓库



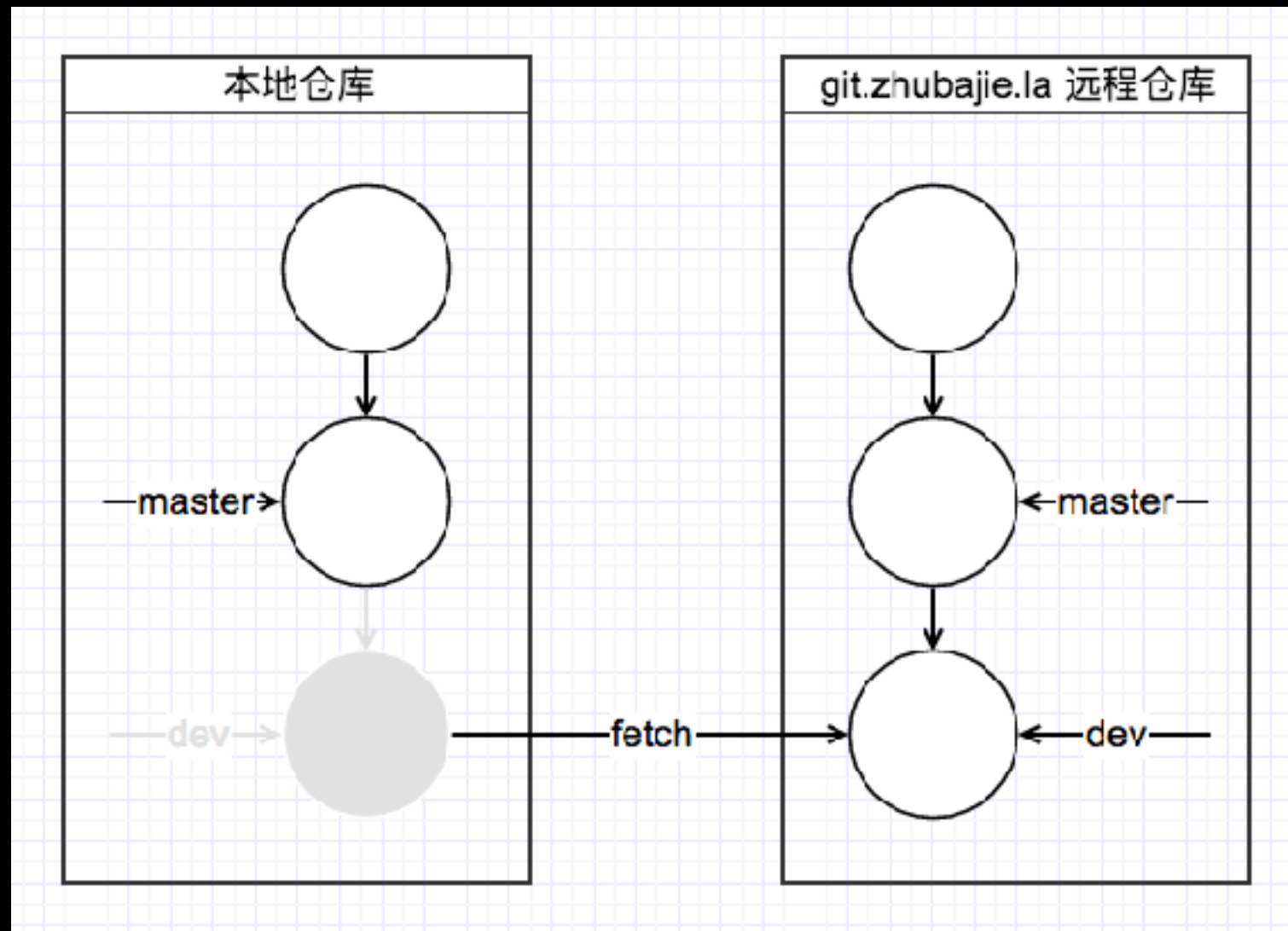
# push



# git push origin dev

- 推送分支 / commit 到远程仓库
- git clone 之后，创建 origin 来指向该远程仓库
- origin 可以理解为是一个指针
- 本地仓库可以同时指向多个远程仓库
- 推送本地仓库的 dev 分支到远程仓库里

# fetch



# 本地分支 / 远程分支

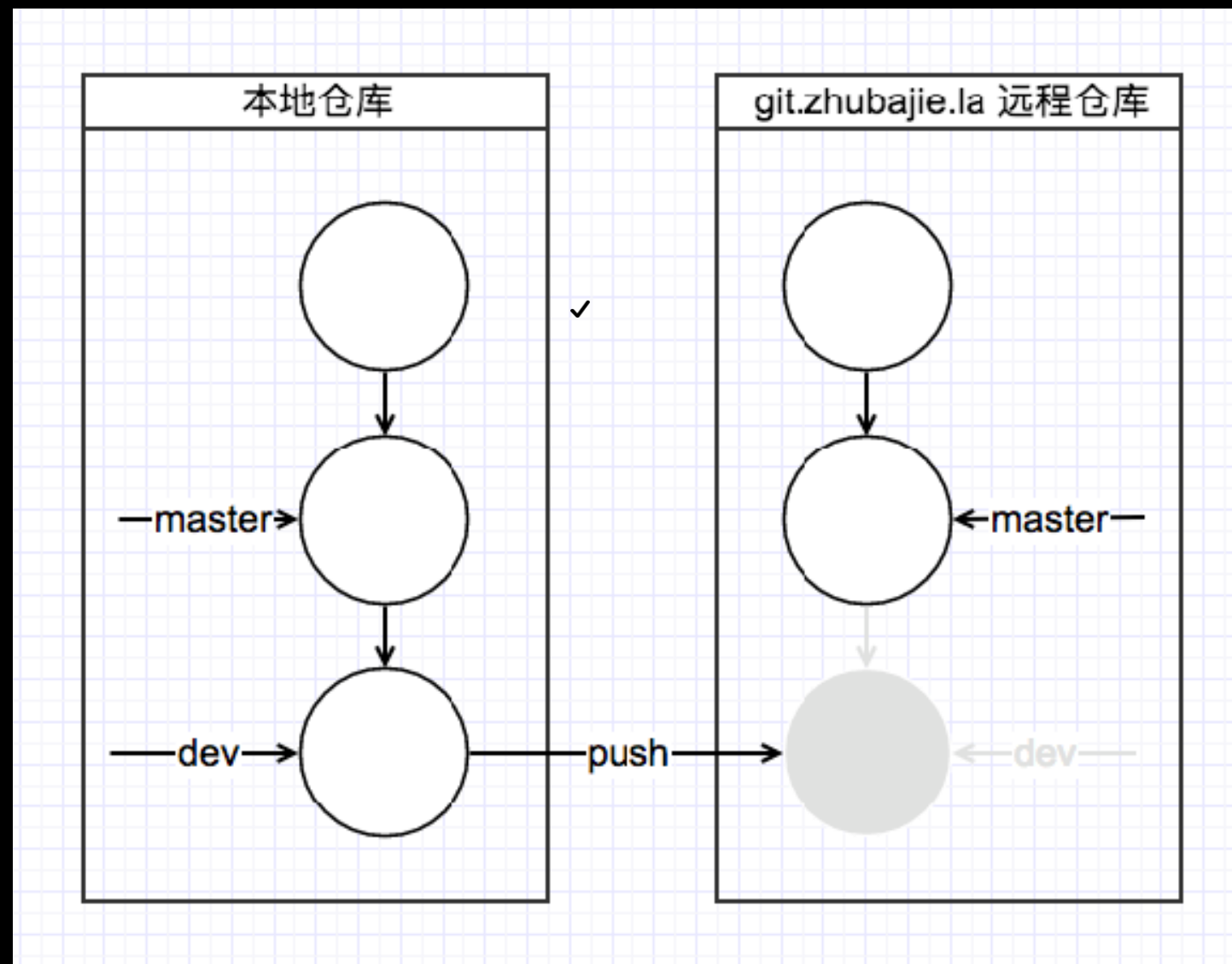
- 本地仓库创建的分支、远程仓库里有的分支
- `git branch xxx`、`git fetch`
- `git branch`、`git branch -a`
- 本质上都是指向 `commit` 的引用

# 回顾

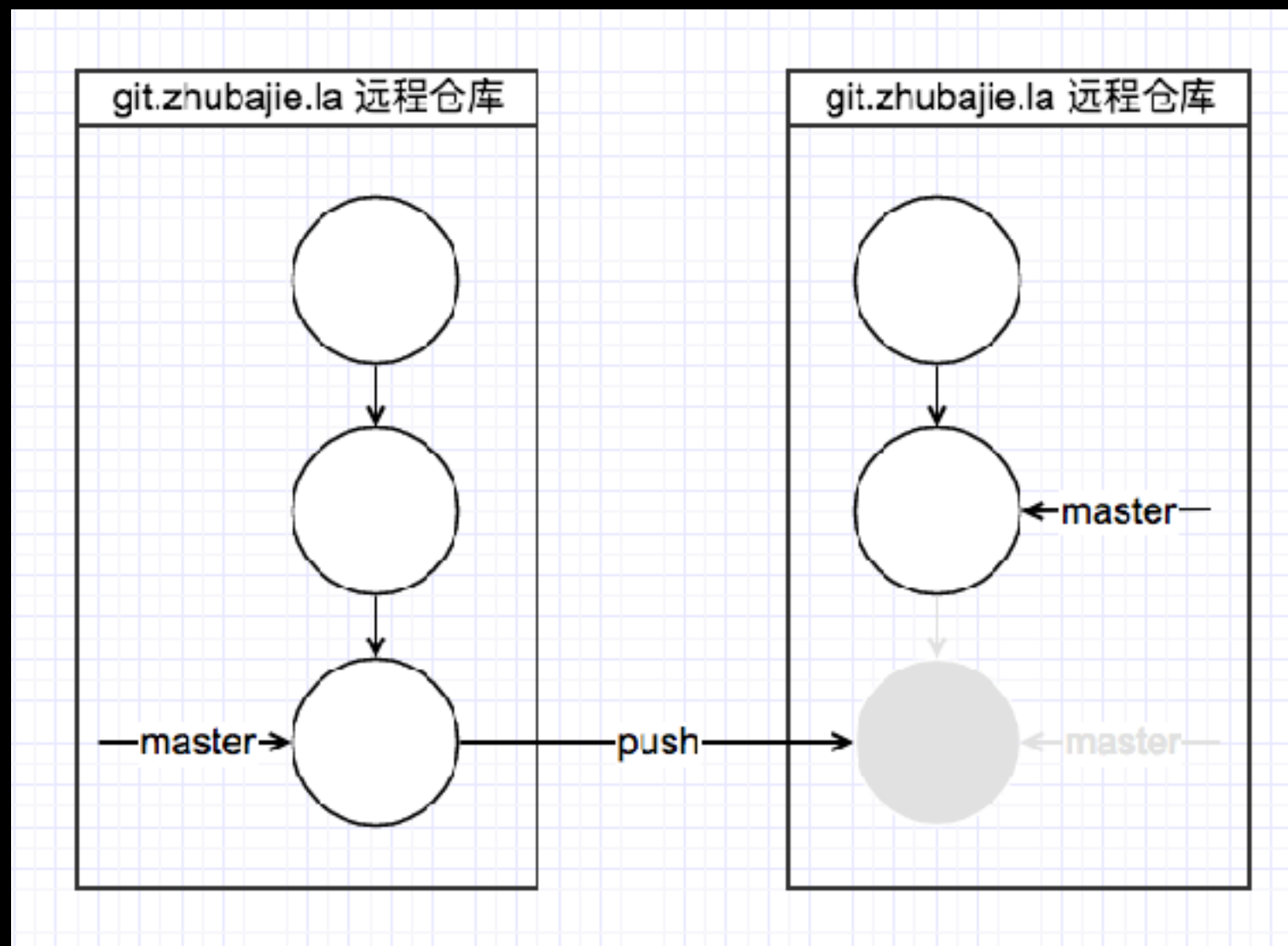
- git clone
- git push
- git fetch



# push master



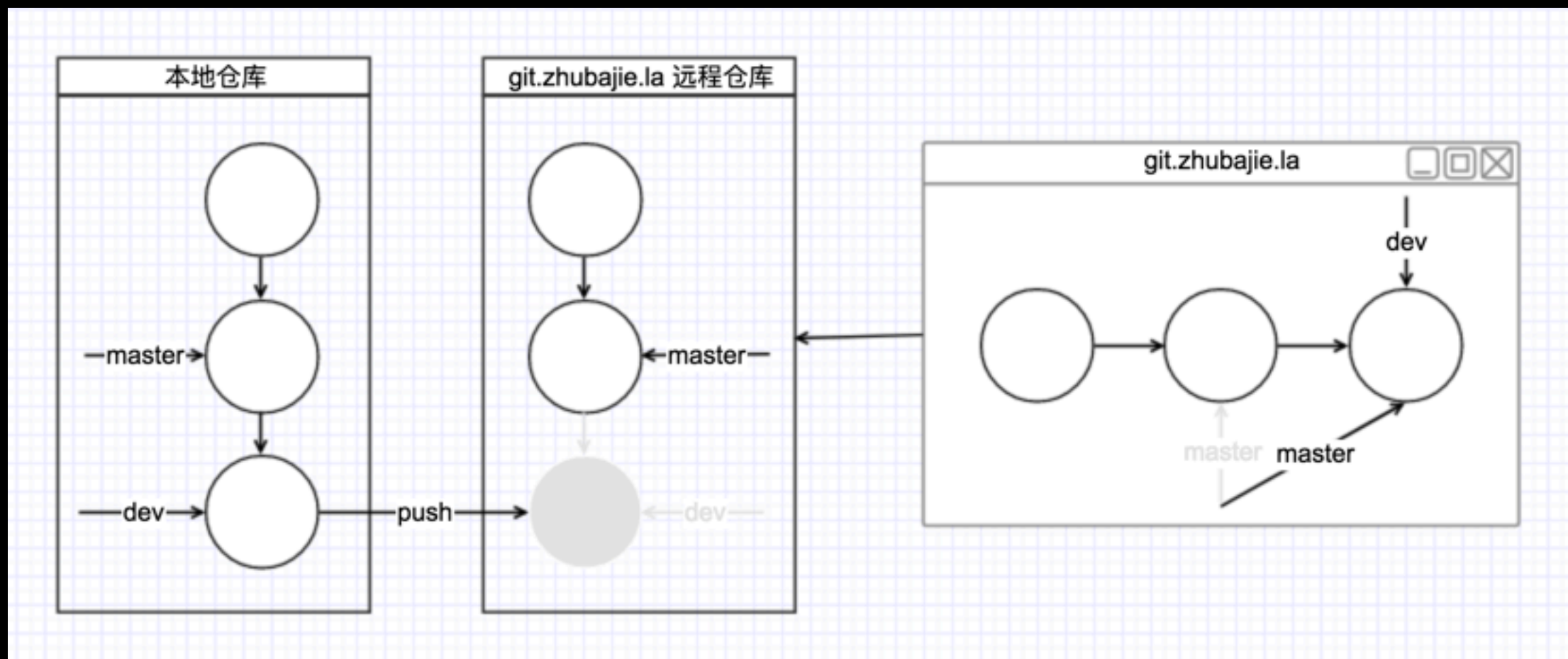
# push master



# push master

- 开发者都需要有 master 推送权限
- master 不可控
- 无法 code review

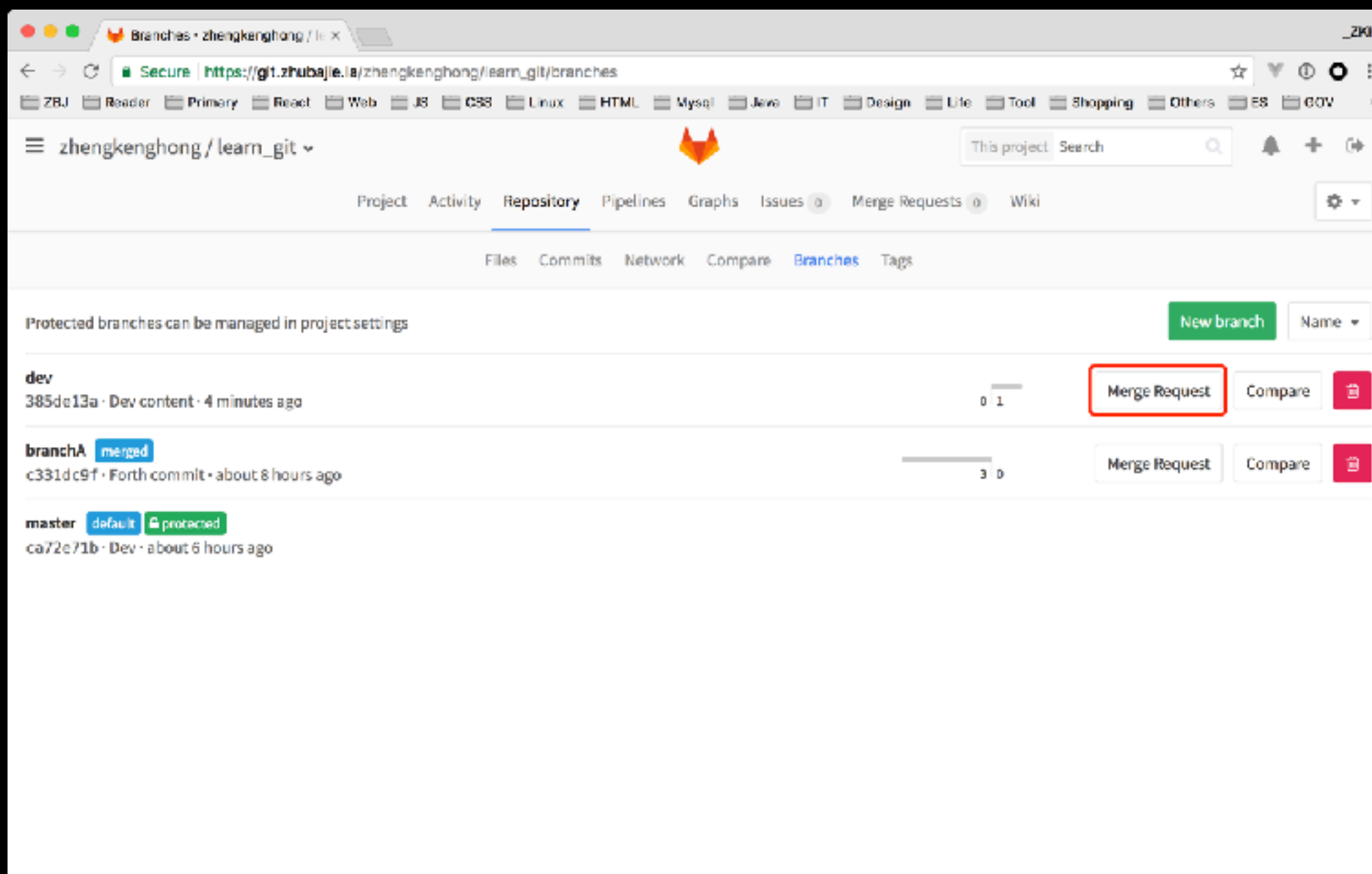
# MR 流程



# MR 流程

```
~/P/learn_git > git push origin dev
Counting objects: 3, done.
Writing objects: 100% (3/3), 271 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@git.zhubajie.la:zhengkenghong/learn_git.git
 * [new branch]      dev -> dev
~/P/learn_git > █
```

# MR 流程



# MR 流程

New Merge Request · zhengkenghong/learn\_git

Secure | [https://git.zhubajie.la/zhengkenghong/learn\\_git/merge\\_requests/new?merge\\_request%5Bsource\\_branch%5D=dev&merge\\_request%5Btarget\\_branch%5D=master](https://git.zhubajie.la/zhengkenghong/learn_git/merge_requests/new?merge_request%5Bsource_branch%5D=dev&merge_request%5Btarget_branch%5D=master)

zhengkenghong/learn\_git

Project Activity Repository Pipelines Graphs Issues Merge Requests Wiki

### New Merge Request

From **dev** into **master** [Change branches](#)

Title

[Start the title with WIP:](#) to prevent a **Work In Progress** merge request from being merged before it's ready.

Description

**Write** Preview [Go full screen](#)

Write a comment or drag your files here...

Styling with [Markdown](#) is supported [Attach a file](#)

Assignee

[Assign to me](#)

Milestone

[Create new milestone](#)

# MR 流程

The screenshot shows a web browser window with the title "New Merge Request · zhengkenghong". The address bar displays a secure URL: `https://git.zhubajie.la/zhengkenghong/learn_git/merge_requests/new?merge_request%5Bsource_branch%5D=dev&merge_request%5Btarget_branch%5D=master`. The browser's tab bar shows several open tabs including "ZBJ", "Reader", "Primary", "React", "Web", "JS", "CSS", "Linux", "HTML", "Mysql", "Java", "IT", "Design", "Life", "Tool", "Shopping", "Others", "ES", and "GOV".

The page header for the repository "zhengkenghong / learn\_git" includes navigation links: "Project", "Activity", "Repository", "Pipelines", "Graphs", "Issues", "Merge Requests", and "Wiki". A search bar is also present.

The main content area is for creating a new merge request. It includes a link to "Create new milestone" and a section for "Labels" with the text "No labels yet." and a link to "Create new label".

The "Source branch" dropdown is set to "dev" and the "Target branch" dropdown is set to "master". Both dropdowns are highlighted with red boxes. A link "Change branches" is next to the target branch dropdown. Below these, there is an unchecked checkbox labeled "Remove source branch when merge request is accepted."

At the bottom of the form, there is a green button labeled "Submit merge request" (highlighted with a red box) and a "Cancel" button.

Below the form, there are tabs for "Commits 1" and "Changes 1". The "Commits" tab is active, showing a list of commits. The first commit is from "22 Aug, 2017 1 commit". The commit details show a file named "Dev content" with a commit hash of "385de13a" and the message "Zheng Kenghong authored 8 minutes ago". A "Browse Files" button is next to the commit hash.



# MR 流程

The screenshot shows a GitHub Merge Request (MR) for the repository 'zhengkenghong/learn\_git'. The MR is titled 'Dev content' and was opened 3 minutes ago by the user 'zhengkenghong'. The request is to merge the 'dev' branch into the 'master' branch. The interface includes a sidebar with navigation links (Project, Activity, Repository, Pipelines, Graphs, Issues, Merge Requests, Wiki) and a top bar with a search bar and notification icons. The main content area shows the MR details, including a green 'Accept Merge Request' button, a 'Remove source branch' checkbox, and a 'Modify commit message' checkbox. Below this, there is a section for 'Changes' showing a diff for the file 'track'. The diff shows three additions (content, content2, content3) and one deletion (-dev). The right sidebar contains a 'Todo' section, a 'Notifications' section with an 'Unsubscribe' button, and a 'Reference' section.

Open Merge Request 11 opened 3 minutes ago by zhengkenghong Close Edit

### Dev content

Request to merge `dev` into `master` Check out branch Download as

**Accept Merge Request** ☐ Remove source branch ☒ Modify commit message

You can also accept this merge request manually using the [command line](#).

Discussion 0 Commits 1 **Changes 1**

Showing 1 changed file with 1 additions and 1 deletions Hide whitespace changes Inline Side-by-side

track

1	1	content
2	2	content2
3	3	content3
4		-dev
	4	+dev content
...	...	

Edit View file @385de13

Unsubscribe

You're receiving notifications because you're subscribed to this thread.

Reference: zhengkenghong/learn\_git

# MR 流程

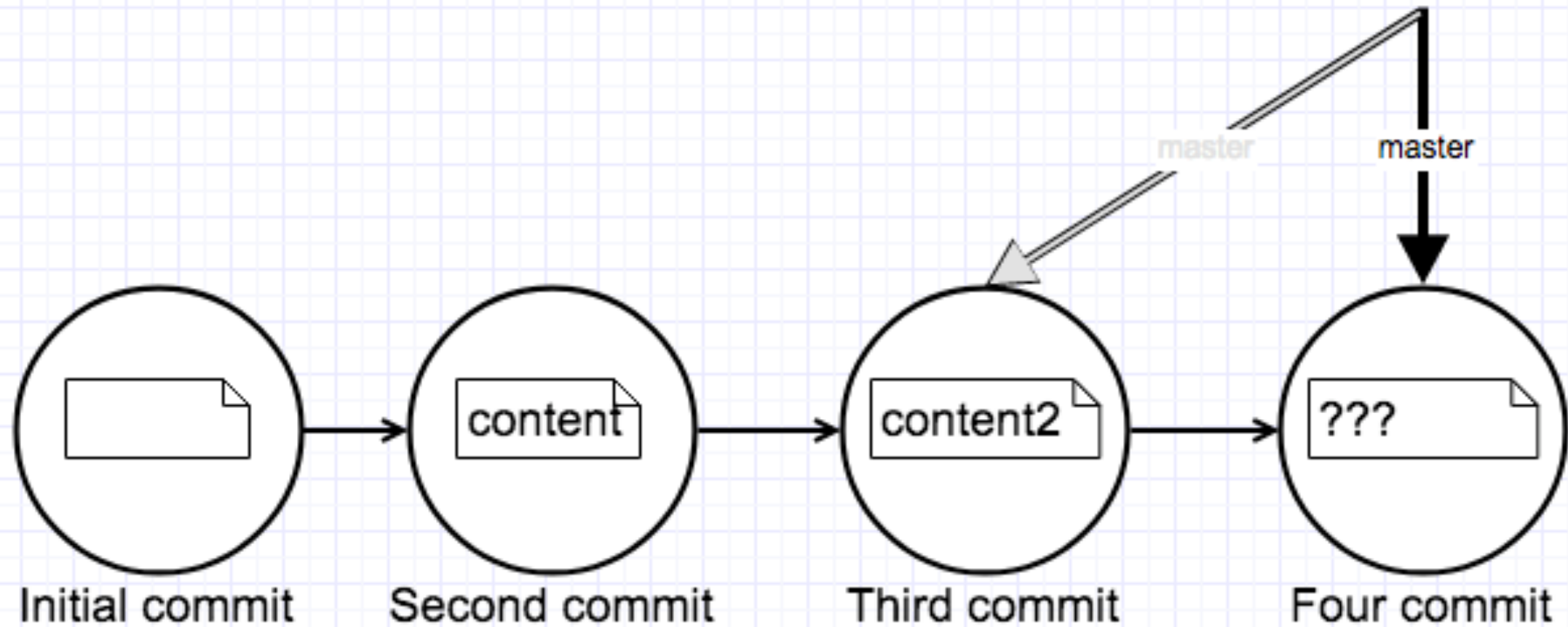
- 推送分支到远程仓库上
- [git.zhubajie.la](https://git.zhubajie.la) 创建一个 <分支 - master> 的 MR
- code review
- 合并 MR

# 回顾

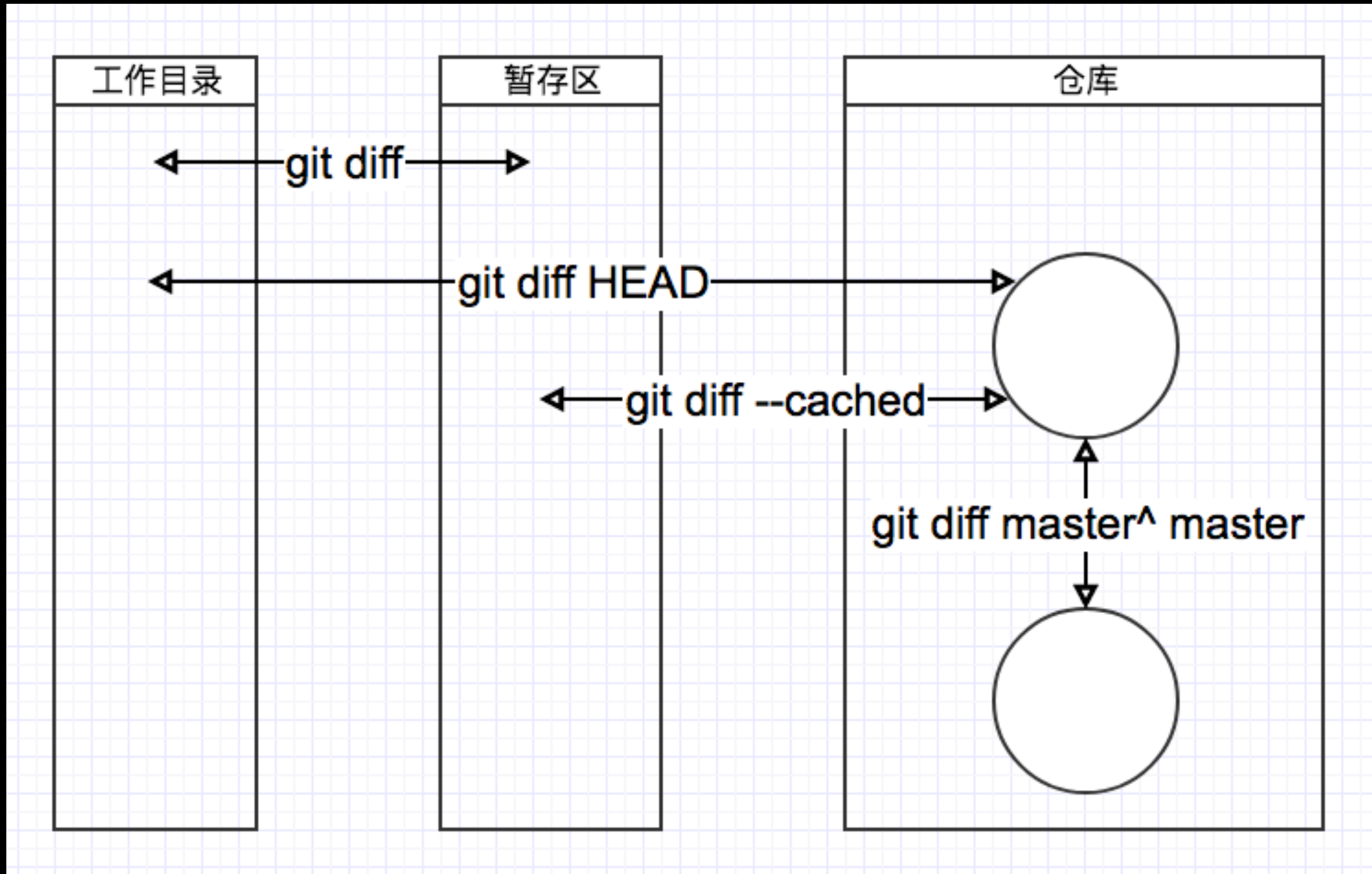
- 把 git merge 交给 [git.zhubajie.la](https://github.com/git-zhubajie) 来做
- 更好的权限控制
- 保证 master 的稳定
- 更好的 code review、更好的代码质量

# 小技巧

# git diff



# git diff



# git checkout -b xxx

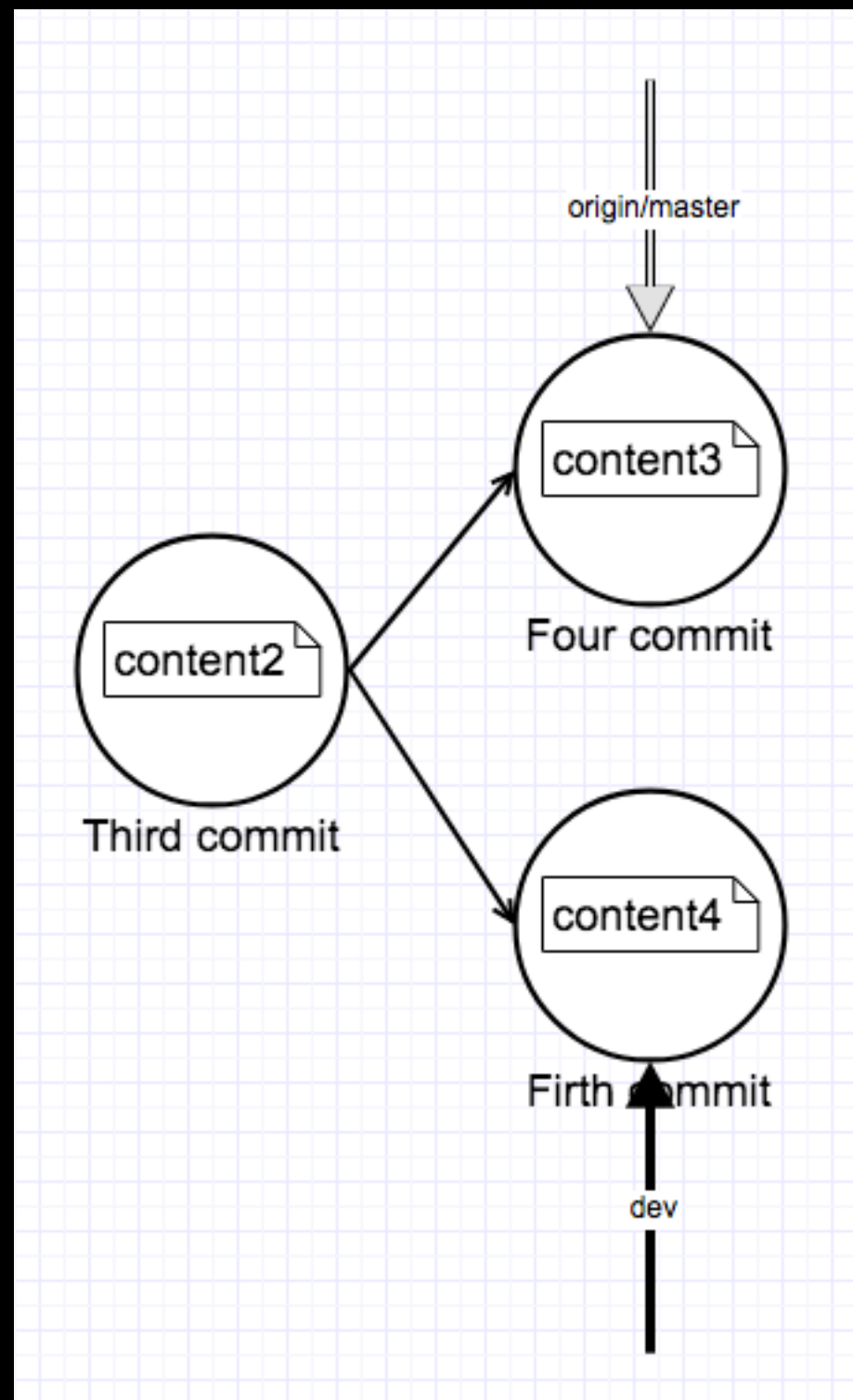
- git branch xx
- git checkout xxx

# git pull origin master

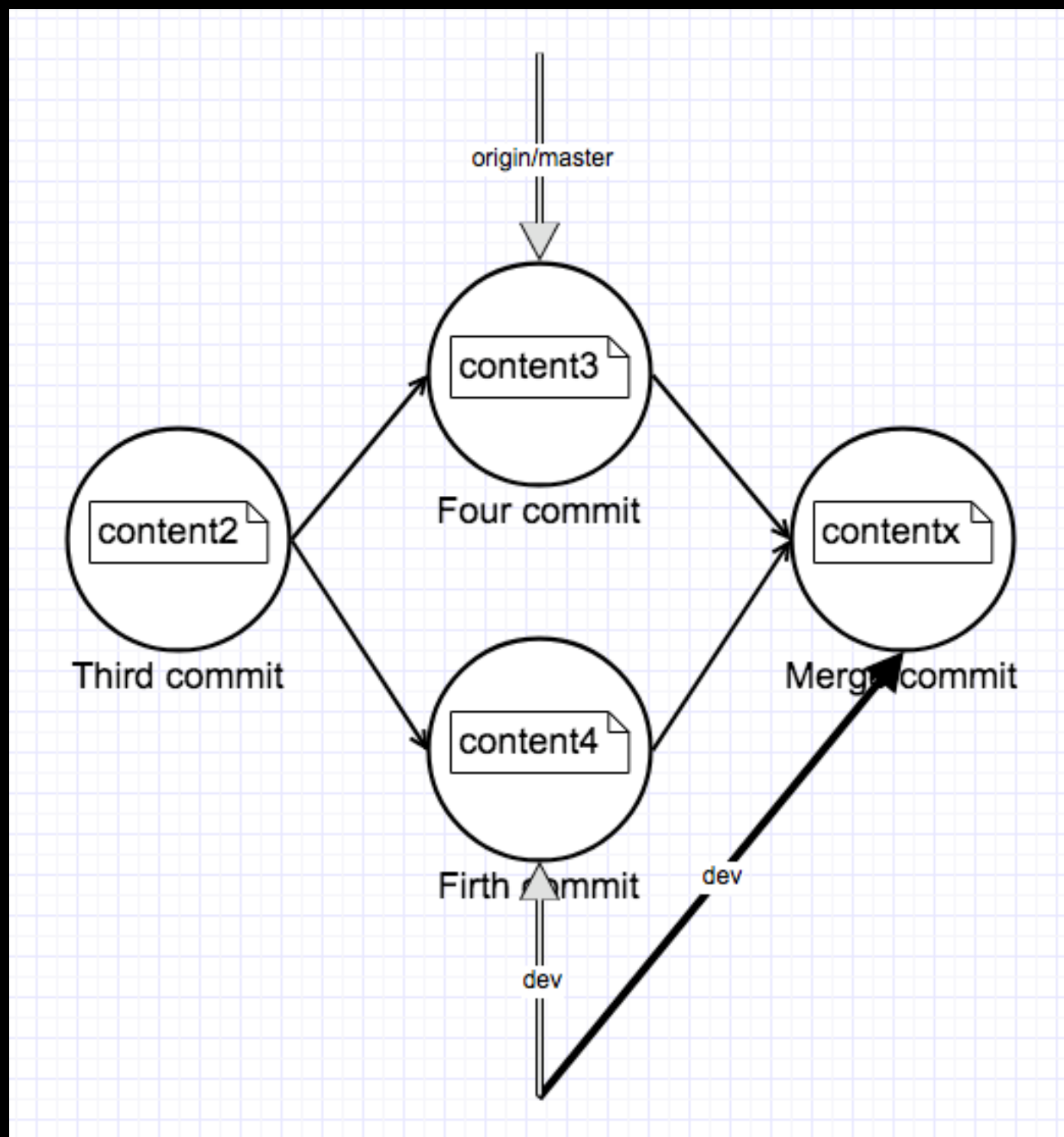
- git fetch origin
- git merge origin/master



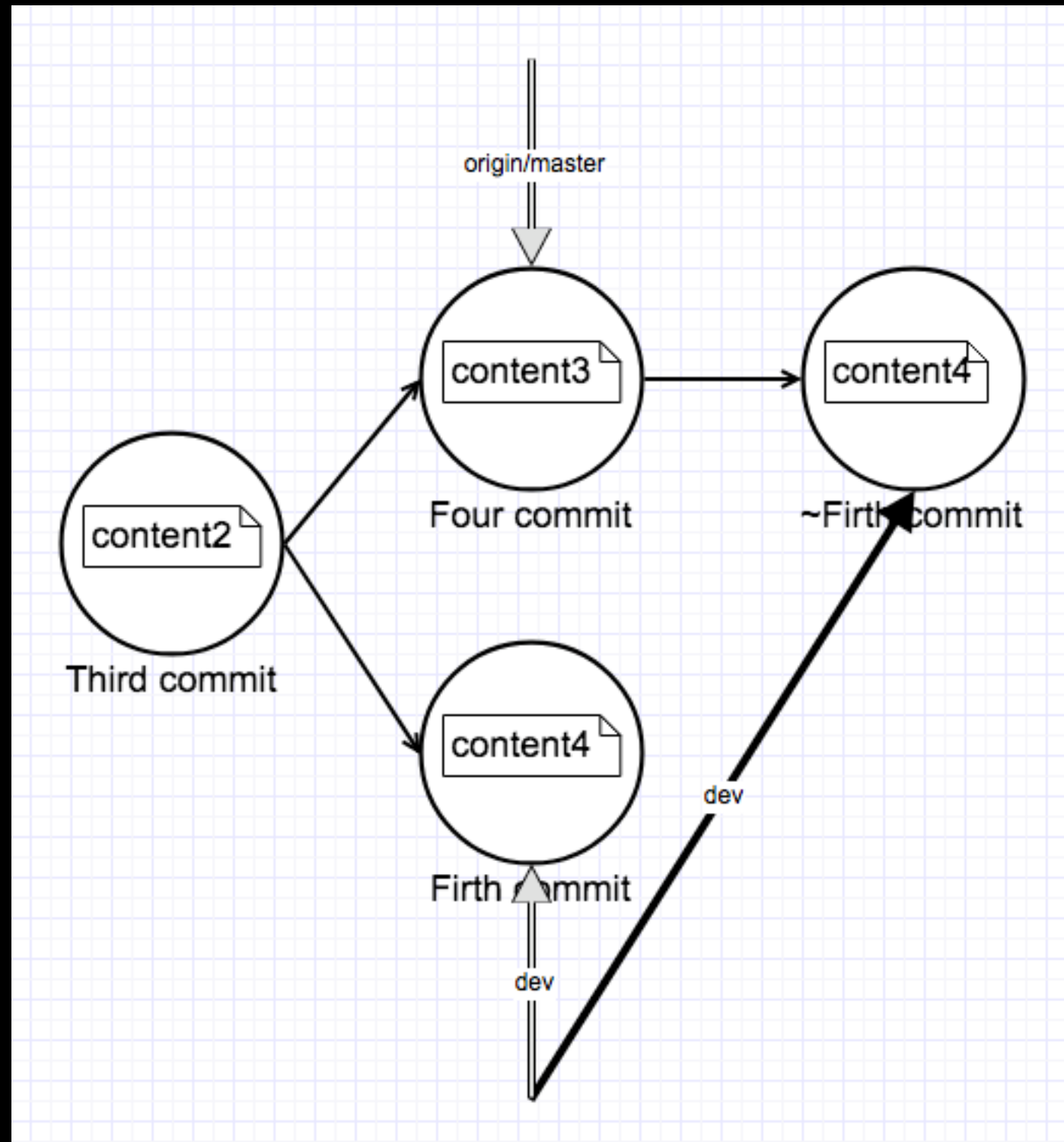
# 基于最新的 master



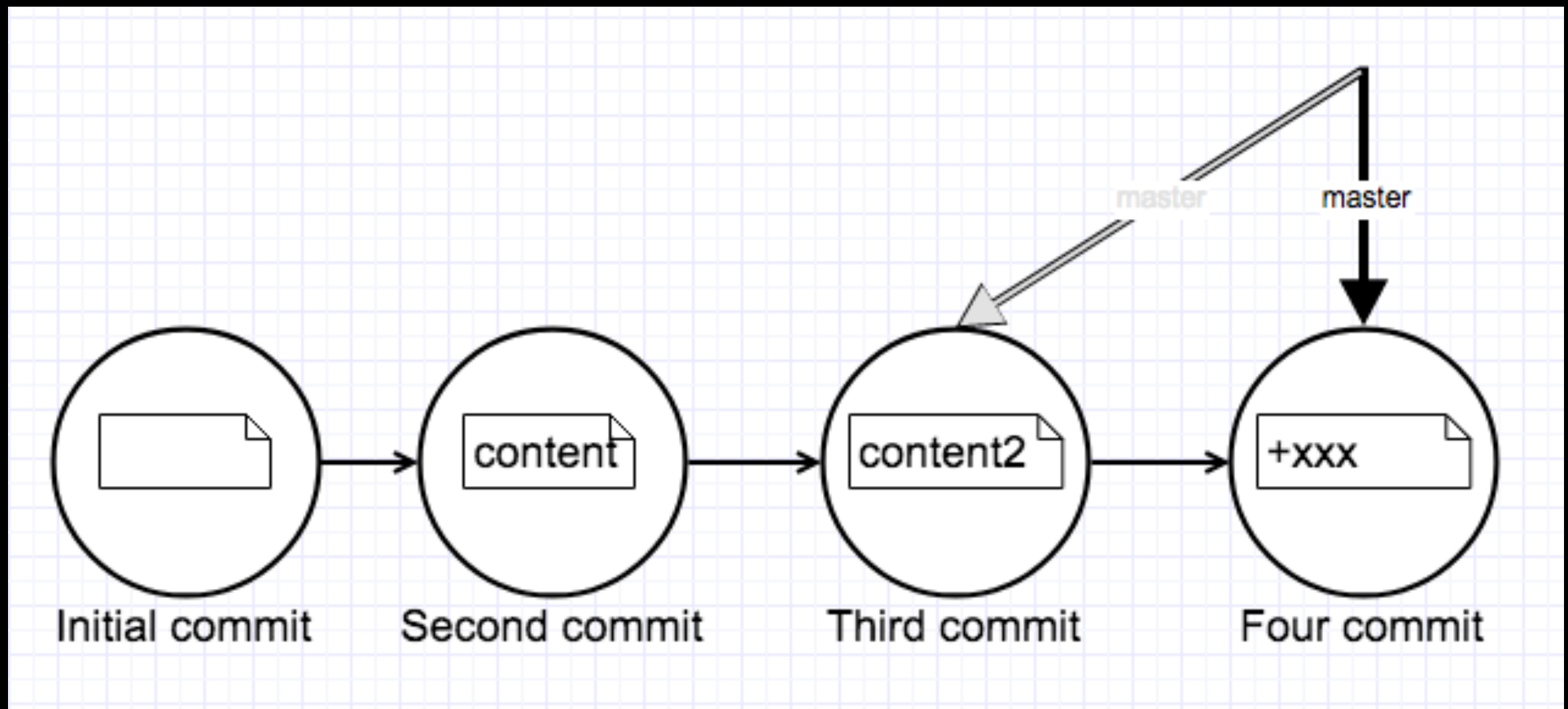
# 基于最新的 master



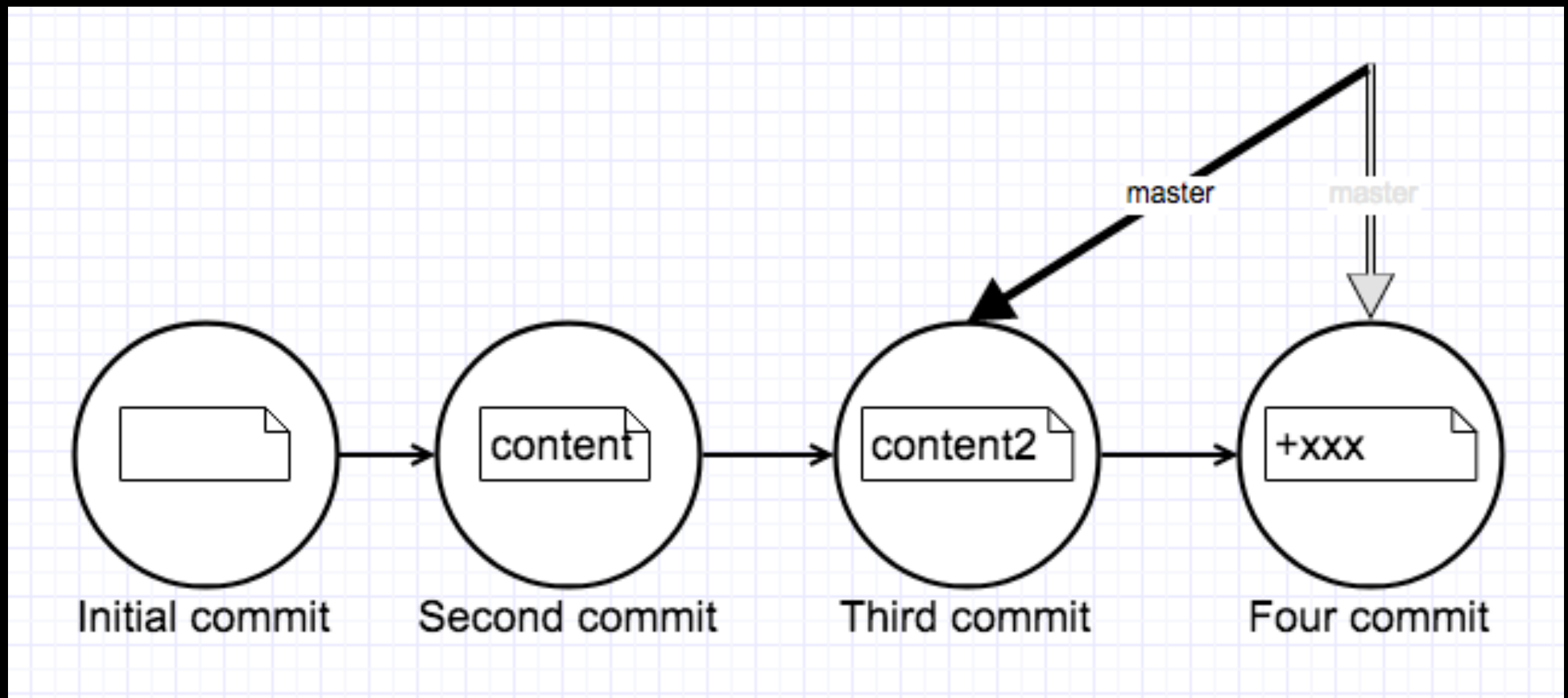
# git rebase origin master



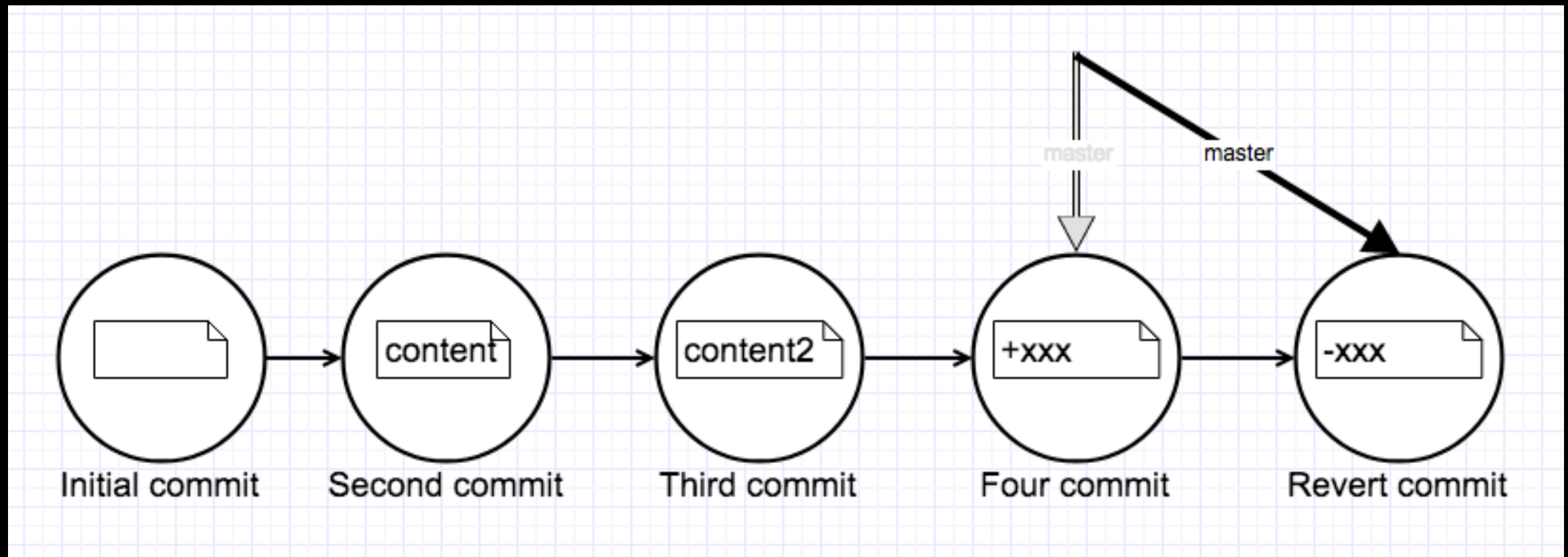
# 撤销



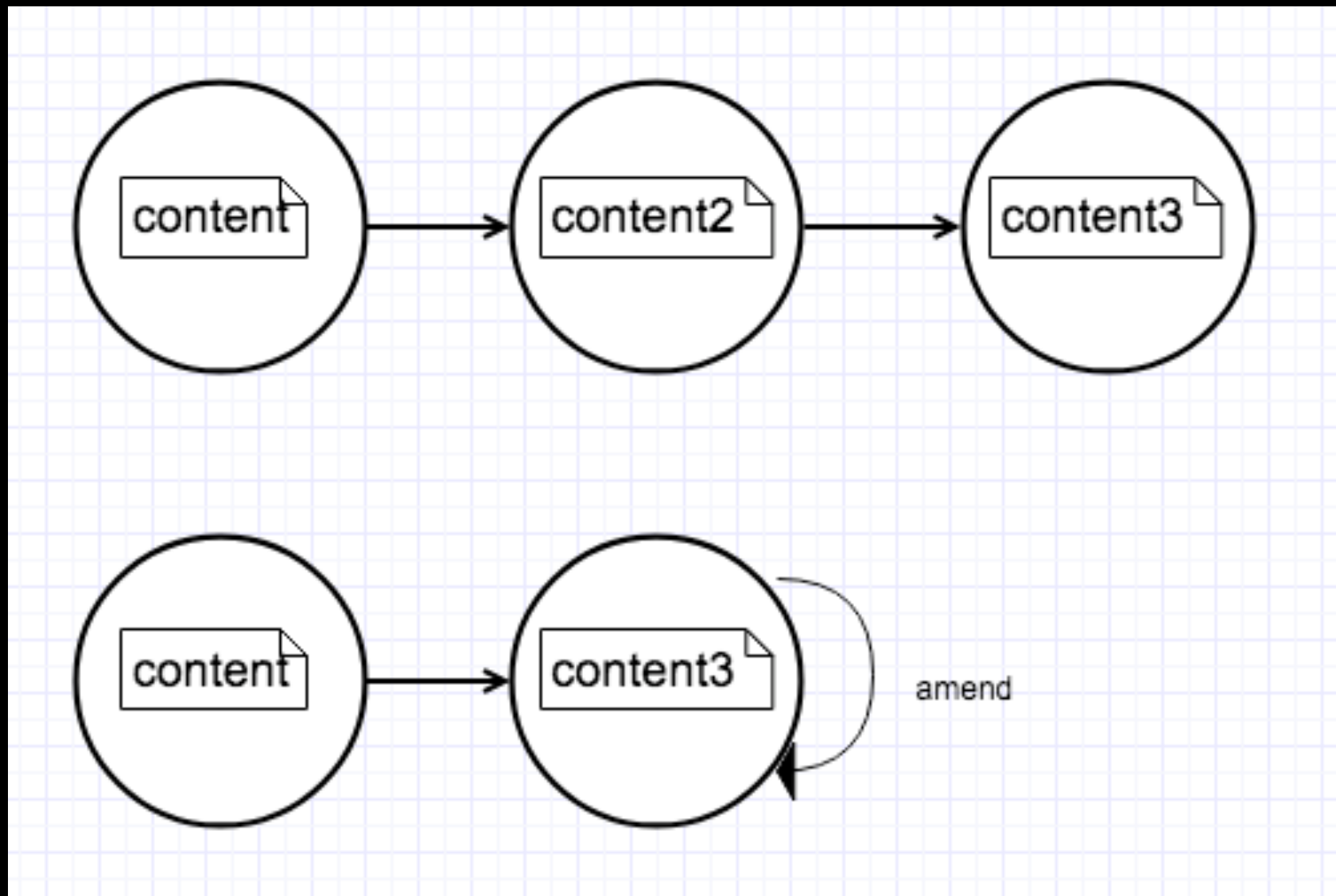
# git reset --hard HEAD^



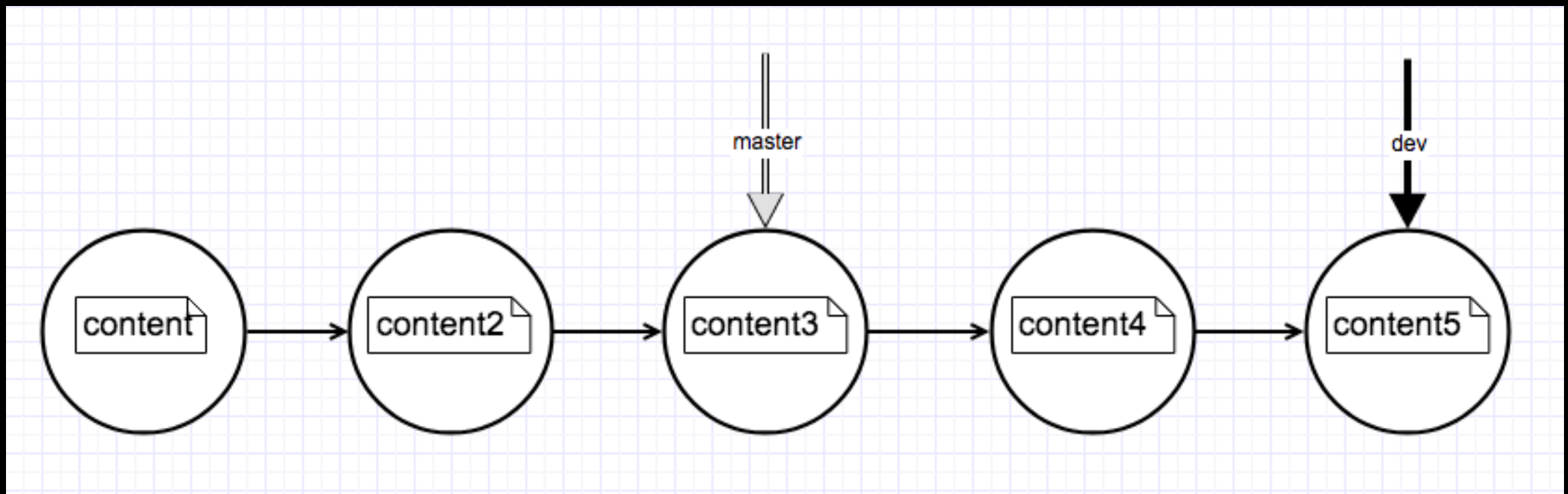
# git revert HEAD



# git commit --amend

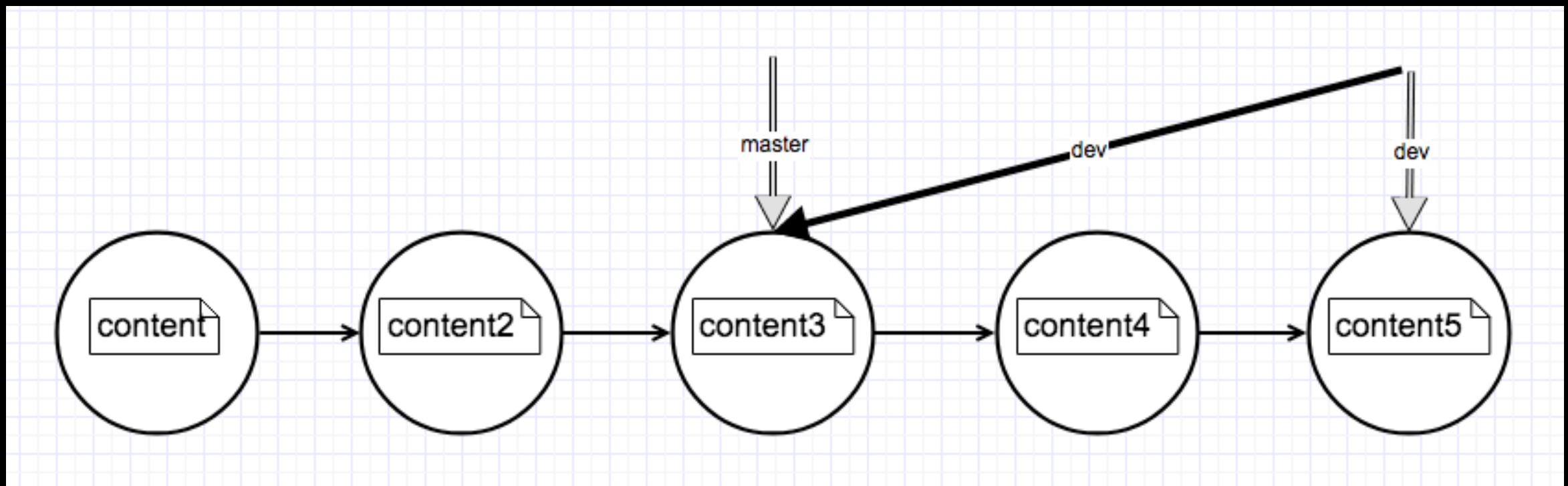


# 压缩 commit

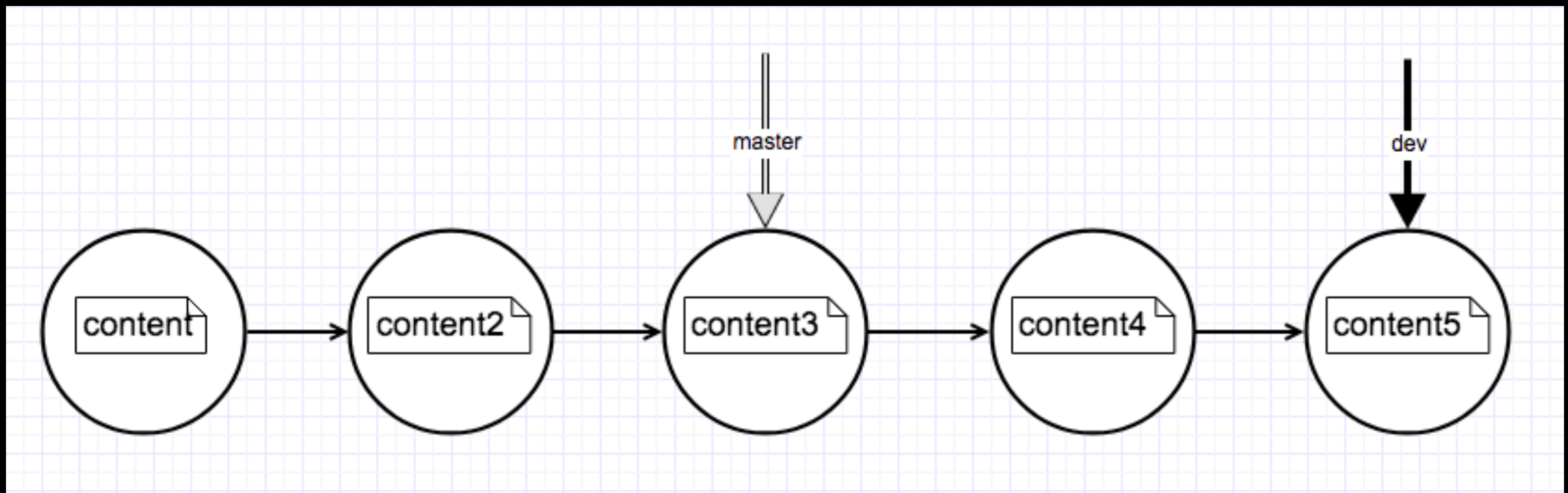




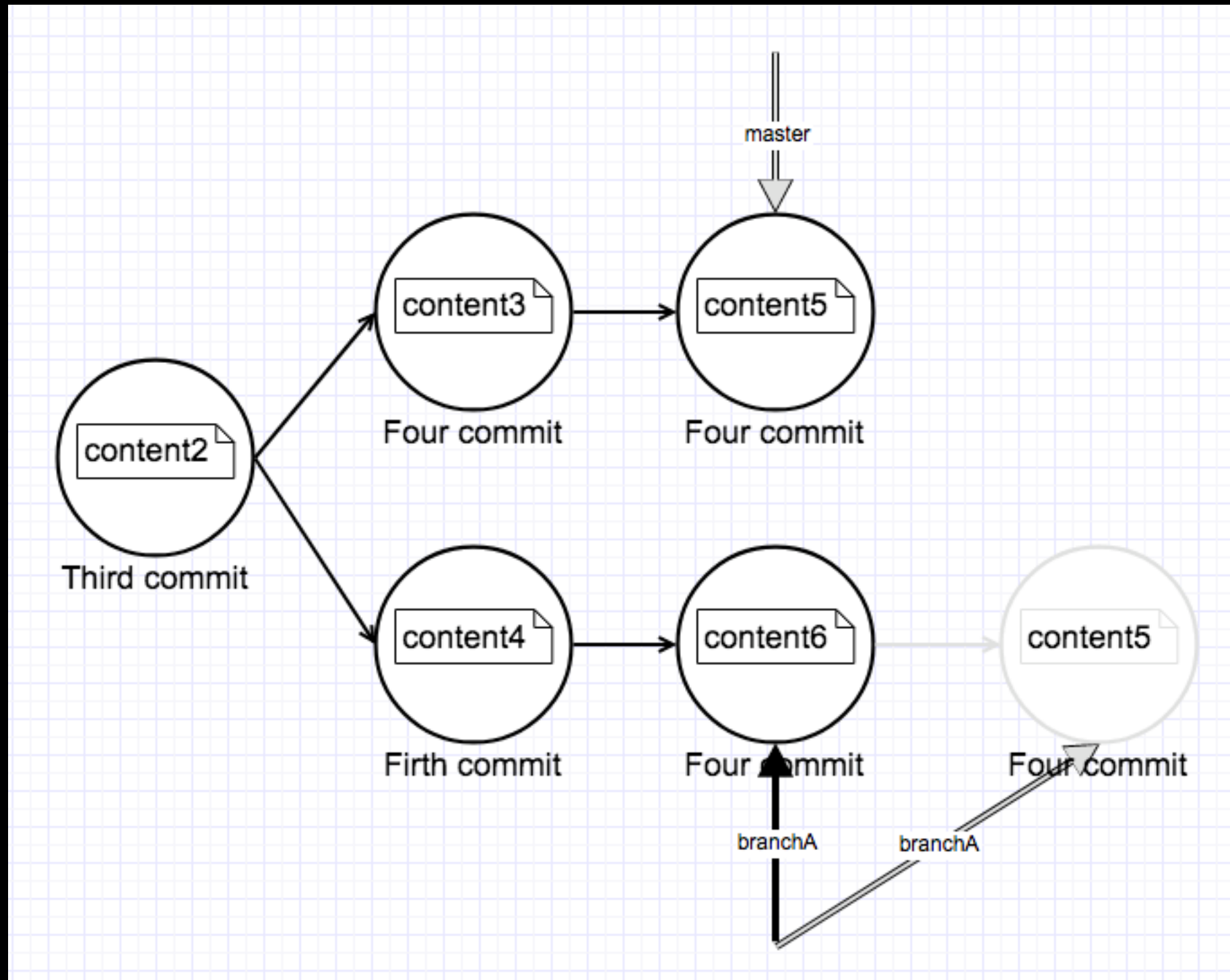
# git reset --soft HEAD^^



# git rebase -i HEAD^^^



# git cherry-pick xxx



Q & A

# Thanks