

El Ataque de Git

Sesión formativa Git y GitHub



Delegación de Estudiantes

Universidad Carlos III de Madrid

Escuela Politécnica Superior

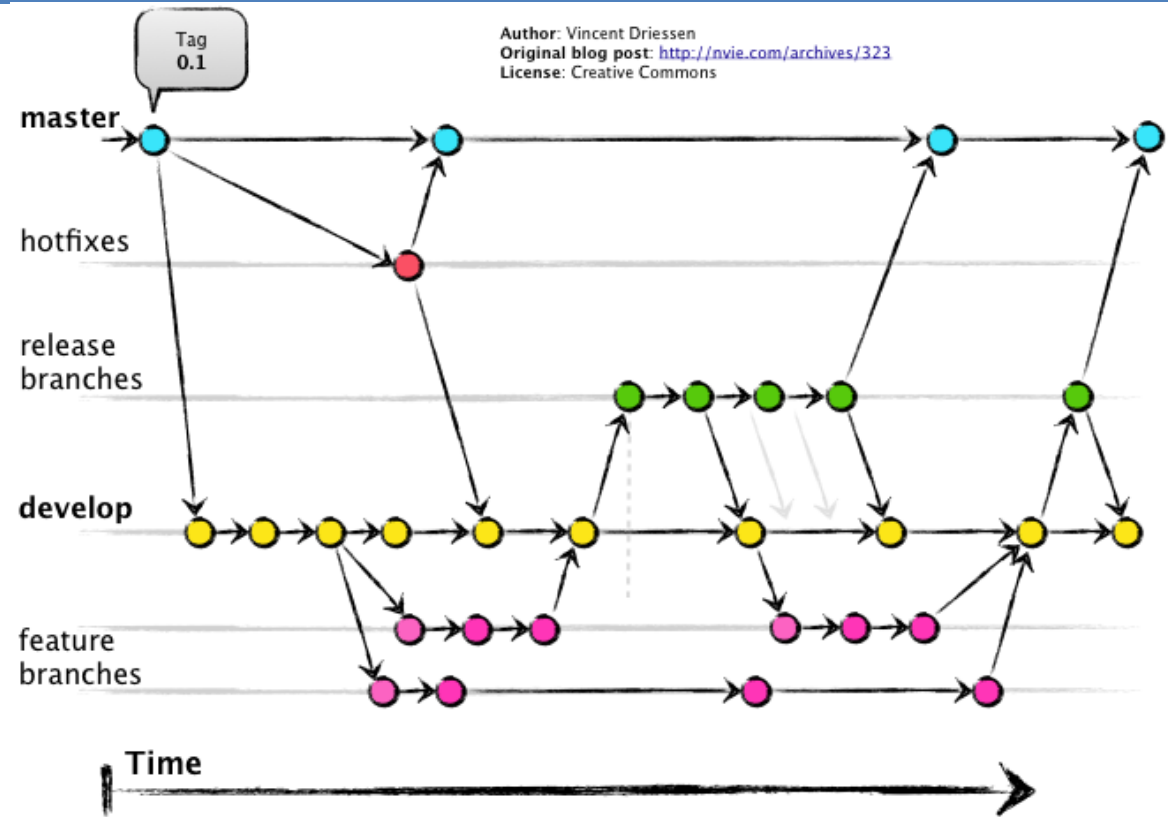
Control de Versiones

El secreto de Git

Comandos, muchos comandos...



¿Cómo afectan al grafo?



Delegación de Estudiantes

Universidad Carlos III de Madrid
Escuela Politécnica Superior

uc3m

La base

Git commit

Cada nodo del grafo anterior será un *commit*

Git almacena *capturas* que revelan los cambios en cada instante. Contiene:

- Hunks: Mensaje que describe los cambios
- Hash identificador (como un DNI)
- Referencia al commit padre



Delegación de Estudiantes
Universidad Carlos III de Madrid
Escuela Politécnica Superior

uc3m

Git diff y patch

Diferencias entre estados del proyecto

El contenido que muestra **git diff** es lo que se guardará en los hunks de un *commit*.

```
diff archivo archivoCambios > archivoSalida.patch
```

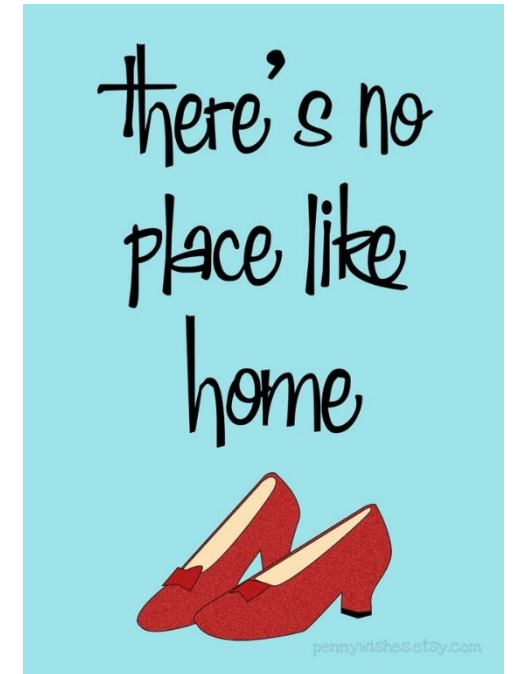
```
patch archivo < archivoSalida.patch
```

Los tres estados

Working directory

Nuestro directorio físico

Podemos añadir y modificar
archivos libremente.



Delegación de Estudiantes

Universidad Carlos III de Madrid
Escuela Politécnica Superior

uc3m

Staging area

Seleccionamos los archivos con cambios

Zona virtual de Git. Aquí moveremos los datos que queremos trackear con Git. Crearemos commits para definir los cambios del proyecto.

- **git add *file***: para pasar del Working Directory al Staging Area

Repository

Almacenamos la información

Zona donde se almacenan todos los *commits* (y muchas otras cosas) con los cambios.

- **git commit:** para pasar del Staging Area al Repository

Diferencias entre estados

Todo bajo control

- `git diff`: entre Working Directory y Staging Area
- `git diff HEAD`: entre Working Directory y Repository
- `git diff --staged`: entre Staging Area y Repository

Subida de archivos

Pero entonces, ¿qué hago?

Simple:

- `git add <file>` `git add .` `git add *.<extensión>`
- `git commit` `git commit -m <Mensaje corto>`
- `git log` Para ver la evolución de los commits (de hijos a padres)

Cuando estemos seguros de que lo que hemos subido está OK

- `git push` para volcar los archivos al repositorio remoto

¿Y si la cago?

Respira
(Casi) todo tiene solución

Es posible deshacer *adds* y *commits* antes de hacer *push* :)



Delegación de Estudiantes

Universidad Carlos III de Madrid
Escuela Politécnica Superior

uc3m

Un poco más de
teoría

Referencias

Referencia HEAD

Define el estado en el que nos encontramos. Apunta la rama en la que estés (por defecto **master**).

Referencia master

Define una referencia a la cabeza (al último commit) de la rama máster. **Toda rama tiene una ref. a su cabeza.**



Delegación de Estudiantes

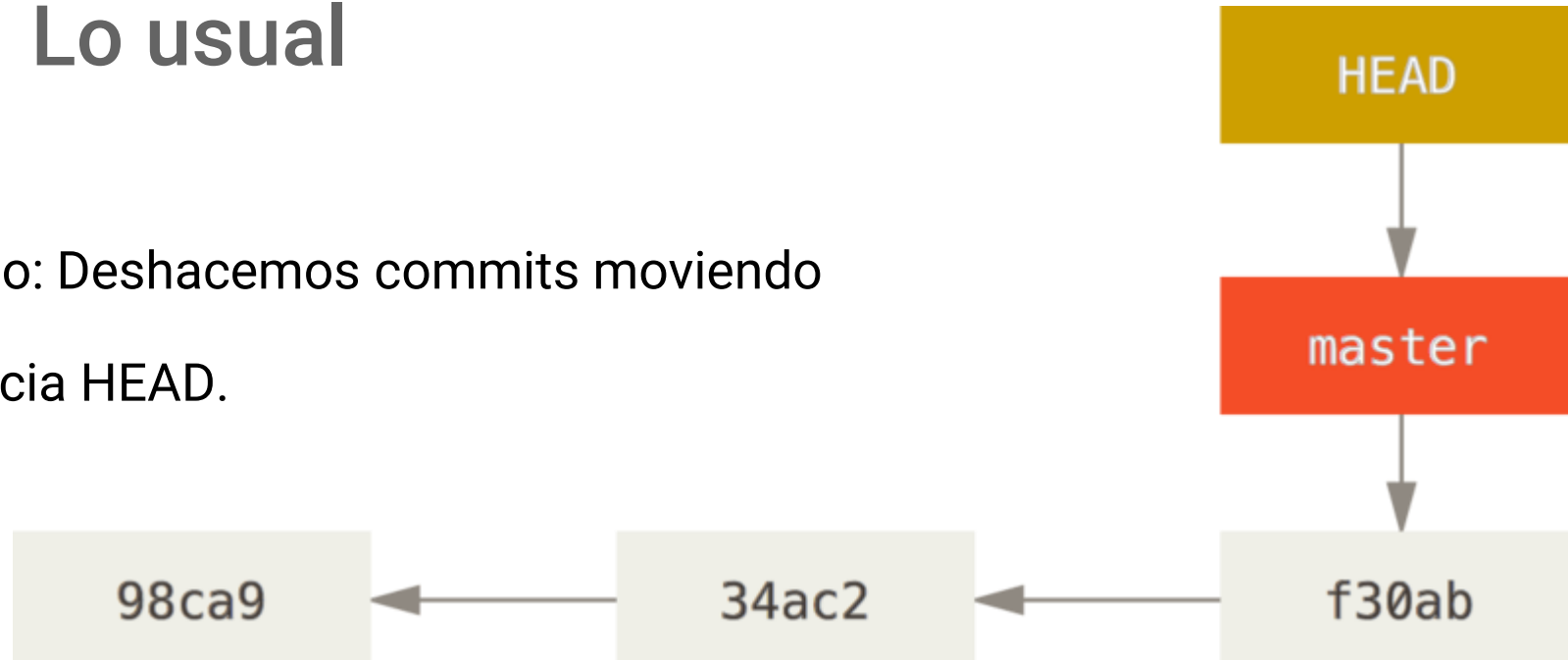
Universidad Carlos III de Madrid
Escuela Politécnica Superior

uc3m

Gráficamente

Lo usual

- * Adelanto: Deshacemos commits moviendo la referencia HEAD.



Delegación de Estudiantes

Universidad Carlos III de Madrid
Escuela Politécnica Superior

uc3m

A jugar!