



SVN - Manual de Referencia

Valparaíso, diciembre 2005

Subversion es un sistema de control de versiones libre y de código abierto (opensource). Esto significa que Subversion maneja archivos y directorios a través del tiempo. Un árbol de archivos es puesto en un repositorio central y se va modificando por distintos usuarios. El repositorio es como un servidor de archivos común, excepto porque recuerda todos los cambios hechos sobre sus archivos y directorios. Ésto le permite recuperar versiones anteriores a la fecha actual, o examinar la historia de cómo sus datos han cambiado. En este sentido, muchas personas ven a un sistema de control de versiones como una especie de “máquina del tiempo”.

En la actualidad, herramientas como ésta son indispensables en el desarrollo de software, especialmente en lo que se refiere a proyectos opensource. Este manual de referencia pretende guiar al usuario final a través de los comandos más usados en el trabajo diario con este sistema.

1. Repositorio

Normalmente no es necesario manipular el repositorio. Las siguientes acciones son realizadas solamente por el administrador y los usuarios trabajan sólo sobre su copia de trabajo.

- Crear Repositorio

```
svnadmin create <nombre_repositorio>
```

- Respaldao o migrando repositorios

El comando **dump** devuelve una versión en texto plano del contenido del repositorio, independiente de su ubicación física. Los siguientes pasos describen cómo migrar un repositorio a una nueva ubicación. Este mismo procedimiento se puede usar para respaldar repositorios (guardando sólo la salida de **dump**).

1. Guardar la salida de **dump** en un archivo

```
svnadmin dump <repositorio> > /tmp/dump
```

2. Crear un nuevo repositorio

```
svnadmin create <repositorio_nuevo>
```

3. Cargar el archivo **.dump** al nuevo repositorio

```
svnadmin load <repositorio_nuevo> < /tmp/dump
```

- Menú ayuda

```
svnadmin help [comando]
```

```
svnlook help [comando]
```

2. Copia de Trabajo

- Paso inicial: Importar árbol con ficheros fuente al repositorio (si se omite el directorio de origen, se asume el directorio actual)

```
svn import [origen] https://svn.inf.utfsm.cl/repos/<nombre_proyecto>
```

Importante: Todos los comandos que siguen se ejecutan desde la copia de trabajo en el cliente.

- Crear copia de trabajo (**checkout**)

```
svn co https://svn.inf.utfsm.cl/repos/<nombre_proyecto> [destino]
```

- Actualizar copia de trabajo y subir al repositorio

Si no se especifica un archivo o directorio, los cambios se hacen recursivamente sobre todo el directorio actual.

Importante: Realizar primero update (up) y después commit (ci). No se permitirá el commit de un archivo del cual haya una versión más reciente en el repositorio.

```
svn update
svn ci
```

```
estados:  A  Added
          D  Deleted
          U  Updated
          C  Conflict  <--- PROBLEMAS!!!
          G  Merged
```

- Resolviendo conflictos

Los conflictos ocurren cuando dos usuarios parten de una misma versión *A* de un archivo y modifican la misma sección de éste. Cuando uno de los usuarios sube sus cambios a la revisión *B* y el otro hace un update, se creará un conflicto en su copia de trabajo.

Al ocurrir conflictos durante un update, se crean los siguientes archivos:

```
lala      : versión en conflicto, con las líneas afectadas marcadas
lala.mine : versión modificada de la copia de trabajo, antes del update
lala.rA   : contenido del archivo en la revisión A, anterior al conflicto
lala.rB   : contenido del archivo en la última revisión del repositorio
```

La idea es comparar las diferencias y arreglar el archivo definitivo (en este caso *lala*). Una vez hecho esto, ejecutar:

```
svn resolved <archivo_arreglado>
```

- Añadir/quitar un archivo/directorio existente

```
svn add <archivo/directorio>
svn rm <archivo/directorio>
```

- Añadir un directorio nuevo

```
svn mkdir <nombre_directorio_nuevo>
```

- Copiar/mover un archivo/directorio

```
svn cp <archivo/directorio>
svn mv <archivo/directorio>
```

- Ver el estado actual de los archivos

```
svn status
```

```
estados:  A Added
          C Conflicted
          D Deleted
          G Merged
          I Ignored
          M Modified
          R Replaced
          X item is unversioned, but is used by an externals definition
          ? item is not under version control
          ! item is missing (removed by non-svn command) or incomplete
          ~ versioned item obstructed by some item of a different kind
```

- Revisar logs y comparar diferencias entre revisiones

```
svn log -r <revision> [archivo]
svn cat -r <revision> [archivo]
svn diff -r <revision> [archivo]
```

Por defecto se compara la revisión actual de la copia de trabajo con la última bajada del repositorio. Para comparar los cambios entre dos revisiones cualesquiera:

```
svn diff -r <revisionA>:<revisionB> [archivo]
```

Alternativa: Interfaz WebSVN

```
https://svn.inf.utfsm.cl/websvn
```

- Ver información acerca de un archivo/directorio

```
svn info <archivo/directorio>
```

- Menú Ayuda

```
svn help [comando]
```

- Predefinir editor para logs (incluir en `.bash_profile`)

```
export SVN_EDITOR=/usr/bin/<su_editor_favorito>
```

Ejemplo:

```
export SVN_EDITOR=/usr/bin/vim
```

3. Referencias y Links

- Sitio Oficial Subversion

<http://subversion.tigris.org>

- SVN Book - Manual Oficial

<http://svnbook.red-bean.com>

- Proyecto WebSVN

<http://websvn.tigris.org>

- Proyecto TortoiseSVN - Herramienta para MS Windows

<http://tortoisesvn.tigris.org>

- Repositorio Departamento de Informática - UTFSM

<https://svn.inf.utfsm.cl/websvn>

<https://svn.inf.utfsm.cl/repos/<REPOSITORIO>>