

UNIVERSIDAD LUTERANA SALVADOREÑA
FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA
LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN
CÁTEDRA DE INGENIERÍA DE SOFTWARE.



“Administración de la configuración”

ALUMNO:

LUIS ROBERTO GARCÍA FLORES

DOCENTE:

INGENIERA LESBIA MARÍA MANCIA DE MARADIAGA

CICLO

II 2020

San Salvador, 24 de Noviembre de 2020

ADMINISTRACIÓN DE LA CONFIGURACIÓN

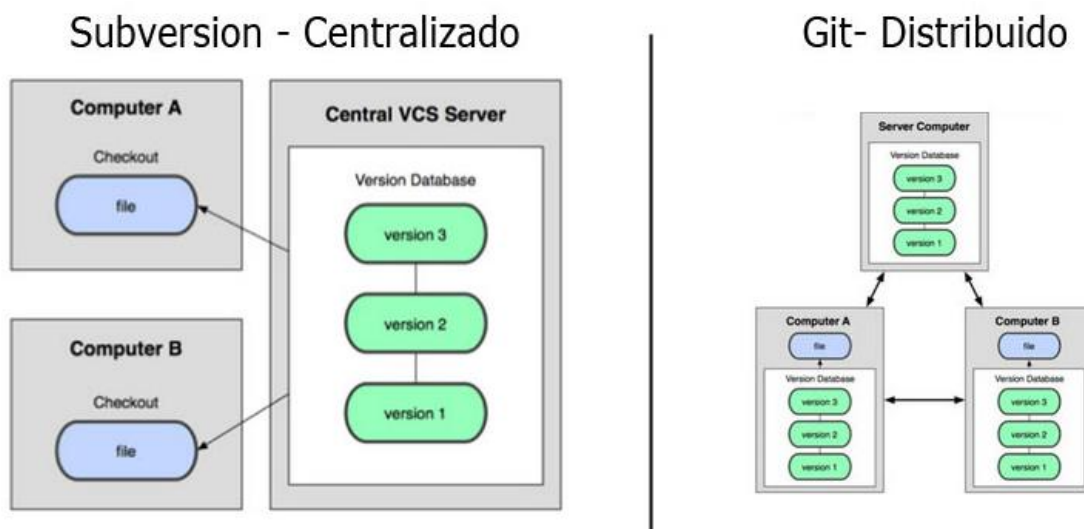
La administración de configuración es la disciplina encargada de identificar la configuración general de un sistema para así mantener su confiabilidad, adaptabilidad y configuración a los diferentes ciclos de vida. Está formalmente definida por la IEEE610.12-90 como "Disciplina aplicada de manera técnica y administrativa para la dirección y supervivencia para: Identificar y documentar las características físicas y funcionales de la configuración de los elementos, control en el cambio de sus características grabar y reportar cambios en el proceso de implementación así como su estado y verificación del cumplimiento de sus requerimientos específicos".



SUBVERSION VS GIT

SISTEMA CENTRALIZADO VS SISTEMA DISTRIBUIDO

Antes de conocer a cada una de las aplicaciones queremos dejar clara la principal **diferencia** entre los dos sistemas **CVS**, y es que mientras **Subversion** (SVN) es un sistema **centralizado**, **Git** es **distribuido**.

Aunque luego explicaremos a que afecta este hecho a ambos sistemas así como sus características, en la imagen de abajo puedes observar dos gráficas una referente a un sistema centralizado como es el que tiene Subversion y otro un sistema distribuido como es el que tiene Git.



 <p>SUBVERSION</p>	 <p>GIT</p>
<p>Commits solo a servidor</p>	<p>Commits a repositorio local y a servidor</p>
<p>Si no tenemos internet no podremos tener acceso al repositorio y por lo tanto no podremos enviar nuestros commits</p>	<p>Si no hay internet podemos acceder al repositorio local y enviar nuestros commits</p>
<p>Menor velocidad</p>	<p>Mayor velocidad</p>

- Git es mucho más rápido que Subversion
- Subversion le permite ver sólo un subárbol de un repositorio; Git necesita clonar el repositorio completo (incluyendo el historial) y crear una copia de trabajo que refleje al menos un subconjunto de los elementos que están bajo control de versiones.
- Los repositorios Git son mucho más pequeños que Subversiones. SVN requiere 30 veces más el espacio en disco para almacenar el mismo historial.
- Git fue diseñado para ser totalmente distribuida desde el principio, lo que permite que cada desarrollador tenga pleno control local.
- Las ramas (branches) Git son más simples y menos pesados que Subversion
- Las ramas (branches) Git llevan toda su historia completa
- La fusión (merge) en Git no obliga a recordar que la revisión que se fusionaron a partir de (este beneficio se evita con el lanzamiento de Subversion 1.5)
- Git proporciona una mejor auditoría de eventos de ramificación (branch) y fusión (merge)
- Los formatos de archivo de los repositorios de Git son simples, por lo que la reparación es fácil y la corrupción es rara.
- La copia de seguridad de los repositorios centrales de Subversion es potencialmente más simple
- El clonado de repositorios Git actúa como repositorio de copias de seguridad completas

- Interfaz de usuario de Subversion es más maduro que Git. Actualmente Subversion tiene una gama más amplia de GUI's que Git. Por ejemplo, hay plugins disponibles de SVN para los IDEs más populares, pero si utilizas Netbeans o Eclipse vas a tener GUI's para ambos sistemas.
- Navegando a través de las versiones es más simple en Subversion, ya que utiliza números de secuencia de revisión (1,2,3, ..); Git utiliza impredecibles códigos hash SHA-1.
- Navegando hacia atrás en Git es muy fácil usando la sitaxis «^», pero no hay manera fácil de navegar hacia adelante.
- Dado que Subversion tiene un repositorio central único, es posible especificar leer y escribir los controles de acceso en un solo lugar y hacer que se respete efectivamente en la totalidad del proyecto.
- Con Subversion se puede extraer un subdirectorio de un repositorio. Esto no es posible con Git. Por lo que para un proyecto grande, esto significa tener siempre que descargar el repositorio completo.
- SVN tiene la ventaja de que es mucho más fácil de aprender mientras que Git es más difícil de aprender porque tiene más conceptos y órdenes.

CONCLUSIÓN PERSONAL

Git no es mejor que Subversion, sólo trabajan de forma diferente. Si necesitamos un control de código fuente offline y tenemos la disposición de gastar más tiempo en aprender un sistema de control de versiones, Git es nuestra elección.

Si tenemos un sistema de control de código fuente estrictamente centralizado y estamos iniciándonos en el control de versiones, la **simplicidad de Subversion** nos facilitará nuestro flujo de trabajo.