

Implantació d'arquitectures web

World Wide Web

- Gestació (1980-1991)
- Eclosió de la WWW (1992-2001)
- Naiximent de les aplicacions web (2002-)

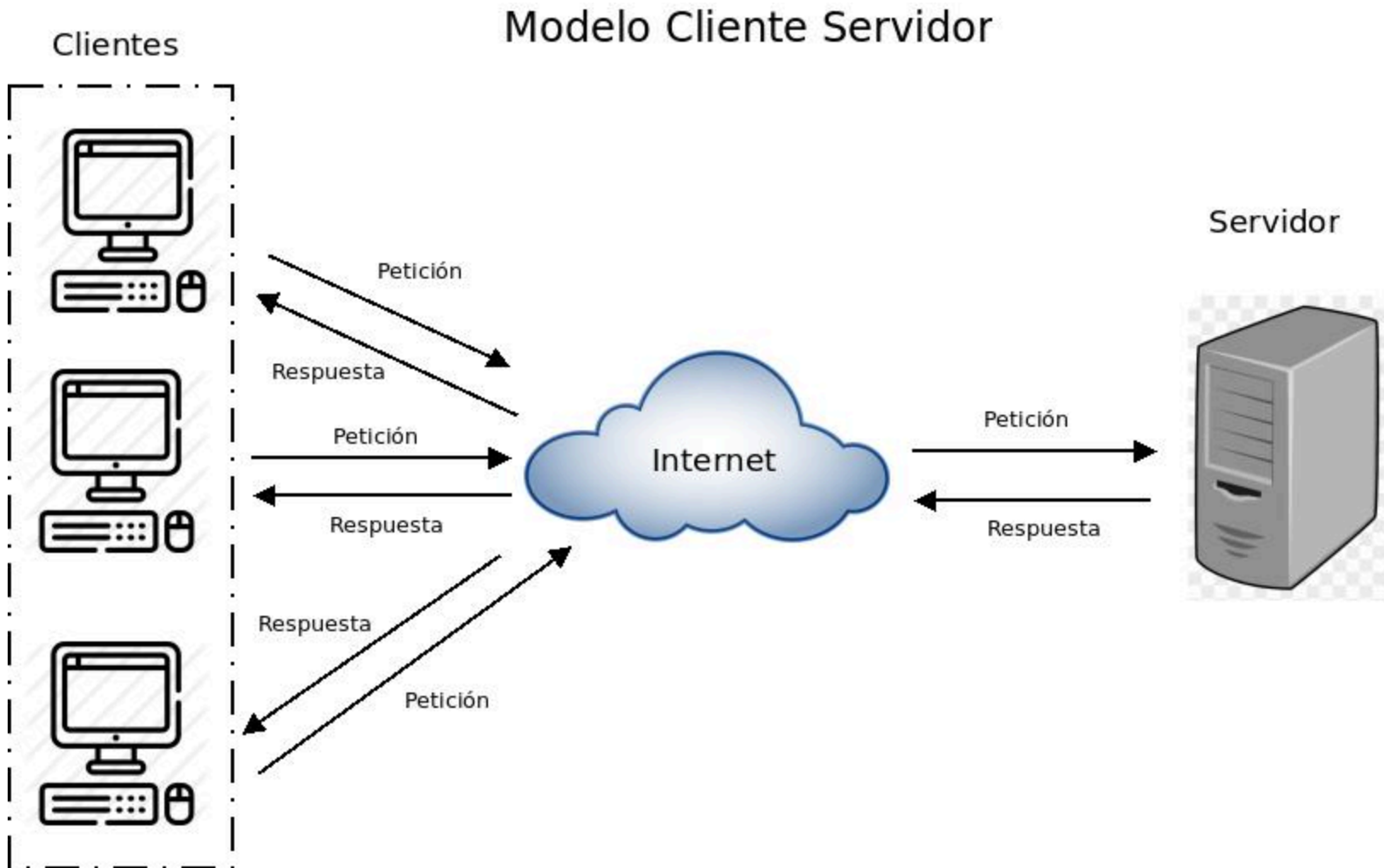
Aplicació web

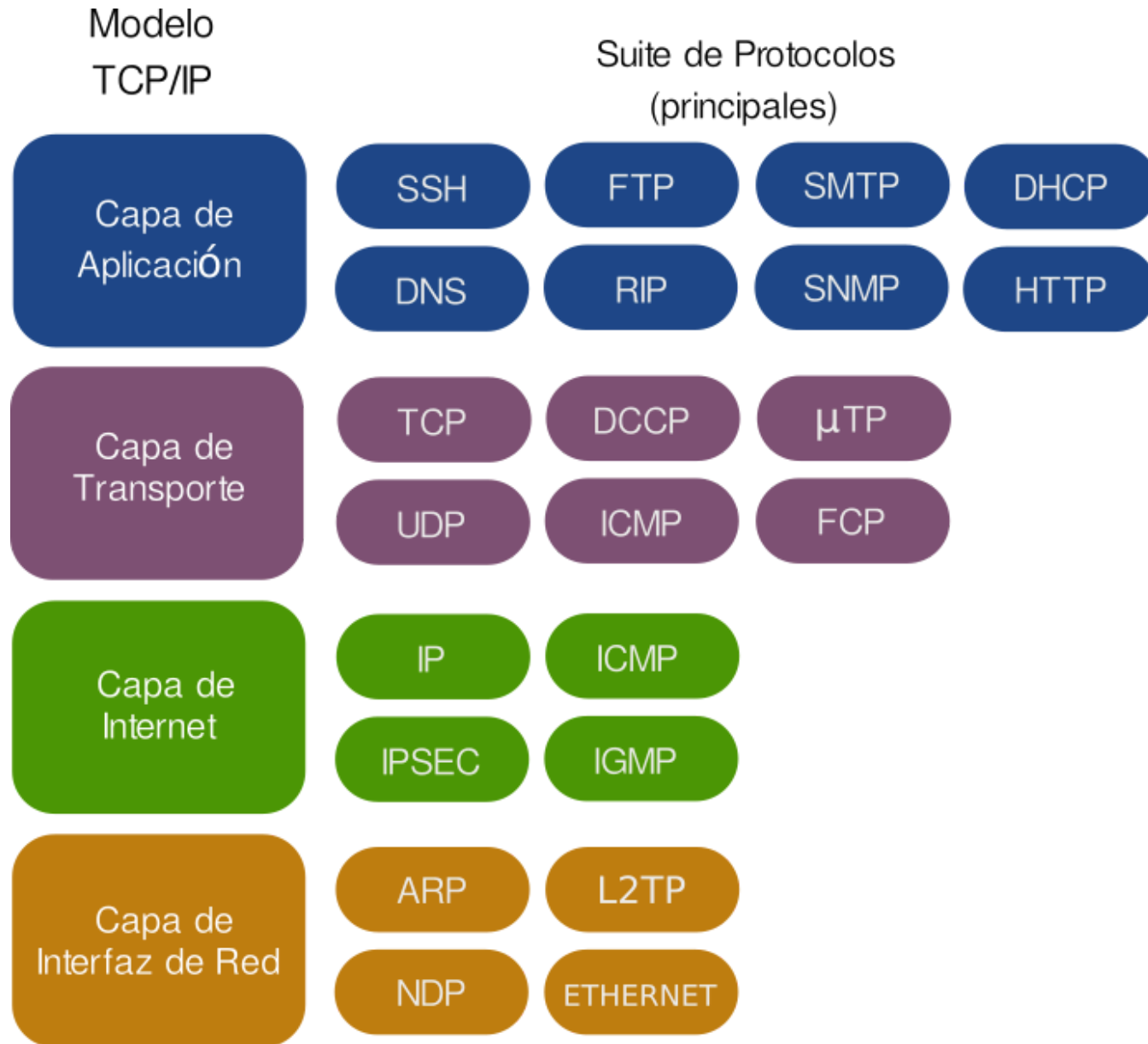
A diferència de les aplicacions d'escriptori, que utilitzen els recursos d'un únic ordinador **les aplicacions web són distribuïdes**, intervenen com a mínim dos equips diferents: el client i el servidor.

La comunicació és du a terme mitjançant **el protocol HTTP**, base de la WWW.

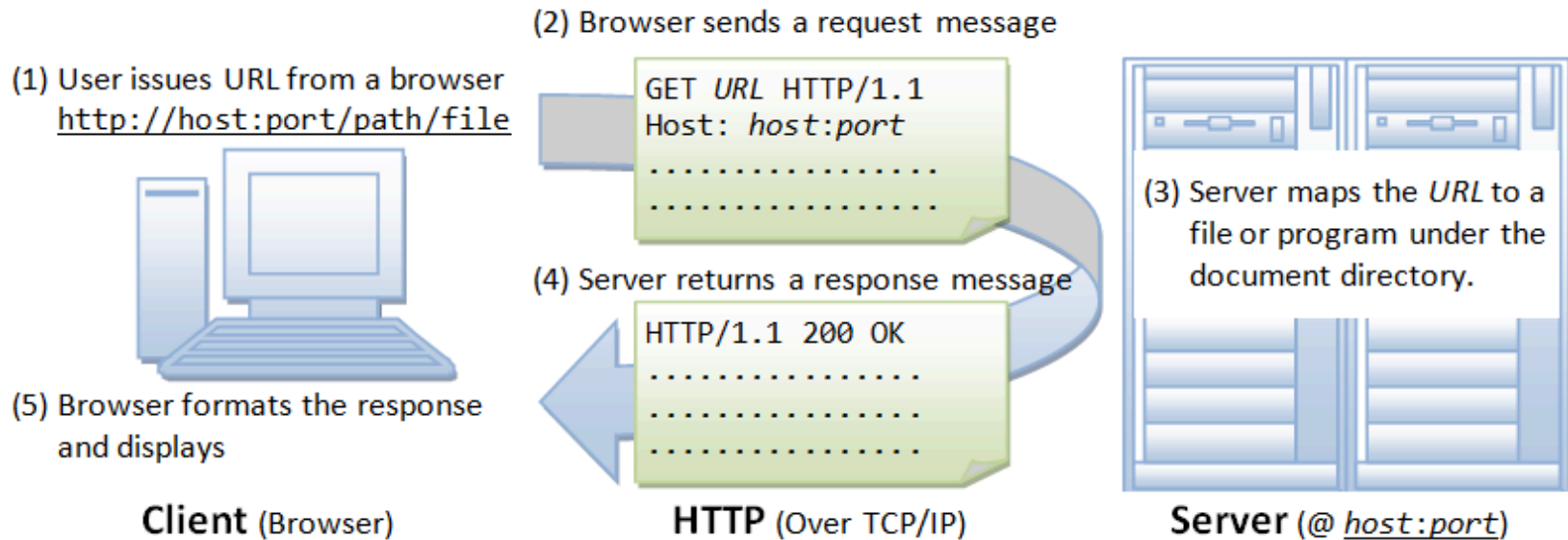
Els serveis d'internet utilitzen protocols de la **família TCP/IP**.

Model client servidor



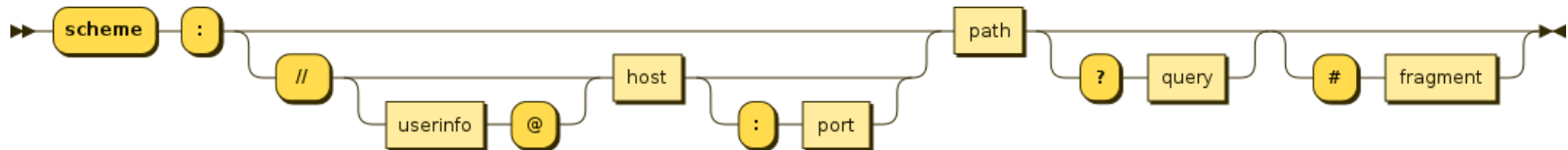


HTTP un servei transaccional

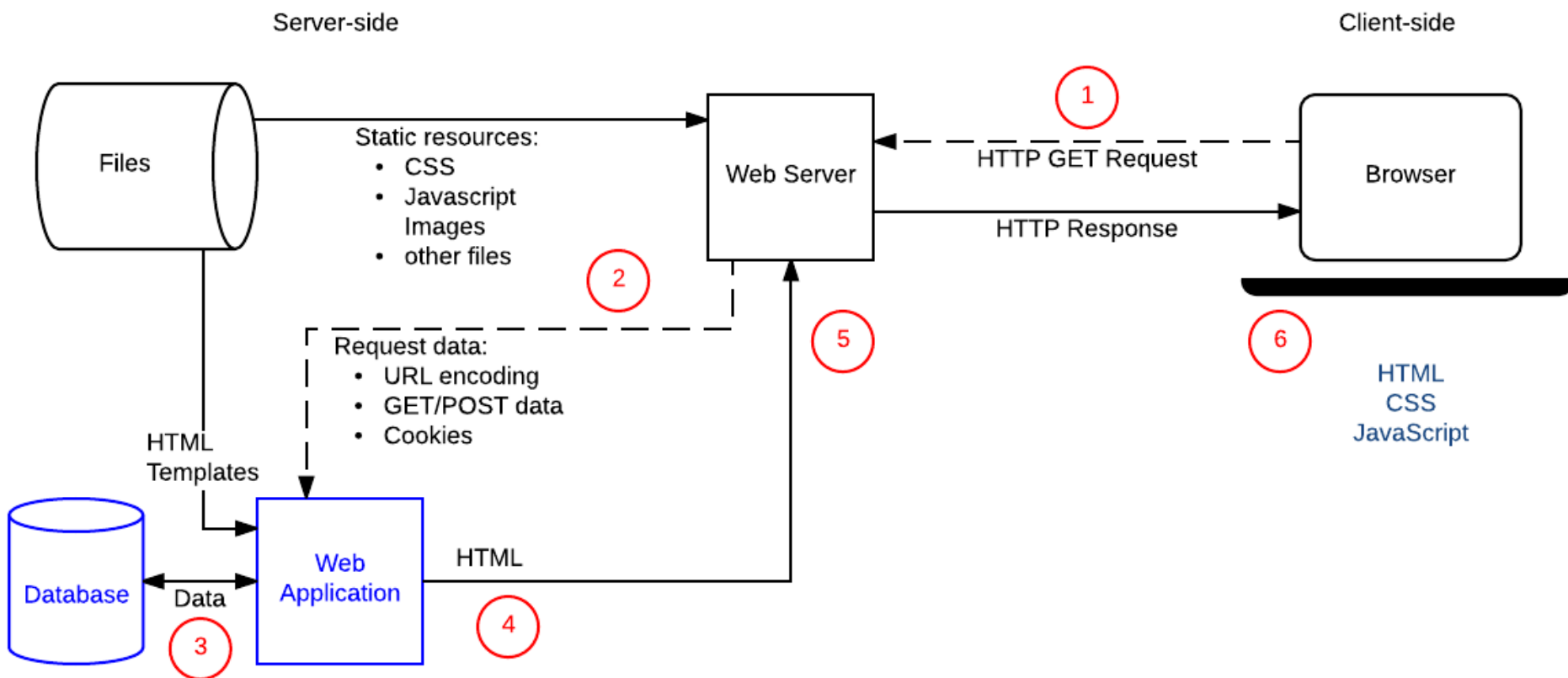


Format de les URL

- https://corriol.github.io/dwes/UD01/3_llenguatges.html#php
- <https://aules4.edu.gva.es/moodle/course/view.php?id=13298>



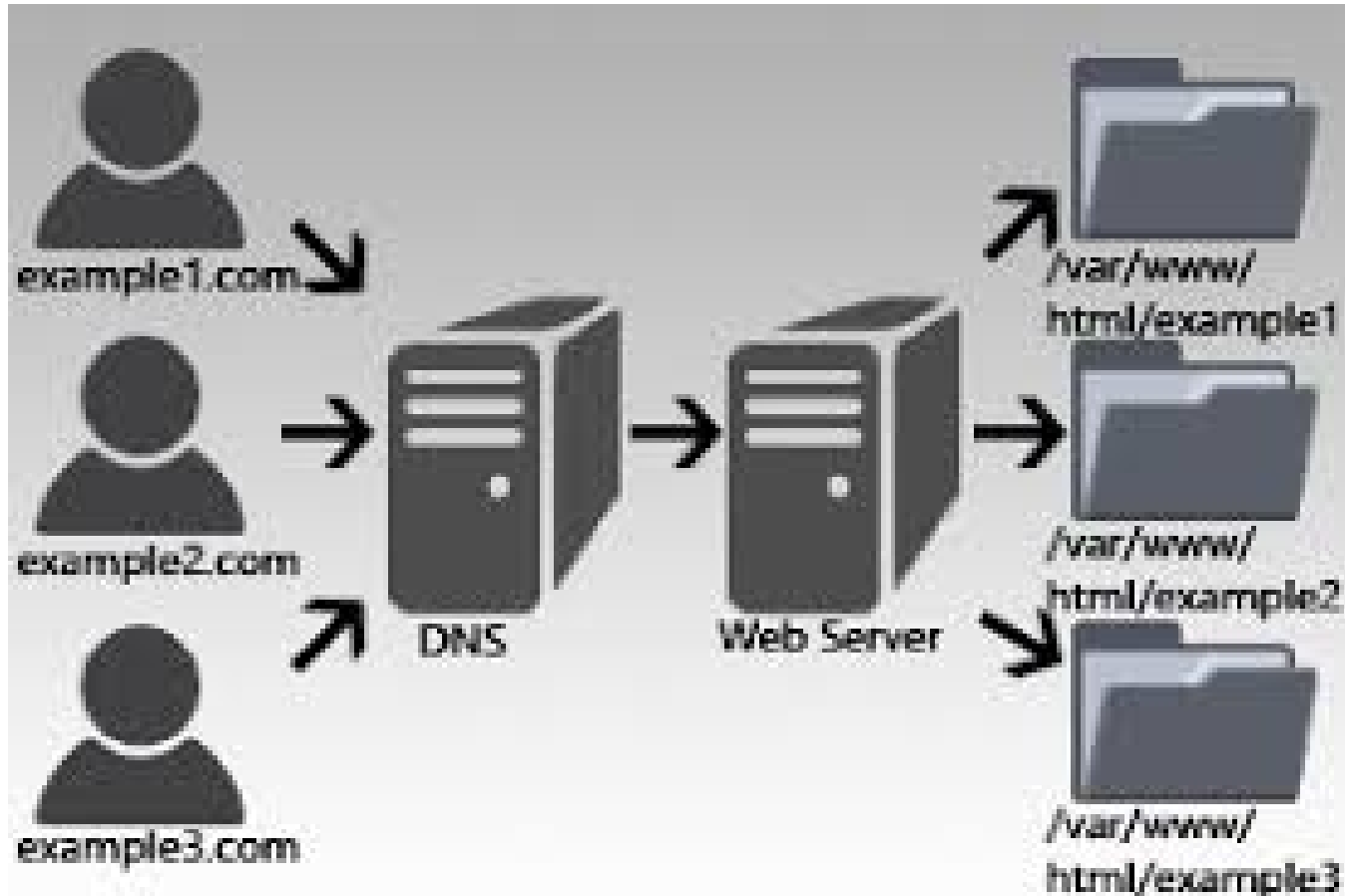
Esquema de funcionament



Apache

- Apache HTTP Server és un servidor HTTP de codi obert multiplataforma desenvolupat per Apache Software Foundation.
- L'arquitectura del servidor Apache és molt modular. El servidor consta d'una secció *core* i molta de la funcionalitat que es podria considerar bàsica per un servidor web està en forma de mòduls.

Apache - Nodes virtuals (virtual hosts)



Apache - Recursos

- <https://httpd.apache.org/docs/2.4/getting-started.html>
- <https://www.digitalocean.com/community/tutorials/how-to-install-the-apache-web-server-on-ubuntu-20-04-es>
- <https://www.ionos.es/digitalguide/servidores/configuracion/instalar-apache-en-ubuntu/>
- <https://httpd.apache.org/docs/2.4/vhosts/examples.html>

Sistema de Control de Versions

Què és un sistema de control de versions?

Un sistema de control de versions és una eina que:

- Gestiona els canvis en el codi font d'un projecte.
- Permet veure l'evolució del projecte al llarg del temps.
- Facilita el treball en equip.

Avantatges del sistema de control de versions

1. Històric de canvis

- Registra qui va fer un canvi, quan i per què.
- Recuperació de versions antigues.

2. Col·laboració

- Permet treball en equip, fins i tot en diferents funcionalitats.

Avantatges del sistema de control de versions (cont.)

3. Rastreig d'errors

- Identificació de canvis problemàtics i solució ràpida.

4. Branques i fusió

- Creació de branques per desenvolupar noves funcionalitats sense afectar el codi principal.
- Fusió de branques posteriorment.

Exemples de sistemes de control de versions

- **Git:** El sistema de control de versions més popular, àmpliament utilitzat.
- **Subversion (SVN):** Una alternativa més antiga però encara usada.
- **Mercurial:** Un altre sistema de control de versions distribuït.

Git

Què és Git?

Git és un sistema de control de versions distribuït que permet gestionar i fer seguiment dels canvis en projectes, especialment de desenvolupament de programari.

- Manté un registre detallat de totes les versions del codi.
- Facilita el treball en equip.
- Optimitza el flux de treball col·laboratiu.

Característiques principals

1. **Distribuït:** Cada usuari té una còpia completa del repositori.
2. **Eficient:** Optimitzat per a projectes grans.
3. **Branques i fusió:** Treballa amb diferents versions del projecte de manera paral·lela i fàcilment fusionable.
4. **Historial complet:** Seguiment de totes les versions i canvis.
5. **Seguretat:** Utilitza hashers criptogràfics per garantir la integritat dels canvis.
6. **Col·laboració:** Sincronitza els canvis entre diferents còpies del projecte.

Components clau de Git

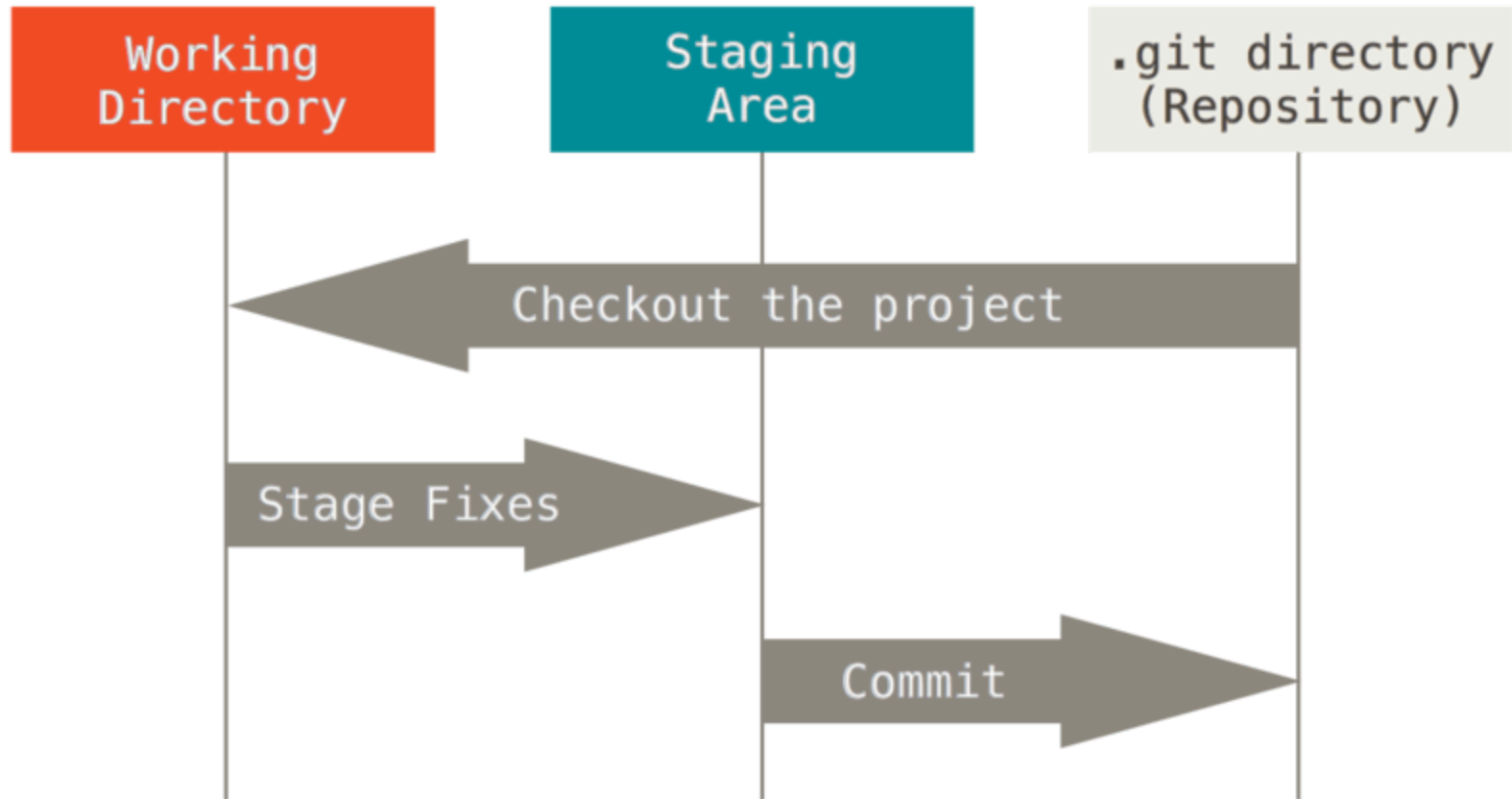
- **Repositori:** Conté l'historial complet del projecte. Pot ser local o remot.
- **Branques:** Línies independents de desenvolupament (com `main` o `develop`).
- **Commit:** Una instantània del projecte que emmagatzema un conjunt de canvis.
- ***Staging area*:** Àrea on es preparen els canvis abans de confirmar-los amb un commit.

Tres estats

Un fitxer pot estar en tres estats bàsics:

- confirmat (*committed*). Guarda't en el repositori.
- modificat (*modified*) o sense seguir (*untracked*). És nou o ha sofert canvis
- preparat (*staged*). Marcat per a guardar-se.

Tres seccions



Github



Comandaments bàsics de Git

1. Inicialització i configuració

- `git init` : Inicialitza un nou repositori de Git a la carpeta actual i crea un àrea coneguda com a *staging* (on es guarden temporalment els fitxers.)
- `git clone [URL]` : Clona un repositori remot a la teva màquina local.

2. Comprovar l'estat del repositori

- `git status` : Mostra l'estat actual del repositori, incloent fitxers modificats, afegits o esborrats.
- `git log` : Mostra l'historial dels commits (versions del codi).

3. Afegir i confirmar canvis

- `git add [fitxer]` : Afegeix els fitxers especificats a l'índex (*staging area*) per ser confirmats.
- `git add .` : Afegeix tots els fitxers modificats i nous a l'índex.
- `git commit -m "[missatge]"` : Confirma els canvis afegits amb un missatge descriptiu.

4. Enviar i obtenir canvis d'un repositori remot

- `git push [origin] [branca]` : Envia els canvis confirmats al repositori remot (per exemple, `git push origin main`).
- `git pull [origin] [branca]` : Descarrega i integra els canvis del repositori remot a la branca local.

5. Treball amb branques

- `git branch` : Mostra totes les branques locals disponibles.
- `git branch [nom]` : Crea una nova branca.
- `git checkout [branca]` : Canvia a una branca existent.
- `git checkout -b [branca]` : Crea i canvia a una nova branca.
- `git merge [branca]` : Fusiona els canvis d'una altra branca a l'actual.

6. Desfer canvis

- `git reset [fitxer]` : Elimina un fitxer de l'índex (treu els fitxers afegits però no confirmats).
- `git reset --hard [commit]` : Torna a una versió específica del codi i elimina tots els canvis des d'aquell punt.
- `git revert [commit]` : Desfà un commit en concret creant un nou commit que el desfà, sense alterar l'historial.

7. Altres útils

- `git stash` : Desa els canvis no confirmats temporalment, netejant l'àrea de treball.
- `git stash pop` : Restaura els canvis desats amb `git stash`.

8. Remot

- `git remote add origin [URL]` : Afegeix un repositori remot.
- `git fetch` : Descarrega els canvis remots sense fusionar-los.