

Control de versiones con Subversion

Introducción

Diseño de Sistemas de Información 2009/10



MADS Group - Departamento de Computación

Víctor M. Gulías (gulias@udc.es)

David Cabrero (cabrero@udc.es)

Laura Castro (lcastro@udc.es)

Javier París (javierparis@udc.es)

David Martínez (dmartinez@udc.es)

24 de febrero de 2010

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

- 1 **Sistemas de control de versiones**
- 2 **Subversion: uso elemental**
- 3 **Subversion: características avanzadas**
- 4 **Conclusiones**
- 5 **Bibliografía**

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

Un mecanismo para:

- **Coordinar las modificaciones** de un conjunto de usuarios sobre un mismo conjunto de ficheros
- Seguir el **rastros de los cambios**, identificando:
 - *Quién* hizo qué
 - *Cuándo* lo hizo
 - *Por qué* lo hizo
 - *Qué más* se hizo

Por ello, también es de utilidad para un solo desarrollador

- **Recuperar versiones** anteriores

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

¿Cómo funciona?

- Existe un **repositorio común** (centralizado o distribuido) donde se almacenan todos los cambios/versiones
- Los desarrolladores trabajan en su **copia local** e incorporan las modificaciones al repositorio
- El sistema **registra las modificaciones** y los comentarios asociados
- El sistema **gestiona las distintas versiones** almacenadas en el repositorio

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

Concurrent Version System (CVS)

Sistemas de control de versiones

Control de versiones
con Subversion

DSI 2009/10

- Creado a partir del proyecto de Dick Grune (1986), siendo aún hoy un desarrollo activo de uso muy extendido
- Tiene algunas limitaciones ligadas a su diseño inicial que empujan a su reemplazo por sistemas alternativos, que solucionan algunos de sus problemas:
 - *Commits* atómicos
 - Identificador de archivo distinto de la ruta de archivo
 - Permiten trabajo *off-line*
 - Soporte de *branches* mejorado

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

- Arch (arch.fifthvision.net)
 - Gran conjunto de funcionalidades
 - Escrito en *shell scripts*
- GIT (www.kernel.org/git)
 - Usado en el desarrollo del *kernel* de Linux
 - Distribuido, no centralizado
 - Substituto de Bitkeeper
- Perforce (www.perforce.com)
 - Comercial
- **Subversion** (subversion.tigris.org)
 - Creado a semejanza de CVS, eliminado sus carencias

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

- Para cada proyecto, se **crea un repositorio**
- Cada desarrollador:
 - 1 **Obtiene una copia** del repositorio
(*copia local*)
 - 2 **Trabaja sobre su copia** local
 - 3 Eventualmente:
 - **Actualiza su copia** local con los cambios que otros desarrolladores han realizado
 - **Resuelve conflictos**
 - **Integra sus cambios**, enviándolos al repositorio



Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

- Administrador (`svnadmin`)
 - 1 Crear un repositorio
 - 2 *Opcionalmente:* poblar el repositorio
- Usuarios/desarrolladores (`svn`)
 - 1 Obtener una copia del repositorio
 - 2 Trabajar sobre la copia local
 - 3 Comprobar el estado del repositorio y de la copia local
 - 4 Integrar los cambios propios al repositorio
 - 5 Actualizar la copia local con los cambios que haya en el repositorio
 - 6 Resolver conflictos

Creación de un repositorio

Subversion: uso elemental

- En la /ruta/al/repositorio, crearemos un subdirectorio SVN

```
mkdir /ruta/al/repositorio/SVN
```

- Se usa la orden `svnadmin create` para inicializar el repositorio:

```
svnadmin create  
/ruta/al/repositorio/SVN/nombre-repositorio
```

- Se generará la estructura de gestión del repositorio

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

Creación de un repositorio (y II)

Subversion: uso elemental

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

- Sólo las órdenes `svnadmin` deben usarse para interactuar con esta estructura de gestión

```
/ruta/al/repositorio/SVN/  
nombre-repositorio/README.txt
```

- **URL** del repositorio:

```
file:///ruta/local/al/repositorio/  
SVN/nombre-repositorio  
http://ruta/remota/al/repositorio/  
SVN/nombre-repositorio
```

- La estructura de directorios es libre

Inserción de datos en el repositorio

Subversion: uso elemental

- Se pueden añadir ficheros al repositorio desde cero
- Pero si ya disponemos de una estructura inicial, podemos usarla para poblar el repositorio

```
svn import /ruta/ficheros/iniciales URL
```

- Registro de comentarios
- Variable de entorno EDITOR
- SVN trata cada tipo de fichero adecuadamente (texto, binario)

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

Descarga de una copia de trabajo

Subversion: uso elemental

- Para comenzar a trabajar es necesario descargar nuestra copia de trabajo personal

```
svn checkout URL [/ruta/de/trabajo]
```

- HEAD (última versión) del repositorio
- Directorio .svn
- Sobre la copia de trabajo los archivos pueden modificarse con normalidad

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

Manipulación básica de archivos:

- Añadir o eliminar ficheros del repositorio

```
svn add ELEMENTO  
svn delete ELEMENTO
```

- Añadir un directorio

```
svn mkdir RUTA
```

- Copiar o mover ficheros

```
svn copy ORIGEN DESTINO  
svn move ORIGEN DESTINO
```

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

Edición de la copia de trabajo (y II)

Subversion: uso elemental

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

- Consultar los cambios que se han hecho sobre la copia de trabajo local

```
svn status      (Modified, Added, Deleted...)
```

- Visualizar los cambios hechos respecto a la copia de partida

```
svn diff [ELEMENTO]
```

- Descartar cualquier cambio y volver a la copia de partida

```
svn revert ELEMENTO
```

Integración/actualización de cambios

Subversion: uso elemental

- Actualización de la copia de trabajo local

```
svn update
```

- Integración de nuestras modificaciones en el repositorio

```
svn commit
```

- Es importante **documentar** los cambios
- La nueva versión **debe funcionar**
- Consulta del registro de cambios

```
svn log [ELEMENTO]
```

Conflicto

Desde la descarga de la copia de trabajo (`checkout`) hasta el momento en que de nuevo interactuamos con el repositorio (`update/commit`), otro(s) desarrollador(es) puede(n) haber modificado y confirmado archivos también alterados por nosotros

Ante esta situación, el sistema no puede decidir, no puede corromper el repositorio:

- un `svn commit` fallará anulando la integración de nuestros cambios

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

Resolución de conflictos (y II)

Subversion: uso elemental

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

- Se debe ejecutar `svn update`, y como consecuencia, para cada archivo en conflicto,
 - en el archivo original se insertan marcadores del conflicto
 - se crean tres copias adicionales, conteniendo
 - una copia del archivo modificado (`.mine`)
 - una copia del archivo en el repositorio (`.rN`)
 - una copia del archivo en el momento del *checkout* (`.rM`, con $M < N$)
- El procedimiento a seguir consiste en resolver a mano el conflicto, eliminar los tres ficheros auxiliares y finalmente integrar los cambios definitivos mediante un `svn commit`

Otras operaciones

Subversion: uso elemental

- Descarga de copias de trabajo de versiones específicas
 - `svn -r 2 checkout URL [RUTA]`
- Comparación de versiones concretas
 - `svn -r 2:4 diff [ELEMENTO]`
- Ayuda:

```
svn help [COMANDO]
svnadmin help [COMANDO]
```

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

Trabajo en red

Subversion: características avanzadas

- El uso normal de Subversion involucra a un conjunto de desarrolladores distribuidos y trabajando de forma concurrente
- Aparte de los repositorios locales, es necesario proveer de modos de acceso en red
 - svnservce
 - Ligero, simple, limitado...
 - svn://
 - Apache + WebDAV
 - Versátil, control de acceso, SSL...
 - http://

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

Ramas de desarrollo

Subversion: características avanzadas

- Una *rama* o *branch* en un desarrollo es una línea de trabajo que coexiste con otras con las que comparte un pasado común
- Ejemplo: una vez alcanzada una versión estable de un programa es deseable dividir el repositorio de forma que una línea de desarrollo de dedique a pequeñas correcciones sobre dicha versión 1 (mantenimiento), mientras que otra avance en el desarrollo de la siguiente versión

```
svn copy RUTA RUTA-NUEVA-RAMA
```

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía

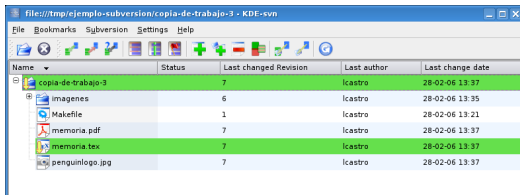
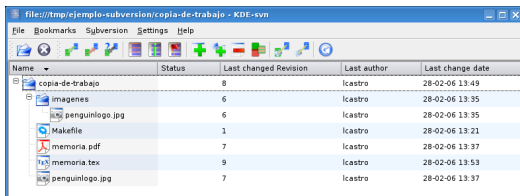
Otras herramientas

Subversion: características avanzadas

Control de versiones
con Subversion

DSI 2009/10

- Clientes gráficos
 - eSvn, rapidsvn, websvn, KDEsvn...



Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

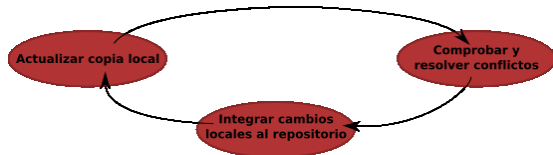
Subversion:
características
avanzadas

Conclusiones

Bibliografía

Prácticas de la asignatura:

- Descarga + ciclo de desarrollo



- Problemas más frecuentes:
 - Olvidarse de integrar ficheros nuevos al repositorio
 - Integrar cambios que dejan al repositorio en situación inestable

Contenido

Sistemas de control de versiones

Subversion: uso elemental

Subversion: características avanzadas

Conclusiones

Bibliografía

Más información:

- [svn05a] [Subversion project home](http://subversion.tigris.org).
<http://subversion.tigris.org>, 2005.
- [svn05b] [Version control with subversion](http://svnbook.red-bean.com).
<http://svnbook.red-bean.com>, 2005.

Contenido

Sistemas de control de
versiones

Subversion: uso
elemental

Subversion:
características
avanzadas

Conclusiones

Bibliografía