

Comparativa entre Git, Mercurial y Subversion

<i>Característica</i>	<i>Git</i>	<i>Mercurial</i>	<i>Subversion</i>
Tipo	Distribuido	Distribuido	Centralizado
Commits atómicos	Sí	Sí	Sí
Ficheros y Directorios – Movimientos y Renombramientos	Sí	Sí	Sí
Ficheros y Directorios – Copias	Sí	Sí	Sí
Replicación de Repositorio	Sí	Sí	Indirecta (los repositorios son de sólo-lectura) Aunque hay herramientas que se ocupan de clonar repositorios
Propagación de cambios a otros repositorios	Sí	Sí	Indirecta
Permisos de repositorio	Sí	Sí	Sí
Listas de cambios	Sí	Sí	Parcialmente
Historial a nivel de línea		Sí	Sí
Puede trabajar en un sólo directorio del repositorio	Sí	Sí	Sí
Historial de cambios no enviados	Sí	Sí	Sí
Mensajes de envío por fichero	No	No	No
Documentación	Regular	Muy buena	Muy buena
Facilidad de instalación	Buena	Excelente	Regular
Conjunto de órdenes	Estilo bin-utils Enorme	Estilo-CVS Algunas no siguen este estilo	Estilo-CVS

<i>Característica</i>	<i>Git</i>	<i>Mercurial</i>	<i>Subversion</i>
Soporte de red	Excelente HTTP: git:	Excelente HTTP: SSH:	Excelente WebDav (HTTP:) SSH: SVN:
Portabilidad	Regular Linux / Unix Posibilidad de funcionar sobre Windows (Cygwin)	Excelente (toda plataforma que soporte a Python)	Excelente Linux Windows Mac OS X
Interfaz web	Sí	Sí	Sí (pero no vienen con subversion)
Existencia de GUIs	Medio	Medio	Muy bueno
Licencia	GNU GPL	GNU GPL	Tipo Apache/BSD
PROS	- Muy rápido - Enorme cantidad de órdenes - Estable - Ramas locales - Buena interfaz web - Muy configurable - Puede importar repositorios Subversion	- Portable - Fácil de entender y usar - Ids Incrementales - Estable - Muy buena documentación	- Uso extremadamente fácil - Estable - Menor cantidad de conflictos - Ids Incrementales - Buena documentación - Copias “baratas”
Contras	- No hay soporte nativo en windows - Conjunto de órdenes confuso para principiantes - Recolección de basura manual. - Ids confusas	- El conjunto de órdenes no es muy extenso. - No ramas por repositorio.	- Insuficiente para proyectos con muchas personas trabajando en él - No ramas por repositorio. - Tratamiento confuso de ramas.

Conclusión

- Para un proyecto con mucho personal trabajando y en distintos horarios, es mejor usar Git o Mercurial. Git es muy complejo de usar pero es más potente que Mercurial. La cuestión de decidir entre uno u otro sería Potencia contra Facilidad de uso.
- Para un proyecto con poco personal, es mejor usar Subversion. Aunque la gestión de ramas es algo confusa, no es difícil de usar. Además, usuarios que hallan usado CVS ya sabrán

prácticamente usar Subversion.