

Ayudantía 1

Ivania Donoso - Antonio Ossa



Ayudantes del curso

Ayudantes del curso

- Irán a las clases a ayudarlos con las actividades
- Les responden las dudas a través de github, como issues

Agenda

- Presentación de git
- Setup
- Ejercicios
 - Subir archivos
 - Solucionar problema típico
 - Trabajar en grupos
 - Trabajar con ramas

¿Qué es git?

Git es un sistema distribuido de versión de control, gratuito y open source, diseñado para manejar de pequeños a enormes proyectos de forma rápida y eficiente ¹



¹ <https://git-scm.com/>

¿Qué es GitHub?

Es una plataforma para alojar proyectos usando el sistema de control de versiones git



Ventajas

- Trabajo en equipo fluido (No hay problemas como en Dropbox)
- Versiones disponibles en cualquier momento
- Control de cambios
- Programar versiones en paralelo y luego juntarlas
- Múltiples backup de sus programas

**Se usa en la vida
real. Es
obligatorio
conocerlo :)**

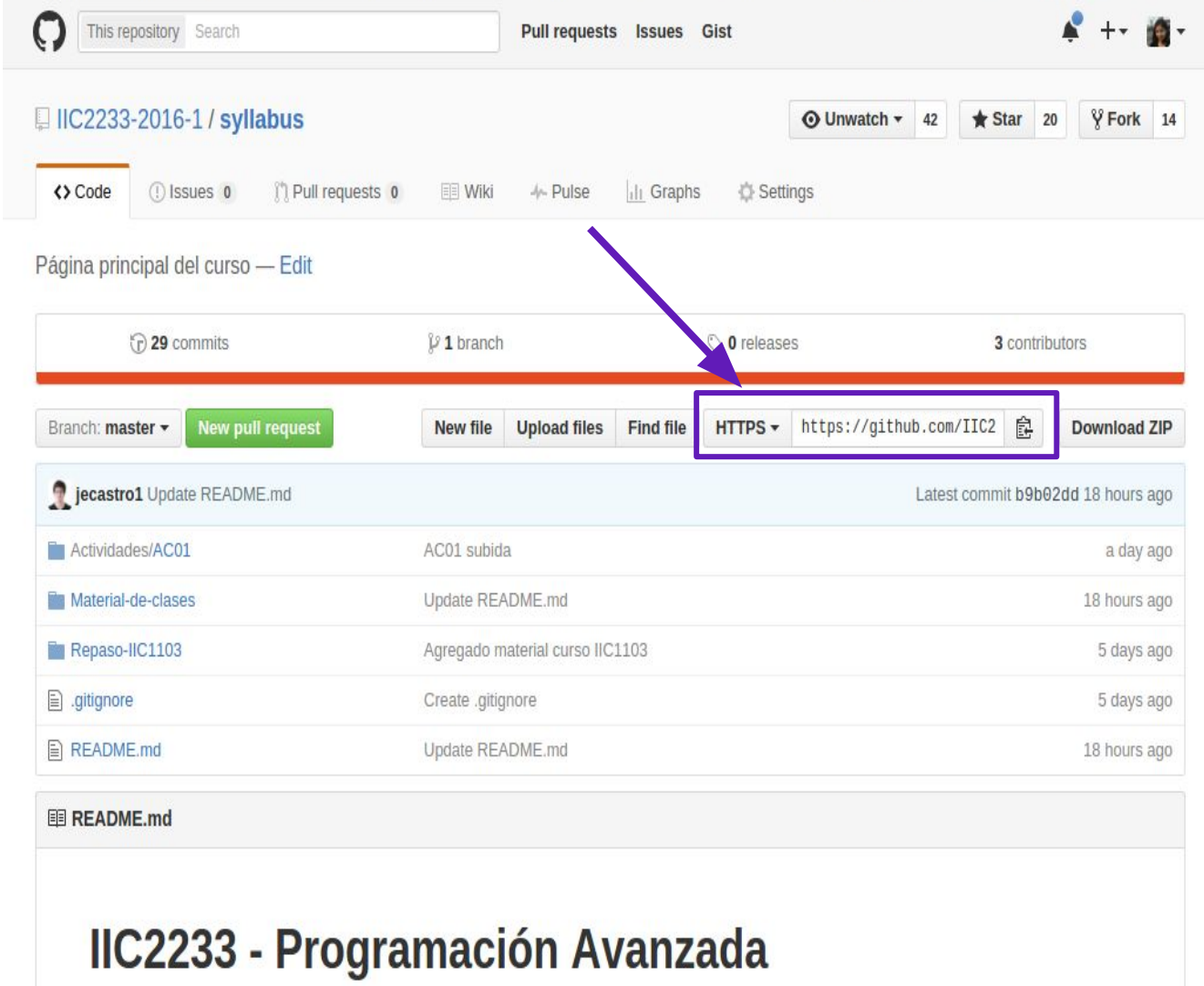
Conceptos

1. **Working directory:** lugar en donde están los archivos que luego serán parte del repositorio
2. **Repositorio local (o repo):** lugar que contiene todos los archivos que han sido “commiteados”
3. **Repositorio remoto:** lugar que contiene todos los archivos que han sido “commiteados” en un servidor de GitHub

Setup



Obtener dirección



This repository Search

Pull requests Issues Gist

IIC2233-2016-1 / syllabus

Unwatch 42 Star 20 Fork 14

Code Issues 0 Pull requests 0 Wiki Pulse Graphs Settings

Página principal del curso — Edit

29 commits 1 branch 0 releases 3 contributors

Branch: master New pull request

New file Upload files Find file HTTPS https://github.com/IIC2 Download ZIP

jecastro1 Update README.md Latest commit b9b02dd 18 hours ago

Actividades/AC01	AC01 subida	a day ago
Material-de-clases	Update README.md	18 hours ago
Repaso-IIC1103	Agregado material curso IIC1103	5 days ago
.gitignore	Create .gitignore	5 days ago
README.md	Update README.md	18 hours ago

README.md

IIC2233 - Programación Avanzada

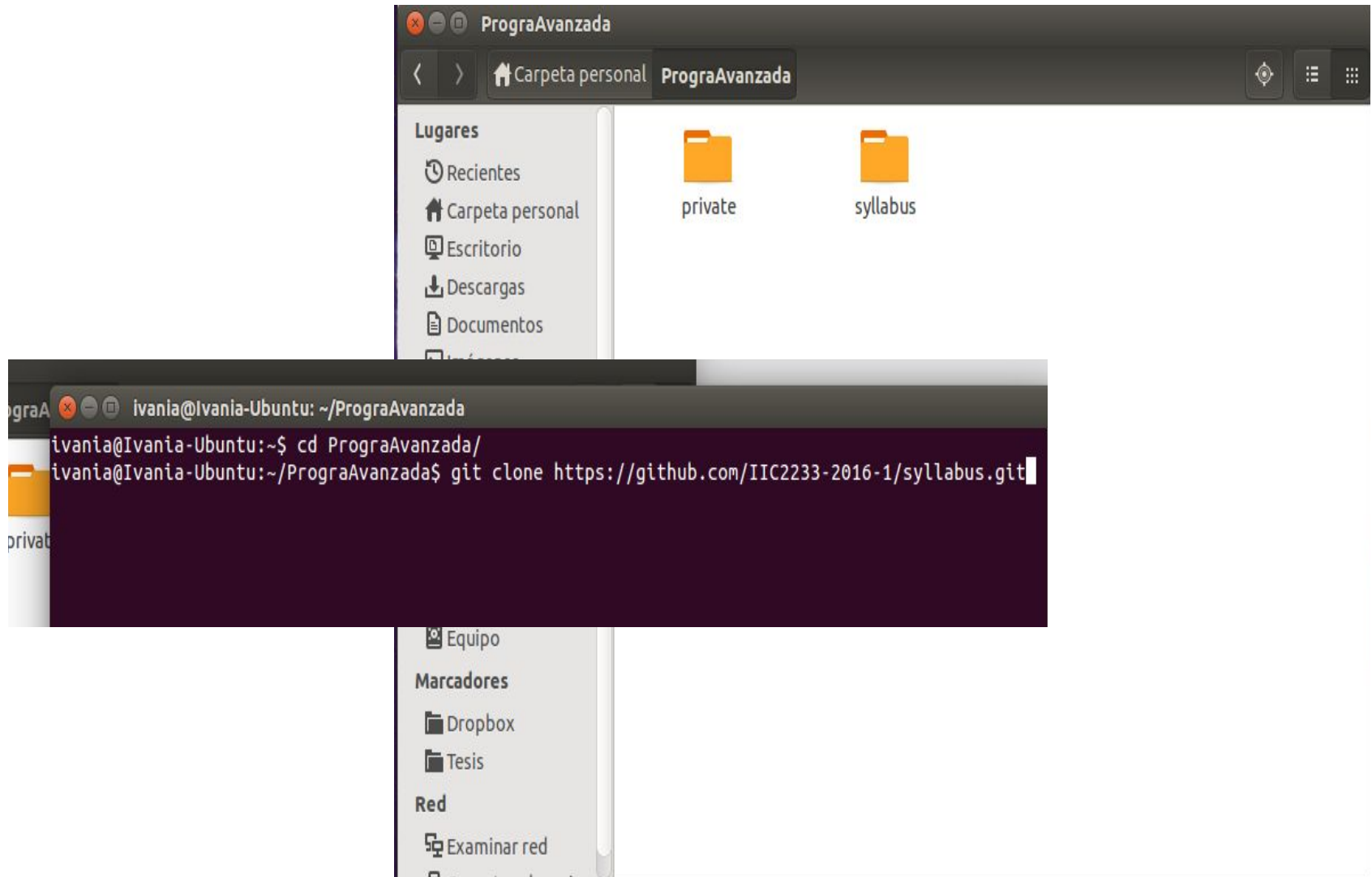
Clonar el repositorio

En la consola escribir

```
git clone https://github.com/IIC2233-2016-1/syllabus.git
```

Recuerda estar en la carpeta en la que quieren mantener el repo

Clonar el repositorio

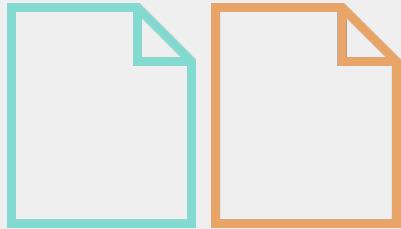


Clonen sus repositorios

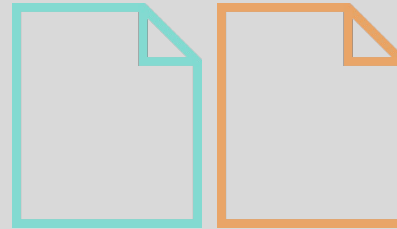
Ejercicio

Vayan al sitio de github, obtengan la dirección y clonen sus repos en sus computadores

Mi working directory



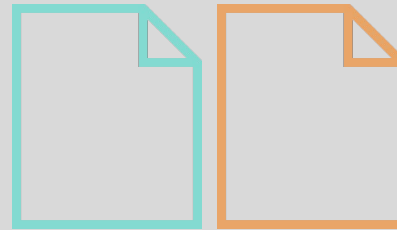
Repositorio remoto



Mi working directory

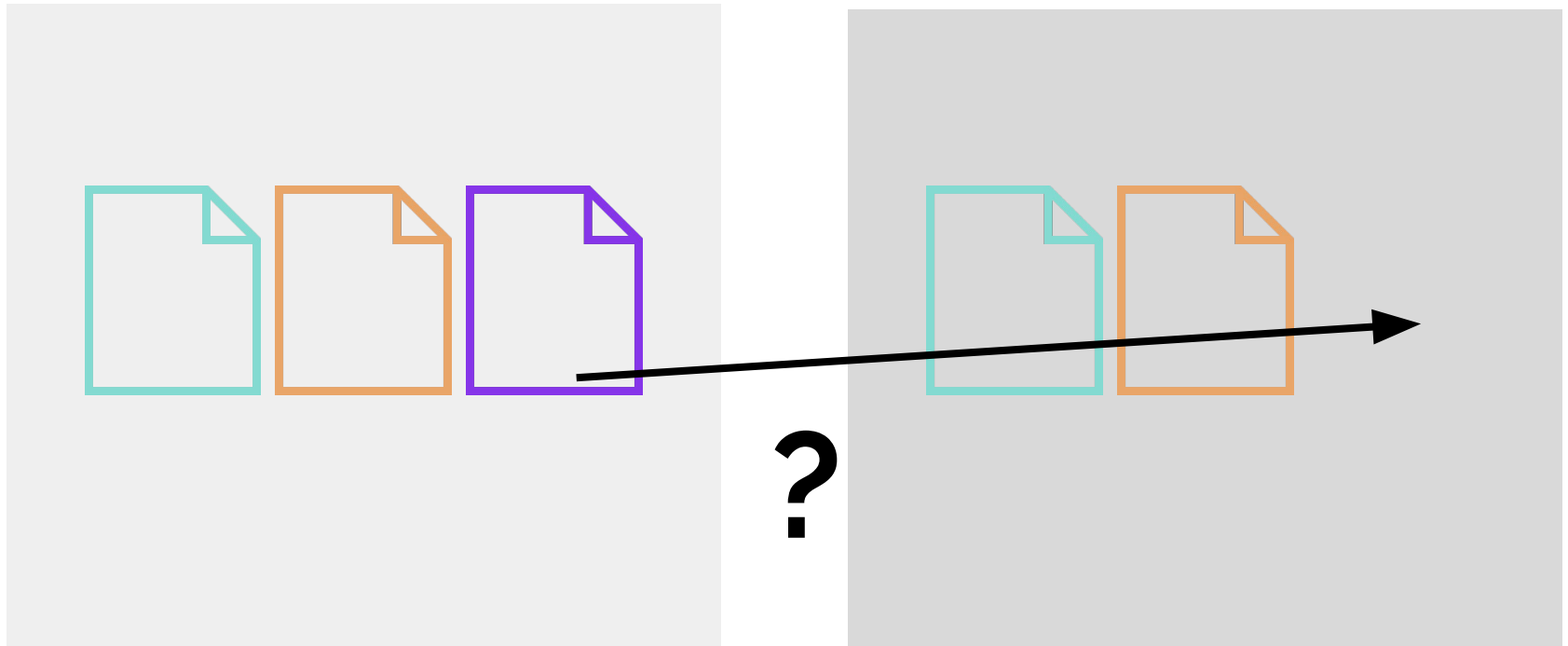


Repositorio remoto

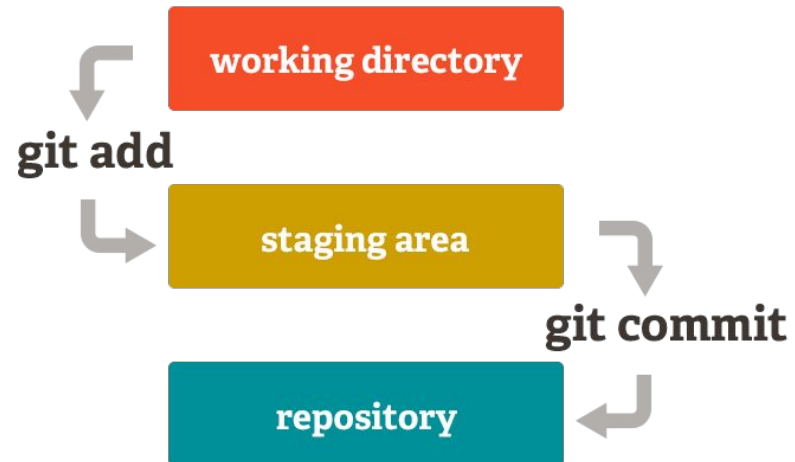


Mi working directory

Repositorio remoto



Subir un archivo

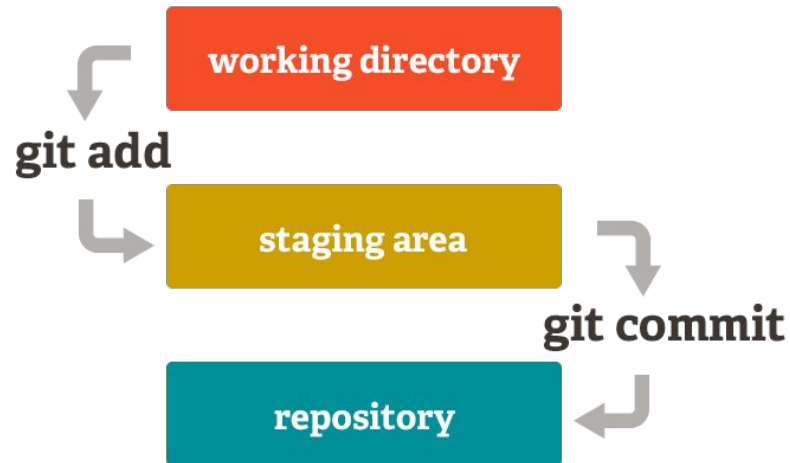


Subir un archivo

`git add --all`

`git add file_name`

`git add *.txt`



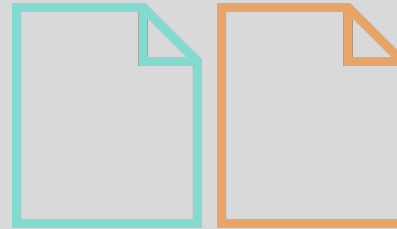
`git commit -m`
"Mensaje descriptivo"

Subir un archivo

**Mi working directory y mi
repositorio local**



Repositorio remoto

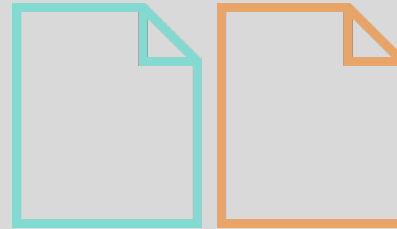


Subir un archivo

Mi working directory y mi
repositorio local



Repositorio remoto



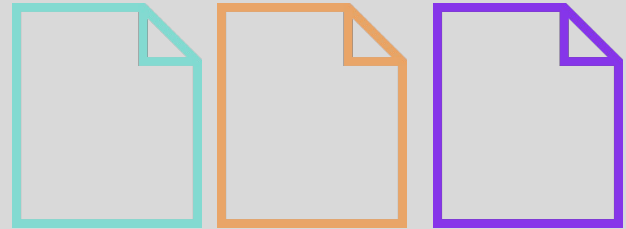
git push

Subir un archivo

**Mi working directory y mi
repositorio local**



Repositorio remoto



Subir un archivo

Ejercicio

1. Crea un archivo llamado hola.txt
2. Agrégalo al staging area
3. Ejecuta el comando git status.
Debería aparecer el mensaje
“Cambios para hacer commit”
4. Agrégalo al repositorio local
5. Agrégalo al repositorio remoto

Cambiar el stage

Son las 16:48. Las instrucciones dicen que no debo subir el archivo “VeryHeavyFile.txt” que pesa 100 MB. Hice `git add --all` y solo me queda un minuto para poder subir la actividad.



I'm having a panic attack.

```
git reset HEAD file_name
```

Ya hice commit



Guarden lo que deben subir en una carpeta fuera del repositorio luego escriben en la consola

```
git reset HEAD~1
```

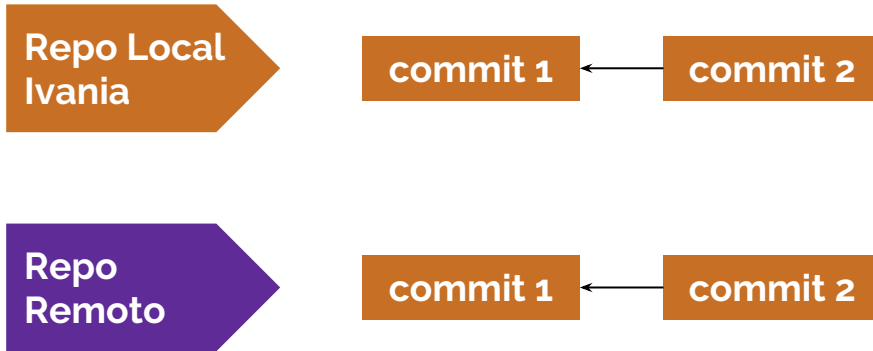
Cambiar el stage

Ejercicio

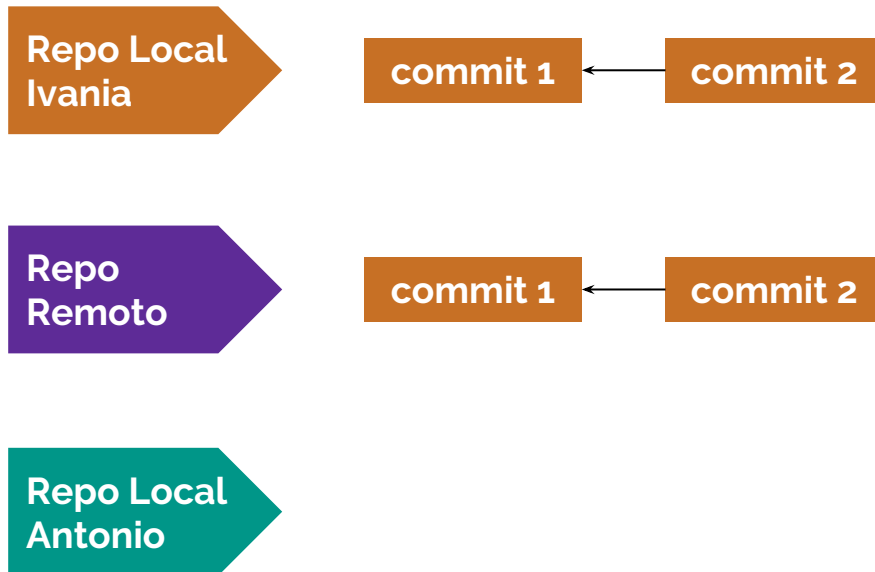
1. Crea un archivo llamado hola1.txt y agrégalo al staging area.
2. Crea un archivo llamado hola2.txt y agrégalo al staging area.
3. Elimina el archivo hola1.txt del staging area.
4. Agrega el stage al repositorio local y luego sube los cambios al repositorio remoto

Trabajar en Equipo

Ivania hizo una aplicación

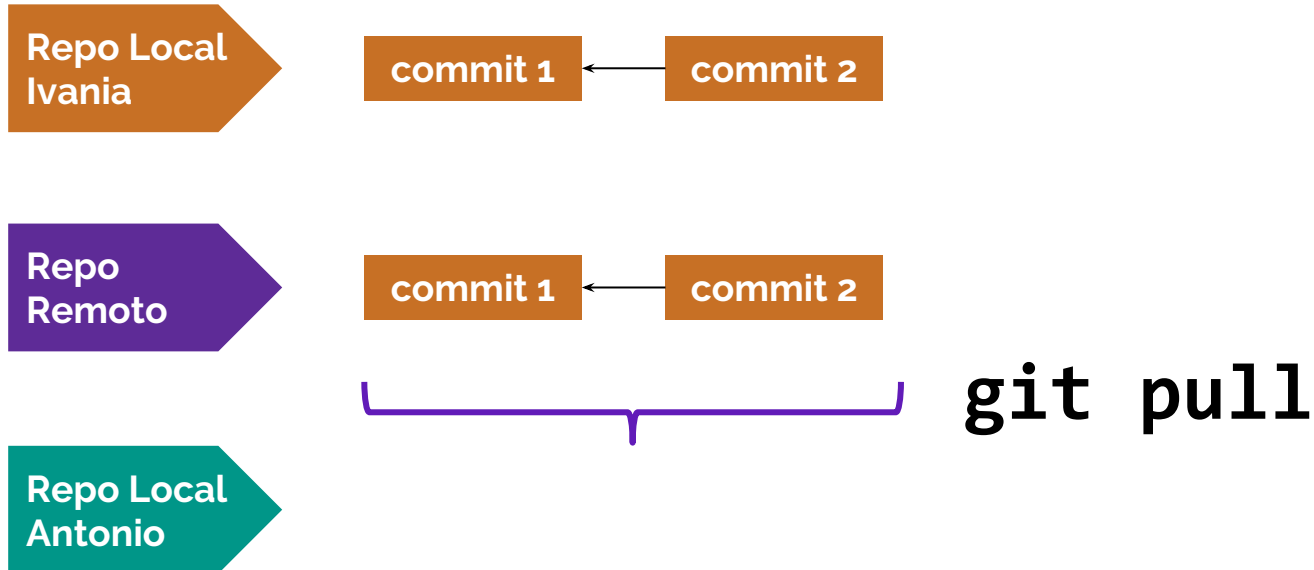


Trabajar en Equipo

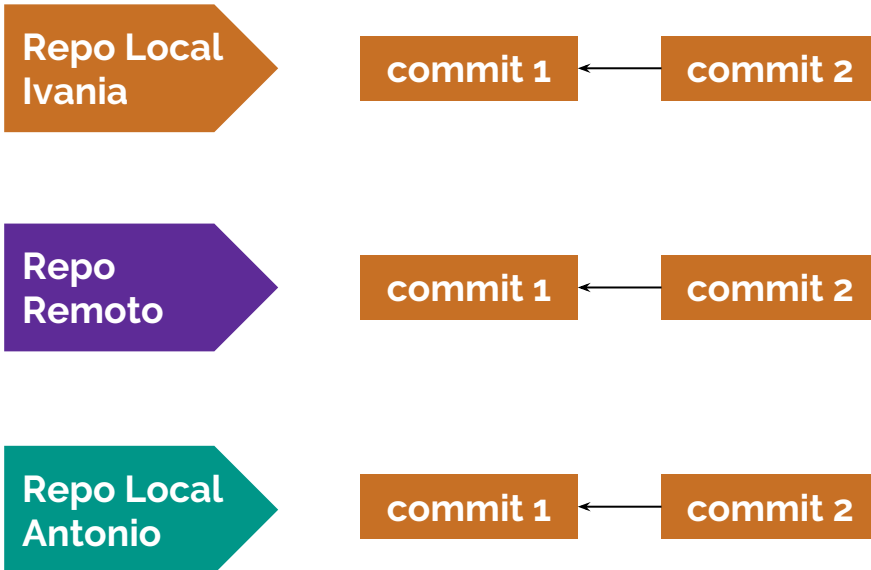


Antonio va a contribuir. ¿Cómo obtiene el código desde el repo remoto?

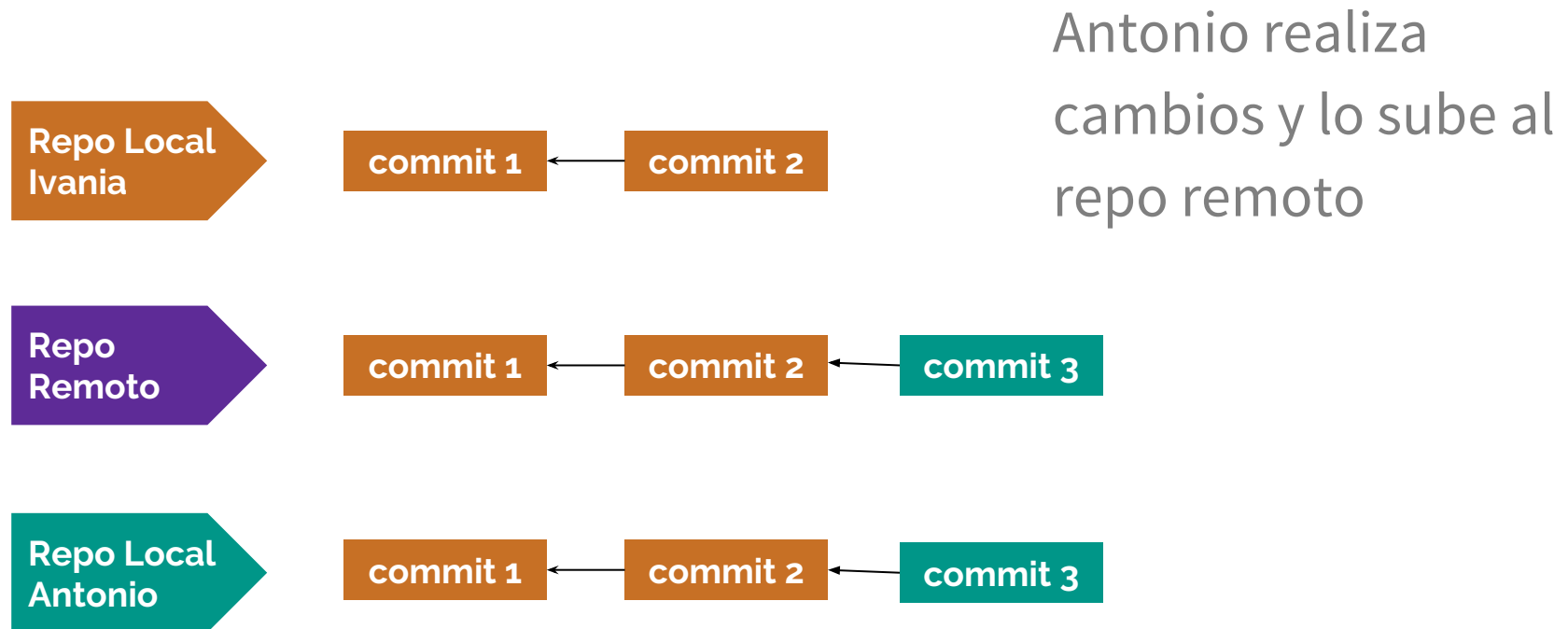
Trabajar en Equipo



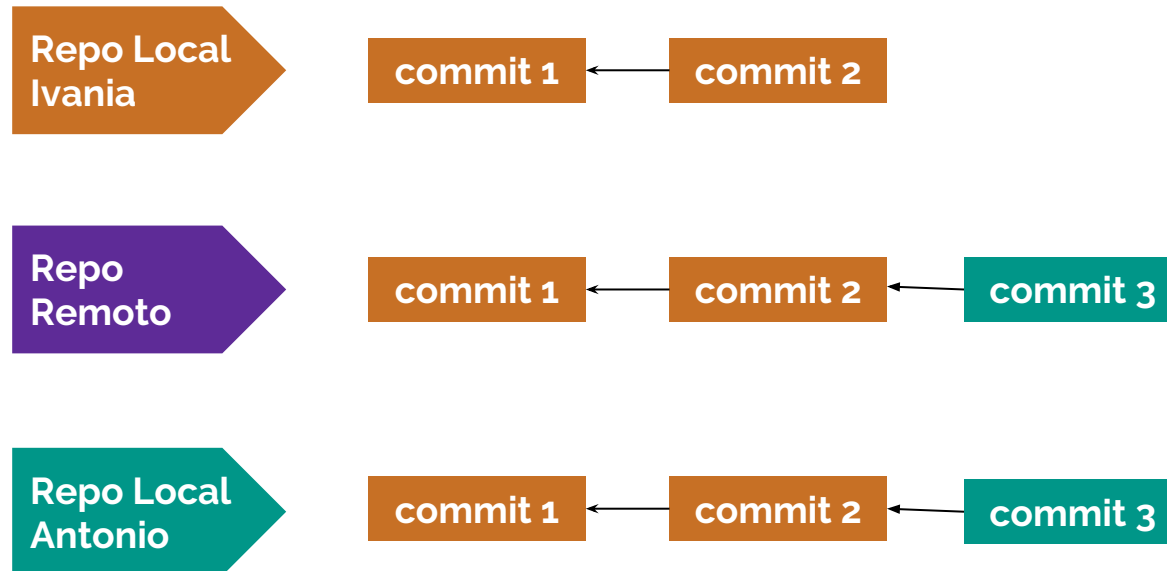
Trabajar en Equipo



Trabajar en Equipo

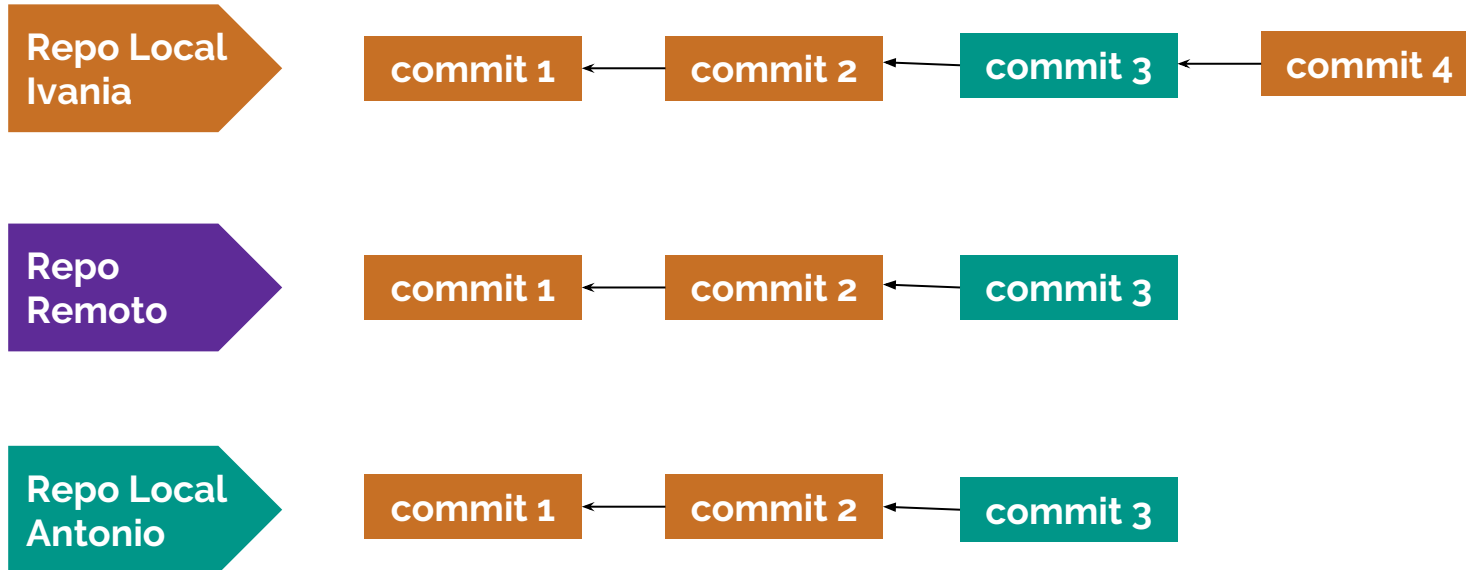


Trabajar en Equipo



Ivania quiere seguir trabajando. ¿Qué comando debe usar para obtener el trabajo que hizo Antonio?

Trabajar en Equipo



Trabajar en equipo

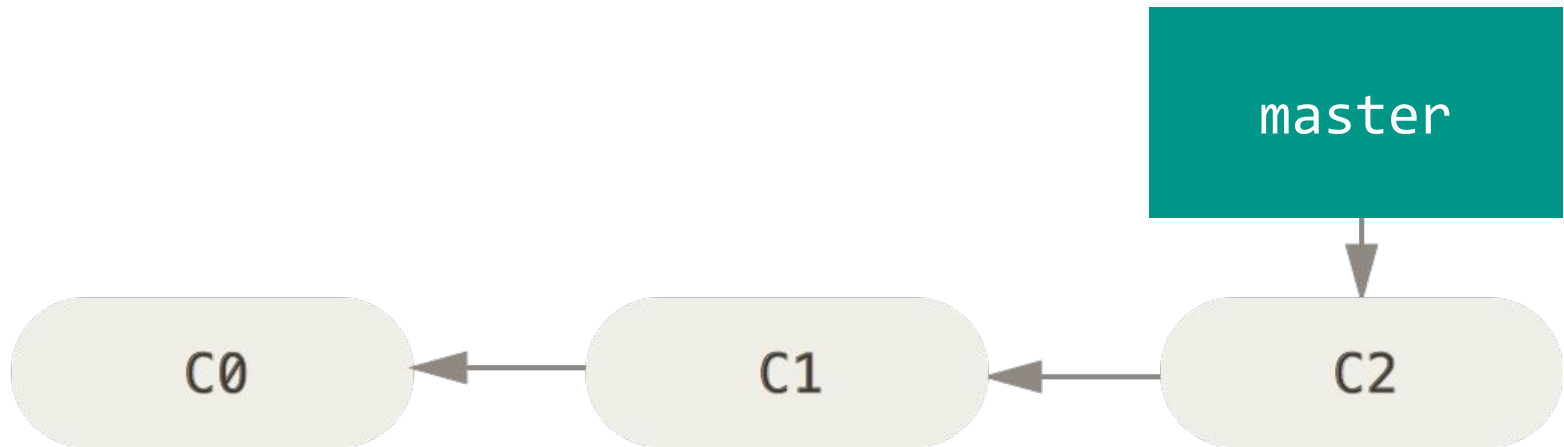
Ejercicio

1. Elimina el archivo hola2.txt del working directory
2. Revisa el archivo README.md del repositorio local. Debe decir “Todavía no has pasado la prueba de trabajo en equipo”
3. Revisa el sitio de tu repositorio. El archivo README.md debe decir “Has pasado la prueba de trabajo en equipo :)”
4. Baja los cambios al repositorio local.

Trabajar con ramas

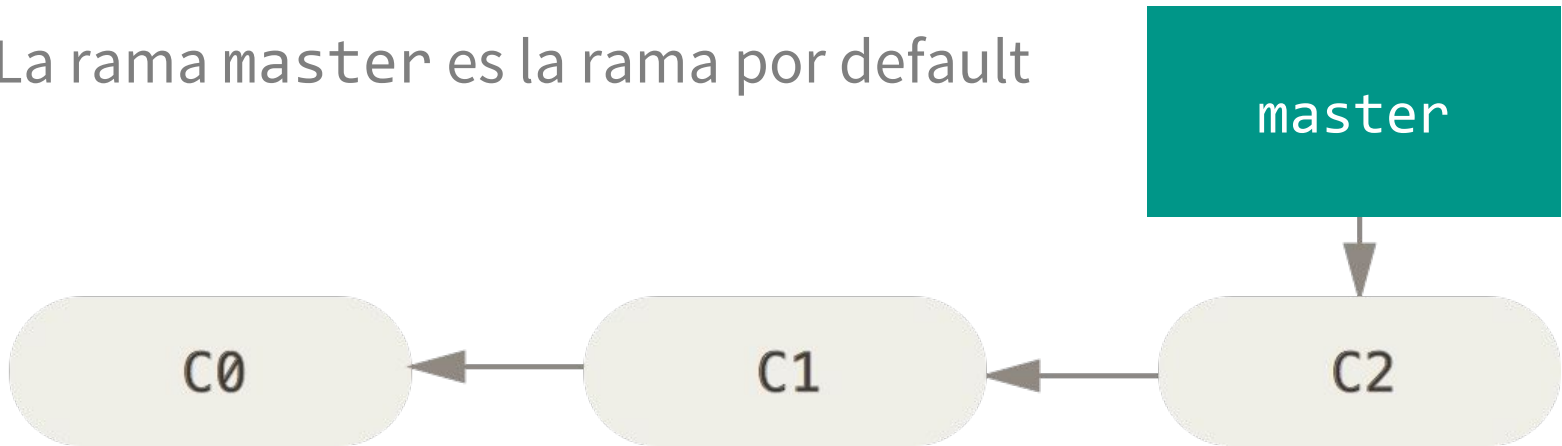
Me quedan dos días para entregar la tarea. Ya terminé todos los items para tener un 7. Esta tarea tiene un bonus, pero no sé si alcanzaré a terminarlo a tiempo. Si modifico el código que tengo hasta ahora puedo perder mi 7. ¿Qué hago? ¿Una copia en otro lado? ¿Y si me doy cuenta de un error?

Trabajar con ramas



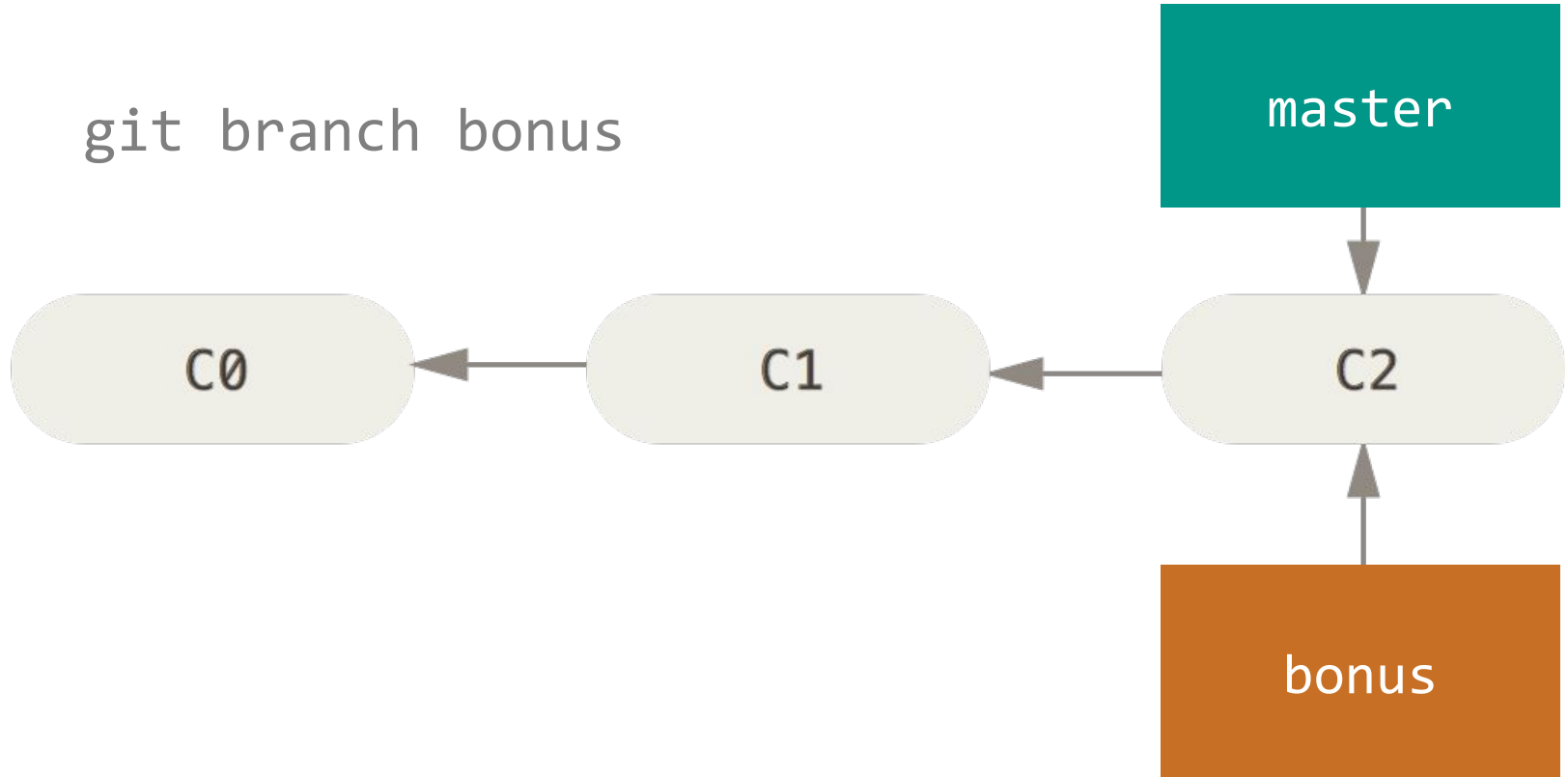
Trabajar con ramas

La rama master es la rama por default



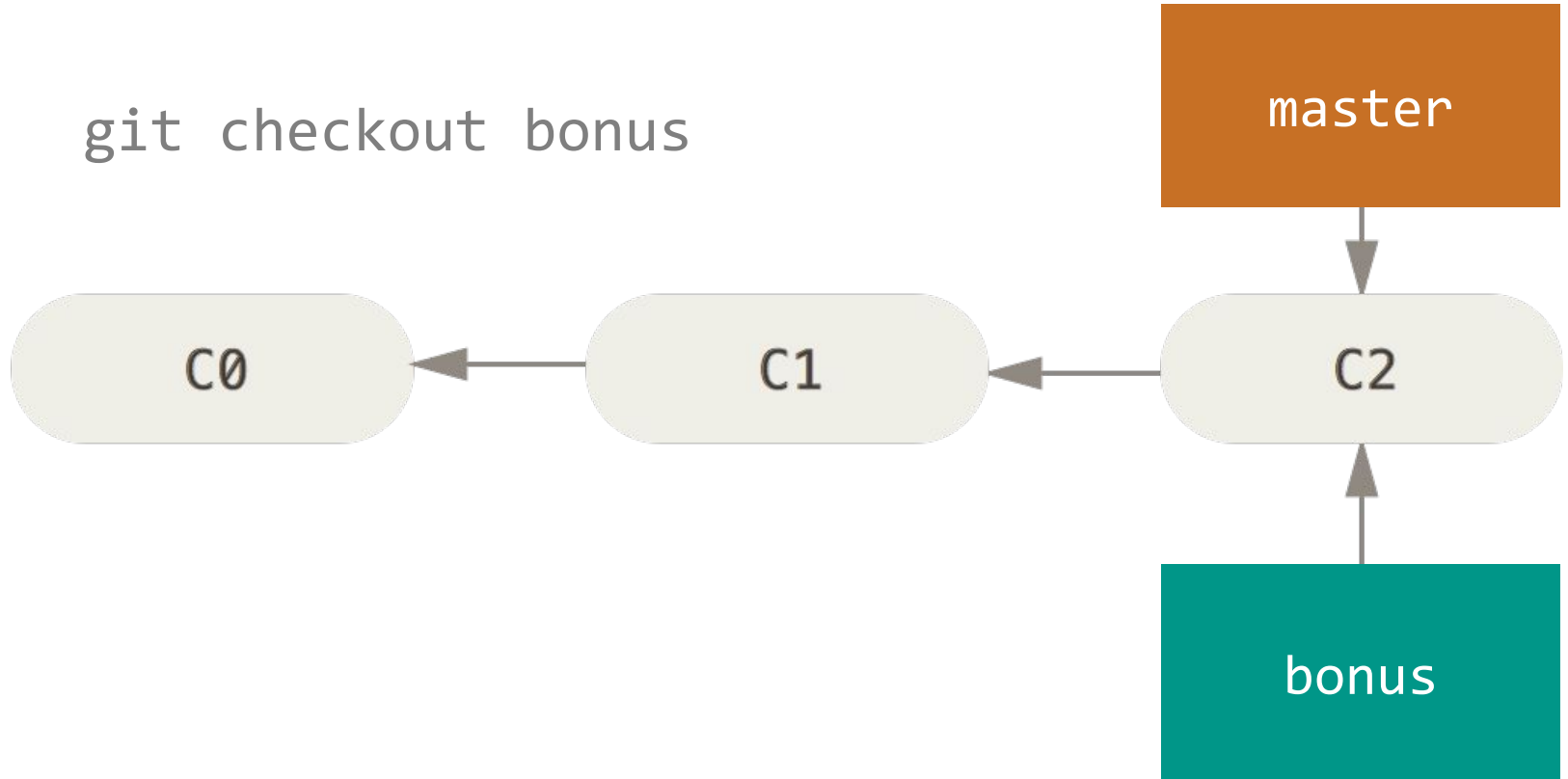
Trabajar con ramas

git branch bonus



Trabajar con ramas

git checkout bonus



Trabajar con ramas

Realizamos cambios en working directory

```
git add new_file.txt
```

```
git commit -m "Commit 3"
```

Trabajar con ramas

Realizamos cambios en working directory

```
git add new_file.txt
```

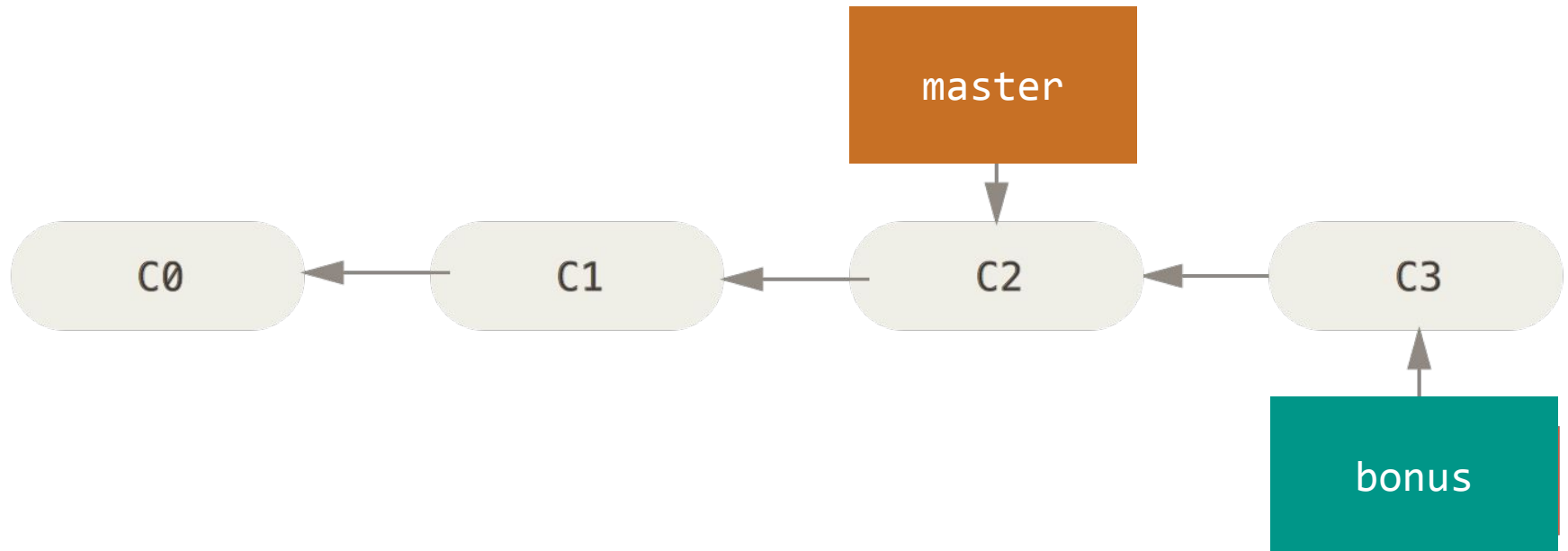
```
git commit -m "Commit 3"
```

A qué commit apunta la rama master? y la rama bonus?

Trabajar con ramas

```
git add new_file.txt
```

```
git commit -m "Commit 3"
```

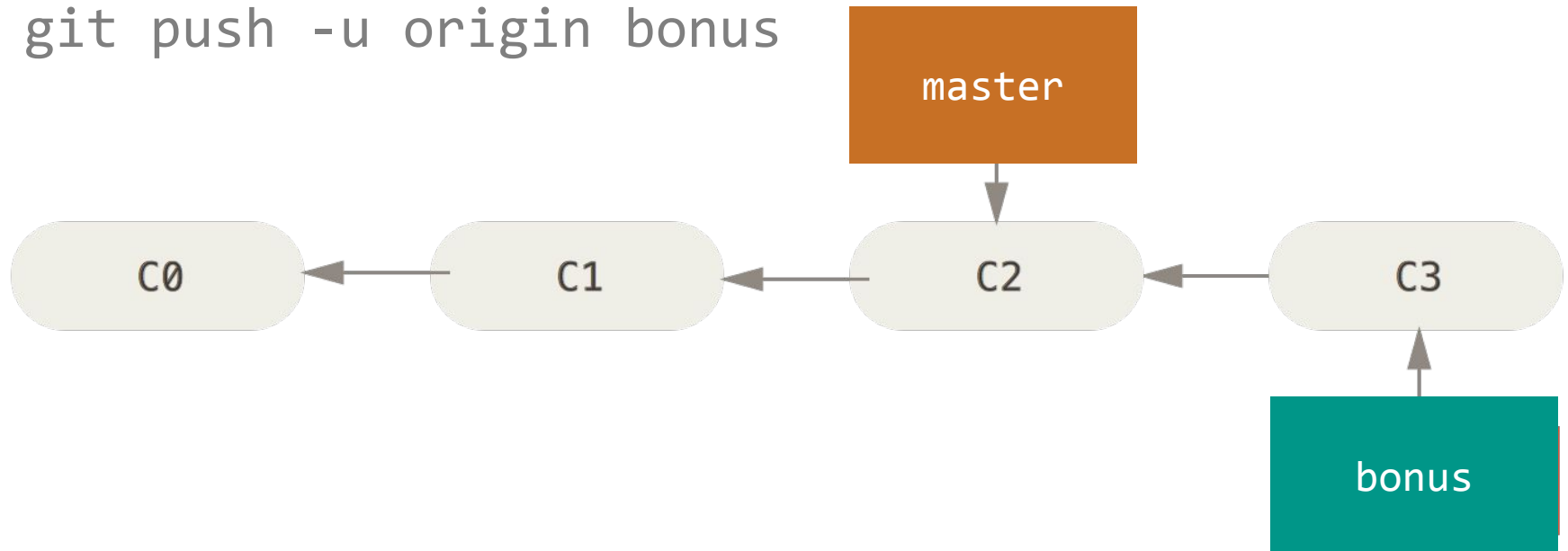


Trabajar con ramas

```
git add new_file.txt
```

```
git commit -m "Commit 3"
```

```
git push -u origin bonus
```



Trabajar con ramas

Va todo ok con el trabajo del bonus, pero te diste cuenta de que tenías un error en la versión para el 7 (la que está en la rama master)



Trabajar con ramas

```
git checkout master
```

```
git add new_file.txt
```

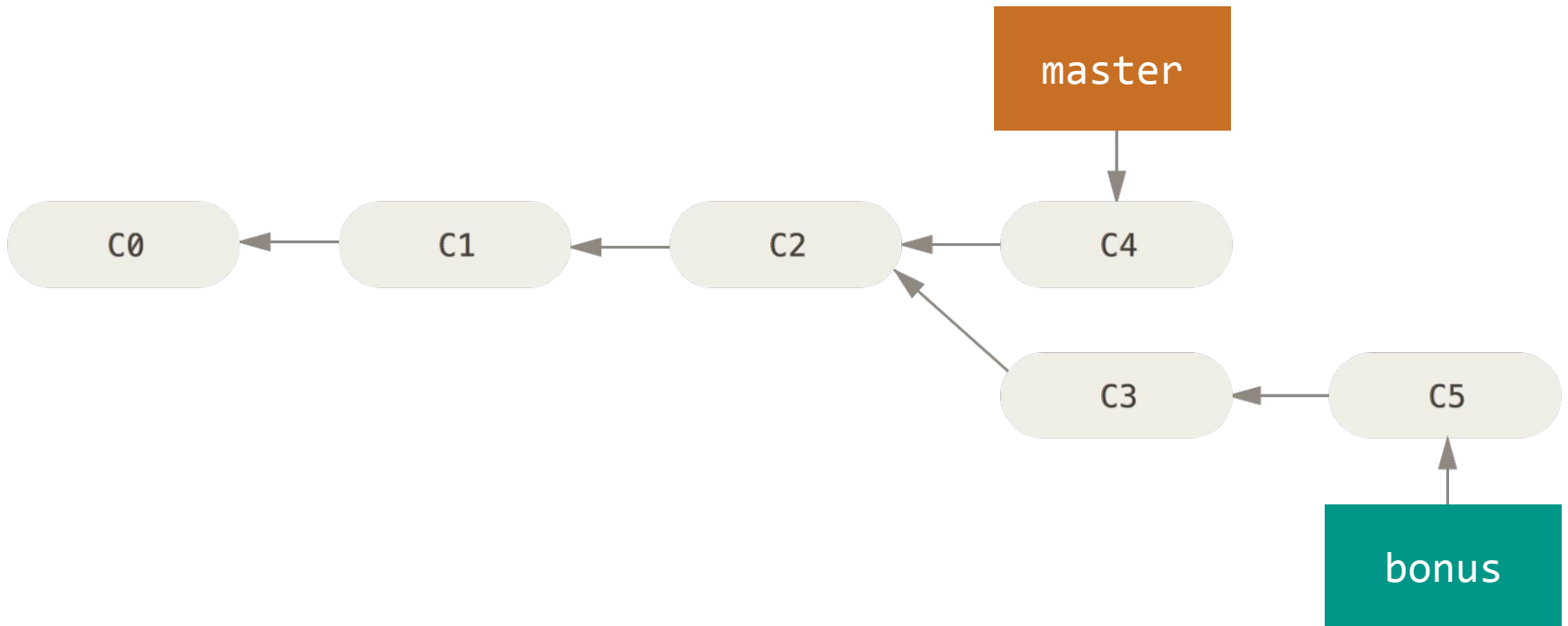
```
git commit -m "Commit 4"
```

```
git checkout bonus
```

```
git add --all
```

```
git commit -m "Commit 5"
```

Trabajar con ramas

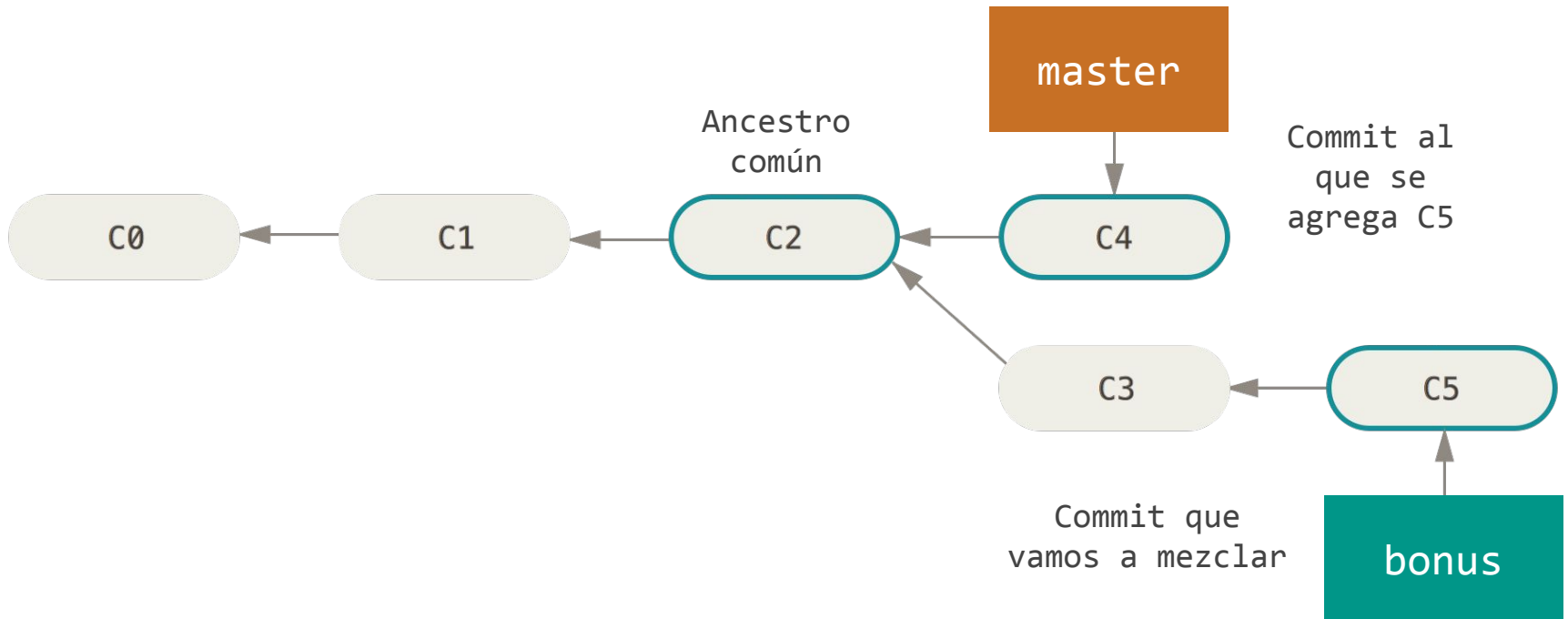


Trabajar con ramas

El bonus está listo :D

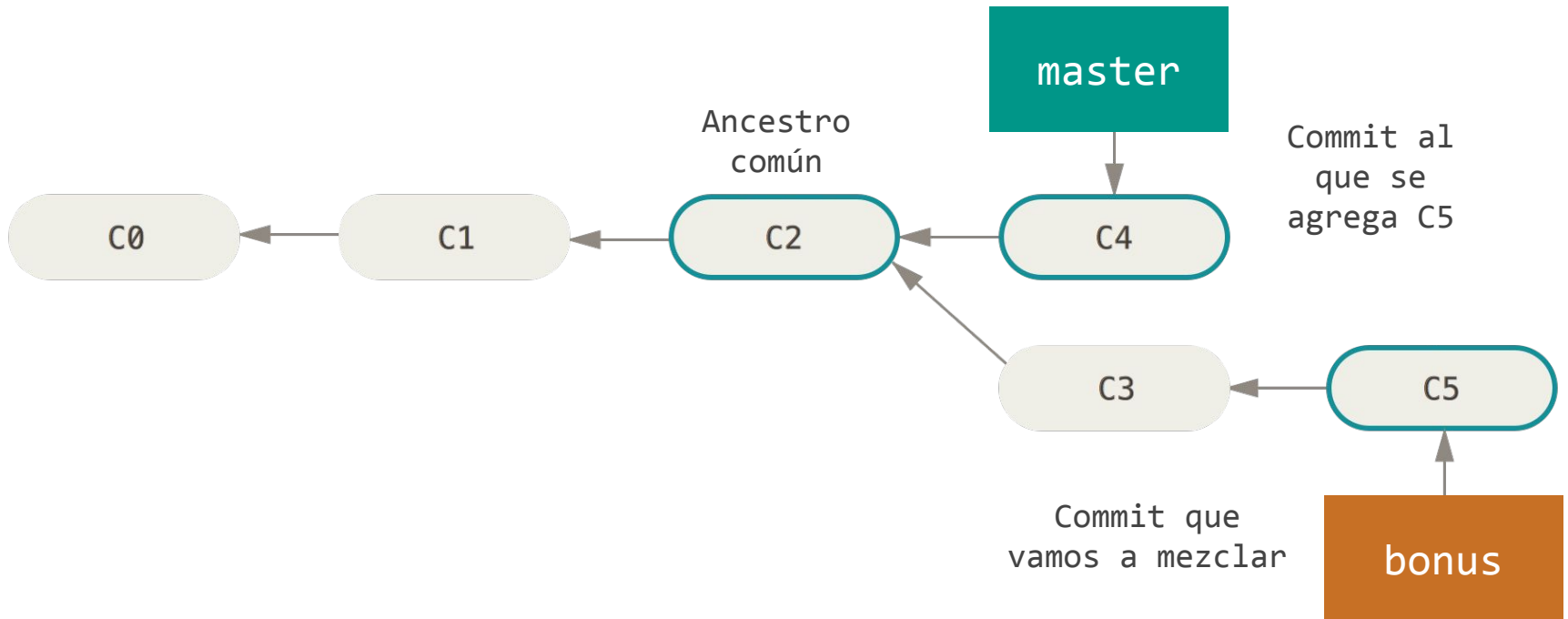
Ahora debes juntar los códigos de ambas ramas.

Trabajar con ramas



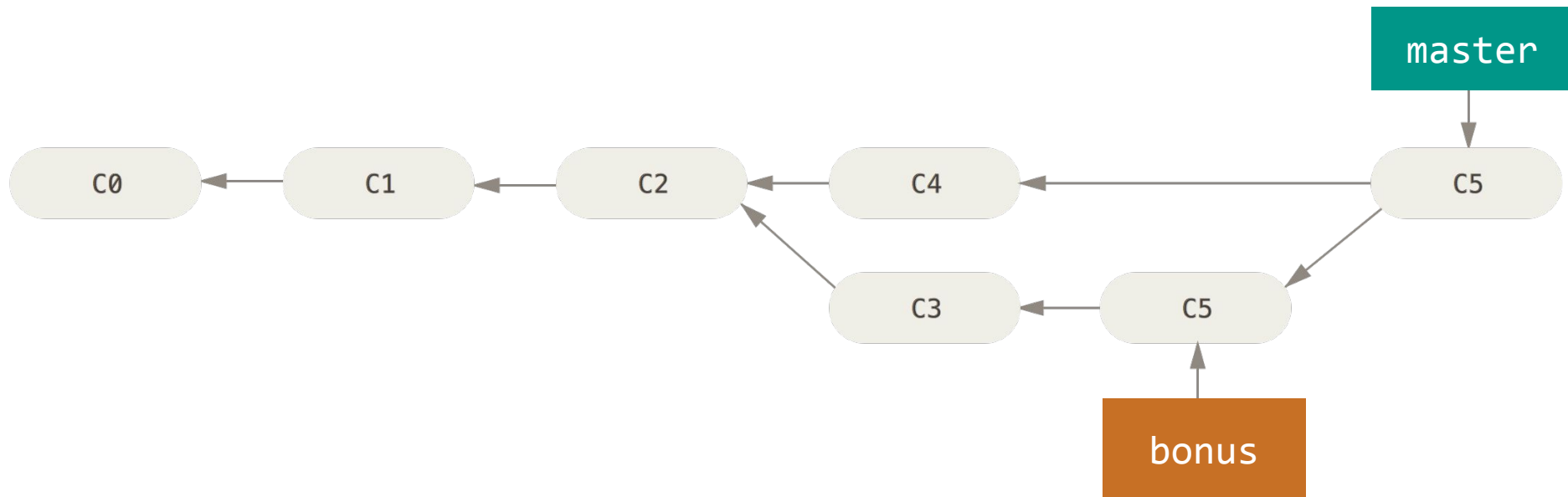
Trabajar con ramas

`git checkout master`



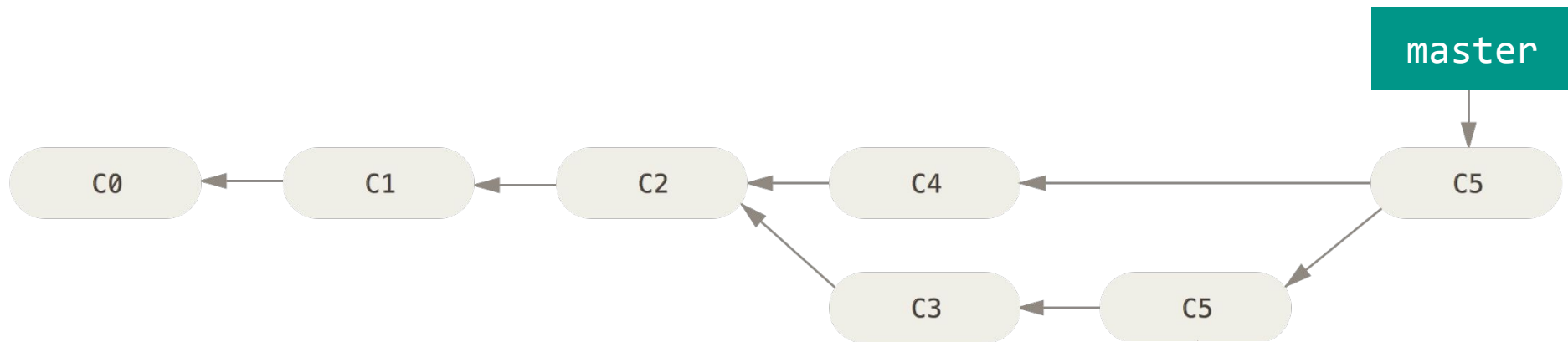
Trabajar con ramas

`git merge bonus`



Trabajar con ramas

```
git branch -d bonus
```



Crear ramas

Ejercicio

1. Crea una rama de nombre “Ayudantia”
 2. Posicionate en la rama
 3. Crea un archivo hola3.txt
 4. Agrégalo al repositorio local
 5. Agrégalo al repositorio remoto
 6. Vuelve a la rama master
 7. Crea un archivo hola4.txt
 8. Agrégalo al repositorio local
 9. Vuelve a la rama “Ayudantia”
 10. Crea un archivo hola5.txt
 11. Vuelve a la rama master
 12. Mezcla la rama master con Ayudantia.
 13. Sube todo al repo remoto
-

Sitios útiles

www.git-scm.com

<http://ndpsoftware.com/git-cheatsheet.html>

<http://rogerdudler.github.io/git-guide/>

<http://rypress.com/tutorials/git/index>

<https://www.udacity.com/courses/ud775>

Bibliografía

- www.git-scm.com
- Imagen de archivo creada por Richard Schumann desde Noun Project