

Prácticas de Procesamiento de Lenguajes. Subversion

Subversion

Página: <http://subversion.tigris.org/>
Libro: <http://svnbook.red-bean.com/>



Referencia rápida de comandos:

<http://www.collab.net/community/subversion/articles/SvnQuickReferenceCard.html>

Wikipedia: <http://es.wikipedia.org/wiki/Subversion>
[http://en.wikipedia.org/wiki/Subversion_\(software\)](http://en.wikipedia.org/wiki/Subversion_(software))

1. Herramientas cliente para la utilización de subversión

Linux:

Línea de comandos, instalación paquetes Debian/Ubuntu:

```
$ apt-get install subversion subversion-tools
```

Cliente subversión multiplataforma con interfaz gráfica:

RapidSVN: <http://rapidsvn.tigris.org/>

Windows:

Línea de comandos:

SlikSVN: <http://www.sliksvn.com/en/download/>

Cliente subversión multiplataforma con interfaz gráfica:

RapidSVN: <http://rapidsvn.tigris.org/>

Cliente subversión implementado como extensión de la shell de windows

TortoiseSVN: <http://tortoisesvn.tigris.org/>

Existen muchas más herramientas que permiten utilizar subversión (clientes para IDEs de desarrollo como Eclipse o Visual Studio, clientes para Mac OS, clientes web, etc.)

2. Introducción

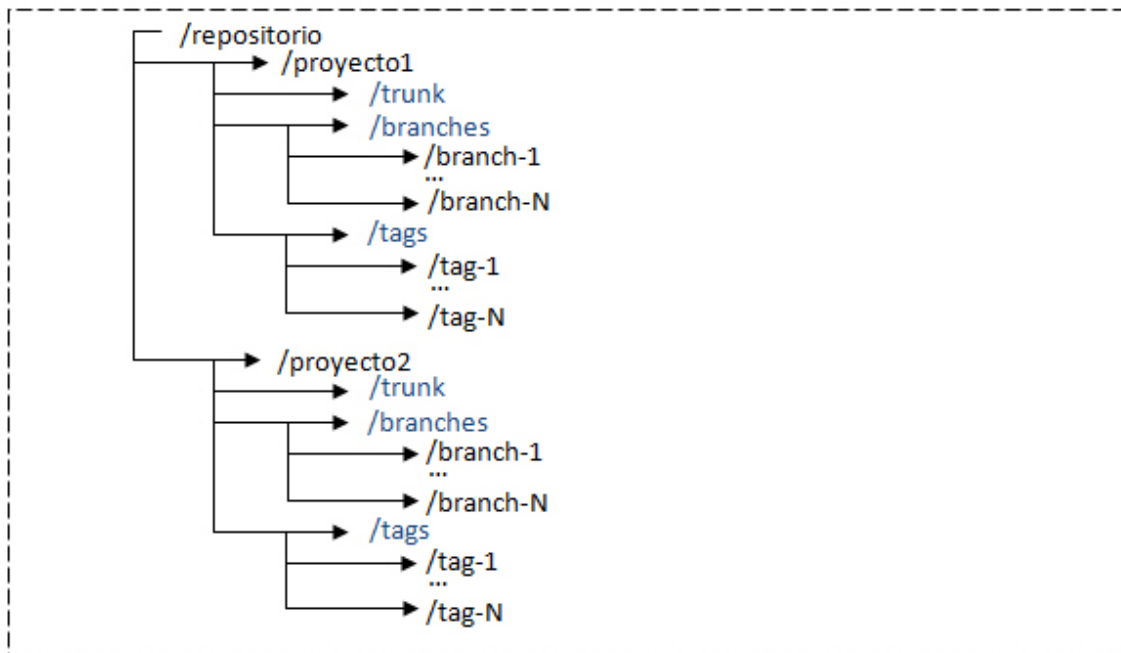
- Los sistemas de control de versiones son herramientas software que permiten la automatización de tareas comunes sobre archivos (guardar, recuperar, registrar, borrar, identificar, mezclar) manteniendo una correcta gestión sobre las versiones de la información almacenada
- Subversion es un sistema de control de versiones libre desarrollado por *CollabNet* como mejora sobre CVS (Concurrent Versions System)

3. Características

- Subversion gestiona ficheros y directorios, y los cambios realizados sobre ellos
- Permite recuperar las distintas versiones de los datos o examinar el histórico de cambios
- Funcionamiento local o en red lo que permite compartir/colaborar en los proyectos desde distintos equipos y con diferentes usuarios de forma concurrente
- Un repositorio es un depósito centralizado para almacenar y mantener información
- La pieza principal de un sistema subversión es el repositorio que está organizado de forma jerárquica en directorios y archivos
- Cada vez que se publica un envío de datos se crea una nueva versión (revisión) pero se puede acceder a las versiones ya existentes

- Se pueden realizar fusiones entre copias pero el usuario tiene que decidir ante los conflictos que puedan aparecer
- El acceso a un repositorio se indica como una URL y se puede realizar de distintos modos dependiendo de la configuración del servidor subversion:
 - file:/// (acceso directo en disco local)
 - http:// (acceso utilizando el protocolo http)
 - https:// (acceso por medio del protocolo https)
 - svn:// (acceso por medio del protocolo propio de subversión)
 - svn+ssh:// (acceso por medio del protocolo svn a través de un túnel ssh)

4. Recomendaciones de organización típica de los datos de un proyecto



5. Ejemplo gráfico de la evolución de un proyecto

6. Comandos básicos

- *svn*, programa cliente desde línea de comandos
- *svnversion*, muestra la versión del programa
- *svnlook*, permite inspeccionar un repositorio
- *svnadmin*, permite crear o configurar un repositorio
- *svn help*, muestra ayuda sobre los comandos permitidos
- *svn help <comando>*, muestra la ayuda sobre un comando concreto
- *svn import*, copia una jerarquía de archivos sin versionar a un repositorio
- *svn checkout*, obtiene una copia de trabajo del proyecto
- *svn commit*, publica los cambios sobre directorios o ficheros
- *svn update*, actualiza los contenidos de la copia local
- *svn list*, muestra los datos almacenados
- *svn status*, muestra el estado de la copia local
- *svn diff*, muestra los cambios
- *svn add*, *svn delete*, *svn copy*, *svn move*,...

Ejemplos:

- Obtener una copia de trabajo de la última versión del *proyecto1* desde el *repositorio* situado en el disco local:

```
$ svn checkout file:///repositorio/proyecto1/trunk
```

- Publicar los cambios realizados en *fichero.c*, opcionalmente se incluye un comentario informativo para los logs del subversión:

```
$ svn commit fichero.c -m "Cambio realizado en fichero.c."
```

- Se realiza una actualización de la copia de trabajo, incluyendo los posibles cambios realizados por otros desarrolladores:

```
$ svn update
```

- Mostrar la lista directorios y archivos existentes en el repositorio y directorio indicado (en este caso se mostraría el contenido del directorio raíz del *proyecto1*):

```
$ svn list file:///repositorio/proyecto1
```

- Crear un nuevo tag o instantánea de la versión actual de la línea principal de desarrollo, opcionalmente se incluye un comentario informativo de la operación realizada para los logs del sistema:

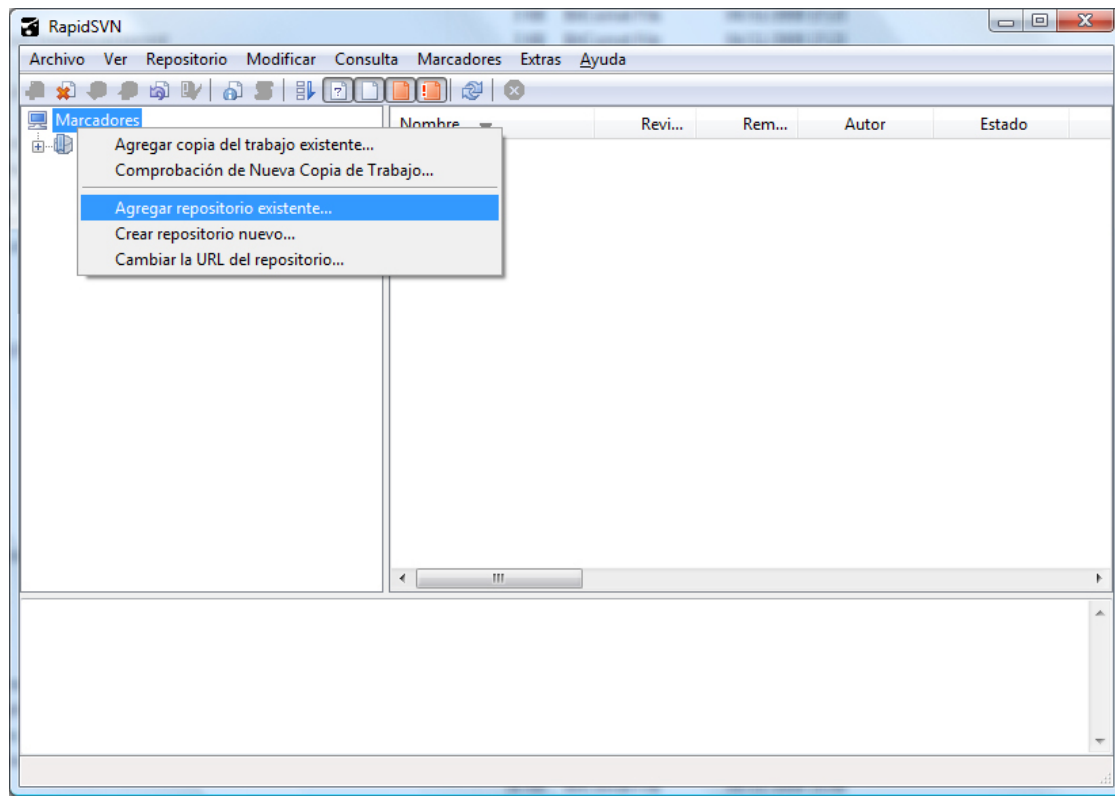
```
$ svn copy file:///repositorio/proyecto1/trunk \
    file:///repositorio/proyecto1/tags/tag-1.0 \
    -m "Etiqueta tag-1.0 del proyecto1"
```

- Como tarea inicial, sobre un repositorio vacío, introducir los archivos y directorios iniciales del proyecto desde una copia local:

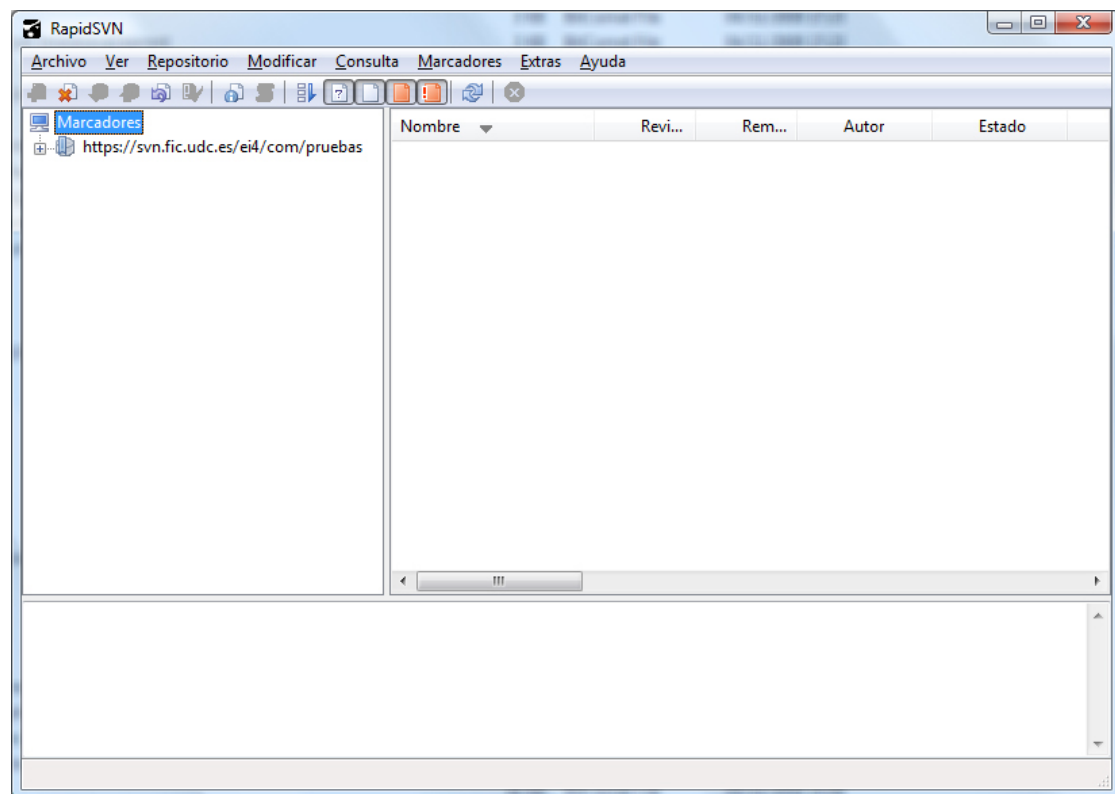
```
$ svn import proyecto3local file:///repositorio/proyecto3 \
    -m "Import inicial"
```

7. Pantallas de clientes gráficos

RapidSVN

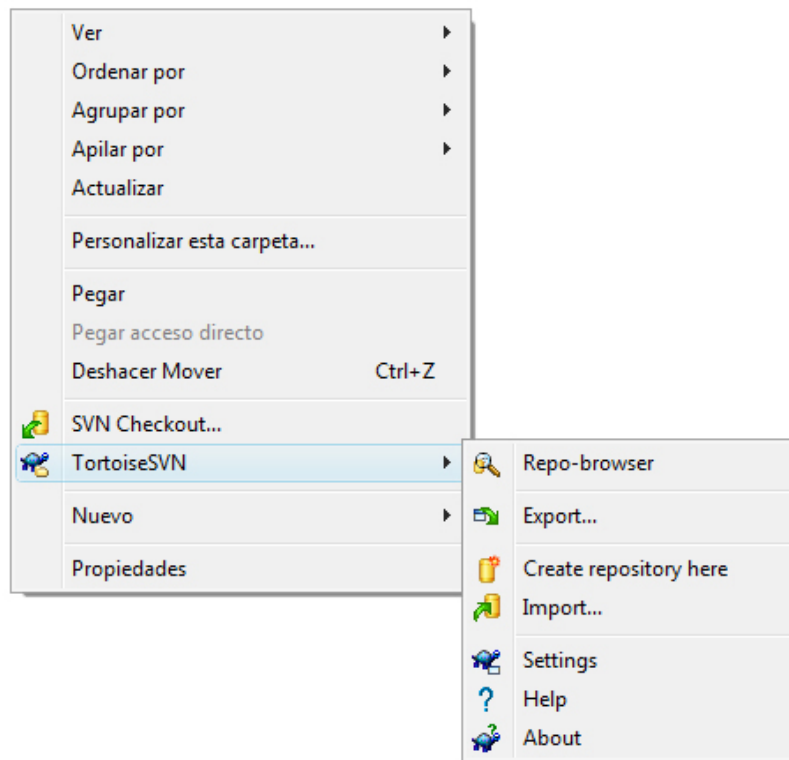


Ventana principal mostrando el submenú para agregar un repositorio

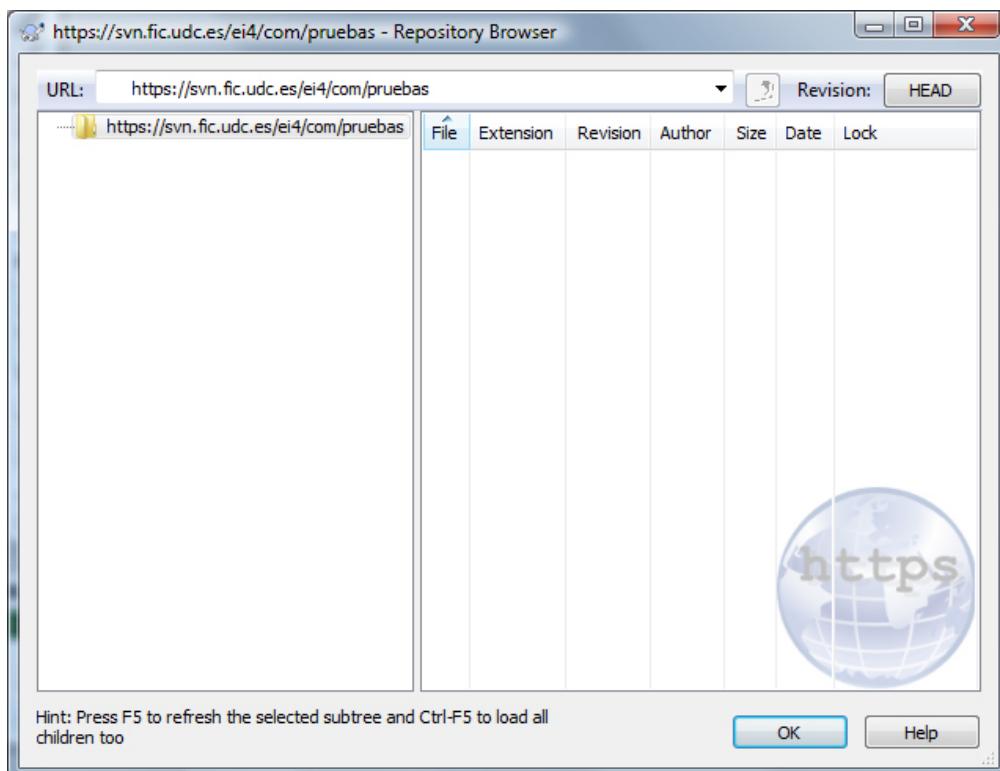


Ventana principal mostrando un repositorio ya agregado

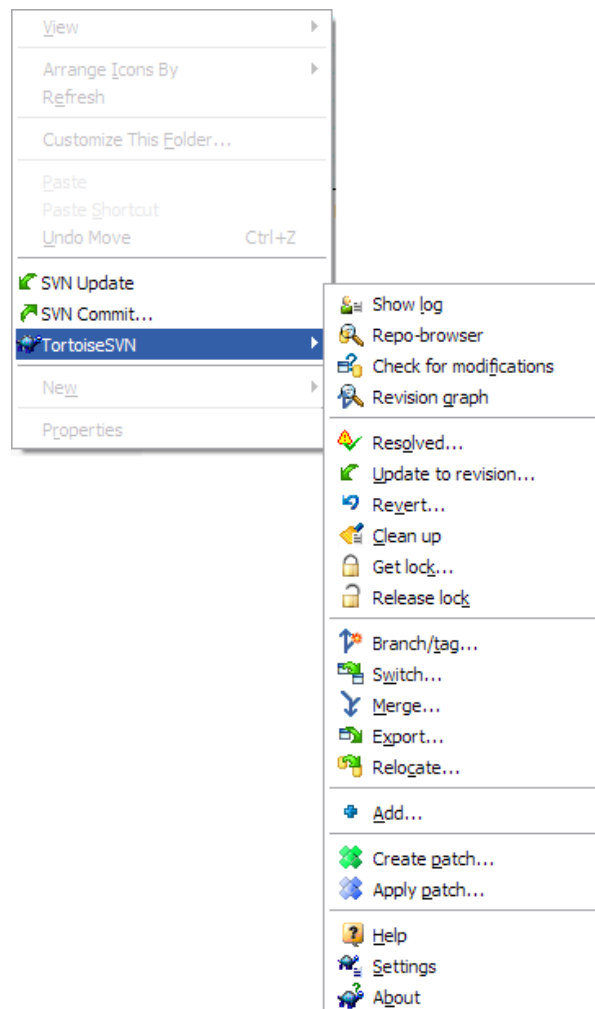
TortoiseSVN



Menú integrado en el explorador de archivos de Windows



Explorador de repositorios, se ha introducido la URL de la dirección del repositorio



Captura opciones menú integrado en el explorador de archivos de Windows