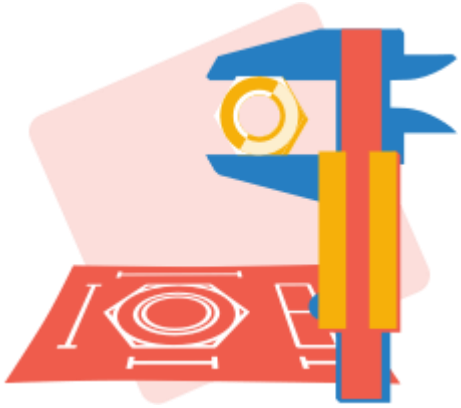
CTIC UNI
Centro de Tecnologías de Información y Comunicaciones
Universidad Nacional de Ingeniería

Programa Ejecutivo de Business Intelligence & Big Data

Marco de trabajo Ágil
Introducción Integración continua
GIT / GIT HUB

Ing. Arturo Rojas Medrano

Sistema de Versionamiento

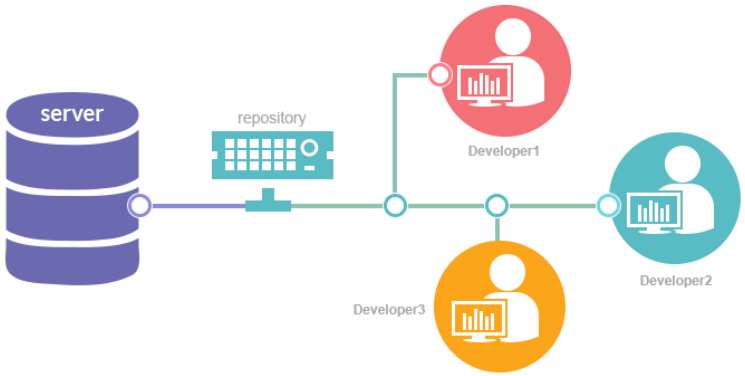


Buenas practicas -> eficiencia en el trabajo.

Versionamiento código:

- Compartir código fuente.
- Mantener registro de cambios (versiones línea por línea).

Sistema de control de versiones



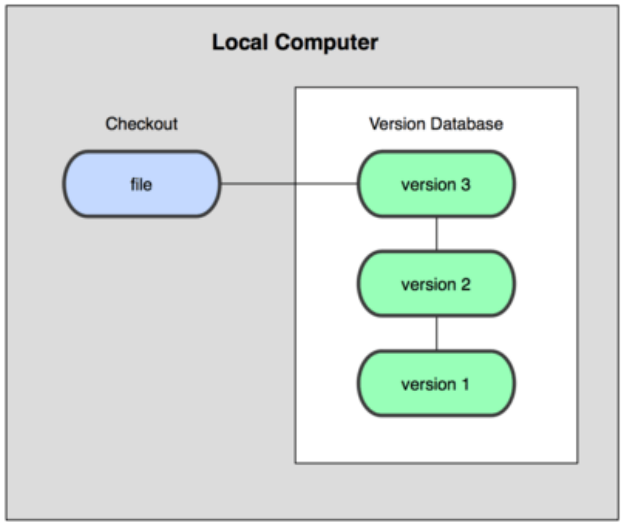
- Registra los cambios.
- Almacena versiones anteriores.

¿Por qué razón acceder a versiones anteriores ?.

- Modificaciones incorrectas (perdidas de funcionalidad).
- Retomar la construcción.

Clasificación

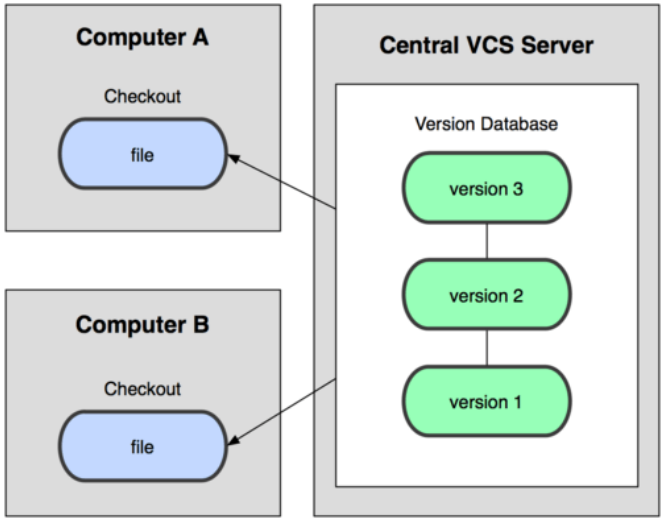
SISTEMA CONTROL DE VERSIONES LOCAL



- Método mas utilizado, menos eficiente.
- Método rudimentario para controlar versiones (copiar archivo en directorio local).
- Utilizado para desarrollo pequeño.
- Para aprender un lenguaje.
- En problemas grandes y complejos se vuelve un problema.

Clasificación

SISTEMA CONTROL DE VERSIONES CENTRALIZADA



PROS :

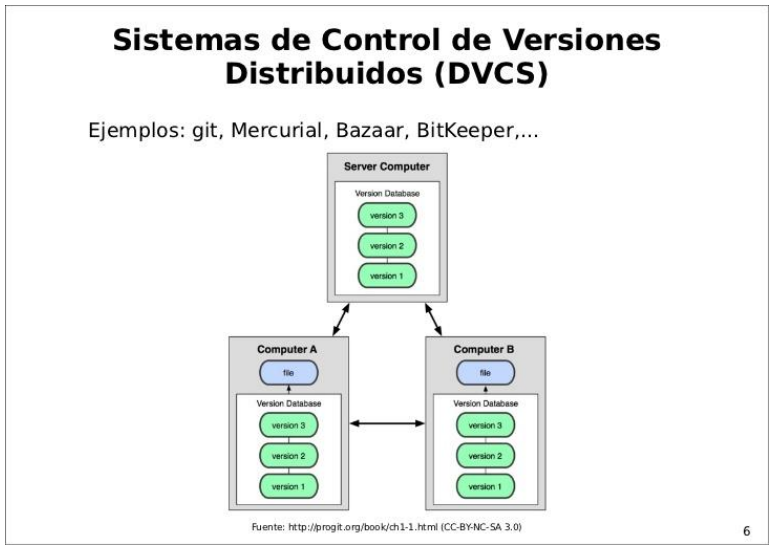
- Bueno cuando formamos parte de un equipo de desarrollo.
- Ofrece un servidor que almacena todos los archivos que son versionados.

CONTRAS :

- Al ser un solo servidor, si existiera un problema toda la información se vé afectada.

Clasificación

SISTEMA CONTROL DE VERSIONES DISTRIBUIDA



- No encontramos un servidor único.
- Cada usuario poseerá una copia completa del proyecto de manera local.
- Si existe un problema, el código versionado se encontrará accesible desde cualquier repositorio cliente.

SUBVERSION mercurial BitKeeper Bazaar PERFORCE git

git

- Sistema de control de versiones distribuido.
- Adecuado para mantener gran cantidad de código para muchos programadores.
- Utilizado en proyectos de cualquier envergadura.
- Herramienta de Código abierto.
- Multiplaforma.

CARACTERISTICAS

Control proyecto
- Control de todo el proyecto

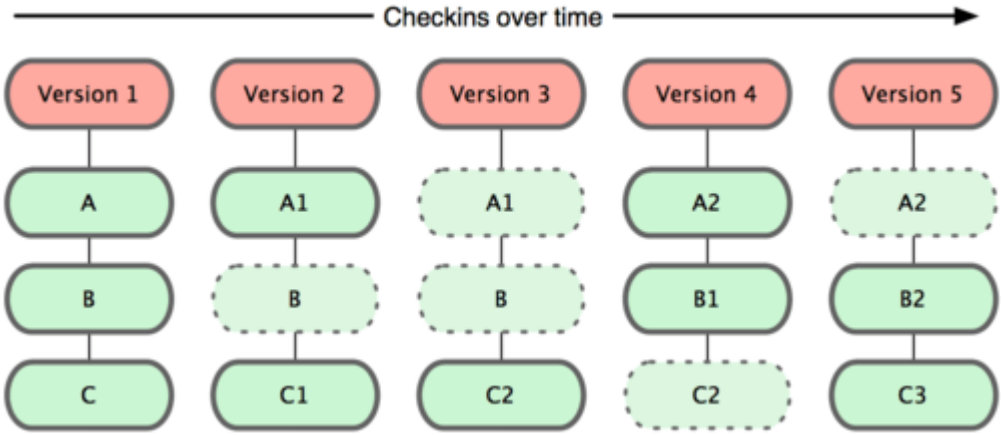
Deshacer
- Volver a pasos previos

Auditoria
- De código (Cuando, donde, quien, que).

git

Etiquetas
- Para el control de versiones

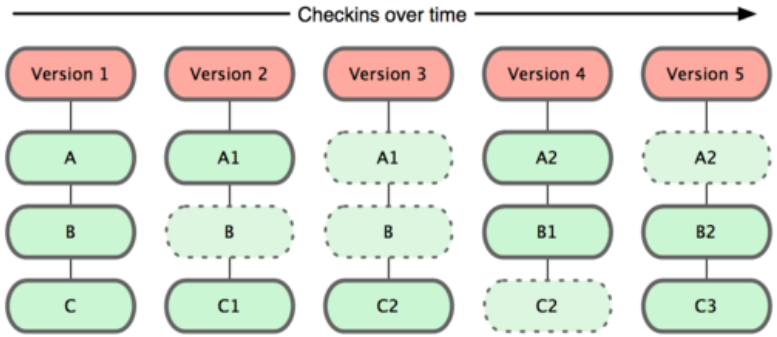
Seguridad
- Cifrado algoritmo SHA1



GIT utiliza instantáneas para almacenar cambios realizados en el tiempo.

- Modificación archivos -> Instantánea archivos, almacena referencia captura.
- NO se modifica -> Sólo se almacena un enlace al archivo anterior

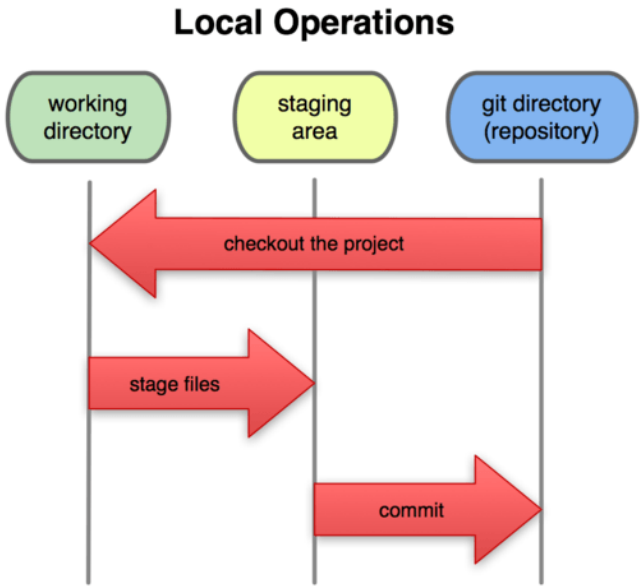
Trabaja con base de datos local.



GIT utiliza instantáneas para almacenar cambios realizados en el tiempo.

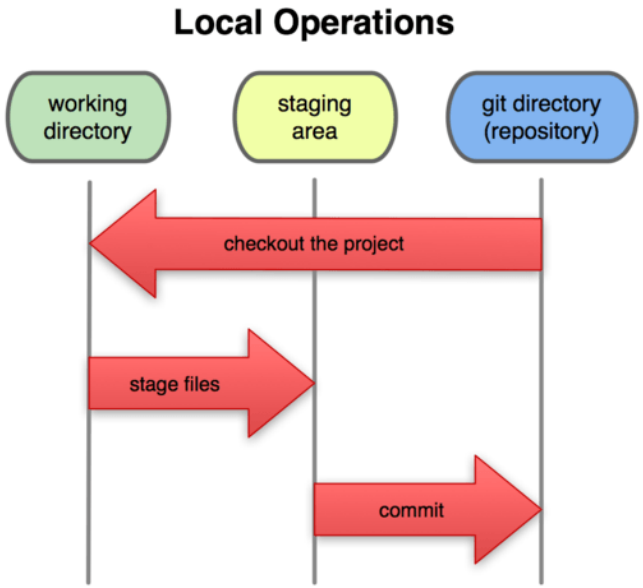
- Modificación archivos -> Instantánea archivos, almacena referencia captura.
- NO se modifica -> Sólo se almacena un enlace al archivo anterior

Trabaja con base de datos local -> Utiliza recursos locales y lo subiremos a la red cuando confirmemos.



FLUJO DE TRABAJO

1. Modificar archivos -> Directorio de trabajo.
2. Agregar los archivos modificados -> Area de preparación.
3. Confirmar cambios realizados para tomar una instantánea y almacenarla en forma permanente en el directorio GIT.



ESTADOS DE ARCHIVOS

1. Confirmado o Commited-> Los datos o cambios se encuentran almacenados en la base de datos local.
2. Agregar los archivos modificados -> Area de preparación.
3. Confirmar cambios realizados para tomar una instantánea y almacenarla en forma permanente en el directorio GIT.

LABORATORIO

1. Instalar GIT, ingresar a la siguiente pagina: https://git-scm.com/downloads

CTIC UNI
Centro de Tecnologías de Información y Comunicaciones
Universidad Nacional de Ingeniería

Programa Ejecutivo de Business Intelligence & Big Data

Marco de trabajo Ágil
Introducción Integración continua
GIT / GIT HUB

Ing. Arturo Rojas Medrano