



Shu Ha Ri



We Love Bug

# Source Code Management with



KBTG - Basic Java

November 2020, 24



<http://shorturl.at/npE07>

bomb0069 / workshop-basic-day-05

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main · workshop-basic-day-05 / slide / SCK-SCM-GIT.pdf Go to file ...

bomb0069 [Added]: Slide for git scm Latest commit 2f52f46 29 seconds ago History

By 1 contributor

32.5 MB Download

The slide features a large orange background. On the left, there's a logo with Khmer text 'សាស្ត្រកម្មវិធី' and English text 'SIEM REAP CHAMONNOM KIT'. Below it is a stylized white 'H' logo with the text 'Shu Ha Ri' underneath. On the right, the text 'Source Code Management with git' is displayed in large white font, with a 'git' icon consisting of a diamond shape and three red dots below it.



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



Shu Ha Ri



SEAL



We Love Bug

## Agenda

- Version Control System
- Introduction to Git
- Working with Git - Local
- Working with Branching
- Working on Remote Repository
- Git for Good



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# Opening



**Share** — copy and redistribute the material in any medium or format.  
**Adapt** — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).



## Karan (bomb) Sivarat

**Agile Practitioner and Technical Journey  
at Siam Chamnankit Co., Ltd  
at Shu Ha Ri Co., Ltd.**

### **Trainer**

**of 3 Days for Agile Development Workshop  
of 2 Days for Agile Project Management Workshop  
of 2 Days for Source Code Management with Git**

Blogger @ [bomb0069.wordpress.com](http://bomb0069.wordpress.com)

Blogger @ [medium.com/@bomb0069](https://medium.com/@bomb0069)

email: [karan@scrum123.com](mailto:karan@scrum123.com)

twitter: [@bomb0069](https://twitter.com/bomb0069)

facebook: [facebook.com/bomb0069](https://facebook.com/bomb0069)



# Version Control System



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# What is Version Control System ?

## Why we need it ?



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# History - Version Control System

First

Second

Third

No-Networking

On File @ Time

Lock for Concurrent

**ex.**

RCS, SCCS

Centralised

Multi-Files

Merge before Commit

**ex.**

CVS, SourceSafe,

Subversion

Distributed

Change Set

Commit before Merge

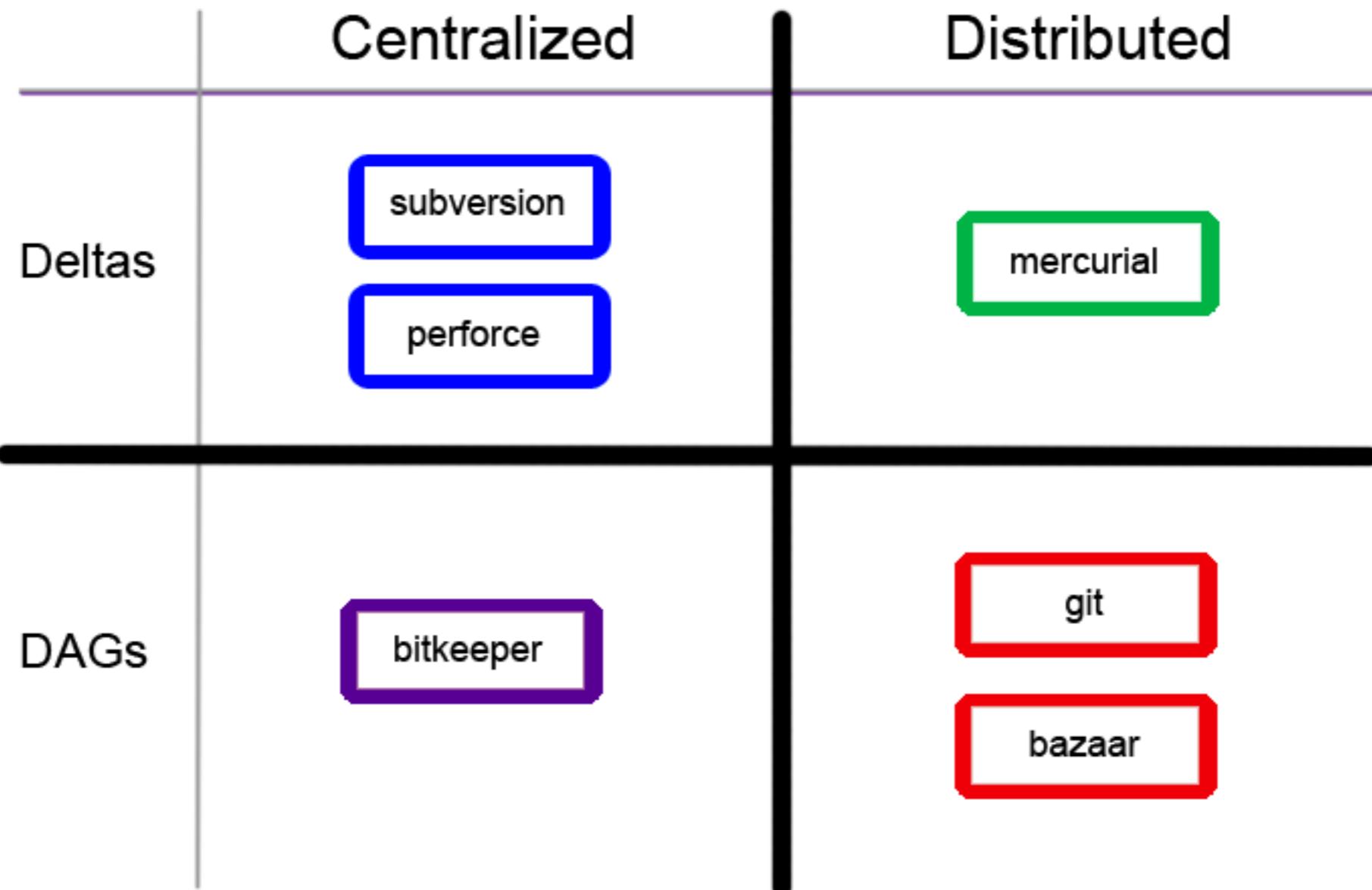
**ex.**

Git, Bazaar,

Mercurial



# Version Control



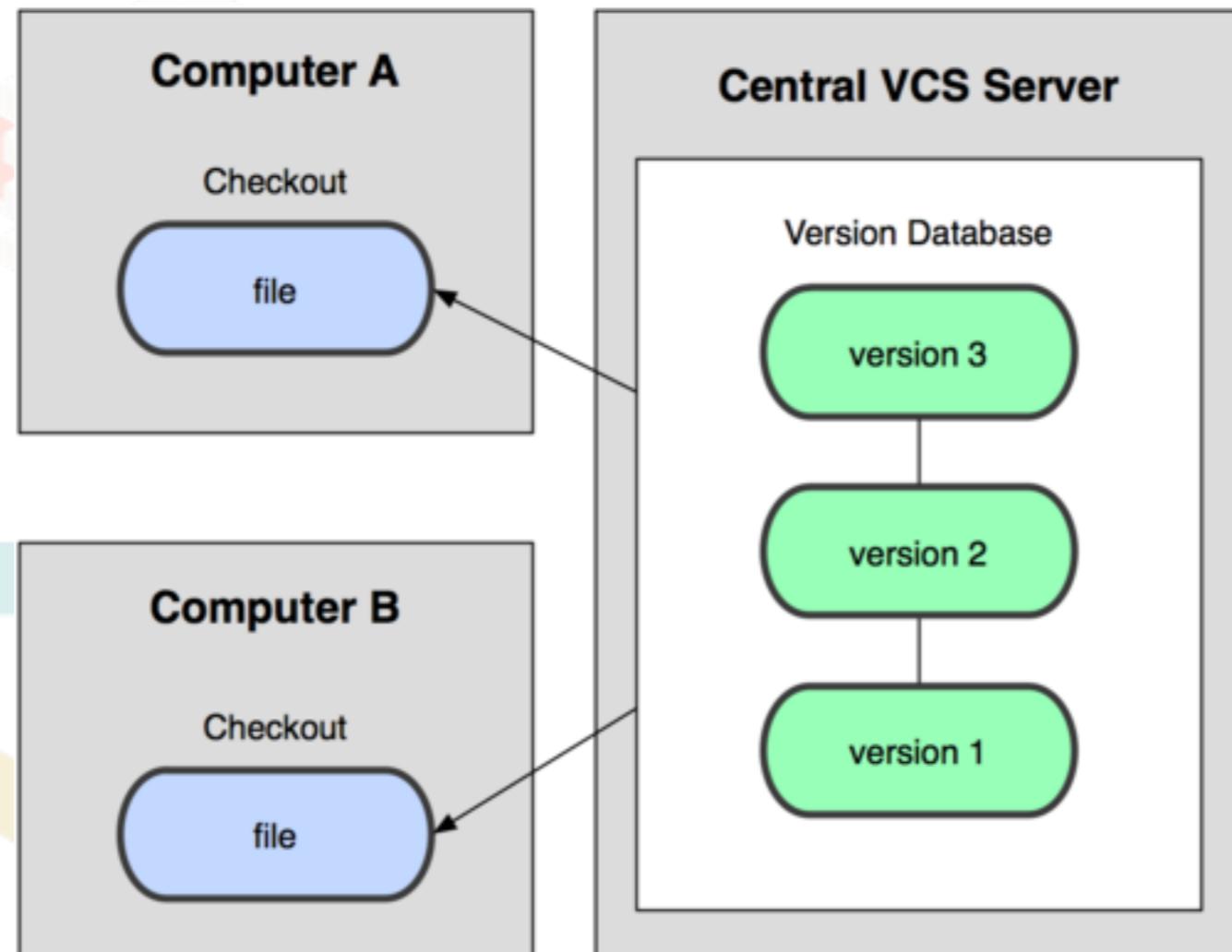
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

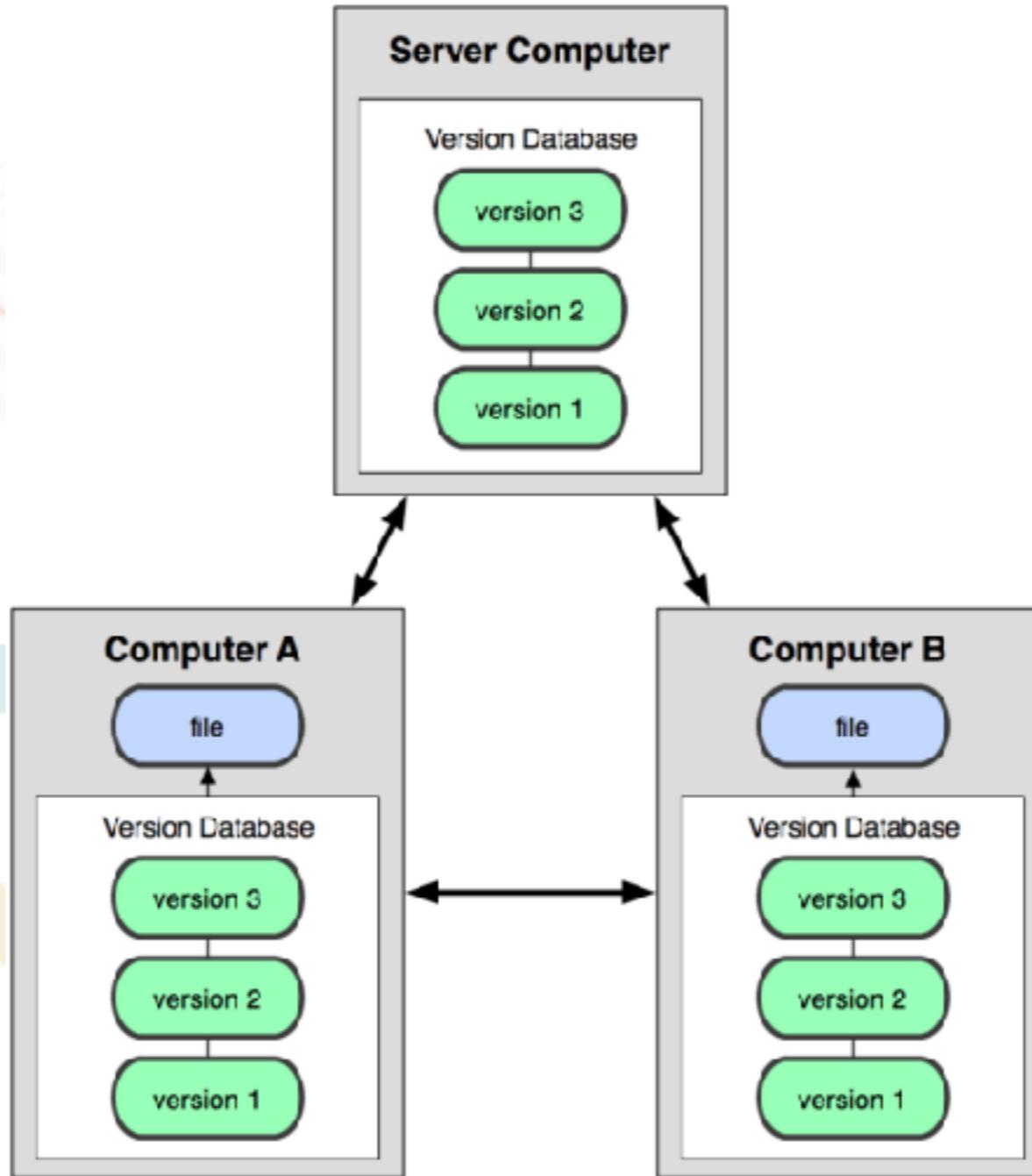
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Centralised Version Control



**Share** — copy and redistribute the material in any medium or format.  
**Adapt** — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Distributed Version Control



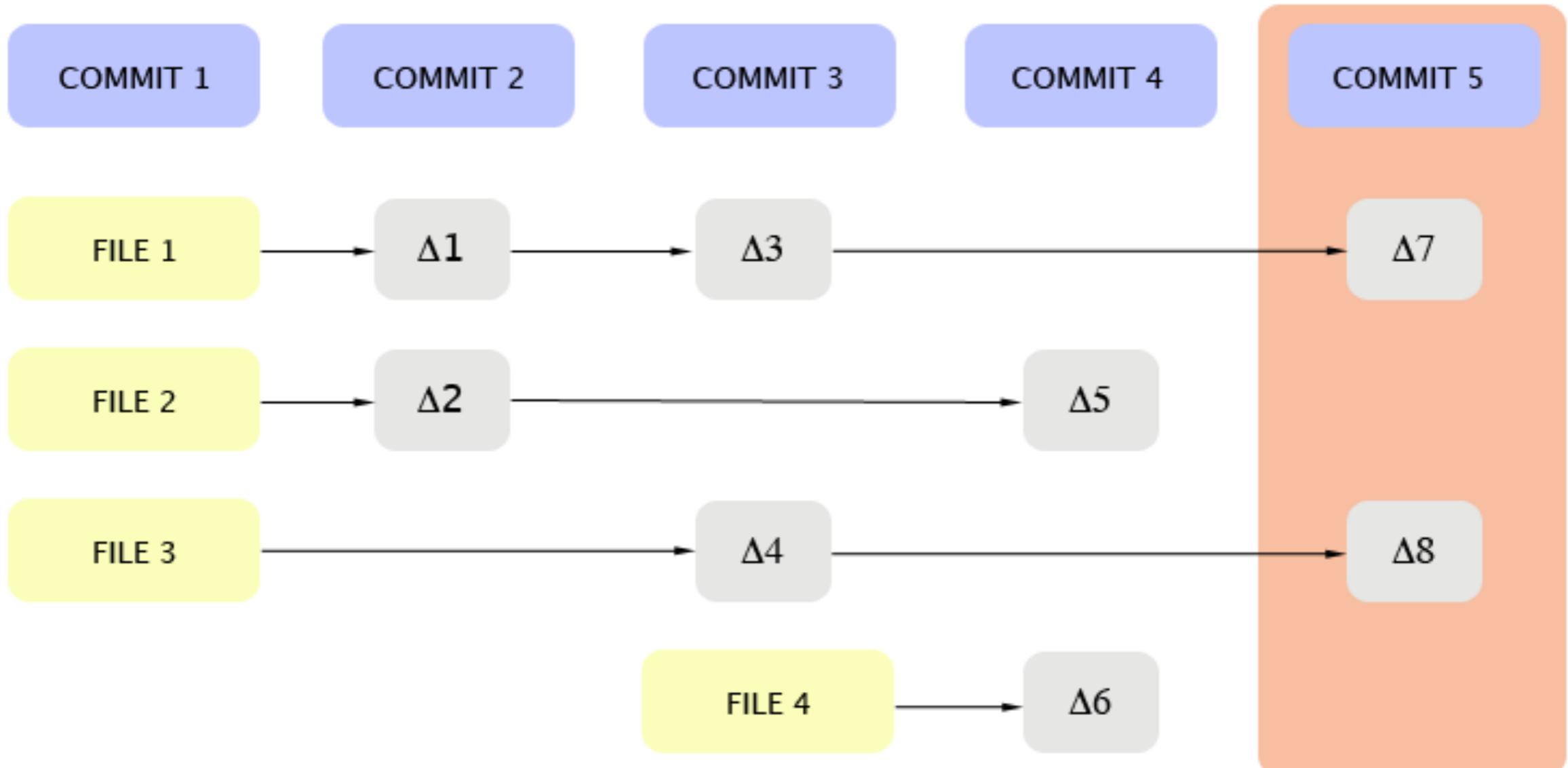
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Delta Repository



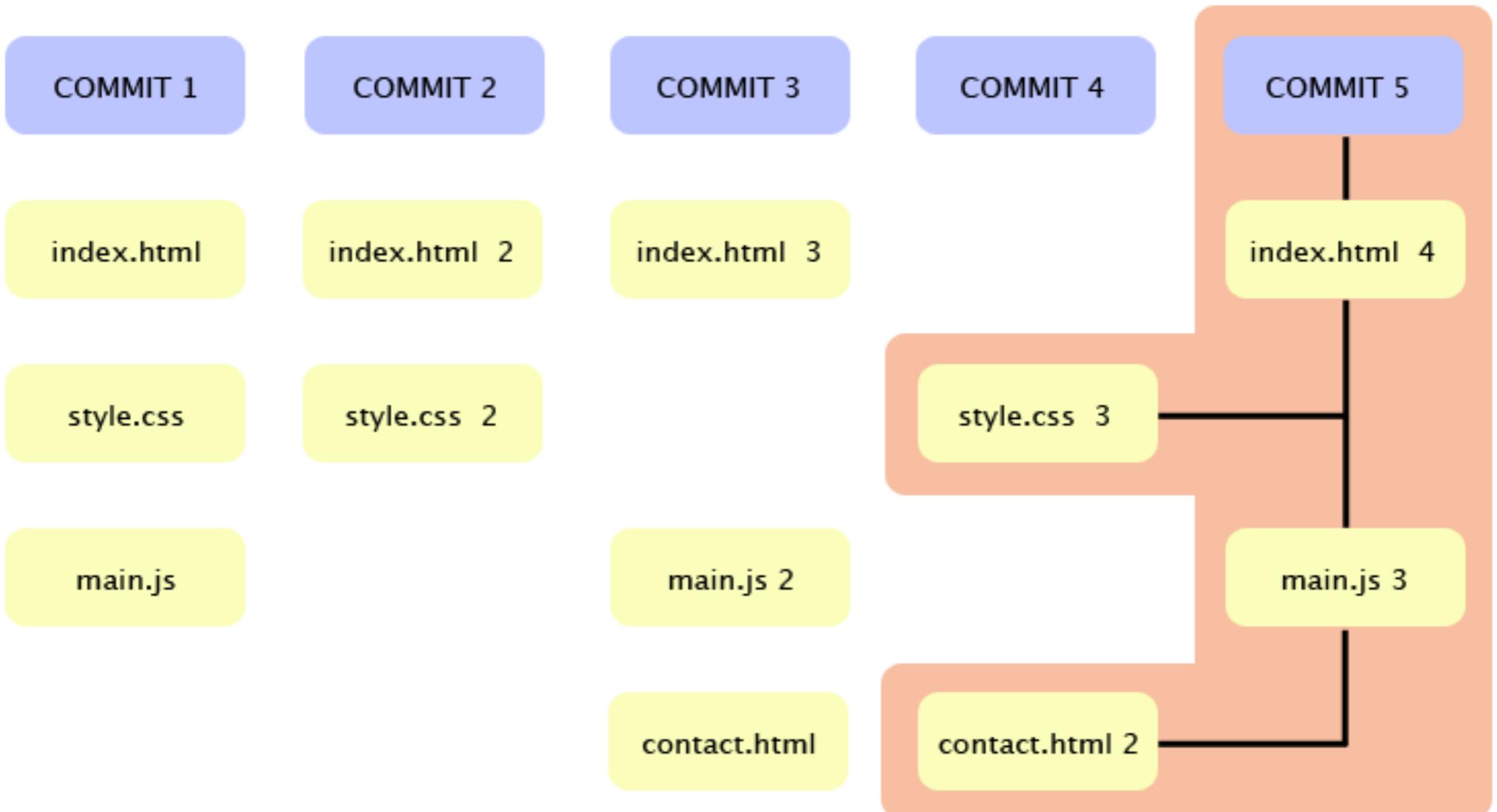
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# DAG Repository



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

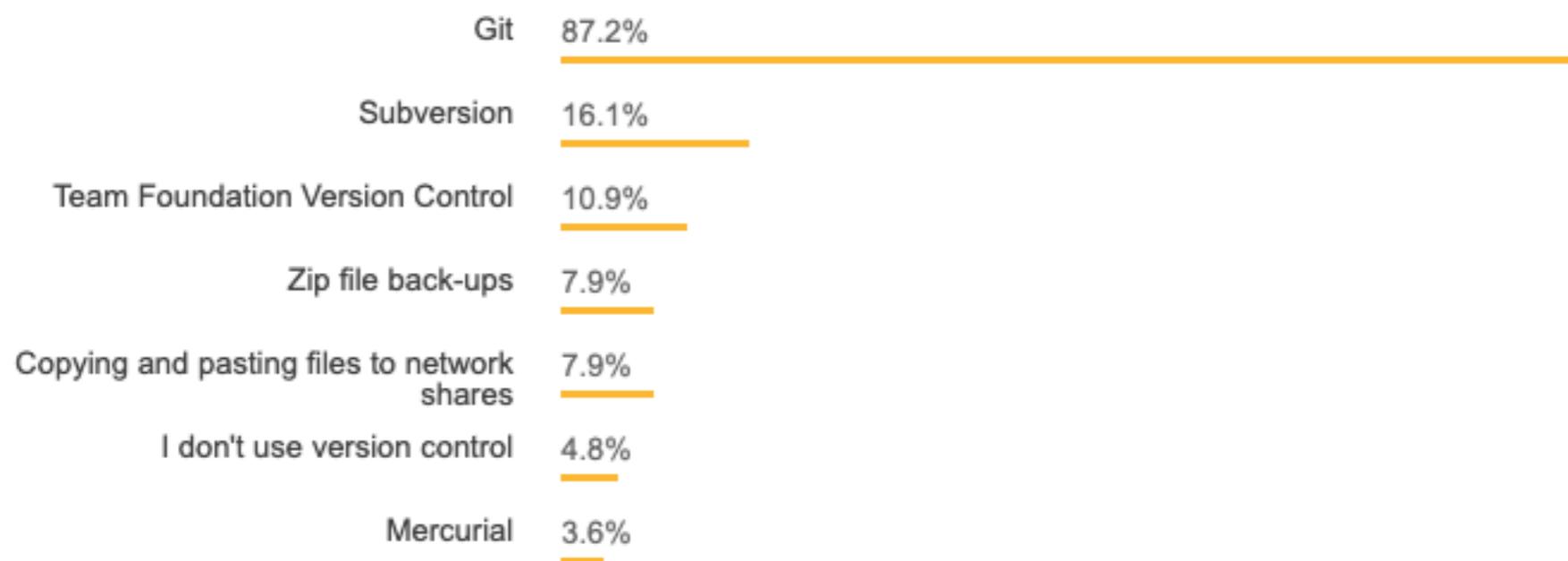
**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Stackoverflow : 2018 Subway – Version Control

## Version Control

All Respondents Professional Developers



74,298 responses; select all that apply

Git is the dominant choice for version control for developers today, with almost 90% of developers checking in their code via Git.

Source: <https://web.archive.org/web/20190530142357/https://insights.stackoverflow.com/survey/2018/#work--version-control>



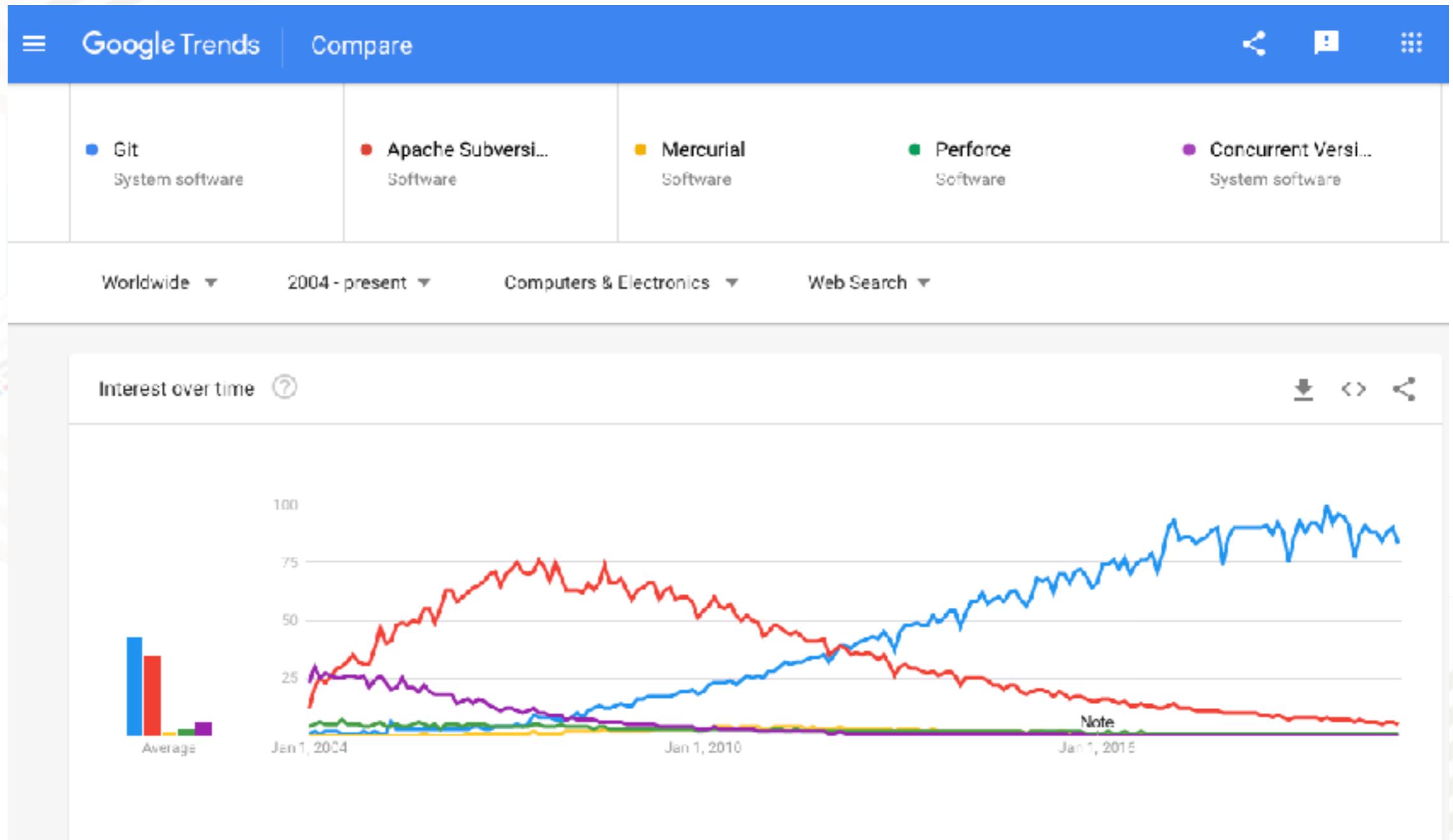
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Google Trend – Version Control System



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Introduction to Git



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# Git History

- o Linux submitting patches model is quite interesting & unique in that time
- o Linus hates CVS so much, he never believe in it
- o He was in doomed maintaining Linux Kernel using Tarball + patch
- o BitMover offer BitKeeper free usage. BitKeeper saved his life, but just for a while
- o BitKeeper has a very strict Software License. People start to get upset
- o Politics happened. Linus needs a replacement for BitKeeper
- o Key criteria are:
  - o Needs to be Distributed SCM, like BitKeeper
  - o Needs to be highly Performance. 3 seconds max to apply a patch
  - o Guaranteed what goes in comes out correctly.



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# GOALS of git

Speed

Simple design

Support for many parallel branches

Fully distributed

To handle large project like Linux kernel



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Who use Git

Google

facebook

Microsoft

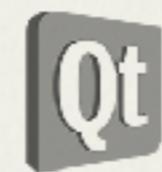
twitter

LinkedIn

NETFLIX



PostgreSQL



## COMPANIES

6227 companies reportedly use Git in their tech stacks, including Netflix, Shopify, and reddit.



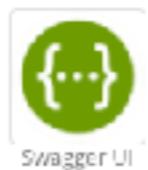
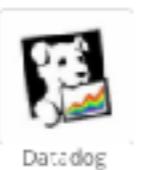
## DEVELOPERS

7188 developers on StackShare have stated that they use Git.



## Git Integrations

Bitbucket, SourceTree, Datadog, Azure DevOps, and Swagger UI are some of the popular tools that integrate with Git. Here's a list of all 128 tools that integrate with Git.



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Practice Git on Browser

<https://try.github.io>

Resources to learn Git

### Learn by reading

- Git Handbook**  
Git, GitHub, DVCS, oh my! Learn all the lingo and the basics of Git.
- Cheat Sheets**  
Keep these handy! Reference sheets covering Git commands, features, SVN migrations, and bash. Available in multiple languages.

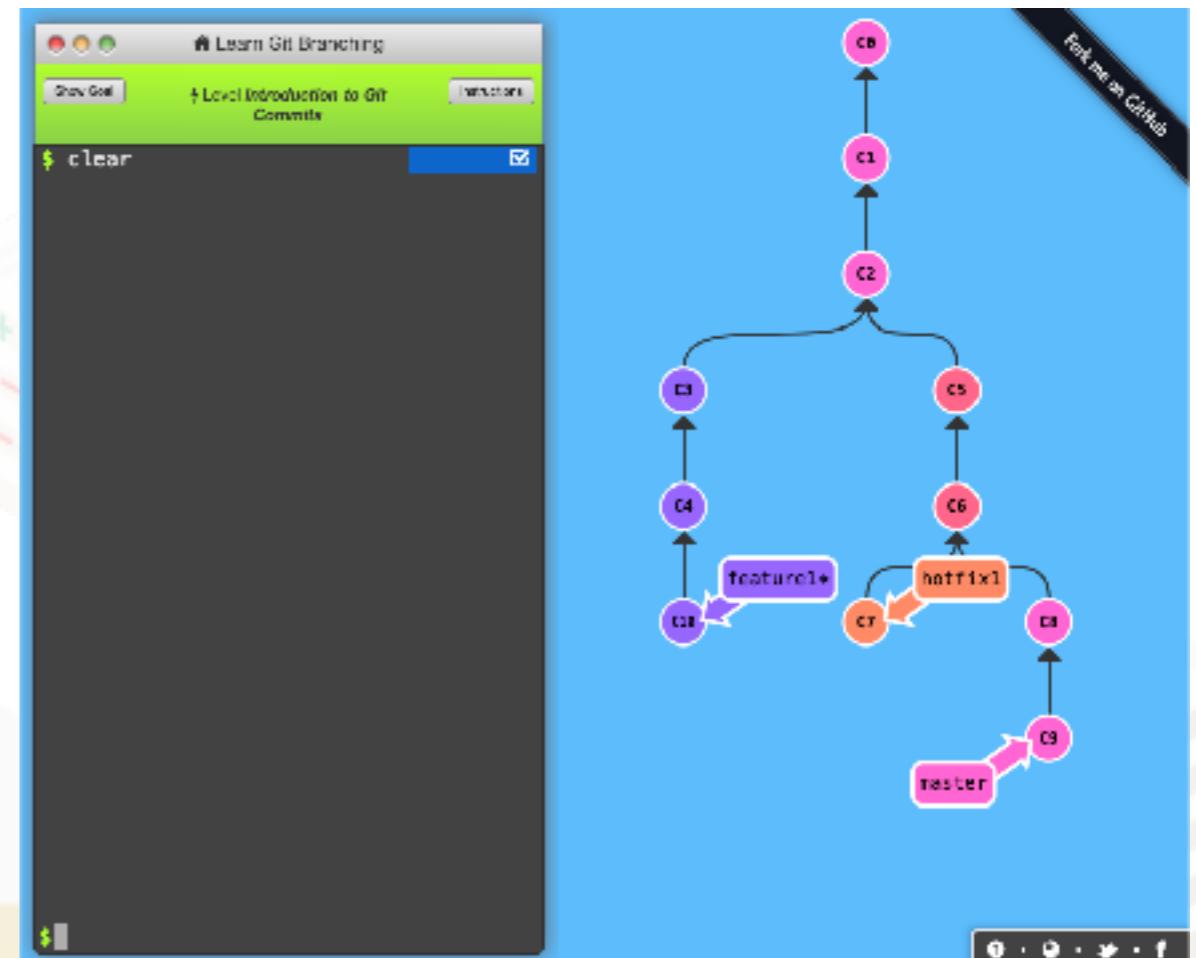
### Learn by doing

- Learn Git branching**  
Try Git commands right from your web browser. Featuring some of your soon-to-be favorites: branch, add, commit, merge, revert, cherry-pick, rebase!
- Visualizing Git**  
Look under the hood! Explore how Git commands affect the structure of a repository within your web browser with a free explore mode, and some constructed scenarios.
- Git-It**  
You've downloaded Git, now what? Download Git-It to your machine and you'll get a hands-on tutorial that teaches you to use Git right from your local environment, using commands on real repositories.

### Looking for GitHub Training?

- GitHub Learning Lab**  
Get the skills you need without leaving GitHub. GitHub Learning Lab takes you through a series of fun and practical projects, sharing helpful feedback along the way.
- Professional training**  
Whether you're just getting started or you use GitHub every day, the GitHub Professional Services Team can provide you with the skills your organization needs to work smarter.

<https://learngitbranching.js.org/>



Share — copy and redistribute the material in any medium or format.

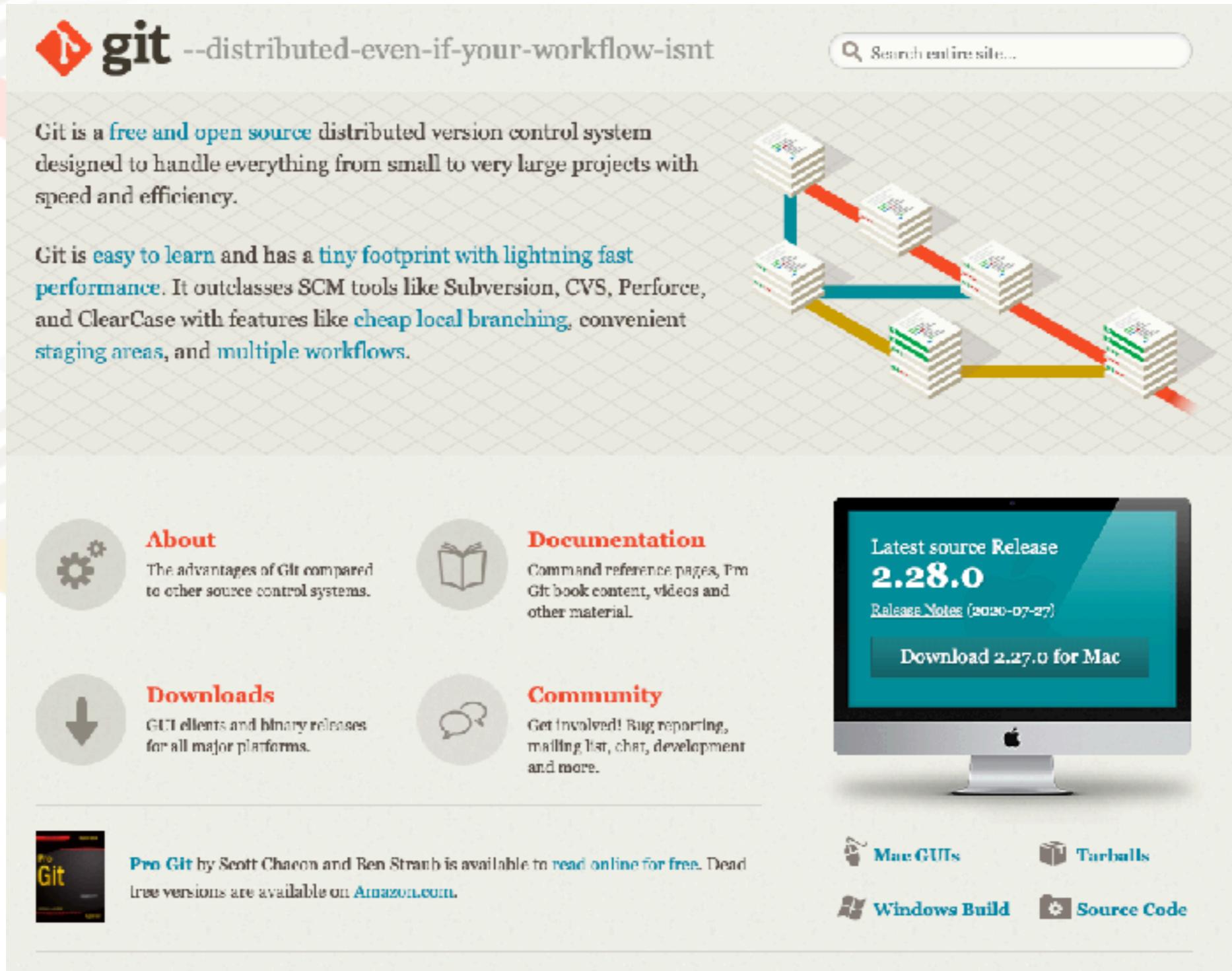
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Install Git

<http://git-scm.com/>



The screenshot shows the official website for Git (<http://git-scm.com/>). At the top left is the Git logo with the word "git" in a bold, lowercase sans-serif font. To its right is the tagline "--distributed-even-if-your-workflow-isn't". A search bar with the placeholder "Search entire site..." is positioned at the top right. Below the header, there are two main sections of text. The first section describes Git as a "free and open source distributed version control system" designed for efficiency. The second section highlights its ease of learning, fast performance, and features like local branching, staging areas, and multiple workflows. To the right of the text is a 3D-style diagram of several white server racks connected by red lines on a grid background, illustrating the distributed nature of Git. The main content area below the header is divided into several sections: "About" (with a gear icon), "Documentation" (with a book icon), "Downloads" (with a download arrow icon), and "Community" (with a speech bubble icon). On the right side, there's a large image of a Mac computer monitor displaying a teal-themed interface for the latest source release (2.28.0) with a "Download 2.27.0 for Mac" button. Below the monitor are links for "Mac GUIs", "Turbos", "Windows Build", and "Source Code". At the bottom left, there are four small decorative icons: a stylized eagle, a vertical bar with horizontal lines, a five-pointed star, and a building with numbers 25 and 62. The footer contains a Creative Commons license notice: "Share — copy and redistribute the material in any medium or format. Adapt — remix, transform, and build upon the material. NonCommercial — You may not use the material for commercial purposes. This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License."



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Configuration Git on Local Machine

```
$git config --global user.name "your name"
```

```
$git config --global user.email "your email"
```

```
$git config -l
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Create a repository

Windows

Select git bash

Mac or Linux

Open Terminal

\$mkdir workspace

\$git init



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Repository structure

\$tree

```
.git
├── HEAD
├── branches
├── config
├── description
└── hooks
    ├── applypatch-msg.sample
    ├── commit-msg.sample
    ├── post-update.sample
    ├── pre-applypatch.sample
    ├── pre-commit.sample
    ├── pre-push.sample
    ├── pre-rebase.sample
    ├── prepare-commit-msg.sample
    └── update.sample
├── info
│   └── exclude
└── objects
    ├── info
    └── pack
└── refs
    ├── heads
    └── tags
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Working with Git - Local



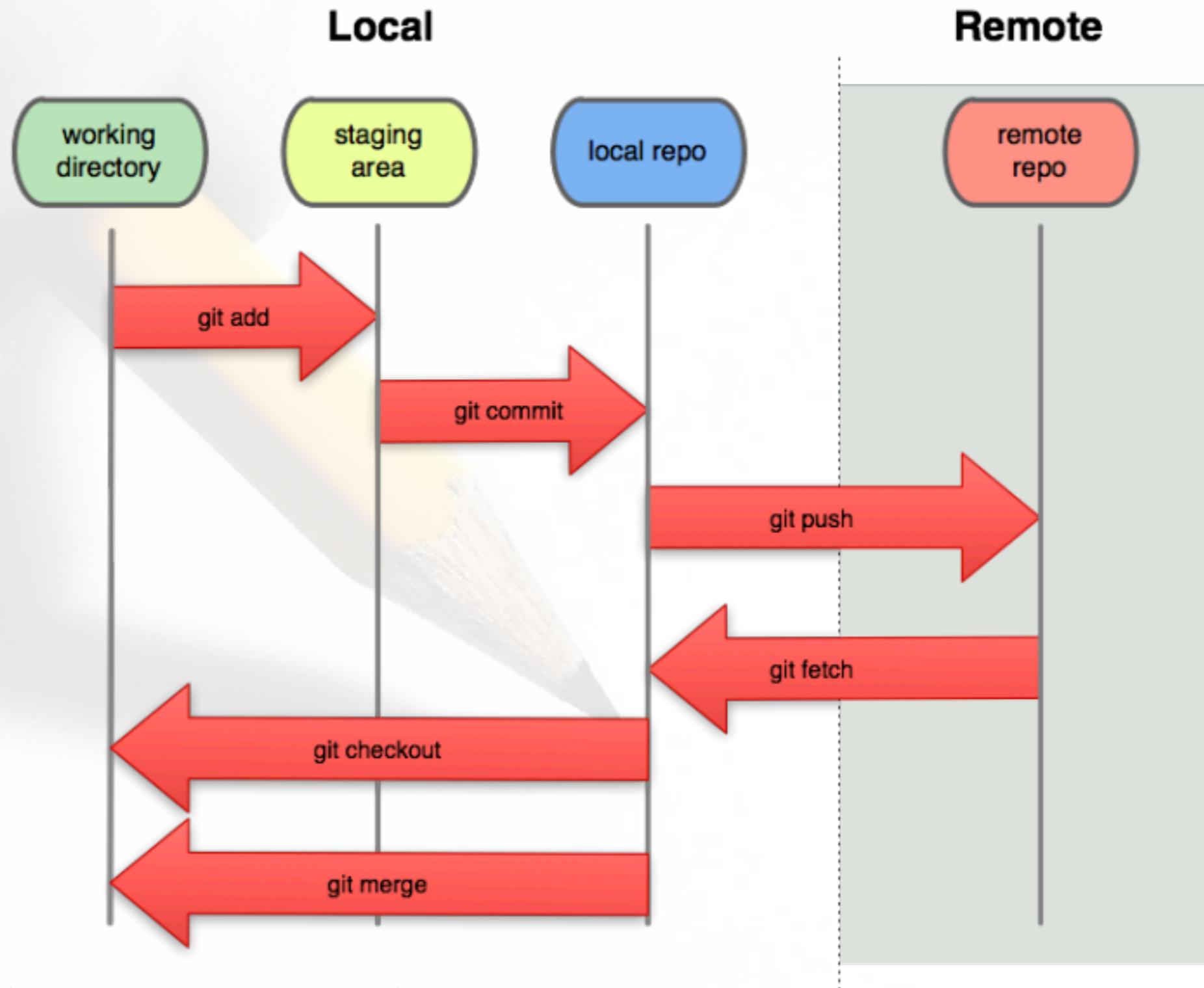
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# Git Workflow



Source: <https://greenido.wordpress.com/2013/07/22/git-101-useful-commands/>



**Share** — copy and redistribute the material in any medium or format.

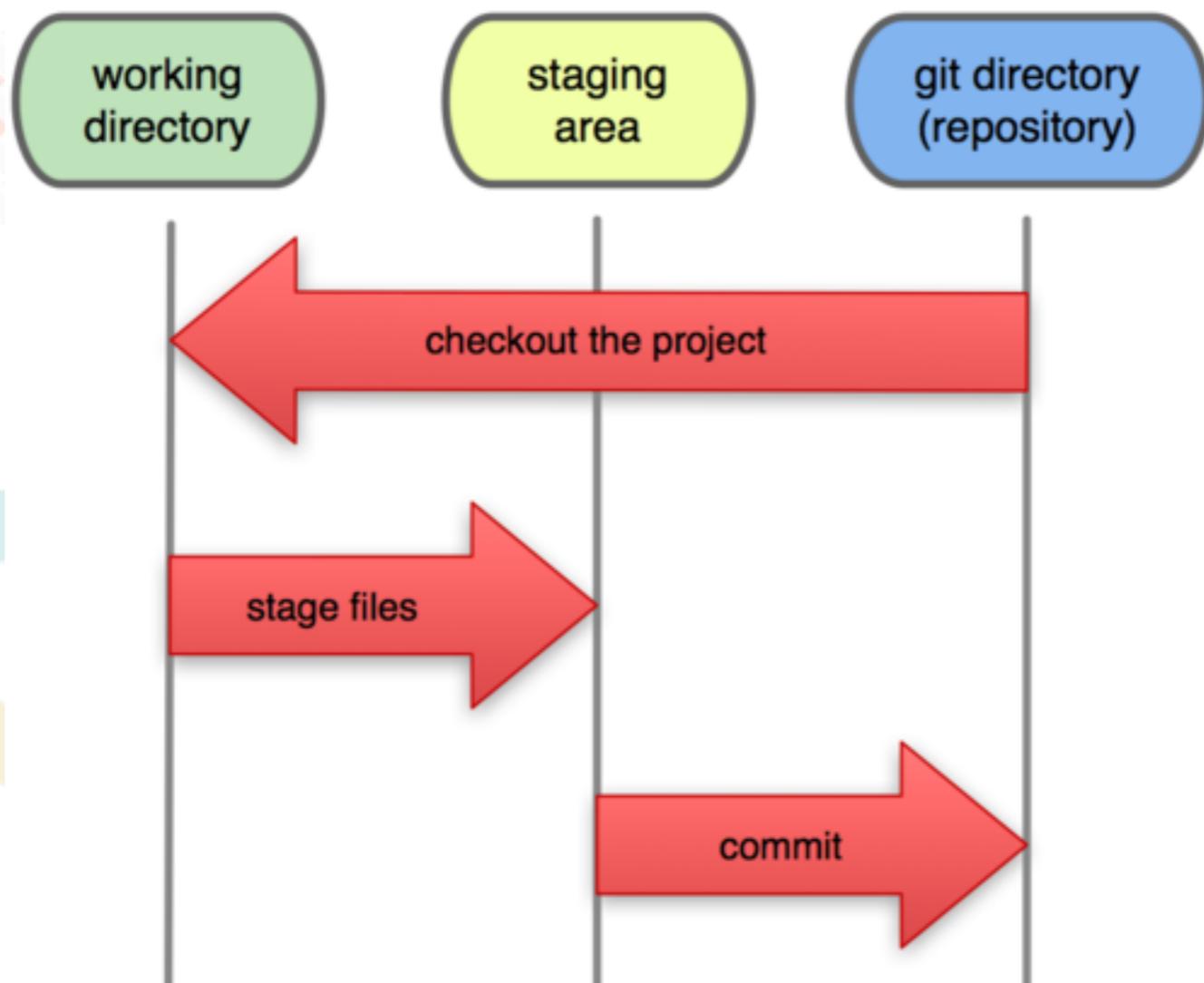
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Git Workflow

## Local Operations



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Checking the status

\$touch *README*

\$git **status**

```
On branch master  
Initial commit  
  
Untracked files:  
(use "git add <file>..." to include in what will be committed)  
    README  
  
nothing added to commit but untracked files present (use "git add" to track)
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status

1. `$touch README`

Untracked

Unmodified

Modified

Staged

Add the file

Edit the file

Stage the file

Remove the file

Commit



README



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Checking the status

```
$git add README
```

```
$git status
```

```
On branch master  
Initial commit  
  
Changes to be committed:  
(use "git rm --cached <file>..." to unstage)  
  
      new file:   README
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status

1. `$touch README`

Untracked

Unmodified

Modified

Staged

Add the file

2. `$git add README`

Remove the file

Edit the file

Stage the file

Commit



README



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# View staged and unstaged change

```
$echo "test" > README
```

```
$git status
```

```
On branch master  
No commits yet  
  
Changes to be committed:  
(use "git rm --cached <file>..." to unstage)  
  new file: README.md  
  
Changes not staged for commit:  
(use "git add <file>..." to update what will be committed)  
(use "git restore <file>..." to discard changes in working directory)  
  modified: README.md
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status

1. `$touch README`

Untracked

Unmodified

Modified

Staged

Add the file

2. `$git add README`

3. `$echo "test" > README`

Edit the file

Remove the file

Stage the file

Commit



README  
"TEST"



README  
""



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# See difference of file

\$git **diff**

```
diff --git a/README b/README
index e69de29..9daeafb 100644
--- a/README
+++ b/README
@@ -0,0 +1 @@
+test
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status

1. `$touch README`

Untracked

Unmodified

Modified

Staged

Add the file

2. `$git add README`

3. `$echo "test" > README`

Edit the file

Remove the file

Stage the file

Commit

`$git diff README`

README  
TEST

README  
" "



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Commit your changes

```
$git commit -m "Message for this change"
```

```
[master (root-commit) c928f65] Message for this change
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
```

```
$git status
```

```
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   README

no changes added to commit (use "git add" and/or "git commit -a")
```



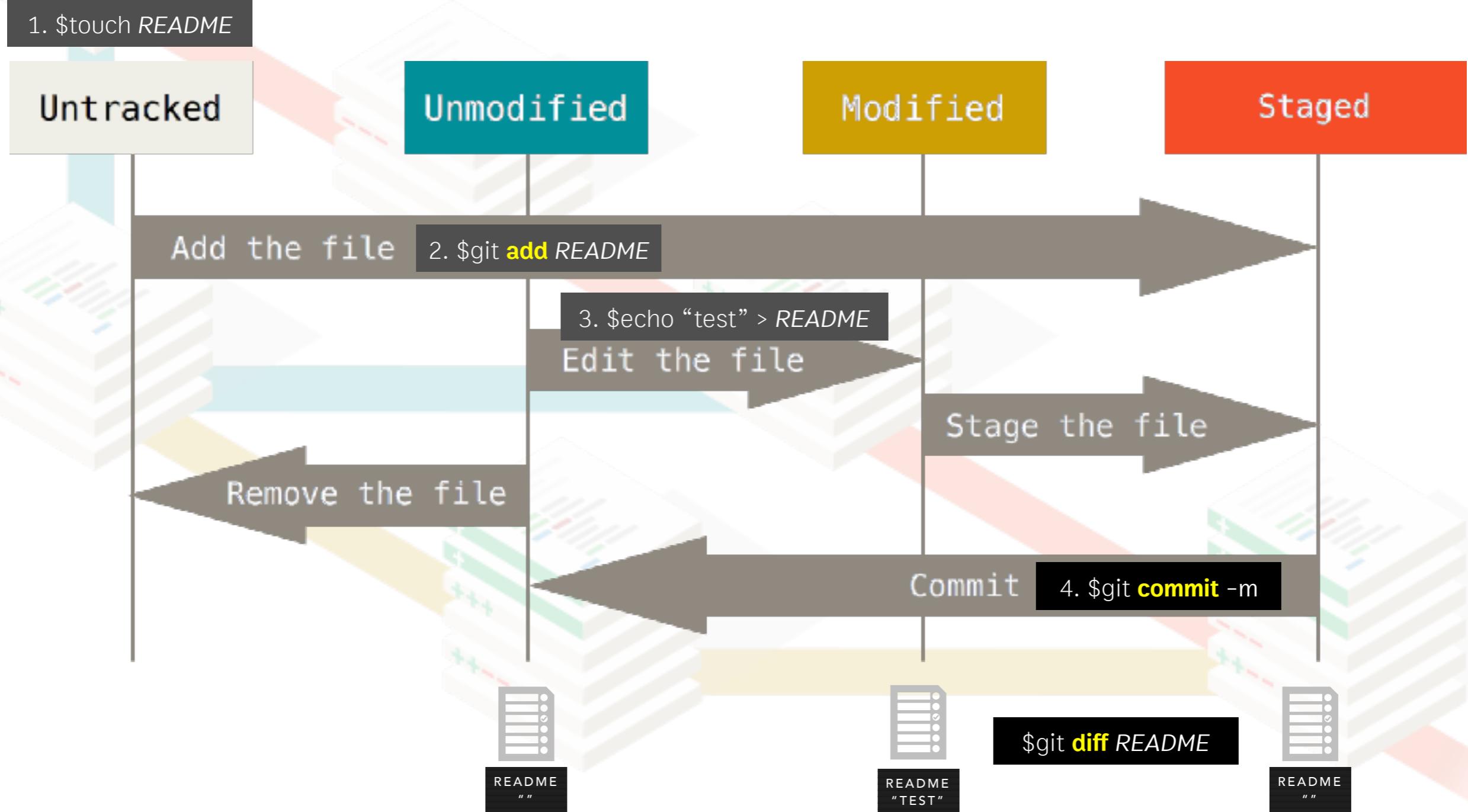
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Checking the status

```
$git add README
```

```
$git status
```

```
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   README
```



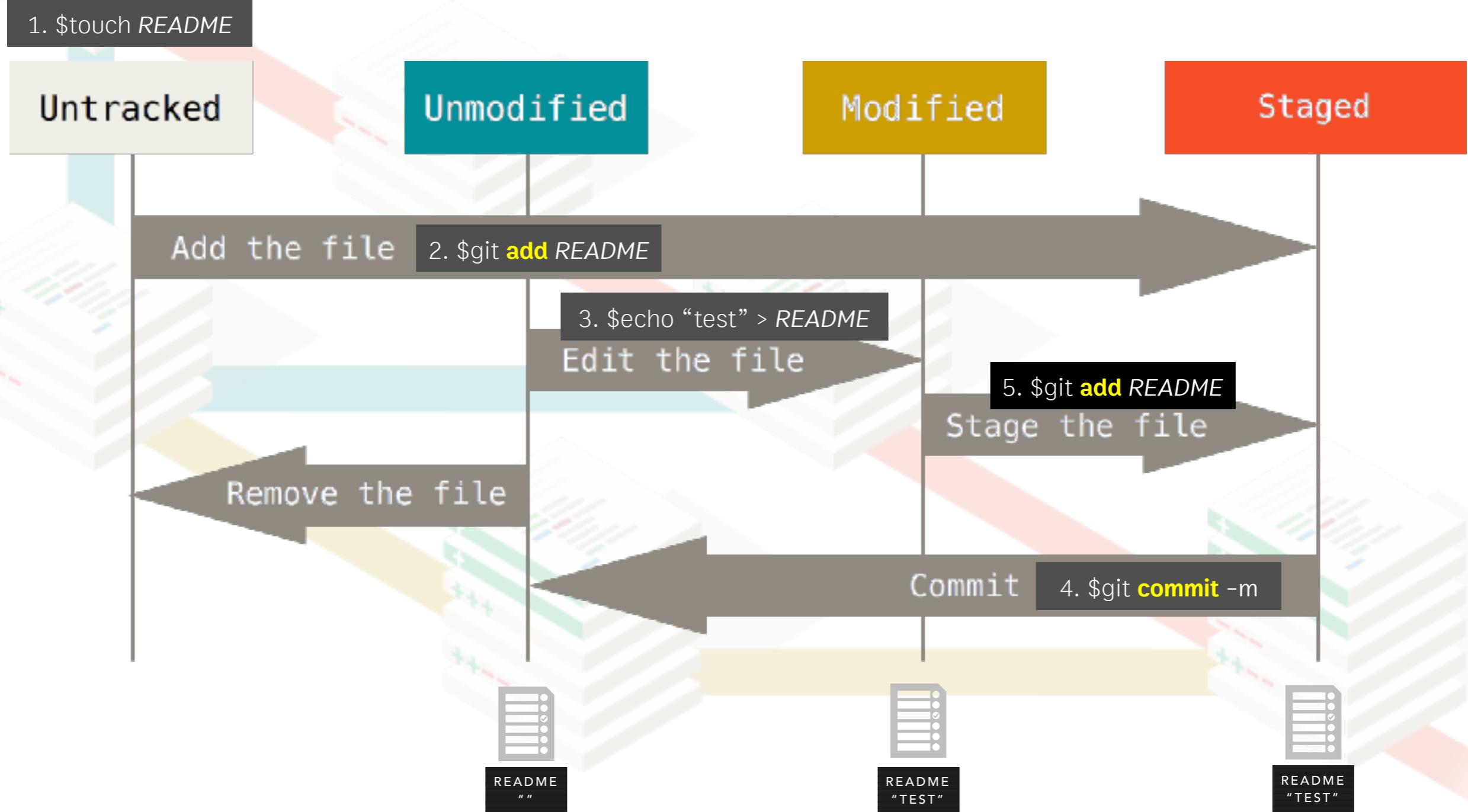
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# See difference of file

\$git **diff**



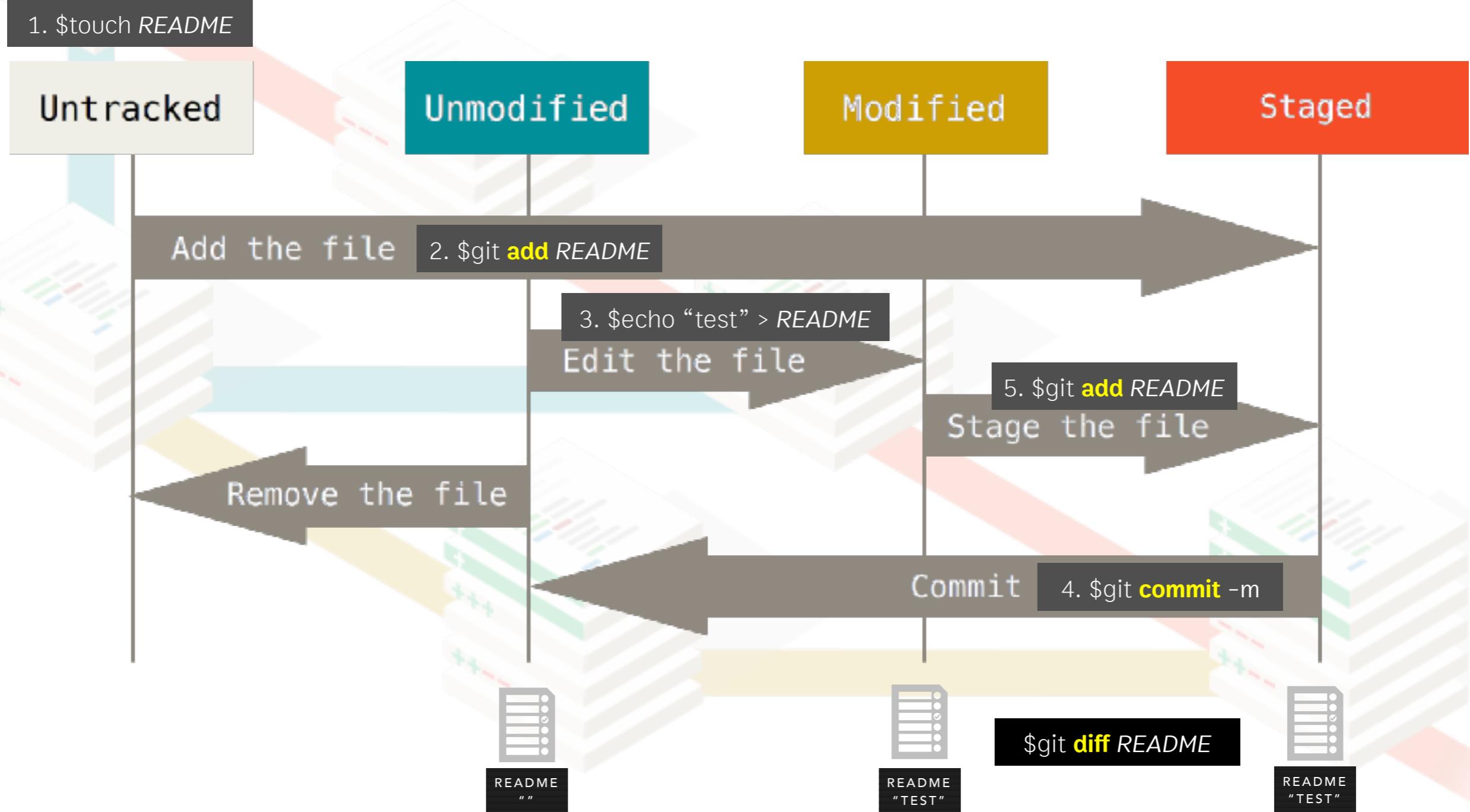
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# See difference of file with history

\$git **diff** --cached

\$git **diff** --staged

```
diff --git a/README b/README
index e69de29..9daeaaf 100644
--- a/README
+++ b/README
@@ -0,0 +1 @@
+test
```



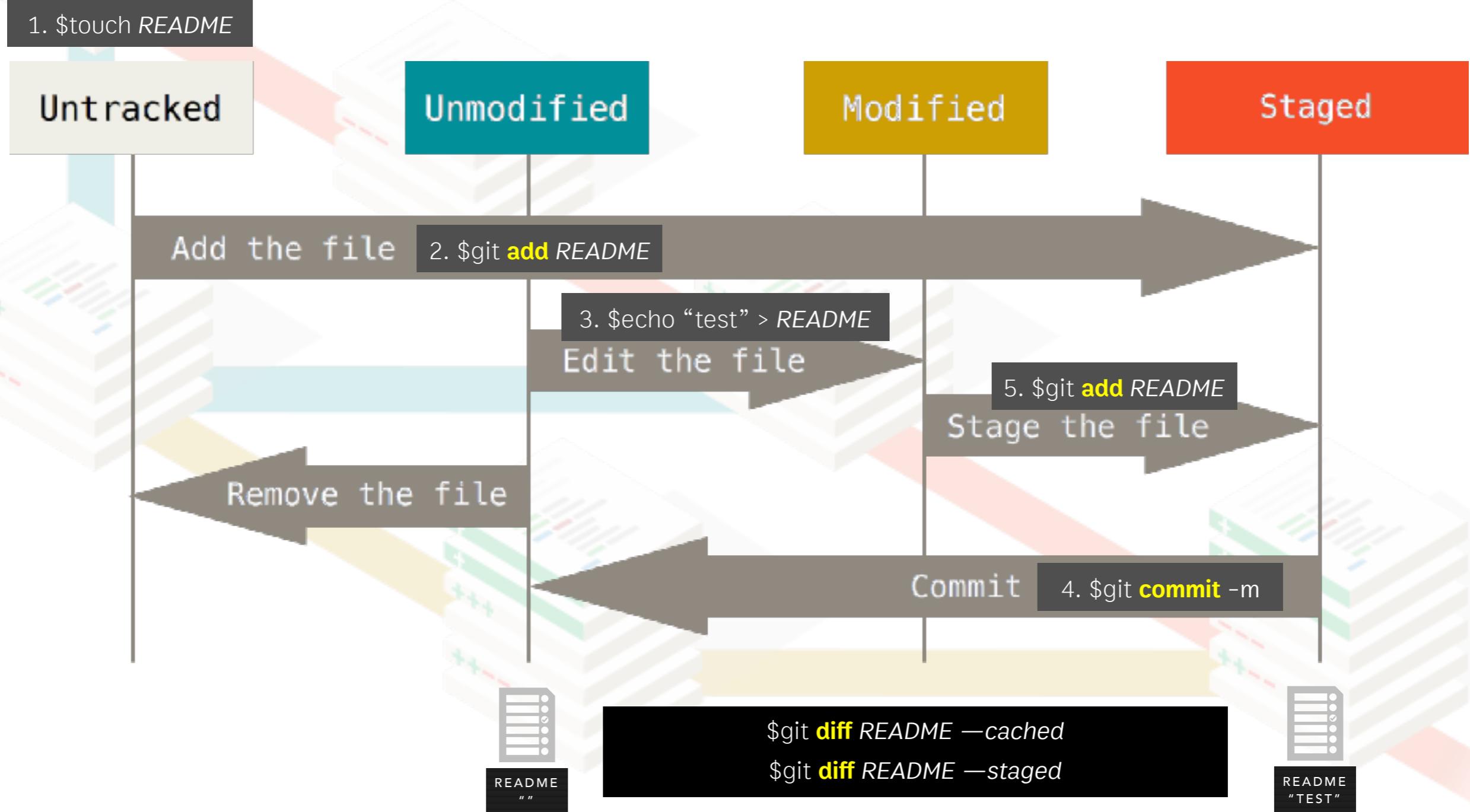
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Modify again ?

\$echo “testing” > README

\$git **status**

\$git **diff**



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status

1. `$touch README`

Untracked

Unmodified

Modified

Staged

Add the file

2. `$git add README`

3. `$echo "test" > README`

Edit the file

6. `$echo "testing" > README`

Remove the file

5. `$git add README`

Stage the file

Commit

4. `$git commit -m`

`$git diff README`



README  
"TEST"



README  
""



README  
"TESTING"

`$git diff README --cached`



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# ADD and Commit

Tips

```
$git commit -a -m "Message for this change"
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status

1. `$touch README`

Untracked

Unmodified

Modified

Staged

Add the file

2. `$git add README`

Remove the file

3. `$echo "test" > README`

Edit the file

6. `$echo "testing" > README`

5. `$git add README`

Stage the file

Commit

4. `$git commit -m`

7. `$git commit -a -m`

`$git diff README`

README  
"TESTING"

`$git diff README --cached`

README  
"TESTING"

README  
"TESTING"



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Checking the status

```
$git status
```

On branch master  
nothing to commit, working tree clean



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Remove your files

\$rm *README*

\$git **status**

```
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    deleted:    README

no changes added to commit (use "git add" and/or "git commit -a")
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Remove your files

```
$git rm README
```

```
$git status
```

```
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    deleted:    README
```



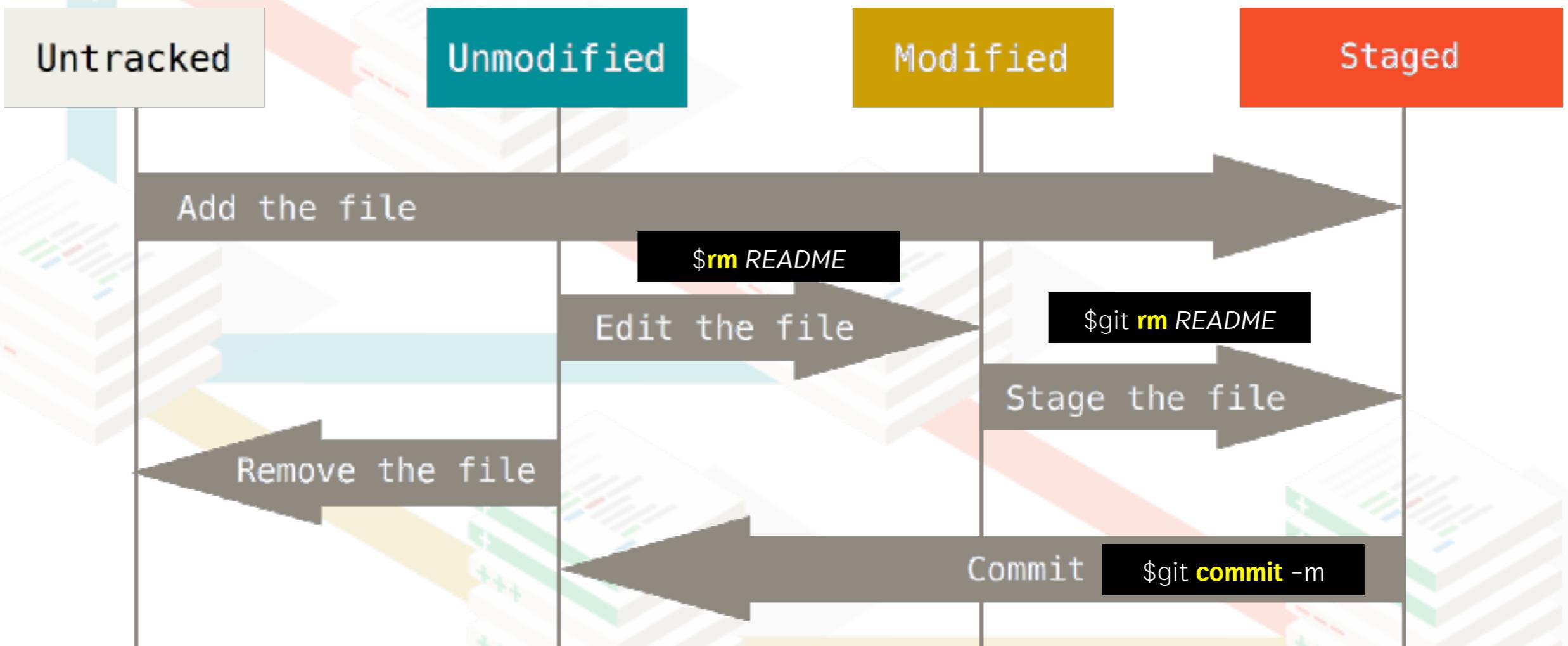
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Lift Cycle of File Status



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Ignore files and folders

\$touch *.gitignore*

\$vi *.gitignore*

\$git **add** *.gitignore*

\$git **status**



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# .gitignore Template

## A collection of `.gitignore` templates

This is GitHub's collection of `.gitignore` file templates. We use this list to populate the `.gitignore` template choosers available in the GitHub.com interface when creating new repositories and files.

For more information about how `.gitignore` files work, and how to use them, the following resources are a great place to start:

- The [Ignoring Files chapter](#) of the [Pro Git book](#).
- The [Ignoring Files article](#) on the GitHub Help site.
- The [gitignore\(5\)](#) manual page.

<https://github.com/github/gitignore>



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Try to reset README file

\$git **WHAT COMMAND?**

\$git **help**

\$git

\$git **status**

On branch master  
nothing to commit, working directory clean



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Move your files

```
$git mv README README.MD
```

```
$git status
```

```
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    renamed:  README -> README.md
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Move your files

\$git **mv** *README README.MD*

**\$mv README README.MD**

**\$git rm README**

**\$git add README.md**



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Working with Branching



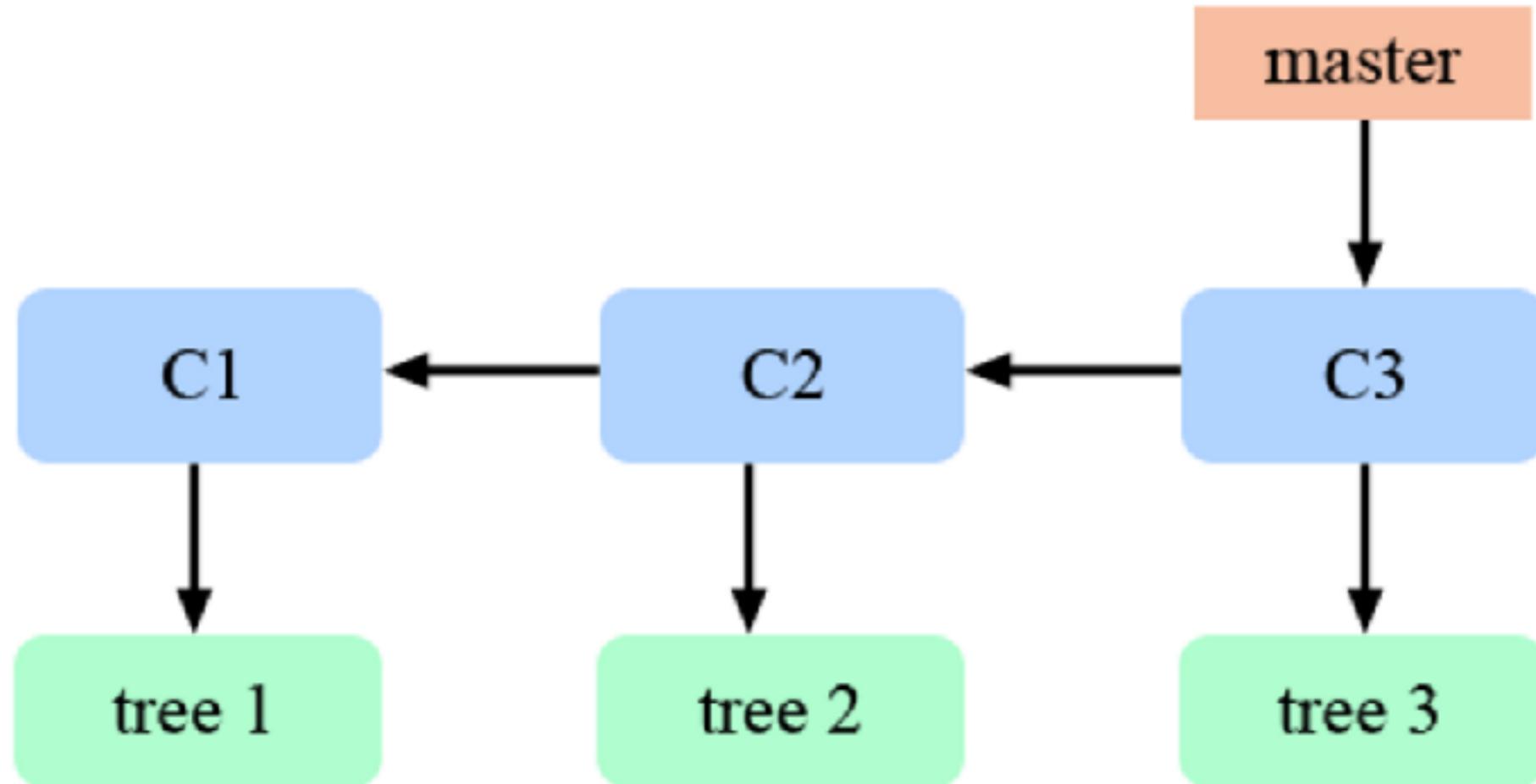
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# Each branch points to a commit



**Share** — copy and redistribute the material in any medium or format.

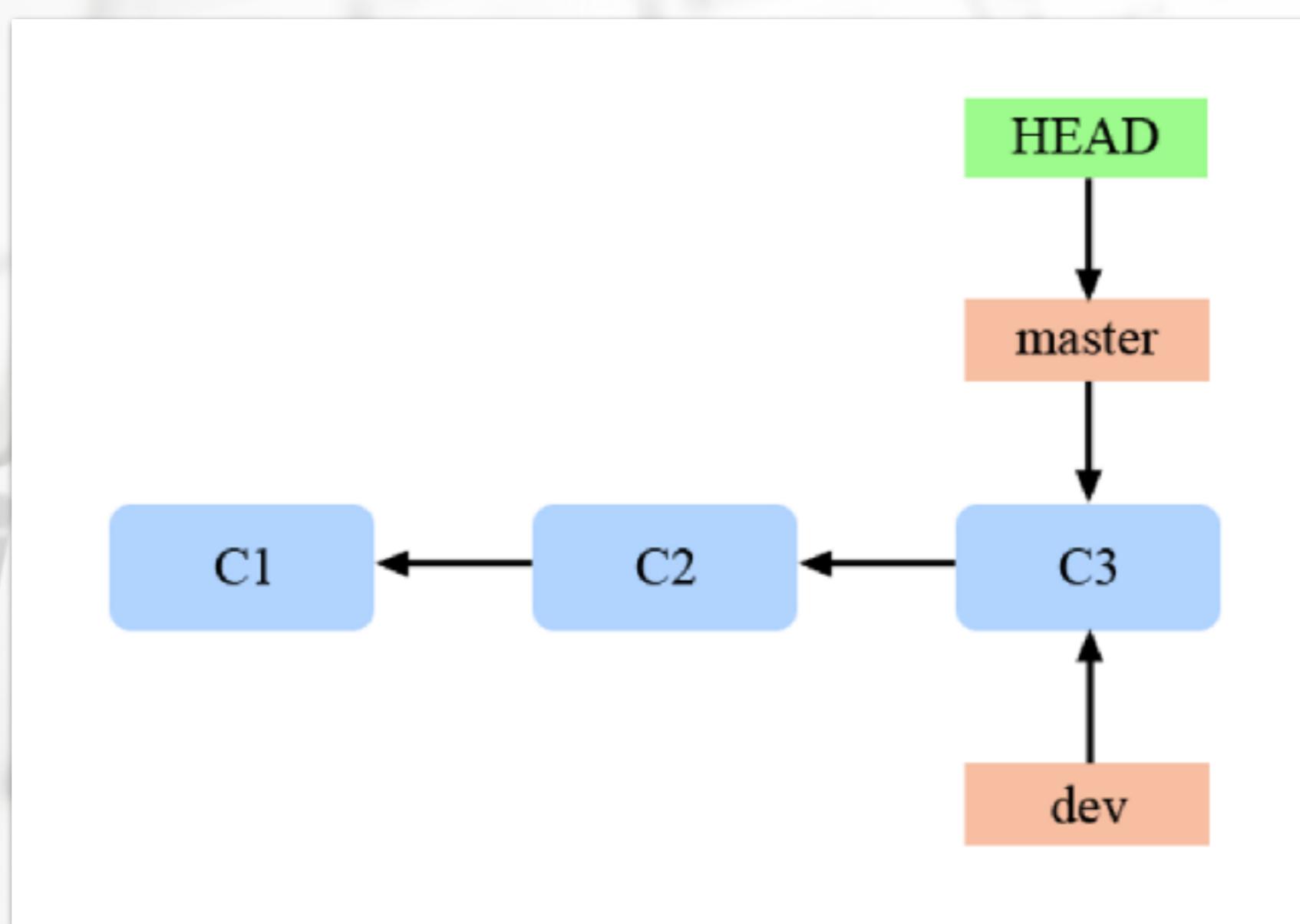
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Create new branch

\$git **branch** *BRANCH NAME*



**Share** — copy and redistribute the material in any medium or format.

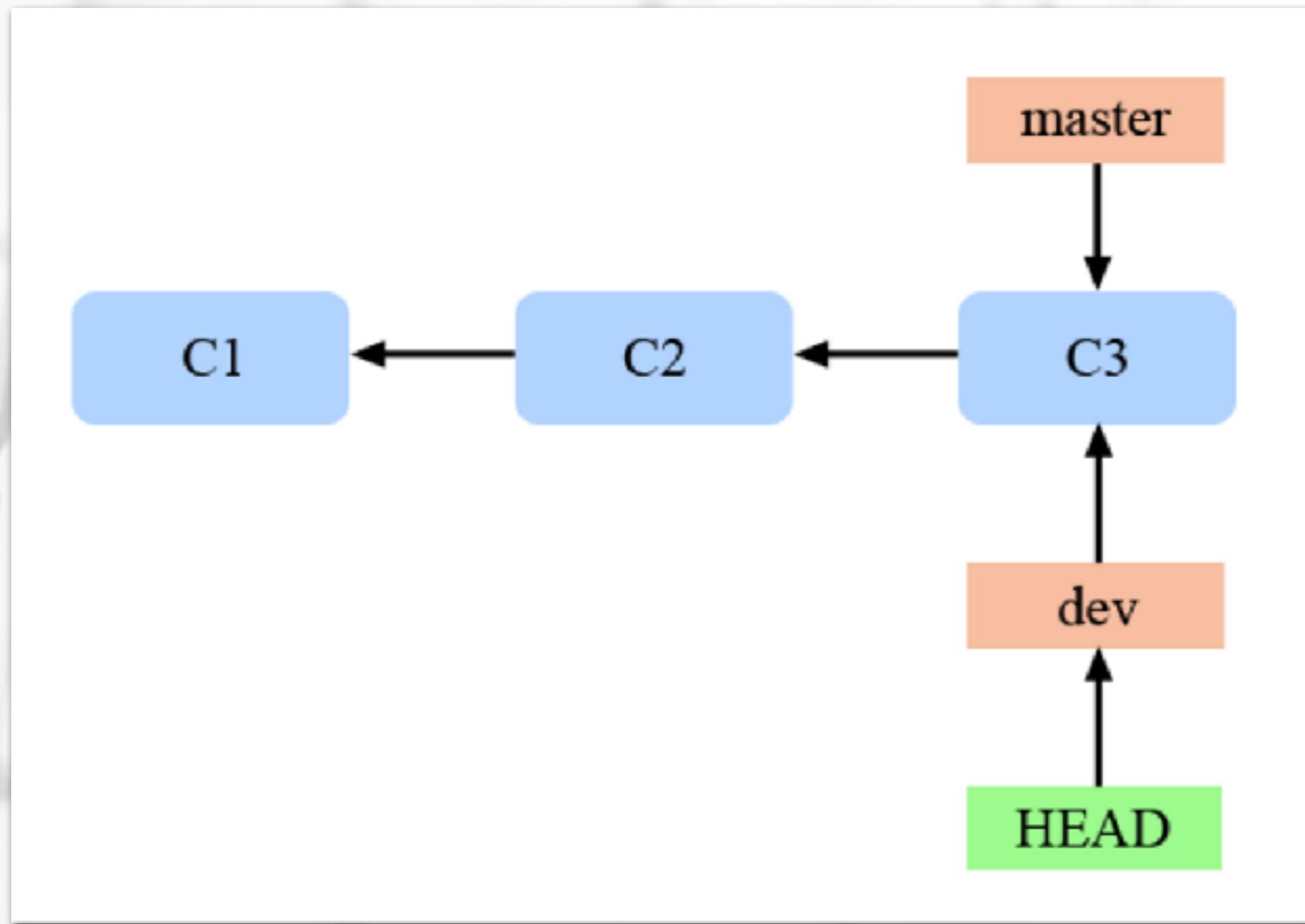
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Switch branch

\$git **checkout** BRANCH NAME



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

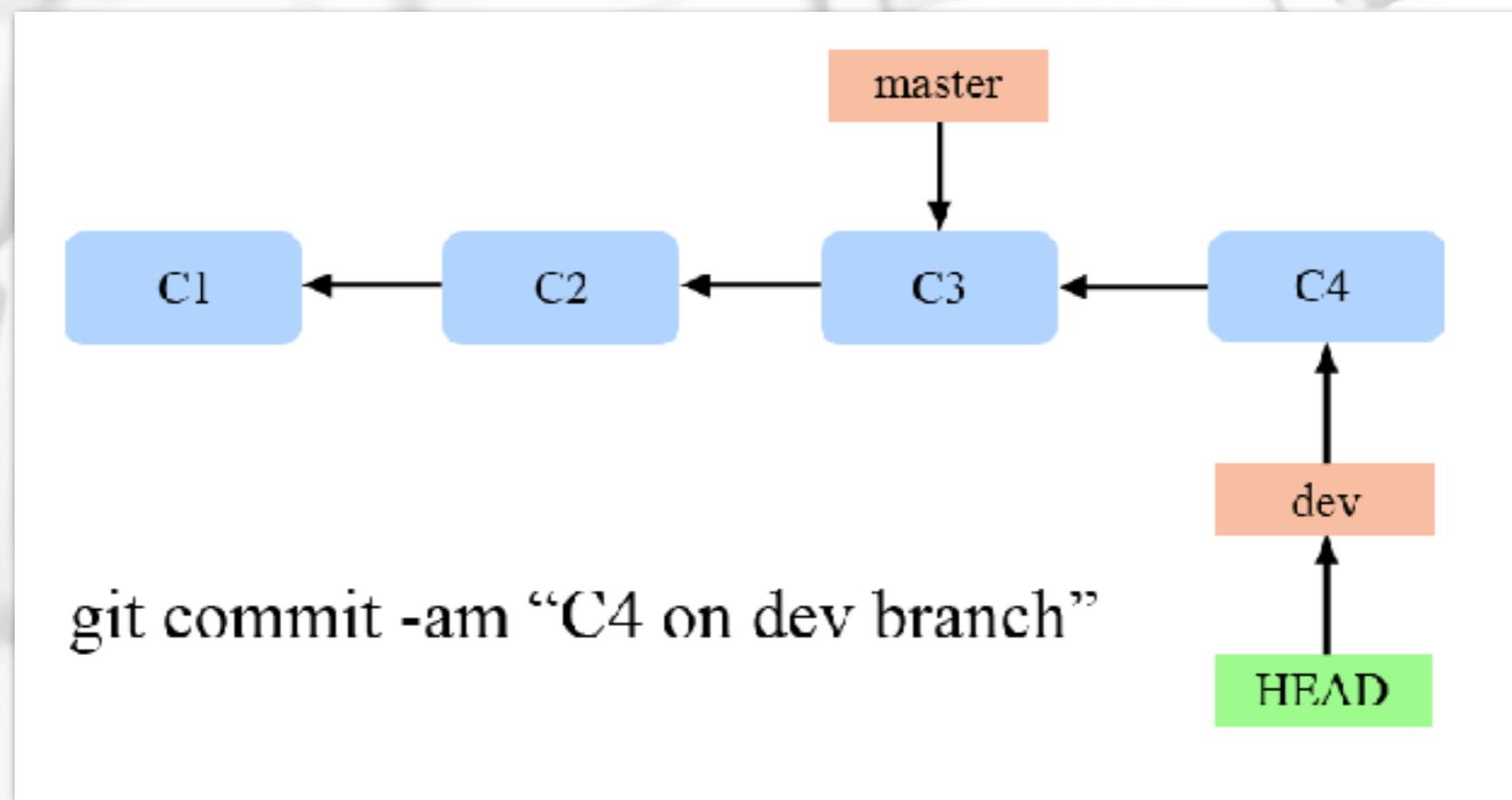
**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Modified and commit on branch

\$echo “On Branch” > *README*

\$git **commit** -am “C4 on dev”



**Share** — copy and redistribute the material in any medium or format.

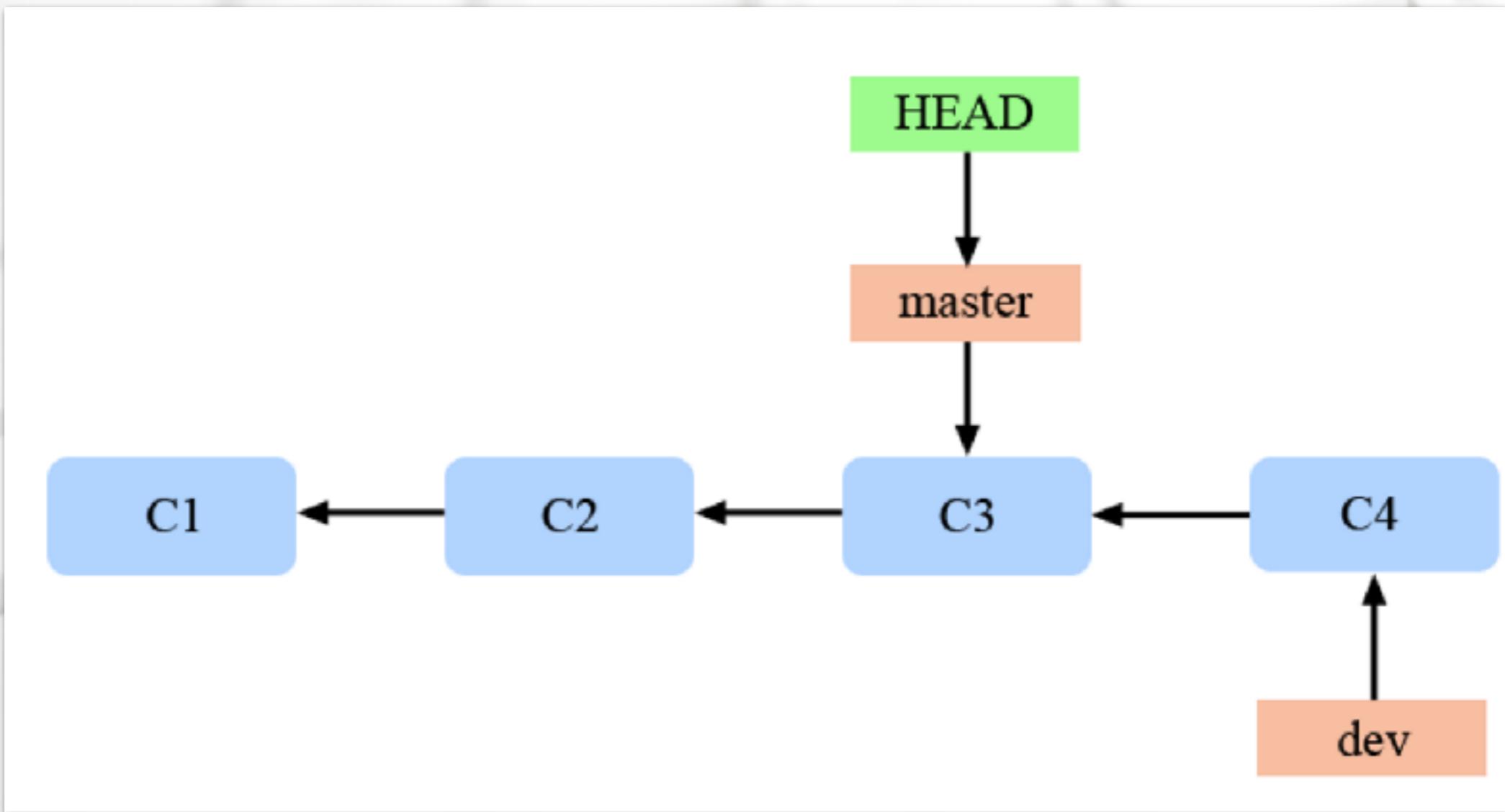
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Switch to master branch

\$git **checkout** master



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Create and Switch branch

`$git checkout -b BRANCH NAME`

`$git branch BRANCH NAME`

`$git checkout BRANCH NAME`



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Delete and Show all branch

```
$git branch -d BRANCH NAME
```

```
$git branch
```



**Share** — copy and redistribute the material in any medium or format.

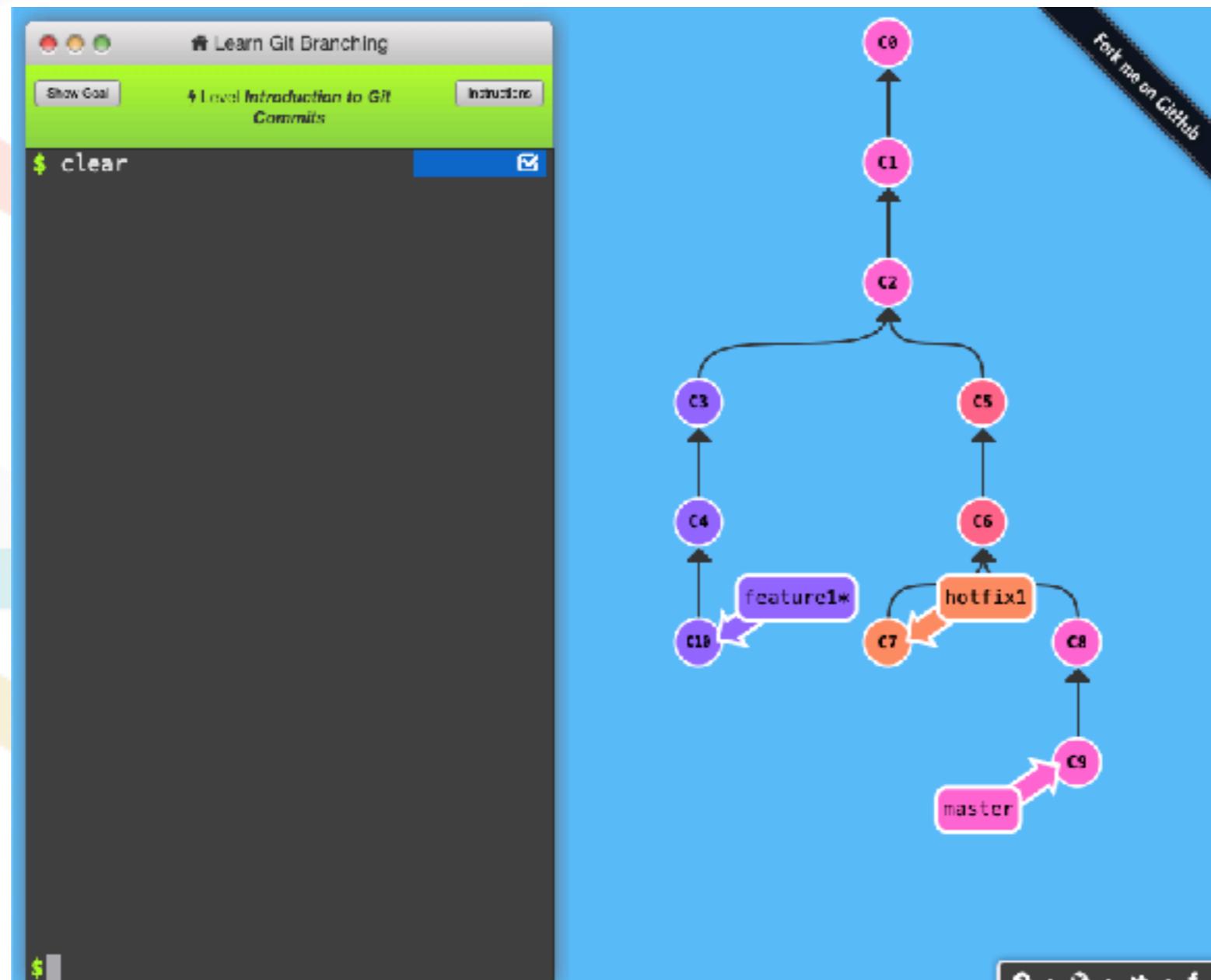
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Let DEMO

<https://learngitbranching.js.org/>



**Share** — copy and redistribute the material in any medium or format.

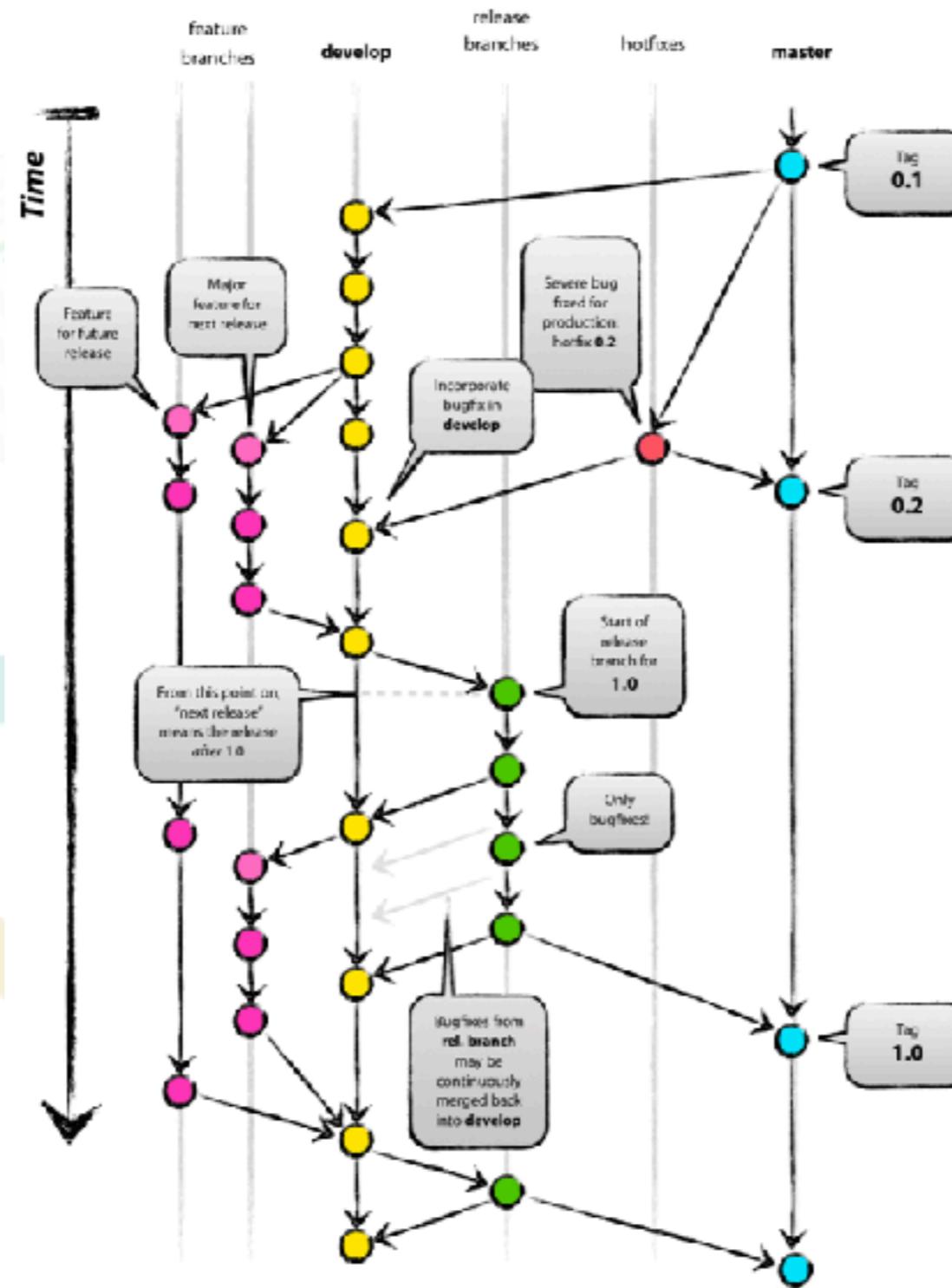
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Git branch

<https://nvie.com/posts/a-successful-git-branching-model/>



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



# keep master running



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use this material for commercial purposes.

บริษัท สยามชานาญกิจ จำกัด และเพื่อนพ้องน้องพี่

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



# Need to fix bug ?



**Share** — copy and redistribute the material in any medium or format.  
**Adapt** — remix, transform, and build upon the material.  
**NonCommercial** — You may not use this material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

บริษัท สยามชานาณกิจ จำกัด และเพื่อนพ้องน้องพี่

# Branch by Feature

\$git checkout **feature**

feature

master



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use this material for commercial purposes.

บริษัท สยามชานมกุ้ง จำกัด และเพื่อนพ้องน้องพี่

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



# \$git **checkout -b feature**



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use this material for commercial purposes.

บริษัท สยามชนาภิการจำกัด และเพื่อนพ้องน้องพี่

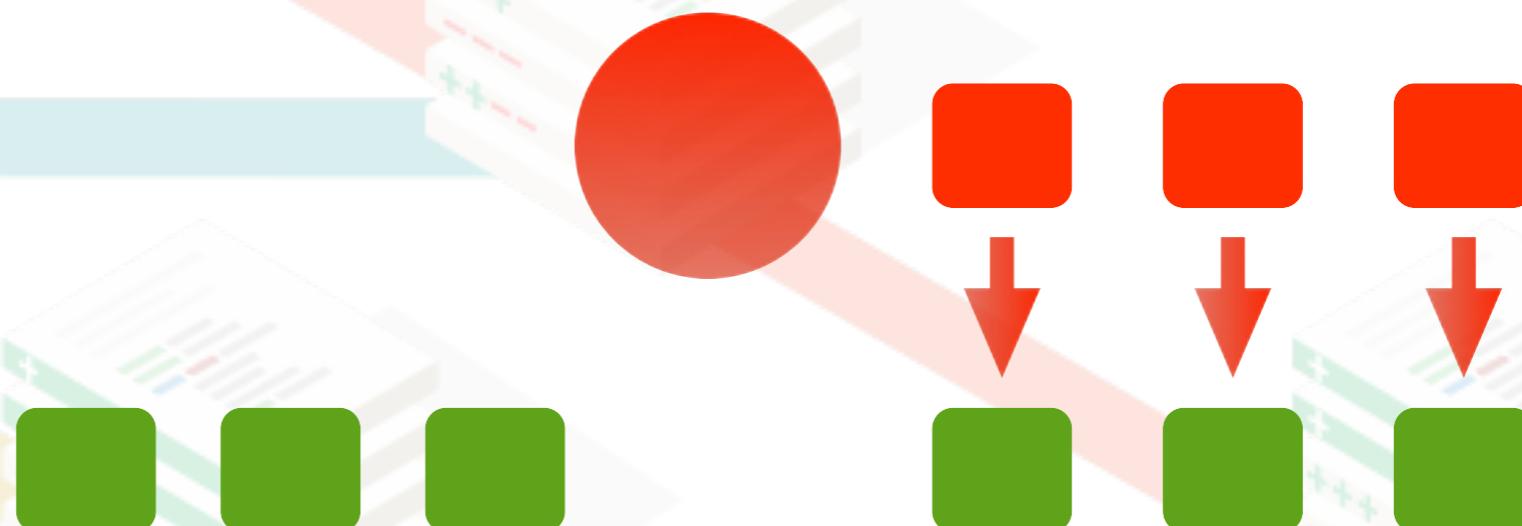
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# When ready to merge into master

```
$git merge feature master
```

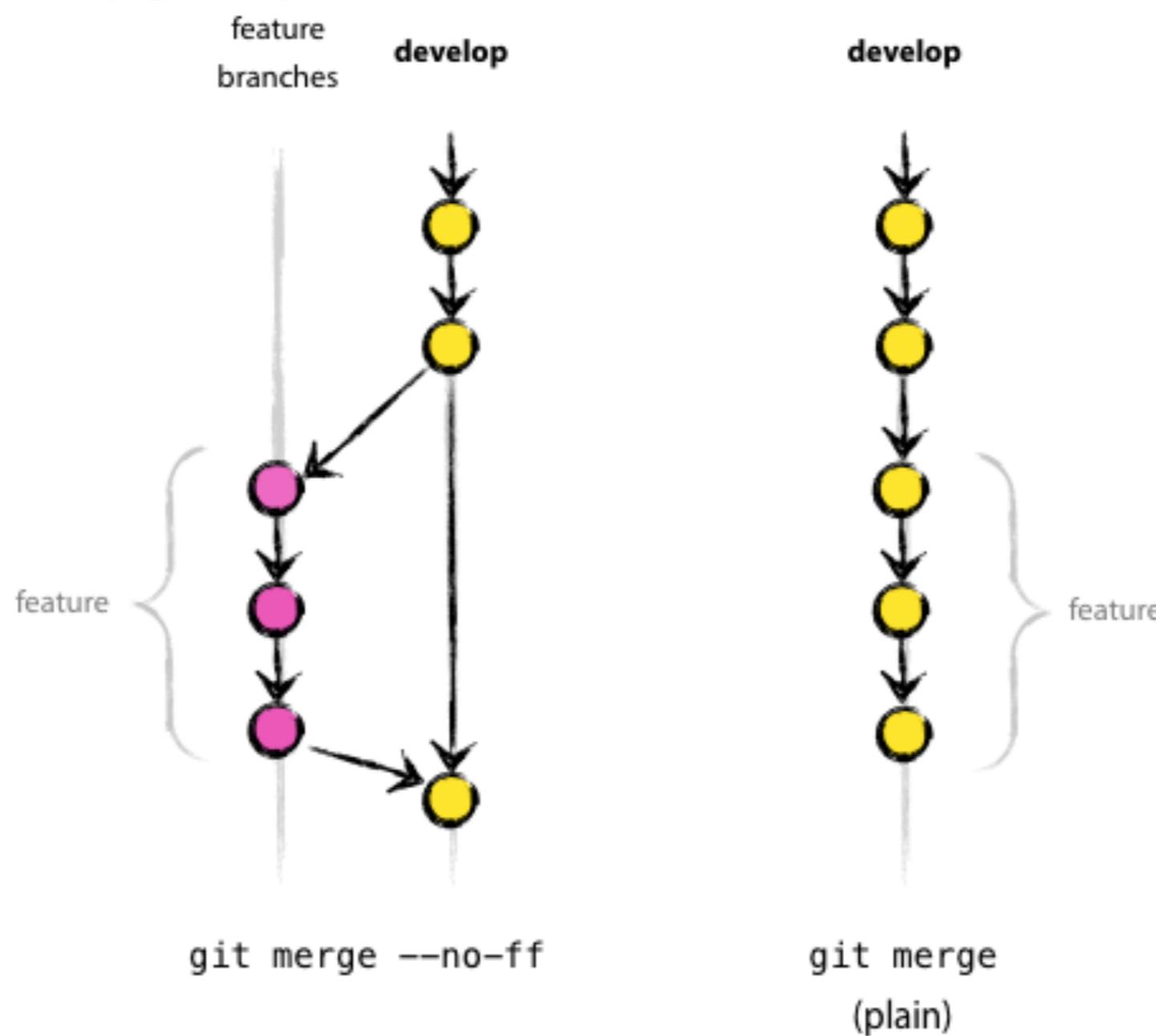
feature

master



# Git Merge – no Fast Forward

\$git **merge** —no-ff



# Git Branch

branching is cheap

painless development

**delete branch if it's not work**

**delete branch if it's not use**



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use this material for commercial purposes.

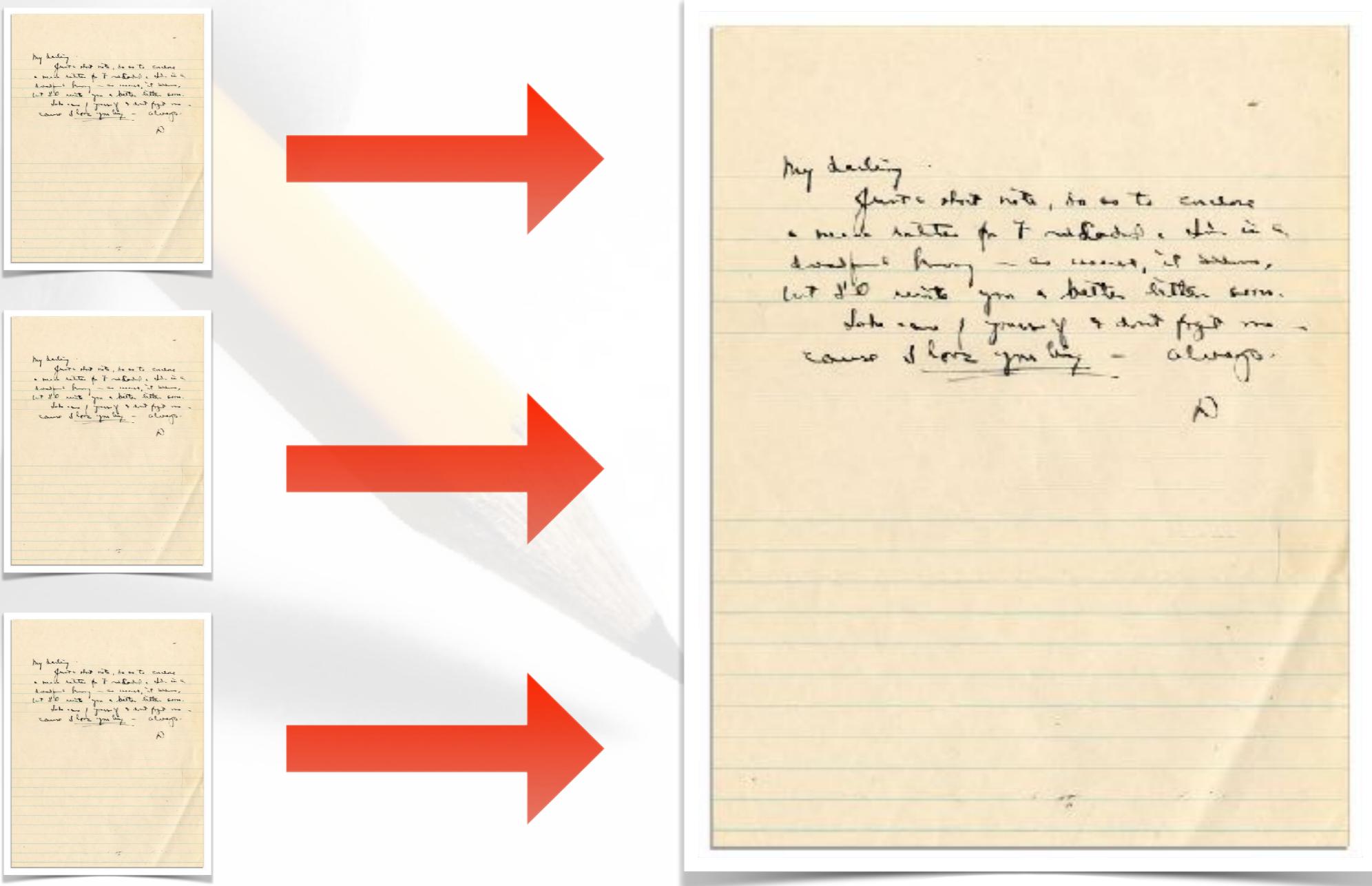
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Working with Remote Repository



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# All become One



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Working on GitHub

<http://github.com/>

The screenshot shows the GitHub homepage with a dark background. At the top, there is a navigation bar with links for "Why GitHub?", "Team", "Enterprise", "Explore", "Marketplace", and "Pricing". A search bar and "Sign in" and "Sign up" buttons are also at the top. The main heading "Built for developers" is displayed prominently. Below it, a paragraph describes GitHub as a development platform for open source and business, mentioning code review, project management, and software development. To the right, a sign-up form is shown with fields for "Username", "Email", and "Password". A note below the password field specifies character requirements: "Make sure it's at least 16 characters OR at least 8 characters including a number and a lowercase letter." A green "Sign up for GitHub" button is at the bottom of the form. A small disclaimer at the bottom of the form states: "By clicking 'Sign up for GitHub', you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails."



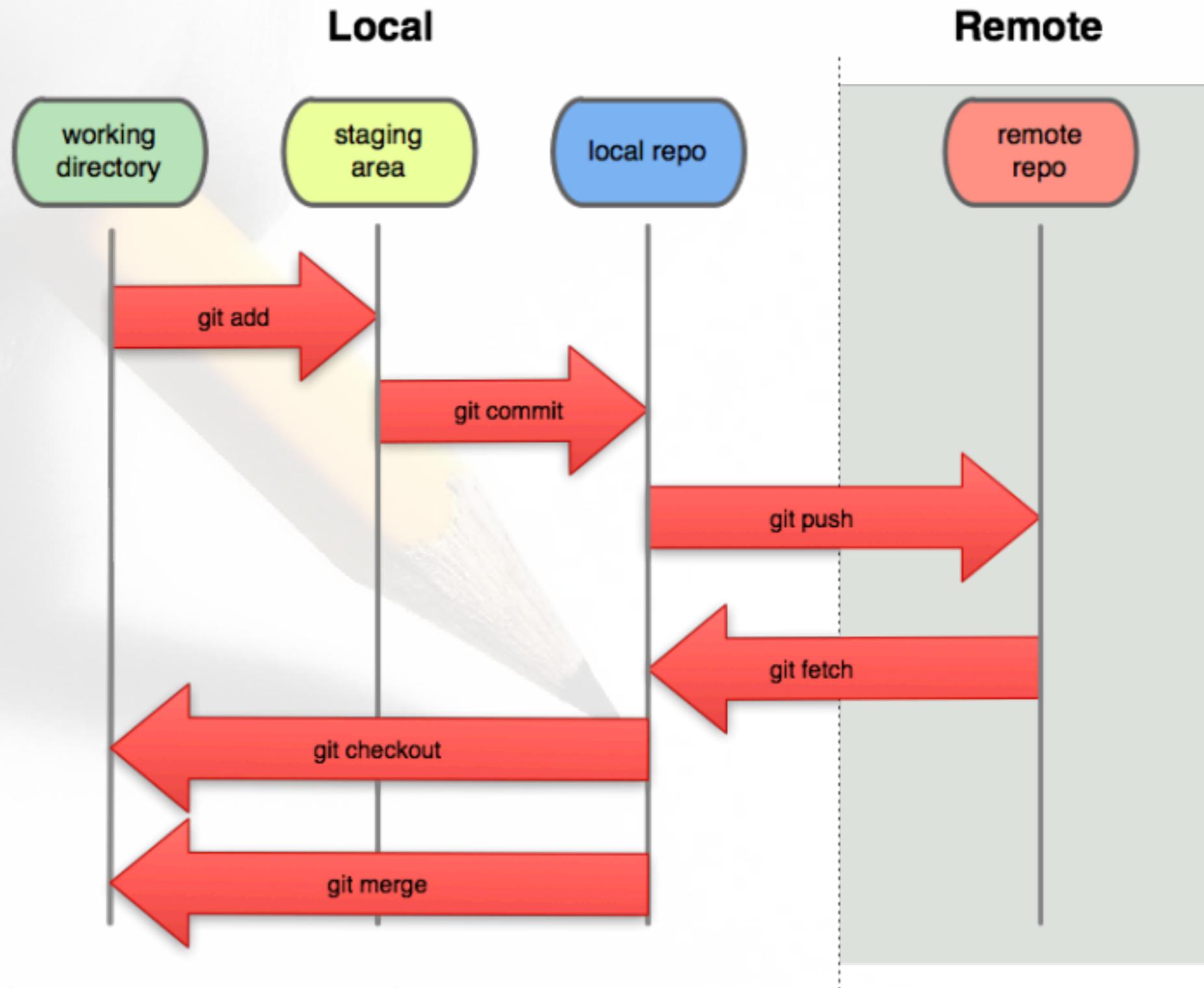
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Git Workflow



Source: <https://greenido.wordpress.com/2013/07/22/git-101-useful-commands/>



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

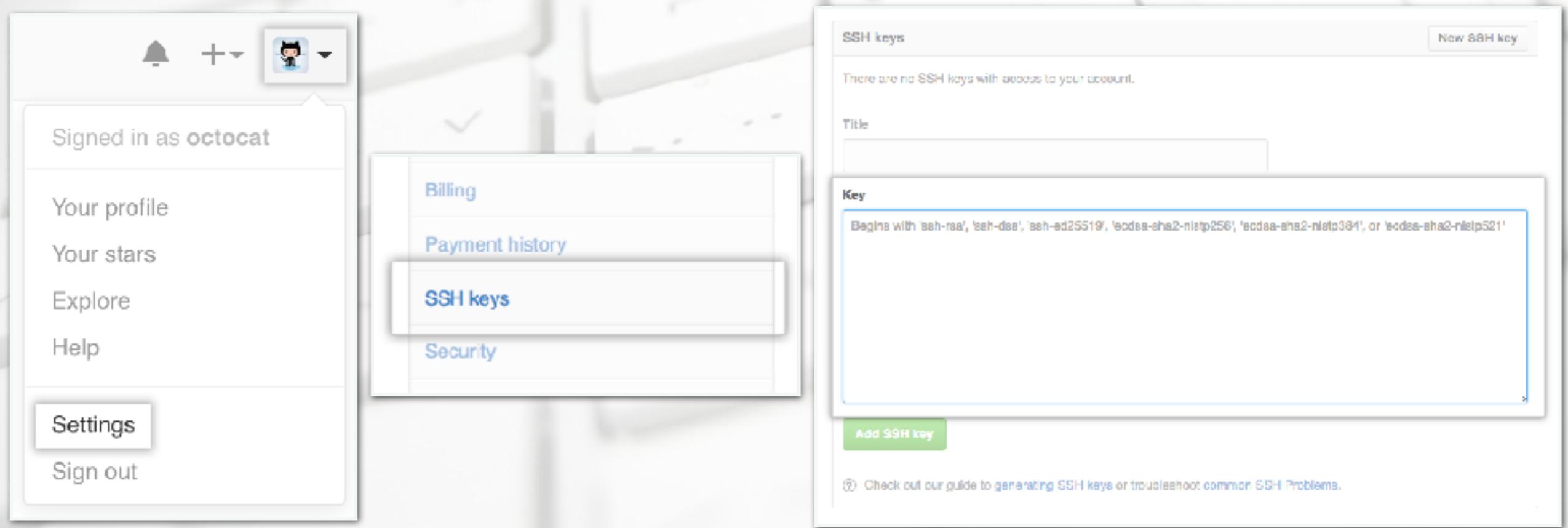
**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Generate ssh key and add to Github

\$ssh-keygen

\$pbcopy < ~/.ssh/id\_rsa.pub



**Share** — copy and redistribute the material in any medium or format.  
**Adapt** — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Testing your SSH connection

```
$ ssh -T git@github.com
```

```
The authenticity of host 'github.com (192.30.252.1)' can't be established.  
RSA key fingerprint is 16:27:ac:a5:76:28:2d:36:63:1b:56:4d:eb:df:a6:48.  
Are you sure you want to continue connecting (yes/no)?
```

```
The authenticity of host 'github.com (192.30.252.1)' can't be established.  
RSA key fingerprint is nThbg6kXUpJWGL7E1IGOCspRomTxoCARLviKw6E5SY8.  
Are you sure you want to continue connecting (yes/no)?
```

```
Hi username! You've successfully authenticated, but GitHub does not  
provide shell access.
```



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Add and Rename remote

\$git **remote add** [NICKNAME] [REMOTE URL]

\$git **remote rm** [NICKNAME]

\$git **remote rm** [OLD NICKNAME] [NEW NICKNAME]



**Share** — copy and redistribute the material in any medium or format.

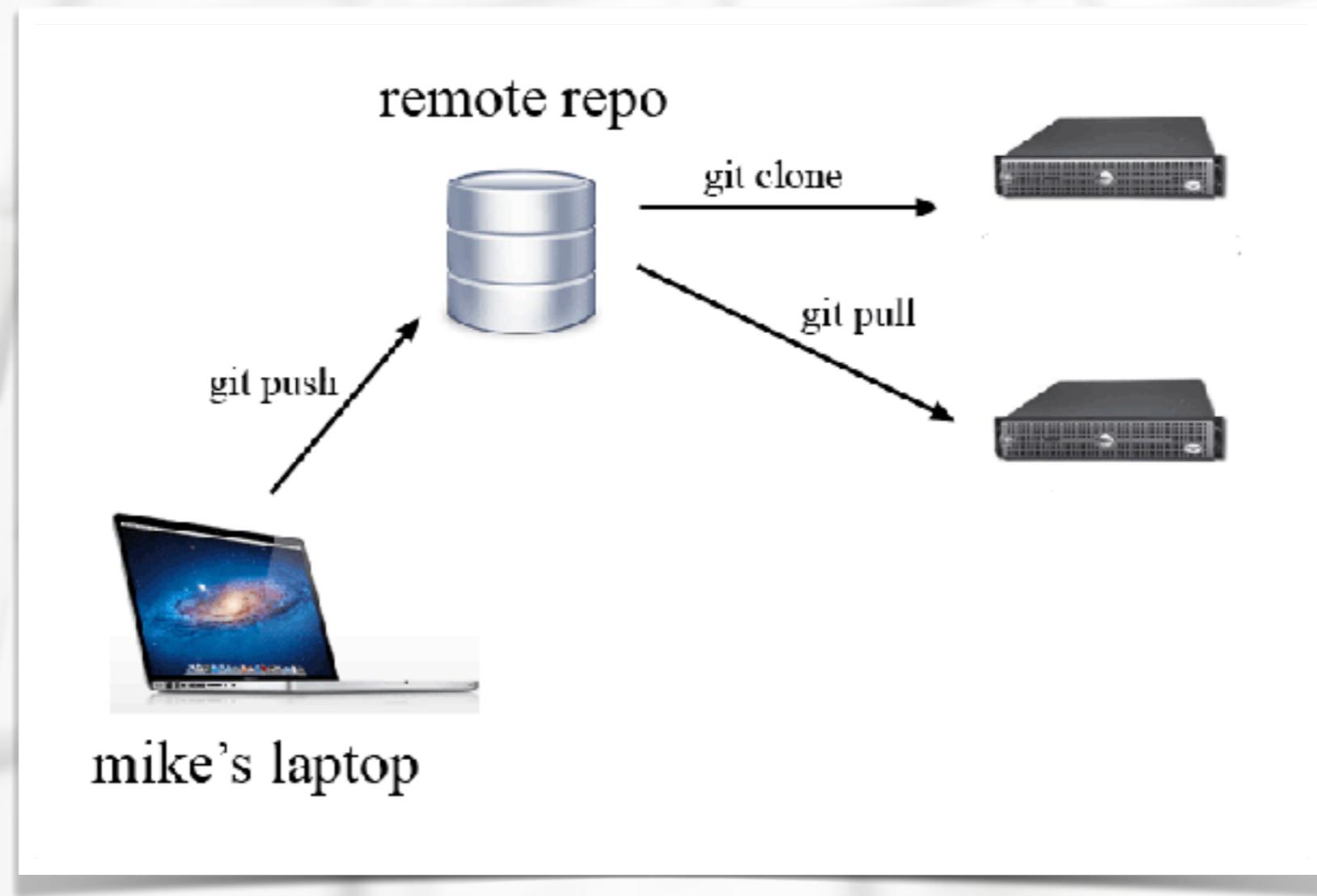
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Push to remote

\$git **push** [NICKNAME] [BRANCH]



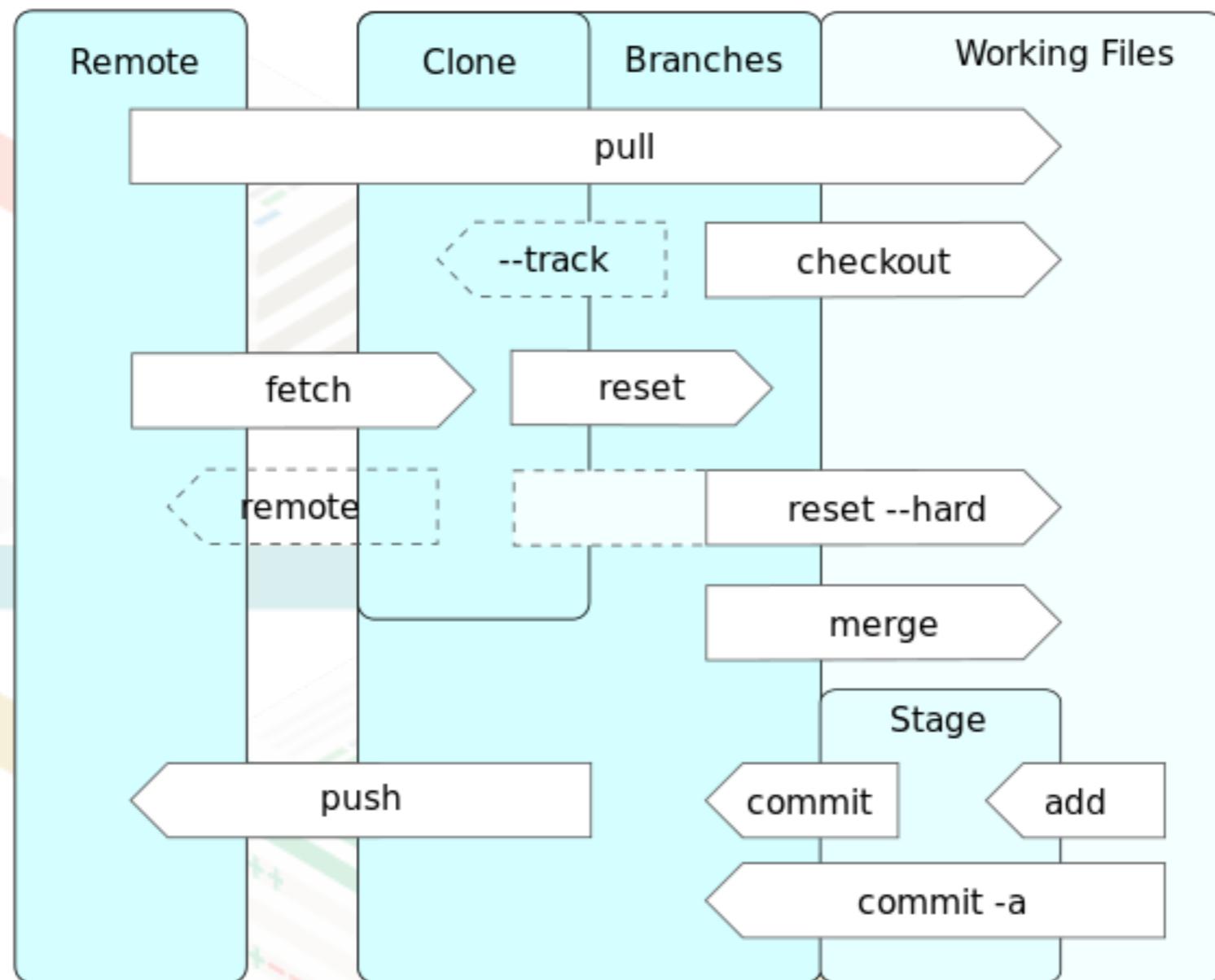
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Git Operation



Source: [https://en.wikipedia.org/wiki/Git#/media/File:Git\\_operations.svg](https://en.wikipedia.org/wiki/Git#/media/File:Git_operations.svg)



**Share** — copy and redistribute the material in any medium or format.

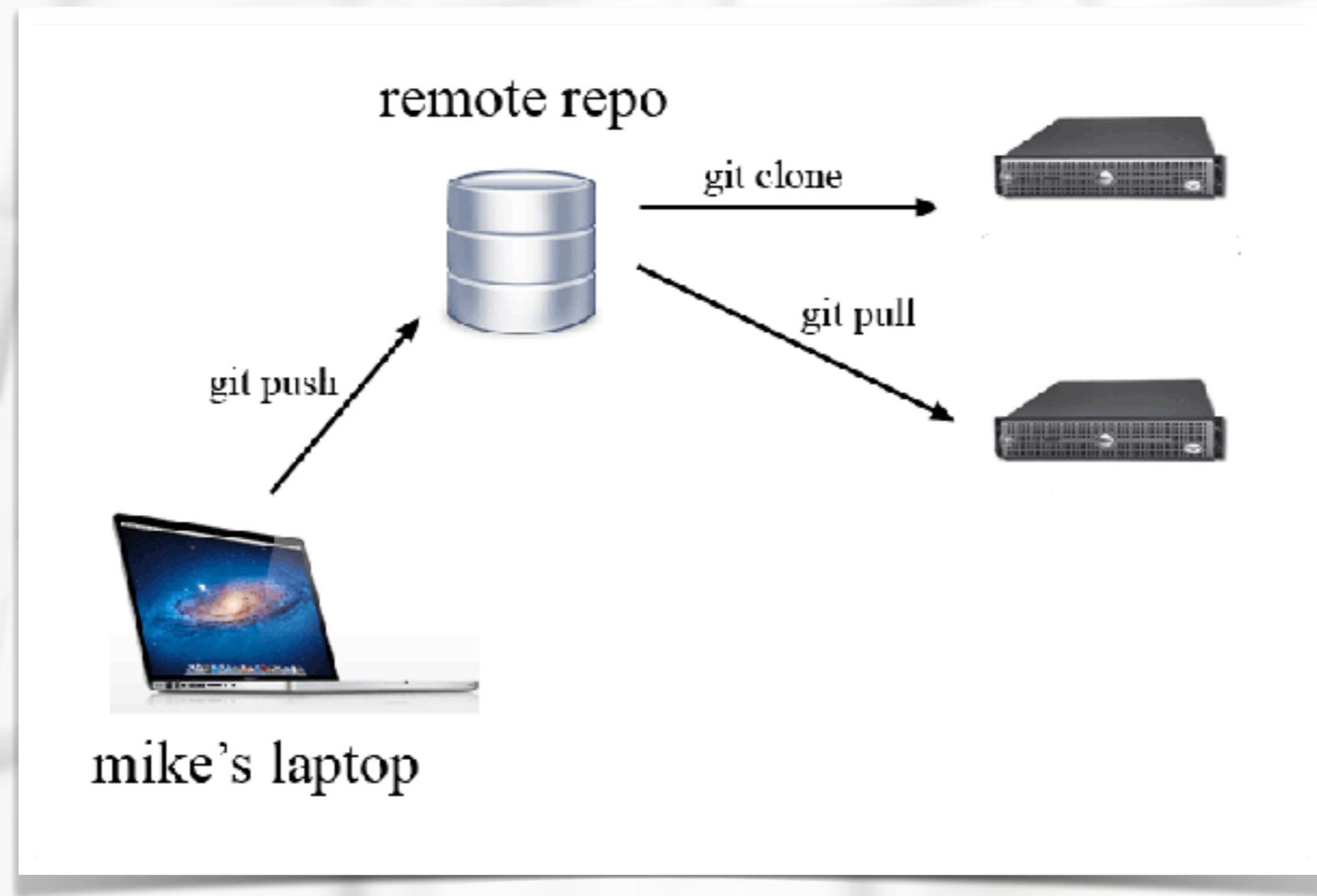
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Fetch and Merge

\$git **pull** [NICKNAME] [BRANCH]



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Fetch and Merge

\$git **pull** [NICKNAME] [BRANCH]

**\$git fetch [NICKNAME]**

**\$git merge [BRANCH]**



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

**BRANCHED FROM MASTER 3  
WEEKS AGO**



**MERGED BACK WITHOUT  
ANY CONFLICTS**



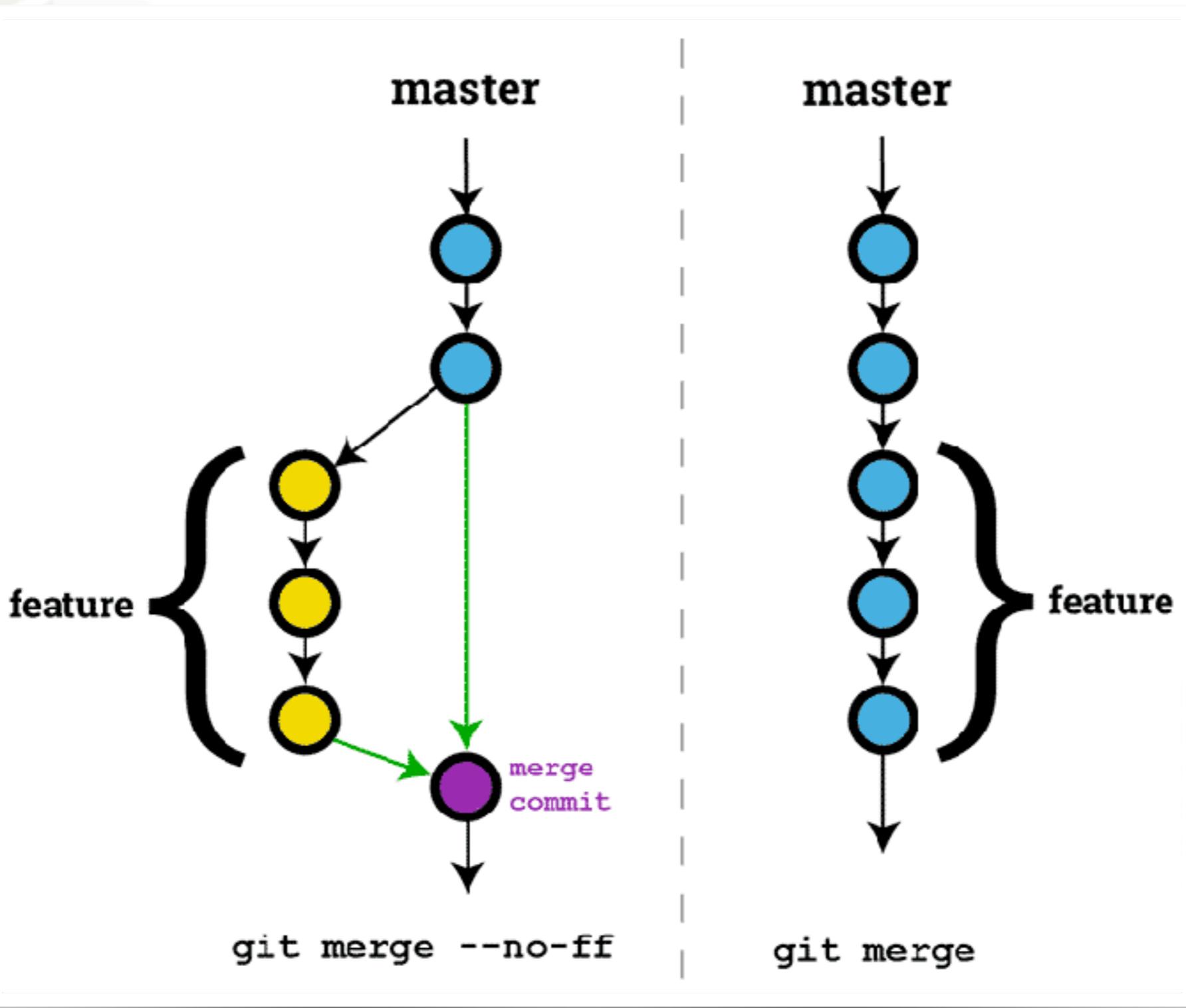
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Merge and Rebase



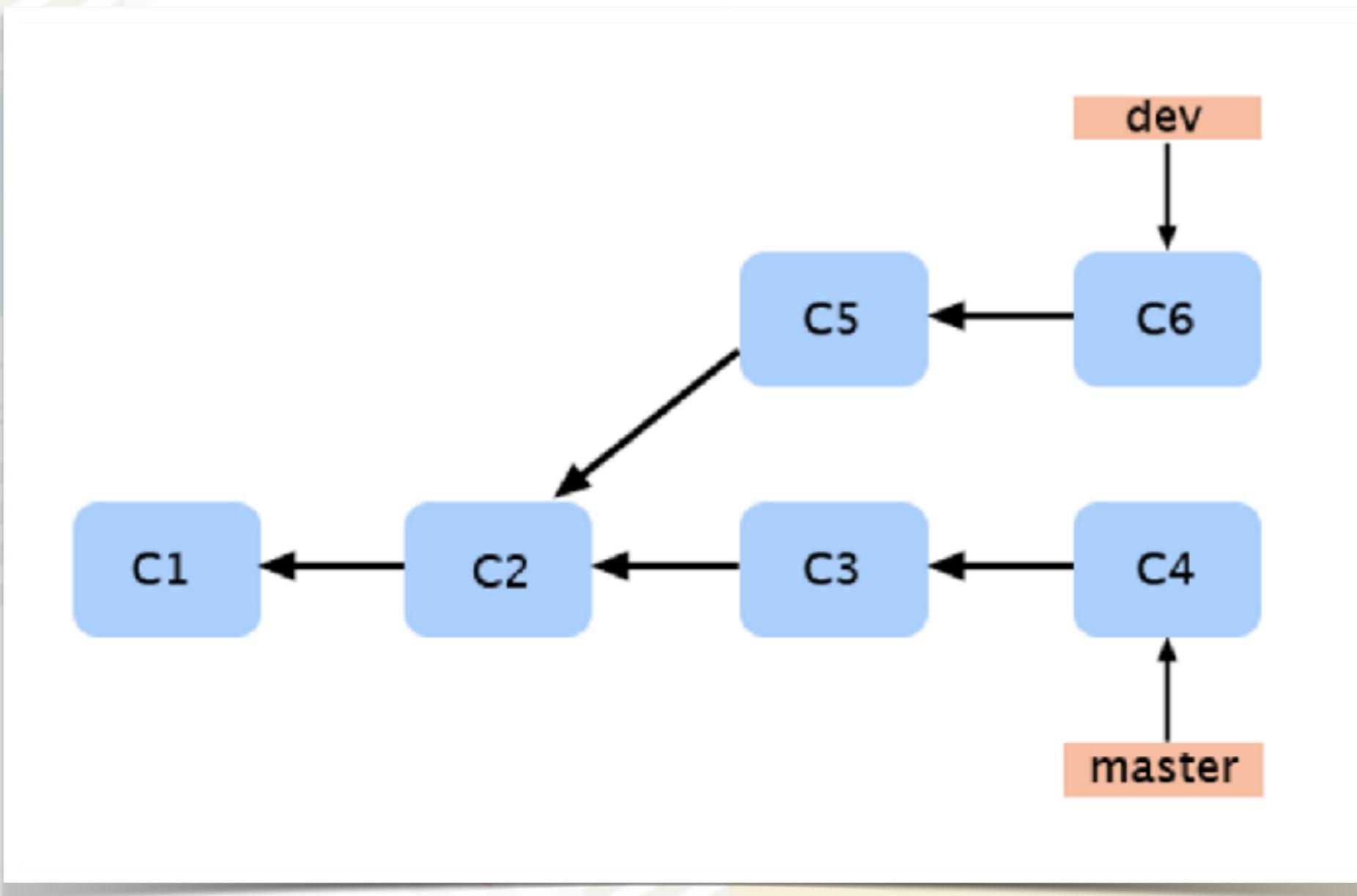
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Start before Merge



**Share** — copy and redistribute the material in any medium or format.

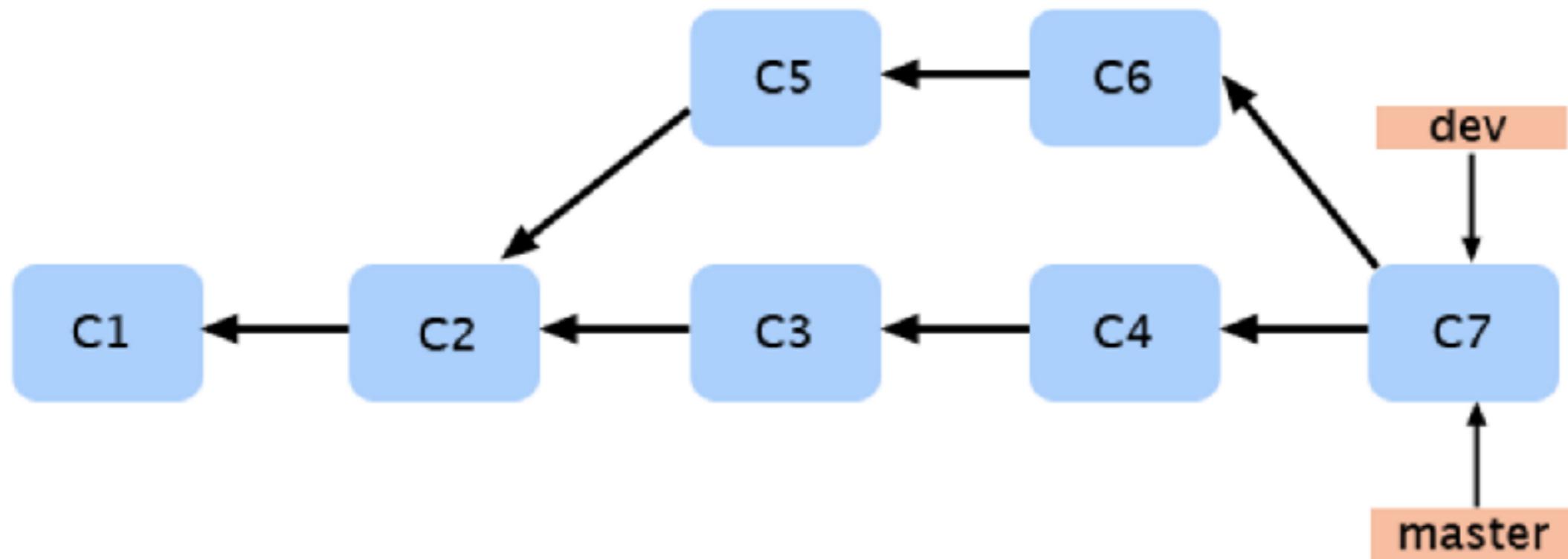
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# After Merge

\$git **merge**



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Avoid Merge Conflict

**Small change and commit**

**Early merge**

**Single Responsibility Principle**

**Communication is a Key**

**Mob programming**



**Share** — copy and redistribute the material in any medium or format.

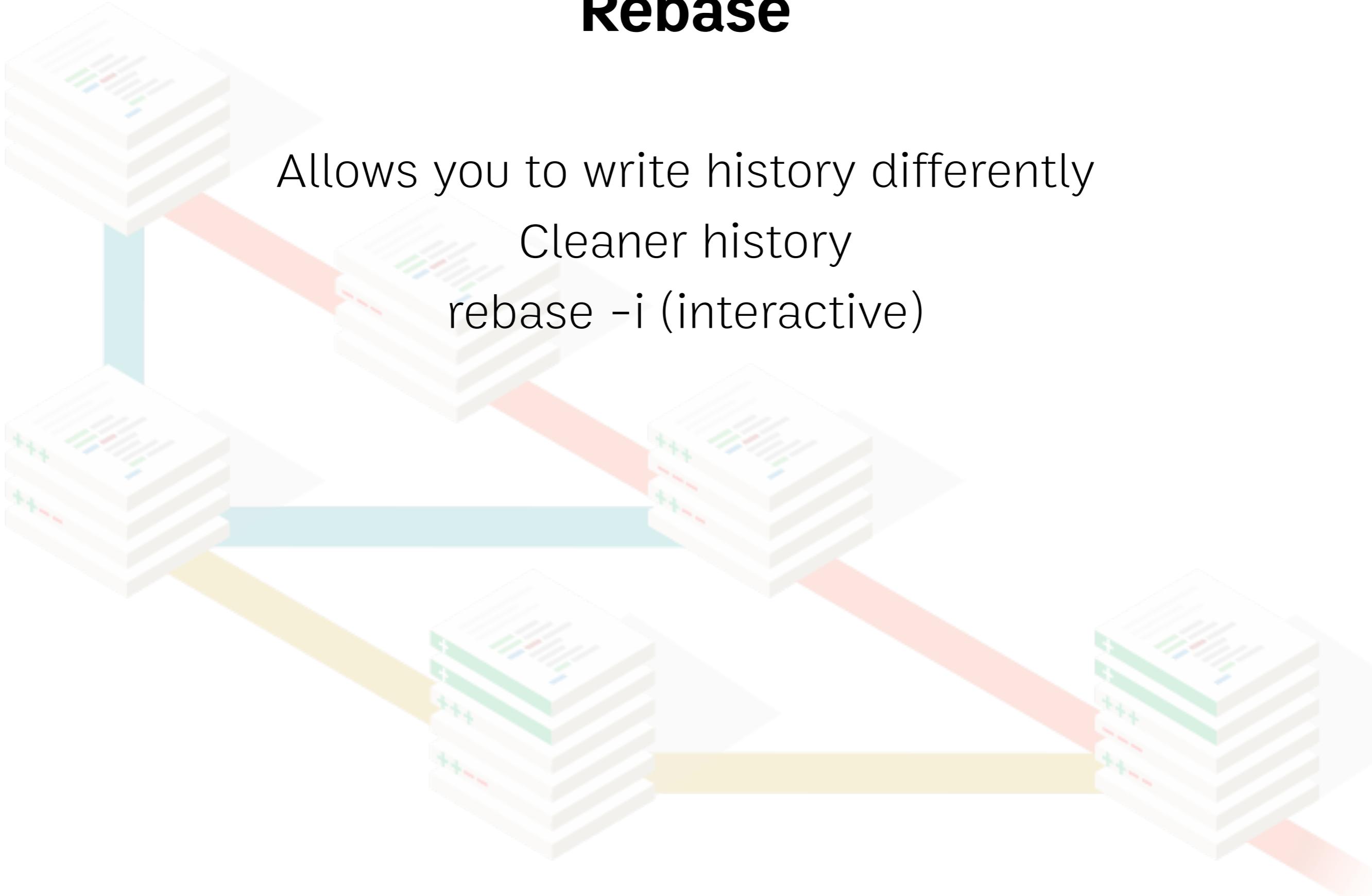
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

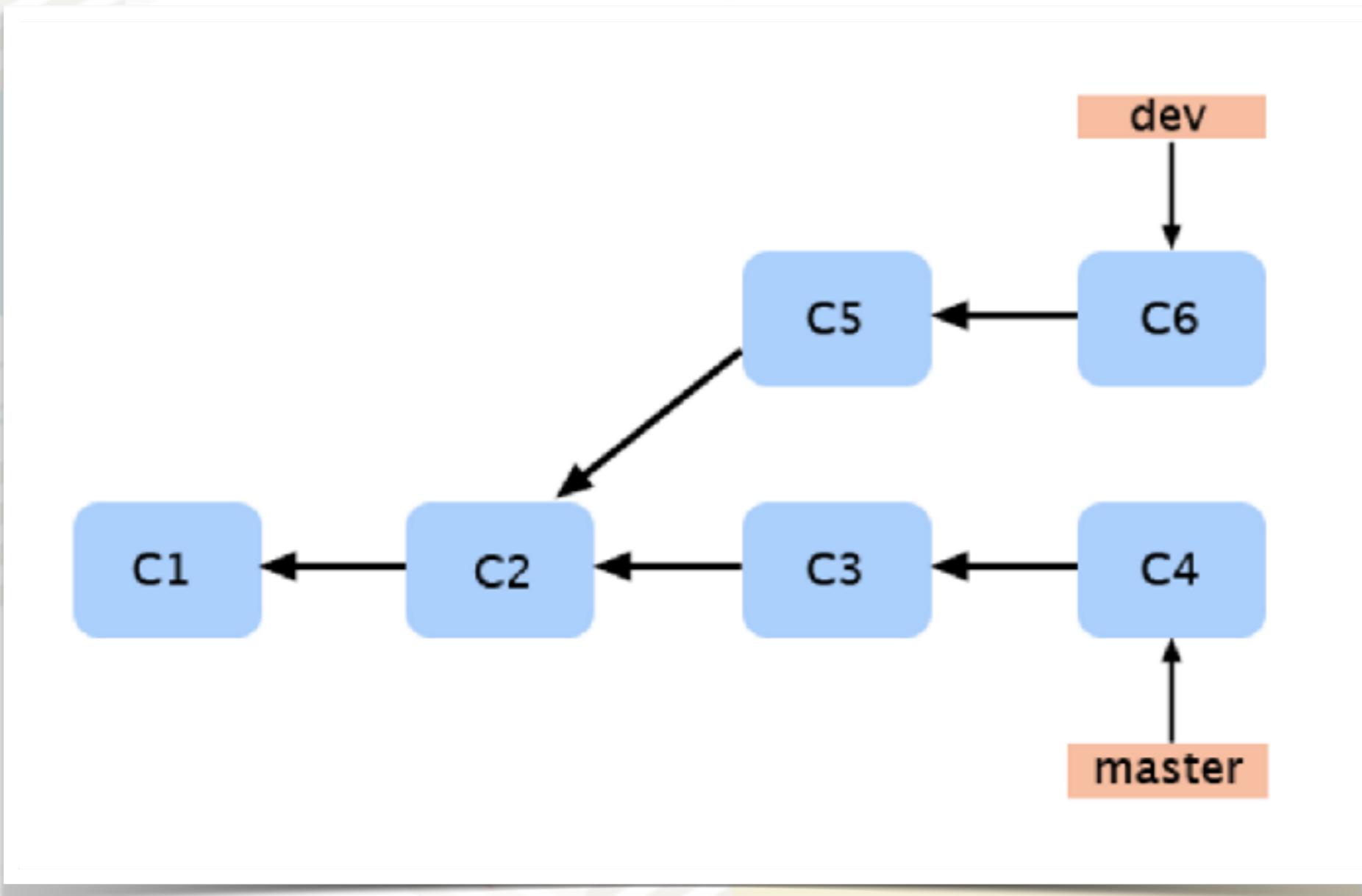
# Rebase

Allows you to write history differently  
Cleaner history  
`rebase -i (interactive)`



**Share** — copy and redistribute the material in any medium or format.  
**Adapt** — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Start before Rebase



**Share** — copy and redistribute the material in any medium or format.

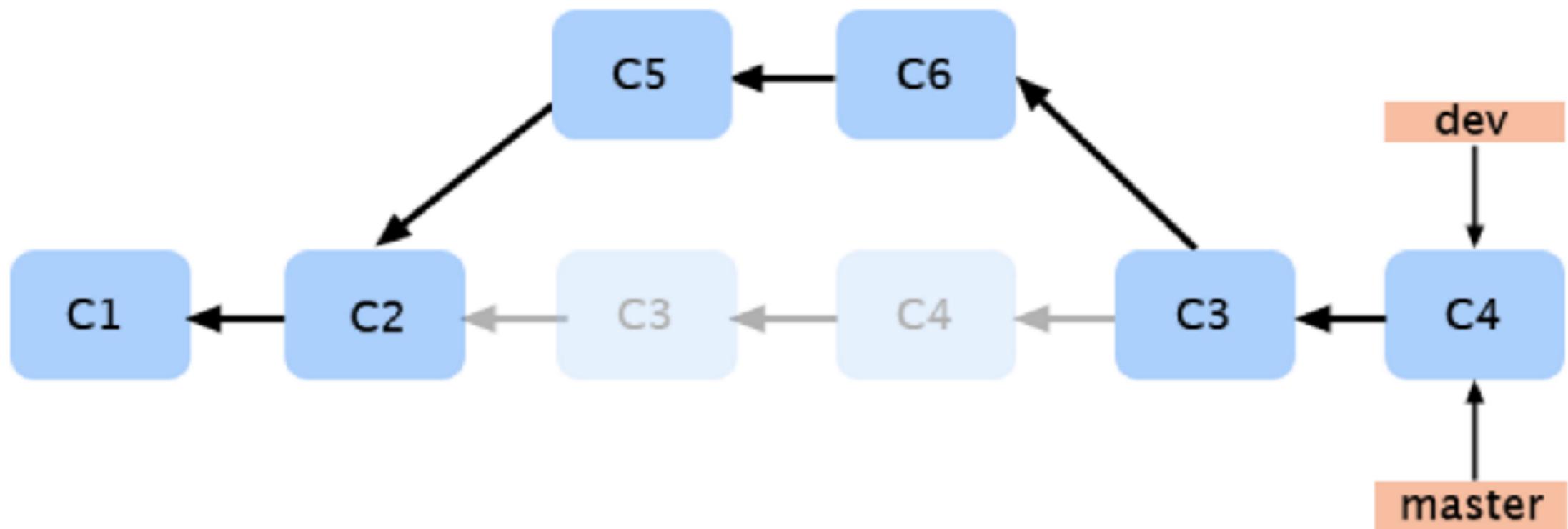
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# After Rebase

\$git **rebase** [BRANCH NAME]



**Share** — copy and redistribute the material in any medium or format.

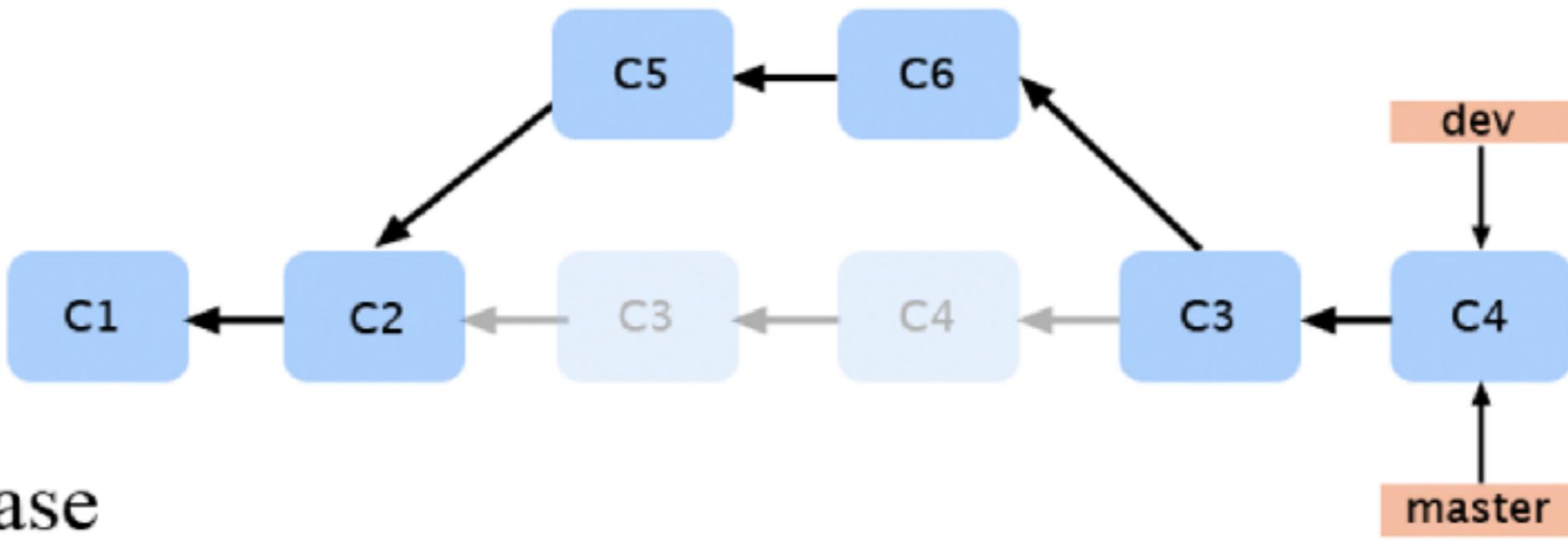
**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

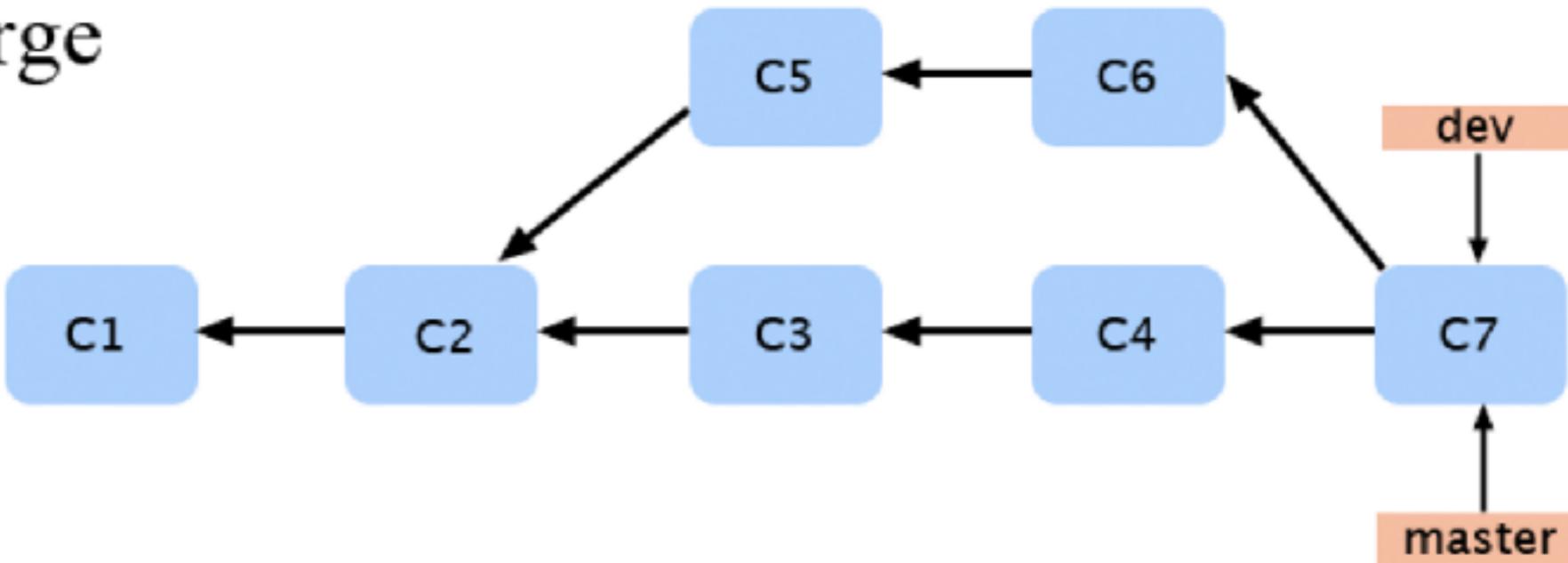
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Merge vs Rebase

rebase



merge



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



# Git Good



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



# Git Status



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Git help

\$git **—help**

```
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           <command> [<args>]
```

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)

clone	Clone a repository into a new directory
init	Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)

add	Add file contents to the index
mv	Move or rename a file, a directory, or a symlink
restore	Restore working tree files
rm	Remove files from the working tree and from the index
sparse-checkout	Initialize and modify the sparse-checkout

examine the history and state (see also: git help revisions)

bisect	Use binary search to find the commit that introduced a bug
diff	Show changes between commits, commit and working tree, etc
grep	Print lines matching a pattern
log	Show commit logs
show	Show various types of objects
status	Show the working tree status

grow, mark and tweak your common history

branch	List, create, or delete branches
commit	Record changes to the repository
merge	Join two or more development histories together
rebase	Reapply commits on top of another base tip



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Commit Message



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Meaningful ?

...

TODO  
asdasfdgafg  
stuff  
fixed bug  
add feature



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Better ?

“Added a user object to the database.  
currently only has a name and email.  
no authentication yet ”



# Readable

“Fixed the bug that would add something  
to everything. Turn to not add something  
to everything”



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Issues on GitHub.com

<https://guides.github.com/features/issues/>

bug 01 #1

 **Closed** up1 opened this issue 5 minutes ago · 0 comments



up1 commented 5 minutes ago

Owner



*No description provided.*

 up1 closed this issue from a commit 4 minutes ago

–  Fixed #1 to remove aomething

90db52f

 up1 closed this in 90db52f 4 minutes ago

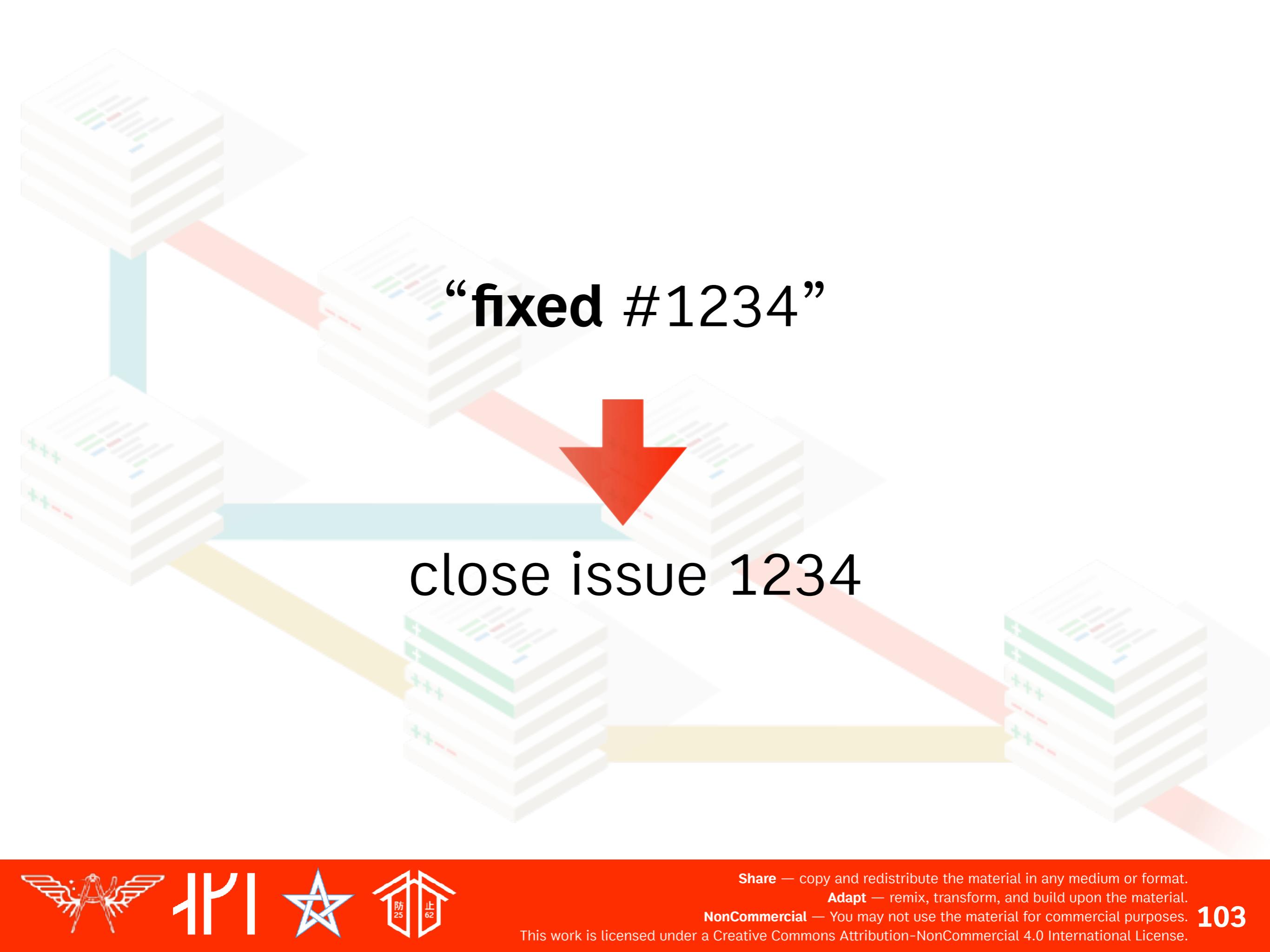


**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



“fixed #1234”



close issue 1234



**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



fixed, fix, fixes  
closed, close, closes

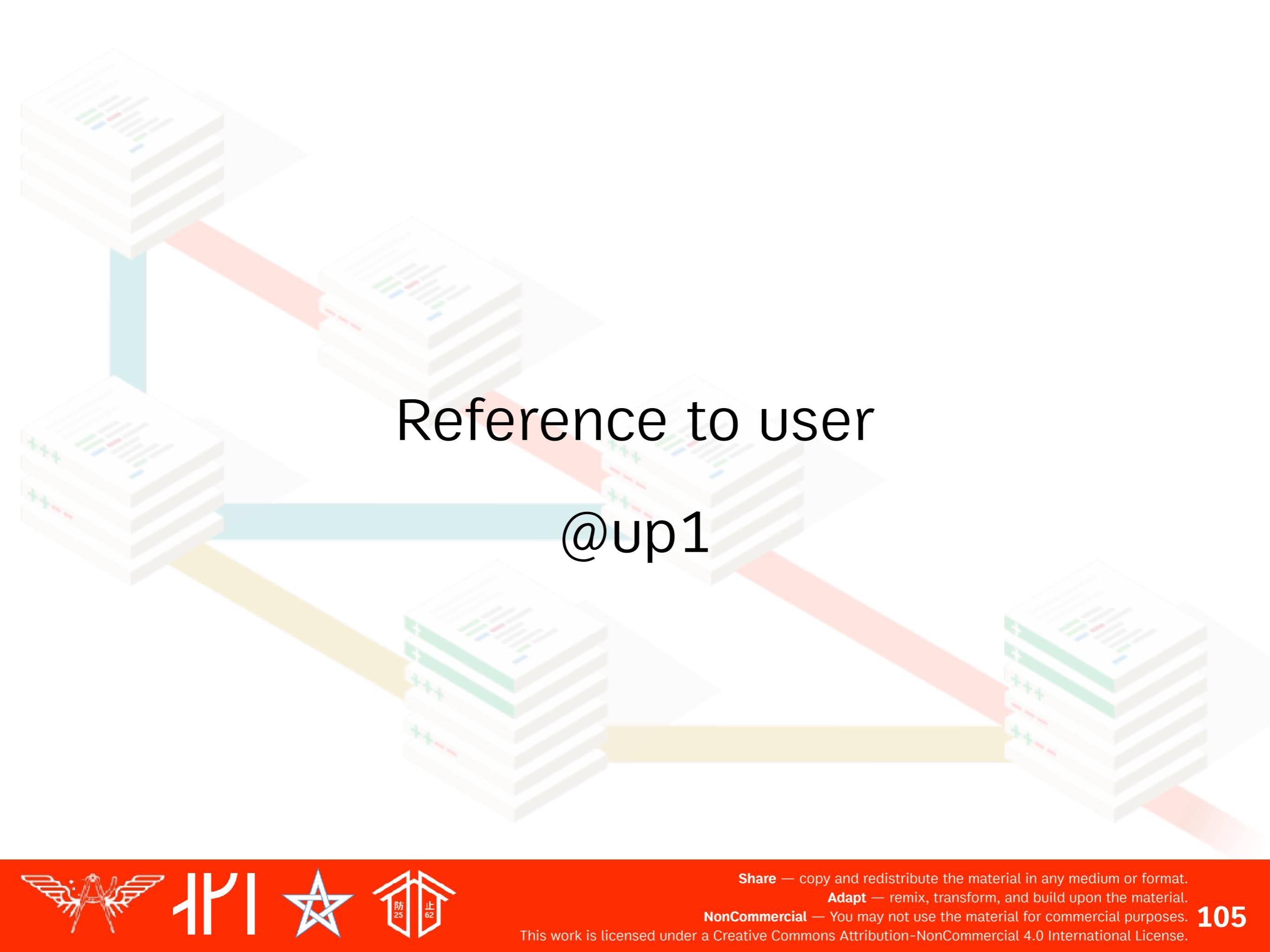


**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



# Reference to user

@up1



**Share** — copy and redistribute the material in any medium or format.  
**Adapt** — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# References



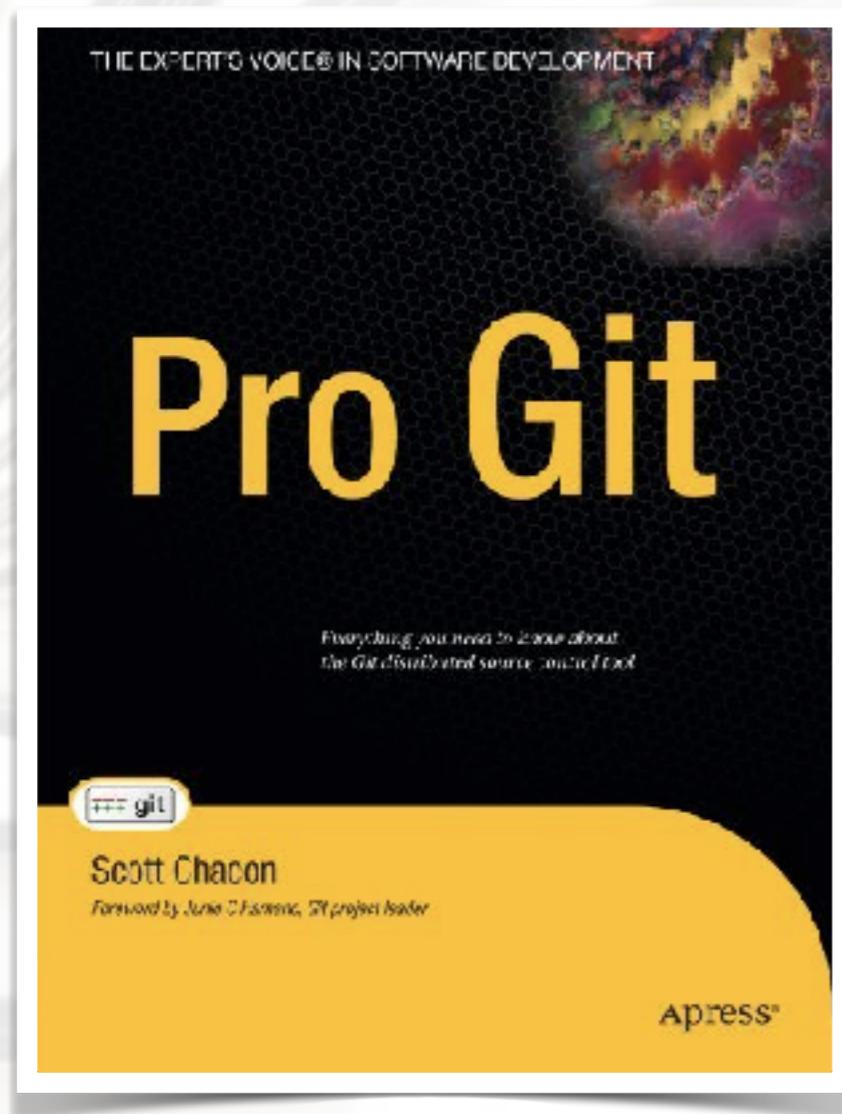
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# Books to Read and Practice



<http://www.git-scm.com/book/en/v2>



<http://shop.oreilly.com/product/0636920034520.do>



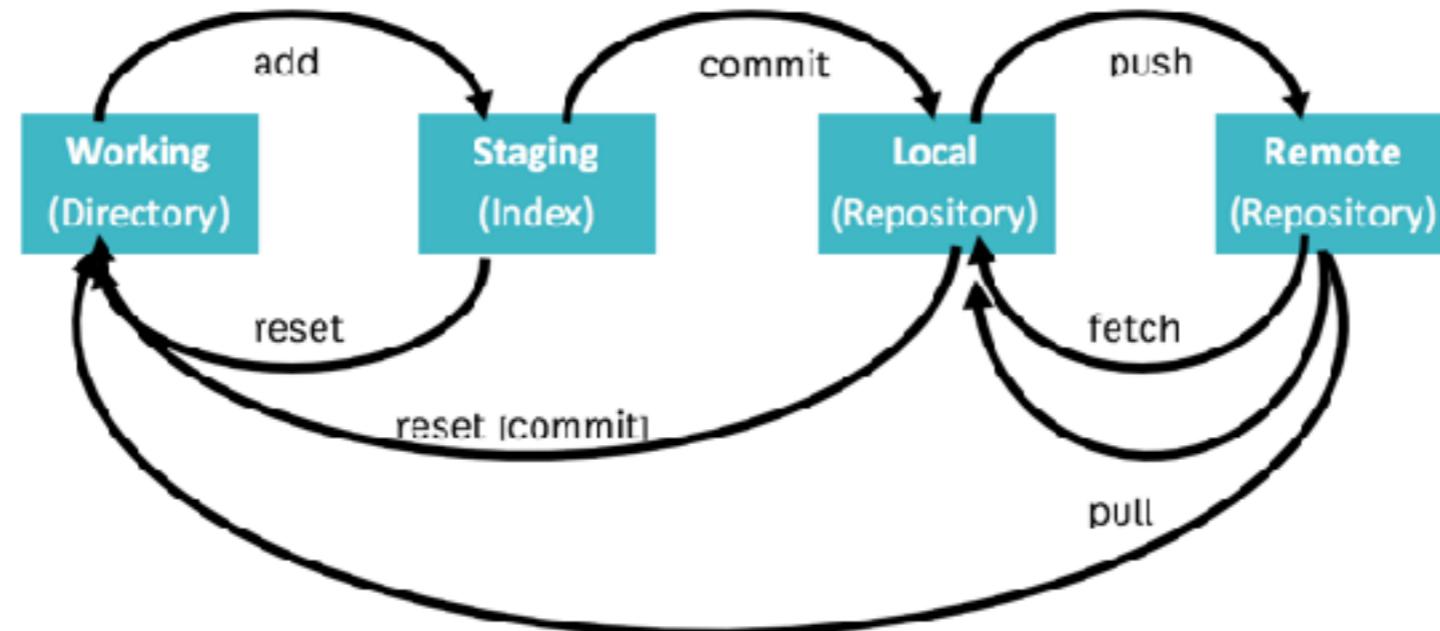
**Share** — copy and redistribute the material in any medium or format.

**Adapt** — remix, transform, and build upon the material.

**NonCommercial** — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# cheatsheet



## CONFIGURE TOOLING

กำหนดข้อมูลสำหรับใช้ใน local repositories

```
$ git config --global user.name "[name]"
```

ตั้งค่าชื่อที่ใช้แสดงผลในการ commit

```
$ git config --global user.email "[email address]"
```

ตั้งค่าอีเมลที่ใช้แสดงผลในการ commit

```
$ git config --global color.ui auto
```

เปิดใช้งานตัวอักษรสีในชุดคำสั่งต่าง ๆ เพื่อจ่ายต่อการ

## CREATE REPOSITORIES

เริ่มต้นสร้าง repository หรือนำเข้าข้อมูลจาก repository URL อื่นๆ

```
$ git init [project-name]
```

สร้างและกำหนดชื่อ local repository

```
$ git clone [url]
```

ดาวน์โหลดข้อมูลจาก repository URL และประวัติการควบคุมการเปลี่ยนแปลง (version history)



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# cheatsheet

## SYNCHRONIZE CHANGES

การปรับ local repository ให้เท่ากับ remote repository

### \$ git fetch [nickname]

ดาวน์โหลดทุกรายการประวัติของเวอร์ชันจาก remote repository สู่ local repository

### \$ git merge [nickname]/[branch]

รวม remote repository จาก branch ที่กำหนดเข้าสู่ local repository

### \$ git push [nickname] [branch]

อัพโหลดข้อมูลจาก local branch เข้าสู่ remote branch

### \$ git pull

ดาวน์โหลดประวัติของ remote repository และการเปลี่ยนแปลงต่าง ๆ เข้าสู่ local repository

## MAKE CHANGES

ตรวจสอบการแก้ไขข้อมูลและ commit

### \$ git status

แสดงสถานะการเปลี่ยนแปลง ณ ปัจจุบัน

### \$ git add [file]

บันทึกไฟล์ข้อมูลเข้าสู่ index เพื่อเตรียมทำเวอร์ชัน

### \$ git reset [file]

ยกเลิกการเปลี่ยนแปลงไฟล์

### \$ git diff

แสดงความแตกต่างของไฟล์ที่ยังไม่ได้ทำ index

### \$ git diff --staged

แสดงความแตกต่างของไฟล์ที่ได้ทำ index แล้ว กับไฟล์ที่ได้แก้ไขครั้งล่าสุด

### \$ git commit -m "[descriptive message]"

บันทึกไฟล์ลงประวัติการแก้ไข



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# cheatsheet

## REVIEW HISTORY

เรียกดูและตรวจสอบวิวัฒนาการของข้อมูล

### \$ git log

แสดงประวัติเวอร์ชันทั้งหมดของ branch ที่กำลังใช้งาน

### \$ git log --follow [file]

แสดงประวัติเวอร์ชันทั้งหมดของไฟล์และการเปลี่ยนแปลงชื่อไฟล์ด้วย

### \$ git diff [first-branch]...[second-branch]

แสดงความแตกต่างของข้อมูลระหว่าง 2 branches

### \$ git show [commit]

แสดง metadata และการเปลี่ยนข้อมูลของ

## REDO COMMITS

ลบข้อผิดพลาด และประวัติการบันทึก

### \$ git reset [commit]

ยกเลิกการ commit ทั้งหมดจนถึง commit ที่ระบุไว้, แต่ยังคงการเปลี่ยนแปลงของข้อมูลใน index

### \$ git reset --hard [commit]

ยกเลิกการ commit ทั้งหมดจนถึง commit ที่ระบุไว้



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# cheatsheet

## GROUP CHANGES

ชุดของการ commits ข้อมูล และการรวมชุดต่าง ๆ เข้าด้วยกัน

### \$ git branch

แสดงรายการทั้งหมดของ local branches ใน repository ที่กำลังใช้งานอยู่

### \$ git branch [branch-name]

สร้าง branch ใหม่

### \$ git checkout [branch-name]

สลับการใช้งาน branch และปรับปรุงข้อมูลของ working directory

### \$ git merge [branch]

รวบรวม branch ที่กำหนด เข้ากับ branch ที่กำลังใช้งาน

## REFACTOR FILENAMES

เปลี่ยนแปลงไฟล์ในเวอร์ชันที่ใช้งาน

### \$ git rm [file]

ลบไฟล์จาก working directory และ index

### \$ git rm --cached [file]

ลบไฟล์ออกจาก Git แต่ไม่ลบออกจาก working directory

### \$ git mv [file-original] [file-renamed]

เปลี่ยนชื่อไฟล์



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



Shu Ha Ri



SEAL



We Love Bug

