

GESTIÓN DE PROYECTOS CON GIT

M.C. Karla Avila Cárdenas

Noviembre 2022



Agenda

- Introducción a la gestión de proyectos
- Introducción a los sistemas de control de versiones
- Introducción a Git y sus variantes
- Comandos básicos en Git
- Intercambio y actualización de proyectos
- Manejo de ramificaciones





Introducción

Las empresas que sobrevivan en el mercado del siglo XXI deben implementar el software como un elemento que permita generar estrategias de diferenciación en sus procesos de negocio.



Ser más competitivas

La paradoja tecnológica

Hoy en día importa que el crecimiento del mercado sea mayor que la disminución de los precios. El desafío está en como ser competitivo en un mundo en el cual la tecnología es virtualmente gratis.

Es necesario hacer una nueva definición de los valores económicos. El valor hoy en día está en establecer una relación de largo plazo con el cliente, aún cuando esto signifique esfuerzos por parte de la empresa, como por ejemplo:
regalar la primera generación de productos.

Administración y gestión de proyectos

Imperio romano



Segunda guerra mundial





La gestión de proyectos

La gestión de proyectos es un proceso continuo. Este proceso requiere de una estrategia global, apoyada por herramientas de trabajo que incrementen la productividad. El propósito de planificar y controlar es proveer una propuesta uniforme para el desarrollo y la administración de los proyectos. Los planes deben apoyar los niveles estratégicos, tácticos y operacionales de las organizaciones con el fin de alcanzar las metas corporativas de largo, mediano y corto plazo.



Definición de Proyecto



Un proyecto es una asociación de esfuerzos, limitado en el tiempo, con un objetivo definido, que requiere del acuerdo de un conjunto de especialidades y recursos.

Los proyectos informáticos obedecen a la definición de proyecto, pero además se caracterizan por el impacto directo e indirecto que provocan en toda la organización.

- El software se desarrolla o modifica con el intelecto; no se manufactura en el sentido clásico.
- El software no se “desgasta”.
- La mayor parte del software se construye para un uso individualizado.

Definición de proyecto informático

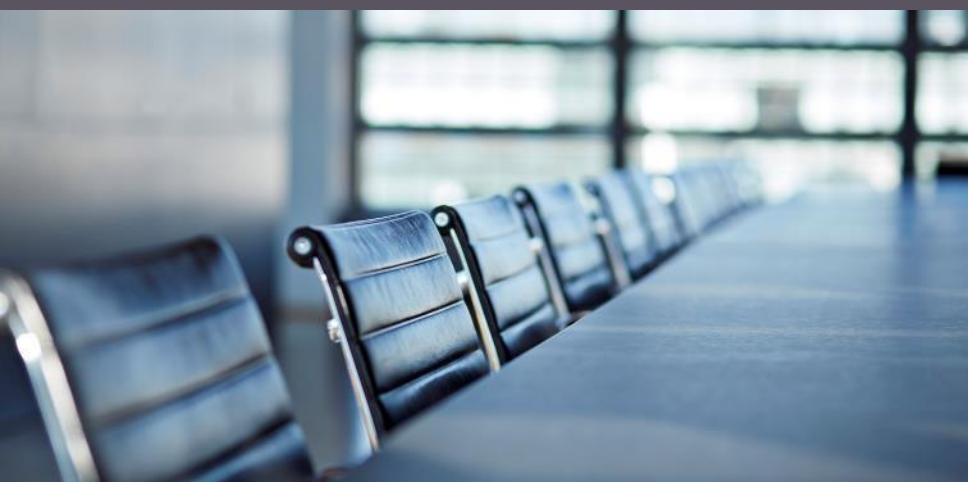
**LA MEJOR
MANERA DE
EMPEZAR ES
DEJAR DE
HABLAR Y
COMENZAR A
ACTUAR.**



Walt Disney

Administración de la configuración





¿Qué es una versión?

Una instancia de un ítem de configuración que difiere, en alguna forma, de otras instancias del mismo ítem. Las versiones siempre tienen un identificador único, que se compone generalmente del nombre del ítem de configuración más un número de versión.



Los sistemas de control de versiones

Un sistema de control de versiones (CVS) te permite realizar un seguimiento de la historia de una colección de archivos y además incluye la funcionalidad de revertir la colección de archivos actual hacia una versión anterior. Cada versión podría considerarse como una fotografía del estado de la colección en un momento determinado de tiempo. La colección de archivos usualmente es código fuente de algún lenguaje de programación, sin embargo, un sistema de control de versiones funciona con cualquier tipo de archivo. Tanto la colección de archivos como su historia completa están guardados en un repositorio.

Sistema de control de versiones

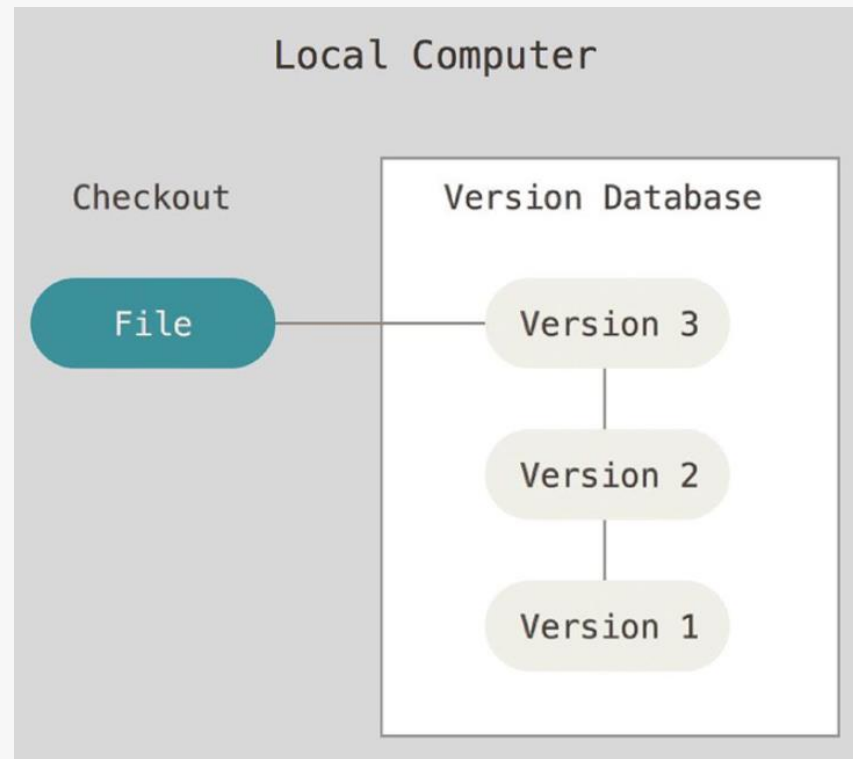
Sistemas de control de versiones locales

El método de control de versiones elegido por muchas personas es copiar archivos en otro directorio. Este enfoque es muy común porque es muy simple, pero también es increíblemente propenso a errores.

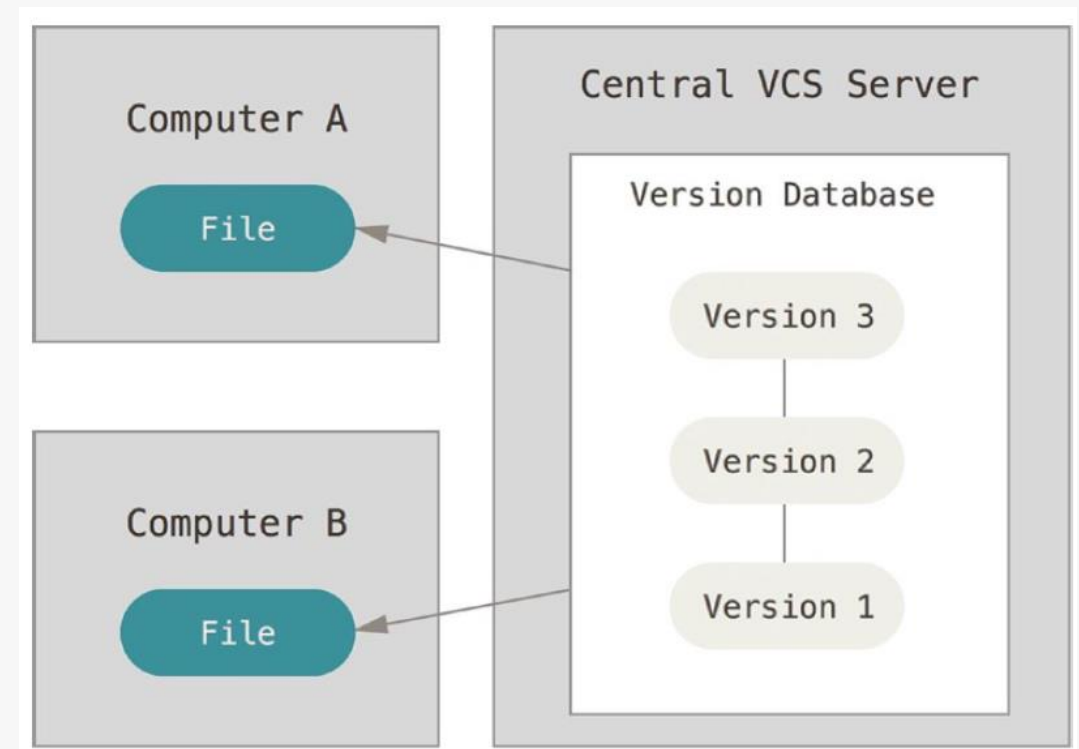
Para lidiar con este problema, los programadores desarrollaron hace mucho tiempo VCS locales que tenían una base de datos simple que guardaba todos los cambios en archivos bajo control de revisión.

Tipos de sistemas de control de versiones

Local

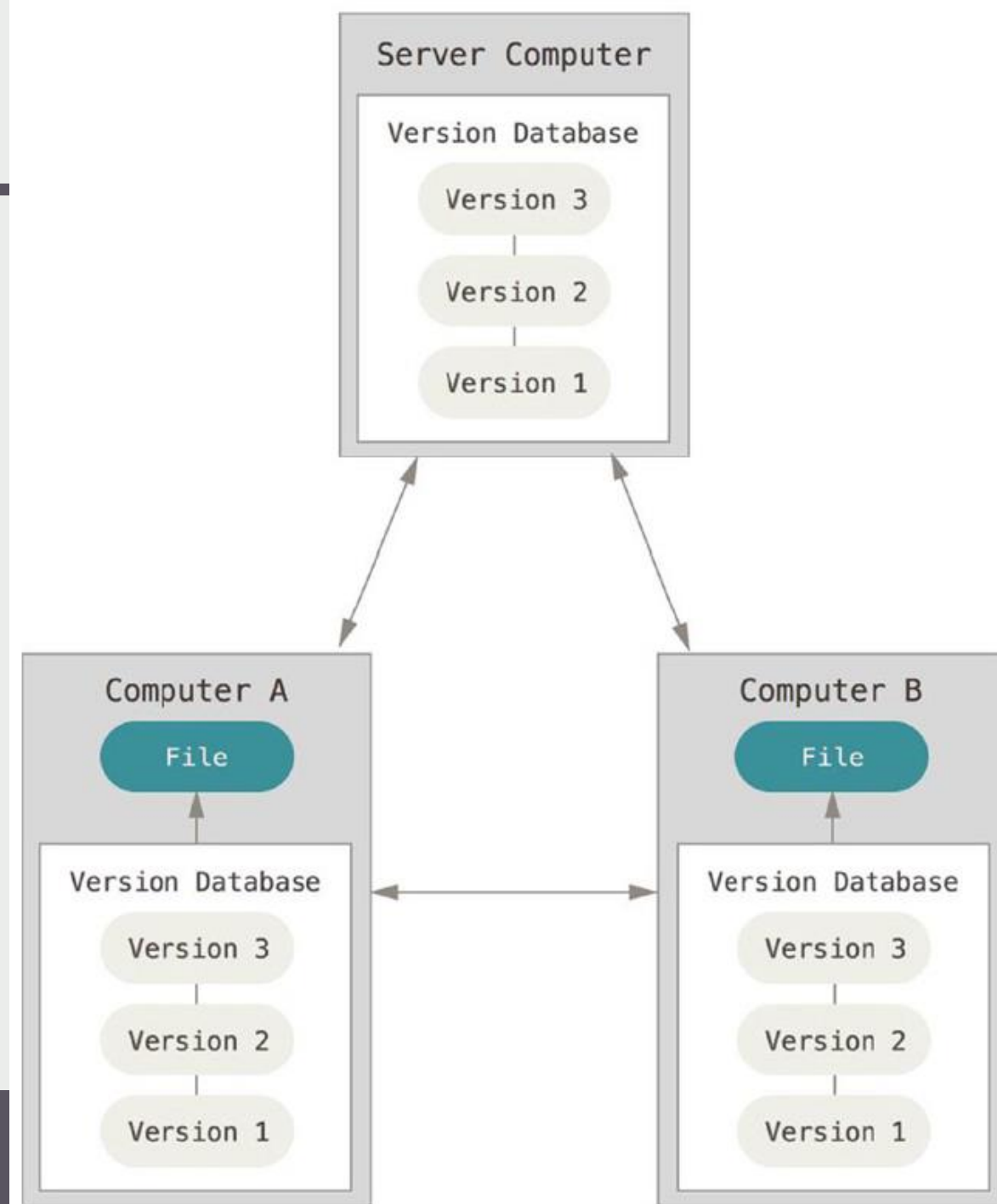


Servidor central



Tipos de sistemas de control de versiones

Distribuido



GIT



git

GIT

Git es un sistema de control de versiones distribuido gratuito y de código abierto diseñado para manejar todo, desde proyectos pequeños hasta proyectos muy grandes, con rapidez y eficiencia. Git es fácil de aprender y ocupa poco espacio con un rendimiento ultrarrápido.

Git fue creado para funcionar en el kernel de Linux, lo que significa que ha tenido que manejar de manera efectiva grandes repositorios desde el primer día. Git está escrito en C, lo que reduce la sobrecarga de los tiempos de ejecución asociados con los lenguajes de nivel superior. La velocidad y el rendimiento han sido un objetivo de diseño principal de Git desde el principio.

Ventajas de GIT

- **Git le permite** y lo alienta a **tener varias sucursales (branches)** locales que pueden ser completamente independientes entre sí. La creación, fusión y eliminación de esas líneas de desarrollo lleva unos segundos.
- **Git es rápido.** Con Git, **casi todas las operaciones se realizan localmente**, lo que le brinda una gran ventaja de velocidad en los sistemas centralizados que constantemente tienen que comunicarse con un servidor en alguna parte.
- **Git es distribuido.** Esto significa que en lugar de hacer una "verificación" del estado actual del código fuente, haces un "clon" de todo el repositorio.

Ventajas de GIT

- Integridad de datos. **El modelo de datos que utiliza Git garantiza la integridad criptográfica de cada parte de su proyecto.** Cada archivo y confirmación se suma y se recupera mediante su suma de verificación cuando se vuelve a desproteger. Es imposible obtener algo de Git que no sean los bits exactos que ingresaste.
- **Git tiene** algo llamado "área de preparación" o "índice" ("staging area" or "index"). Esta es **un área intermedia donde se pueden formatear y revisar** las confirmaciones (commit) antes de completar la confirmación.
- **Git se publica bajo** la Licencia Pública General GNU versión 2.0, que es **una licencia de código abierto.**

Companies & Projects Using Git

Google

Microsoft



LinkedIn

NETFLIX



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



<https://bitbucket.org/>



<https://github.com/>

<https://git-scm.com/download/gui/windows>

Conceptos

PUSH: El comando git push se usa para cargar contenido del repositorio local a un repositorio remoto. El envío es la forma de transferir confirmaciones desde tu repositorio local a un repositorio remoto.

PULL: El comando git pull se emplea para extraer y descargar contenido desde un repositorio remoto y actualizar al instante el repositorio local para reflejar ese contenido.

Conceptos

COMMIT: Consolidar, confirmar o hacer un commit se refiere, en el contexto de la ciencia de la computación y la gestión de datos, a la idea de confirmar un conjunto de cambios provisionales de forma permanente. Un uso popular es al final de una transacción de base de datos.

3 FORMAS DE HACER COMMIT DE TU CÓDIGO



Conceptos

BRANCH: Cuando hablamos de ramificaciones, significa que tú has tomado la rama principal de desarrollo (master) y a partir de ahí has continuado trabajando sin seguir la rama principal de desarrollo.

MERGE: La fusión es la forma que tiene Git de volver a unir un historial bifurcado. El comando `git merge` permite tomar las líneas independientes de desarrollo.

COMANDOS BÁSICOS DE GIT

GIT es un sistema de control de versiones

GIT INIT

Crea un repositorio vacío o reinicia uno existente.

GIT ADD

Agrega archivos al staged (staging area - index).

GIT COMMIT

Registra los cambios en el repositorio.

GIT STATUS

Muestra el estado actual del repositorio.

GIT CLONE

Clona un repositorio en un nuevo directorio.

GIT PUSH

Actualiza las referencias remotas junto con los objetos asociados

GIT PULL

Actualiza las referencias locales junto con los objetos asociados

GIT CHECKOUT

Cambia de rama o restaura archivos del espacio de trabajo.

GRACIAS



M.C. Karla Avila Cárdenas
karla.acardenas@academicos.udg.mx

