

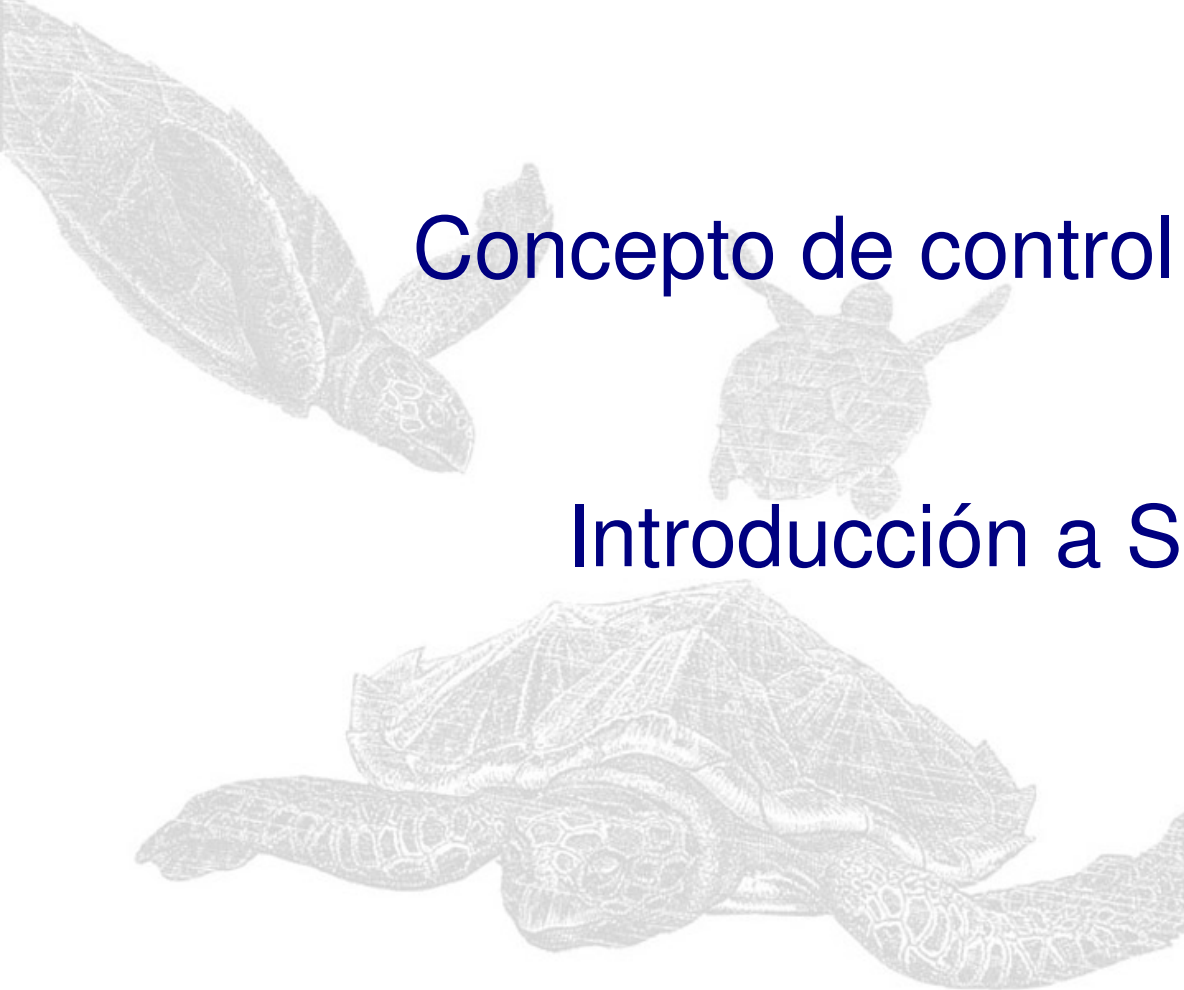
Control de versiones con Subversion



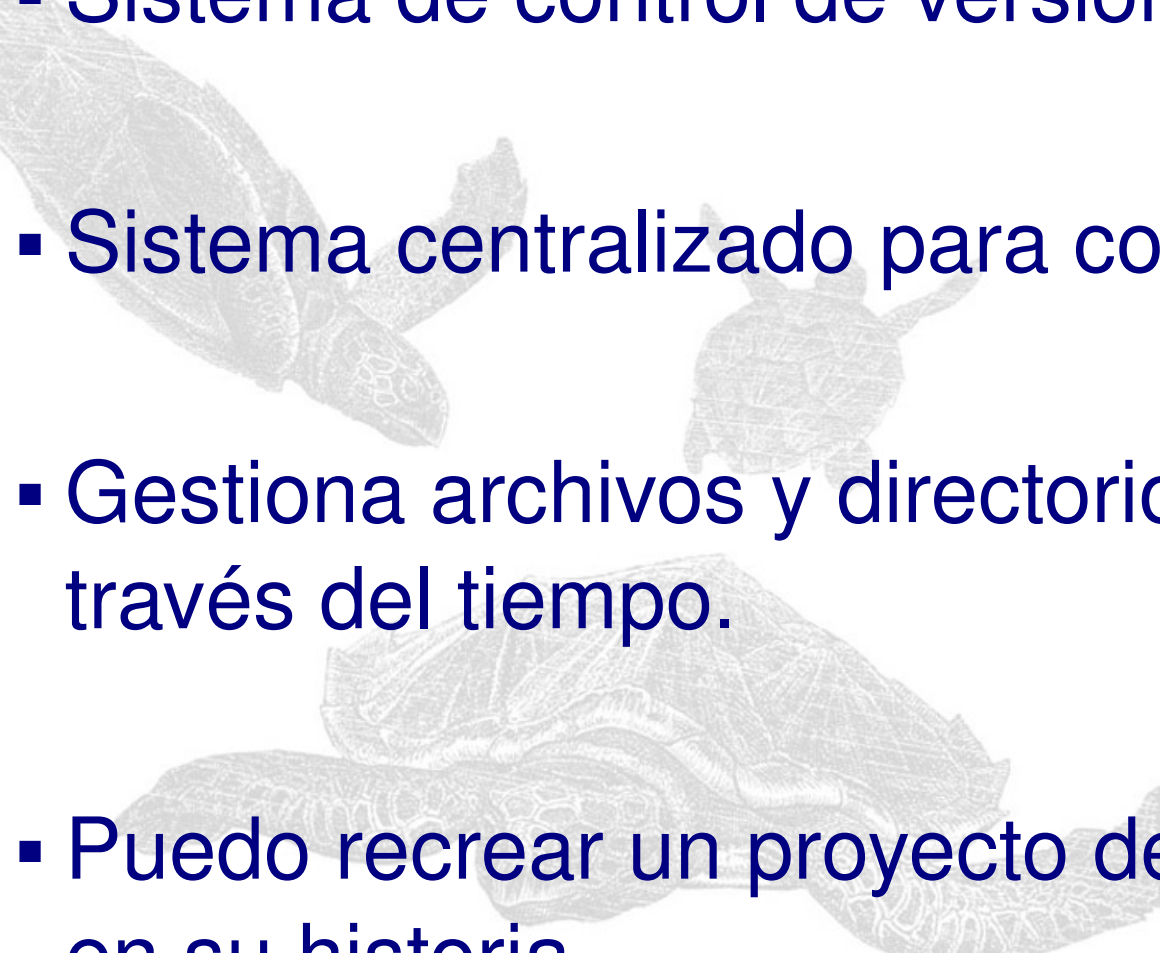
Conceptos fundamentales

Concepto de control de versiones.

Introducción a Subversion



¿Qué es Subversion?

- Sistema de control de versiones.
 - Sistema centralizado para compartir información.
 - Gestiona archivos y directorios, y sus cambios a través del tiempo.
 - Puedo recrear un proyecto desde cualquier momento en su historia
- 

Conceptos

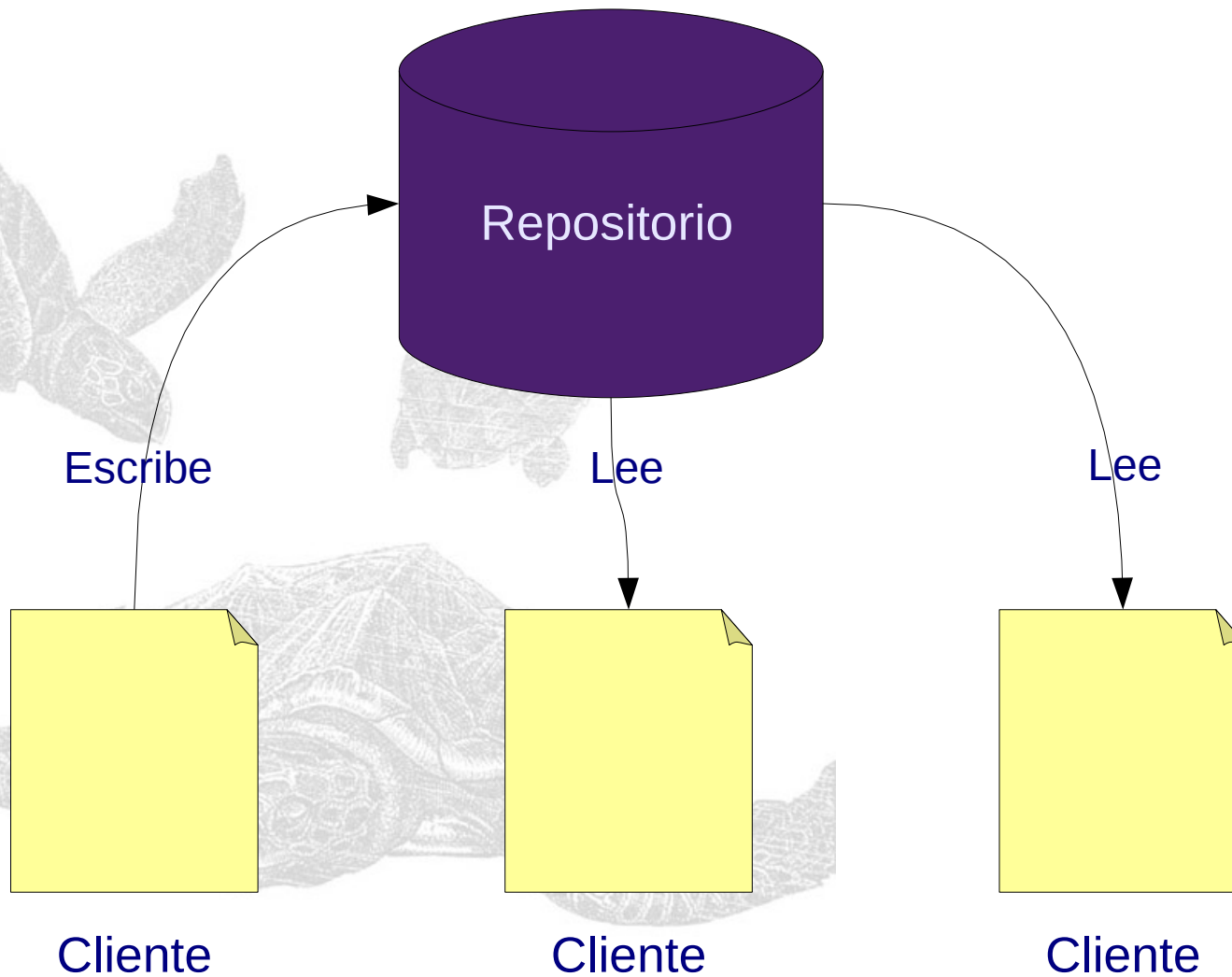
- Un **repositorio**, depósito o archivo es un sitio centralizado donde se almacena y mantiene información digital, habitualmente bases de datos o archivos informáticos.
- Subversion es un repositorio en forma de árbol con una jerarquía de directorios y archivos.

Conceptos

- n clientes se conectan al repositorio para leer o escribir.
- Al escribir, un cliente hace disponible la información para otros.
- Al leer, el cliente recibe información de otros.



Modelo cliente-servidor



Gestionar cambios a través del tiempo

- Subversion recuerda cada cambio que se haya realizado en el repositorio.
- Recuerda cambios realizado a cada archivo así como cambios en el árbol de directorios:
 - Archivos y directorios nuevos
 - Archivos y directorios borrados
 - Archivos y directorios modificados o cambiados de lugar.

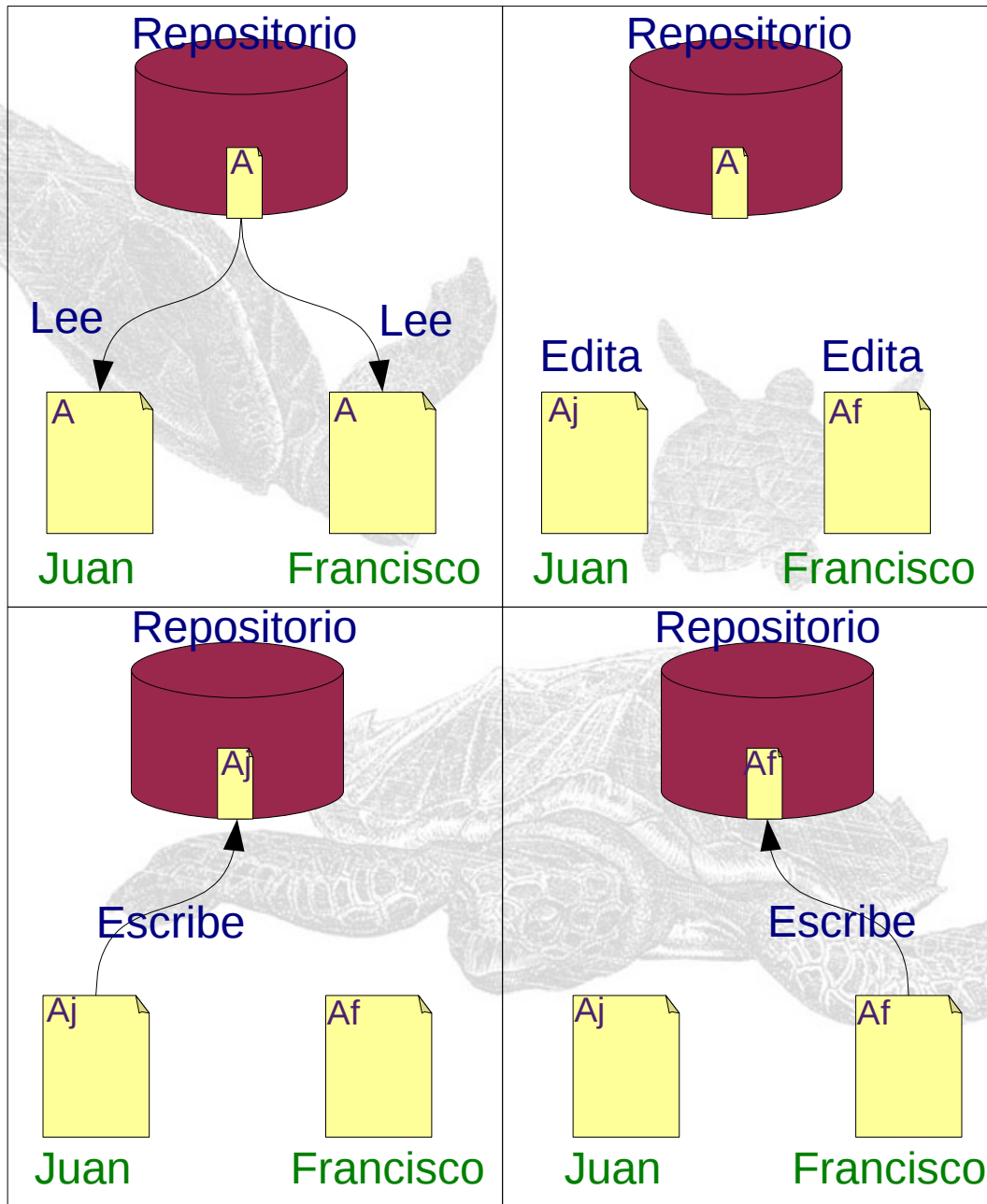
Gestionar cambios a través del tiempo

- Generalmente un cliente lee la versión más reciente del árbol de directorios y archivos.
- Subversion provee la habilidad de leer estados anteriores del sistema de archivos.
- Se pueden conocer los cambios realizados, cuándo se realizaron, y quién realizó dichos cambios.

Trabajo colaborativo

- El objetivo de un sistema de control de versiones es el de permitir editar de forma colaborativa y compartir información.
- Existen diferentes estrategias.
- **Un problema a resolver:** Editar la misma información sin pisar el trabajo de otro.

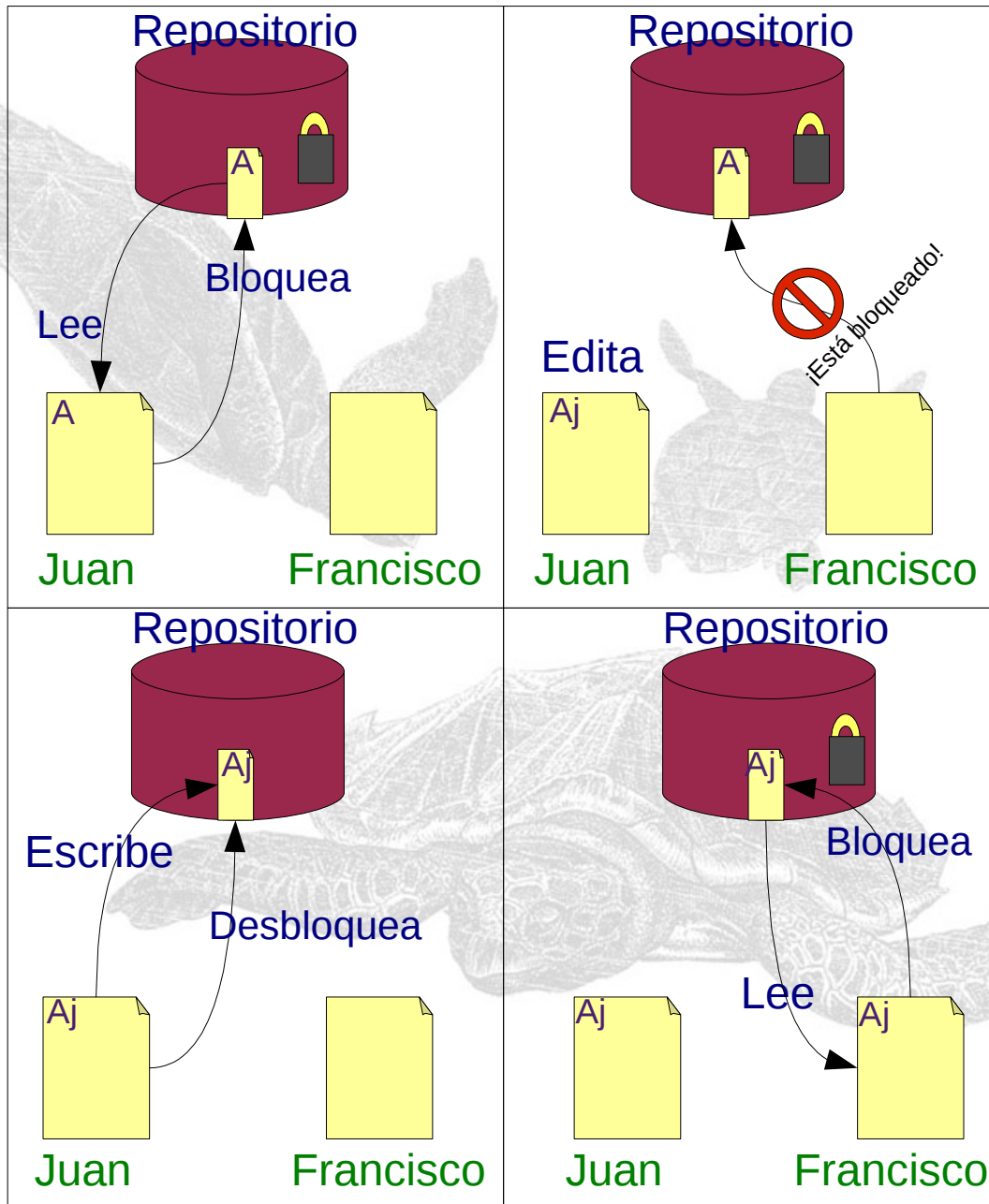
Problema a resolver



- Juan y Francisco leen un archivo en su versión más reciente desde el repositorio.
- Juan edita el archivo con sus cambios.
- Francisco edita el archivo con sus cambios.
- Juan manda sus cambios al repositorio.
- Francisco manda sus cambios al repositorio y sobrescribe los cambios de Juan.

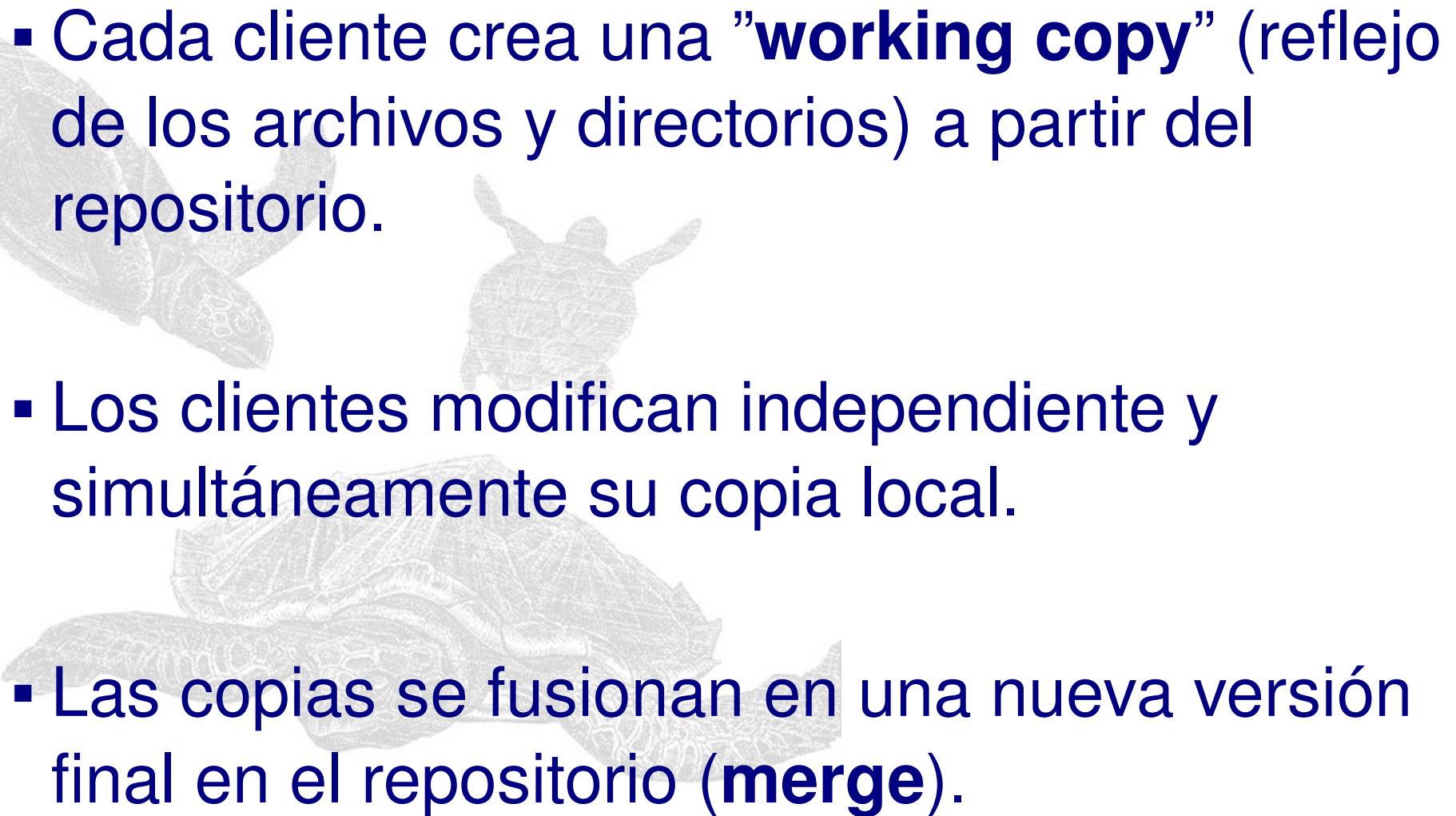
El repositorio recuerda los cambios que hizo Juan, pero la versión de Francisco no los incluye. La última versión del archivo en el repositorio tiene los cambios hechos por Francisco, pero no los cambios hechos por Juan.

Opción: bloquear-modificar-desbloquear



- Una sola persona puede trabajar en cada archivo a la vez.
- Es una solución con limitaciones:
 - **Problemas administrativos:** Mientras Juan edita el archivo, si Francisco lo necesita, tiene que esperar a que Juan termine. Pierde tiempo.
 - Francisco podría querer editar el final de un archivo de texto y Juan el principio del mismo. Podrían trabajar simultáneamente sin necesidad de esperar que el otro termine.
 - Falso sentido de seguridad. Francisco podría editar un archivo B y Juan un archivo A, que dependan uno del otro. El bloqueo simula seguridad, cuando los cambios podrían comprometer la integridad del código.

Solución: Copiar-Modificar-Merge

- Cada cliente crea una "**working copy**" (reflejo de los archivos y directorios) a partir del repositorio.
 - Los clientes modifican independiente y simultáneamente su copia local.
 - Las copias se fusionan en una nueva versión final en el repositorio (**merge**).
- 

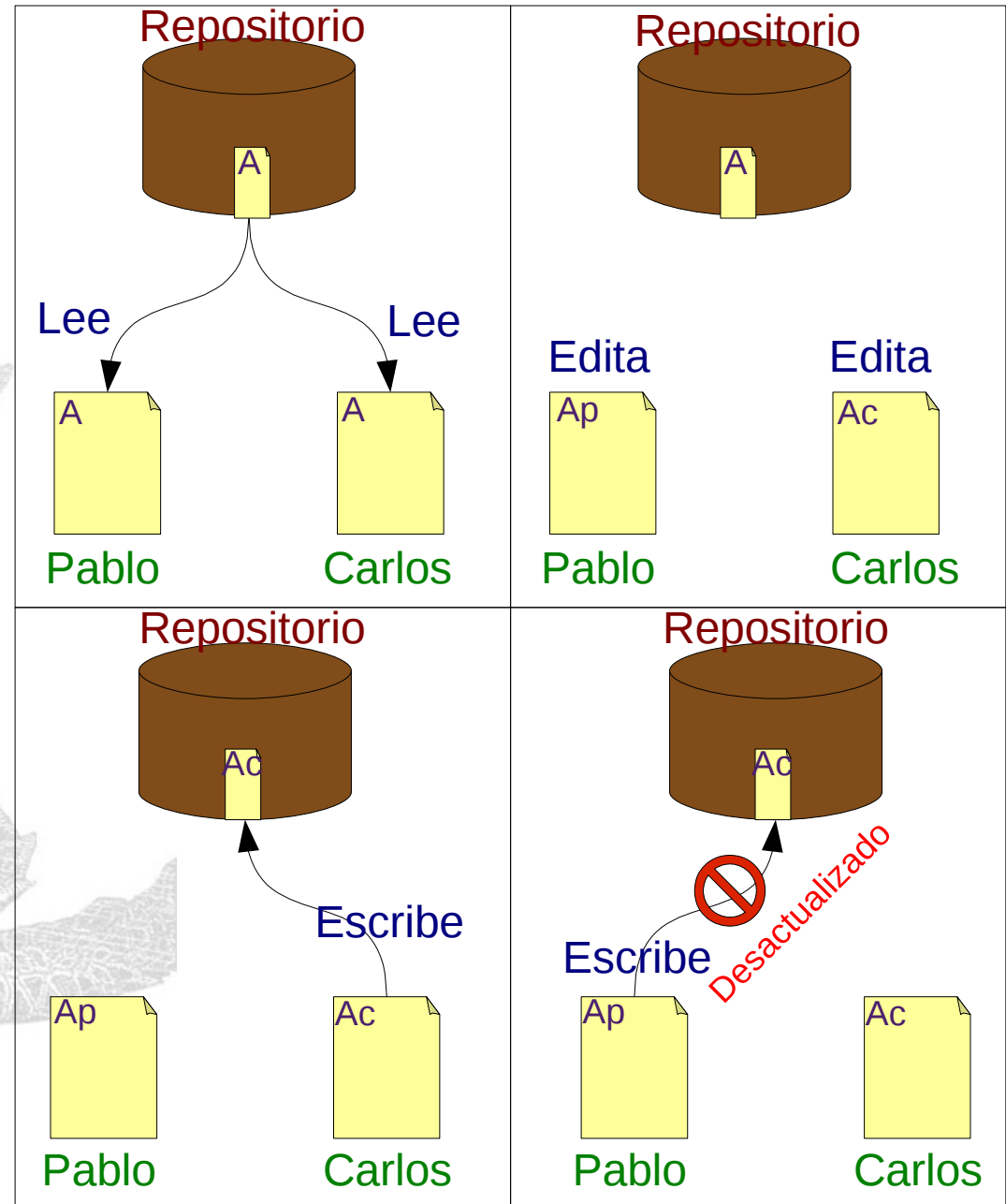
Solución: Copiar-Modificar-Merge

- El sistema de control de versiones es un asistente para la fusión (**merge**), pero **el usuario es el responsable de que la fusión se realice correctamente.**



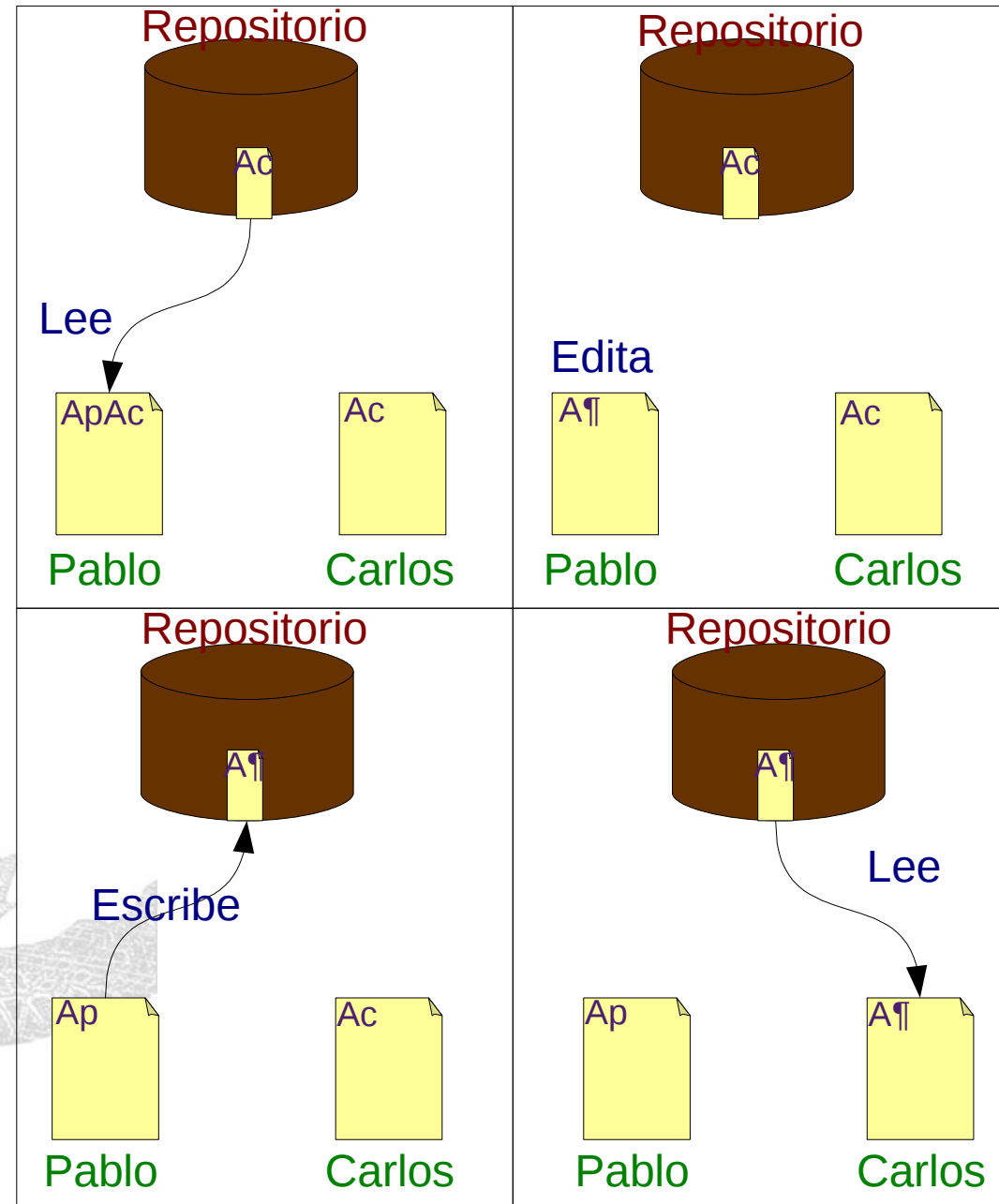
Solución: Copiar-Modificar-Merge

- Pablo y Carlos copian el mismo archivo.
- Editan sus archivos.
- Carlos publica su versión antes.
- Pablo obtiene un error de "desactualizado"



Solución: Copiar-Modificar-Merge

- Pablo compara la última versión con la suya.
- Se crea una nueva versión A¶ "mergeada" (fusionada).
- Se publica la versión "mergeada".
- Ahora los dos usuarios tienen los cambios de cada uno.

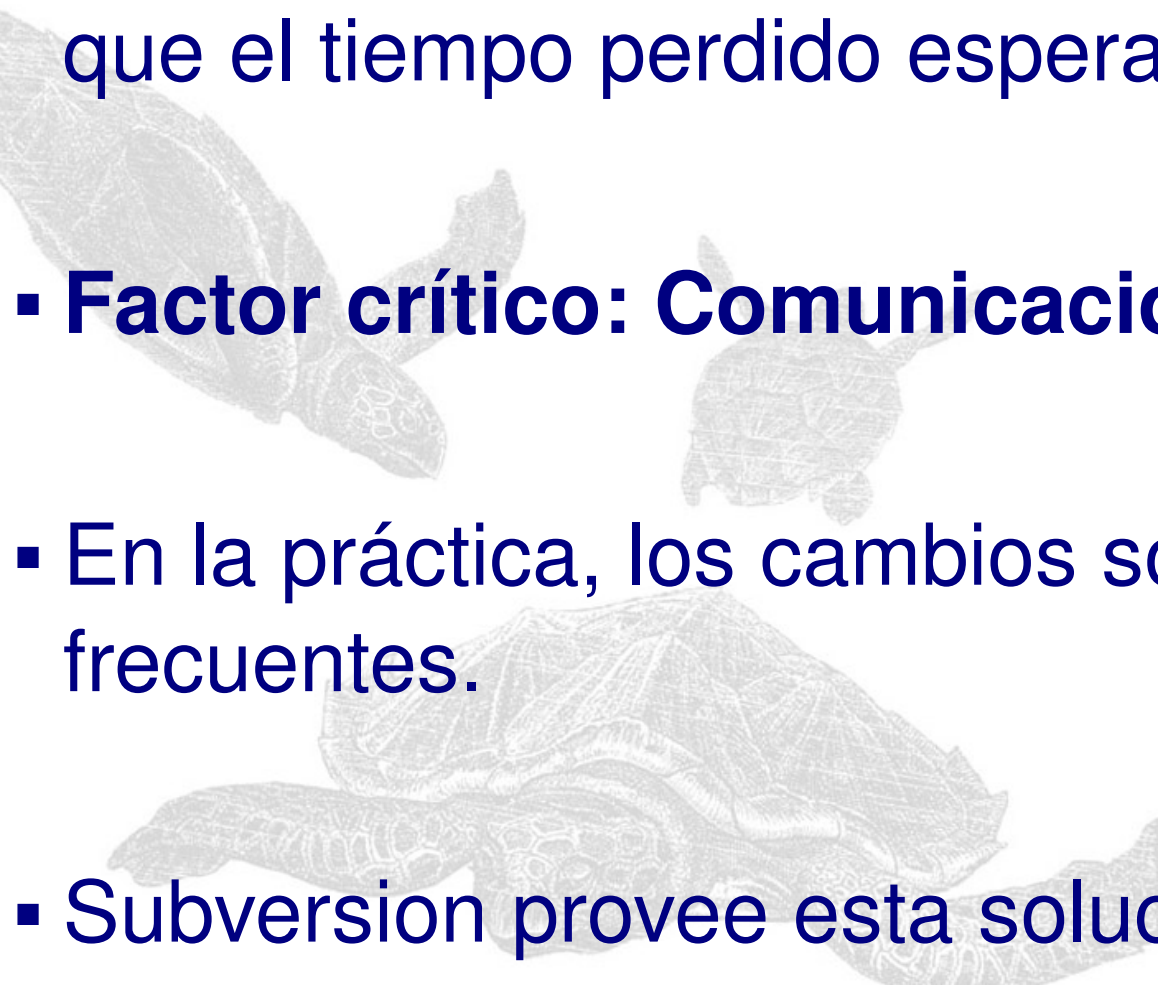


Solución: Copiar-Modificar-Merge

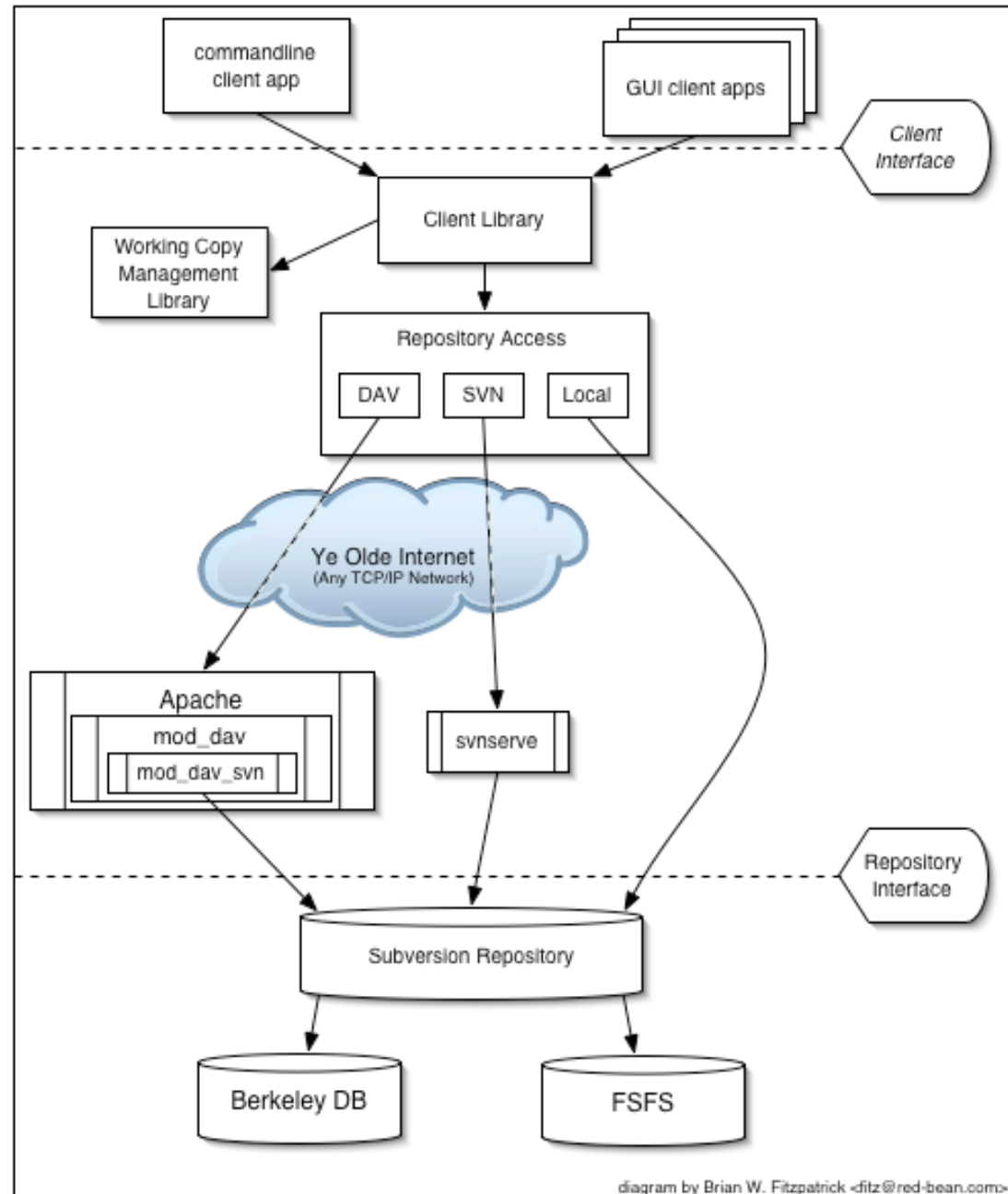
- Los cambios de ambos usuarios se fusionan.
- En caso de haber solapamiento en los cambios, el sistema informa de un conflicto.
- El archivo es marcado por el sistema, y el usuario puede ver los cambios en conflicto y elegir el cambio manualmente.

Nota: El software no es capaz de resolver conflictos automáticamente (todavía), solo los programadores pueden tomar las decisiones "inteligentes" en estos casos (por ahora). Esta afirmación puede ser rebatida por usuarios de GeneXus...

Solución: Copiar-Modificar-Merge

- El tiempo usado en resolver conflictos resulta menor que el tiempo perdido esperando por bloqueos.
 - **Factor crítico: Comunicación entre los usuarios.**
 - En la práctica, los cambios solapados son muy poco frecuentes.
 - Subversion provee esta solución, y la opción de bloquear archivos (conveniente por ejemplo para archivos binarios).
- 

Arquitectura de Subversion



Componentes de Subversion

- **svn**

El programa cliente de línea de comando.

- **svnversion**

Programa para reportar el estado (en términos de revisiones de los ítems presentes) de una copia en funcionamiento.

- **svnlook**

Herramienta para inspeccionar un repositorio Subversion.

Componentes de Subversion

- **svnadmin**

Herramienta para crear, configurar, o reparar un repositorio.

- **mod_dav_svn**

Módulo plugin para el servidor Apache, para usar el repositorio a través de la red.

- **svnserve**

Otro programa para usar el repositorio a través de la red.

Componentes de Subversion

- **svndumpfilter**

Programa para filtrar dump streams de repositorios de Subversion.

- **svnsync**

Programa para hacer mirrors de un repositorio a otro por la red.



URL de repositorios

Los repositorios Subversion son accesibles a través de diferentes protocolos en un disco duro, a través de la red, etc.

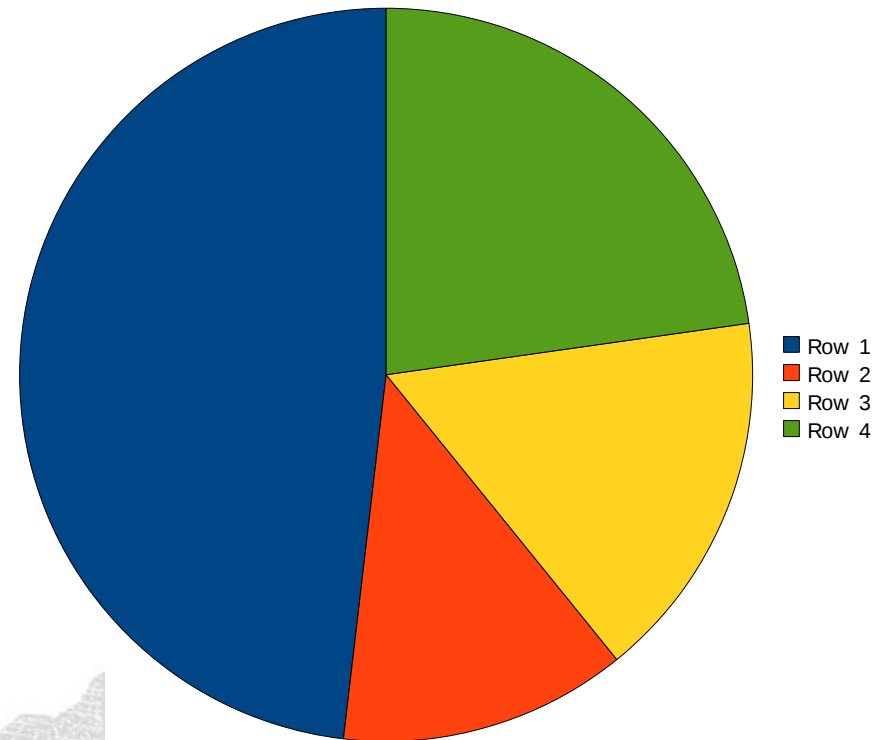
En última instancia, la dirección de un repositorio Subversion es siempre un URL.

Tipo de URL – Forma de acceso

file:///	Acceso directo en disco local.
http:// Subversion.	Acceso via protocolo WebDAV a servidor Apache conciente de
https://	Igual que http://, pero con cifrado SSL.
svn://	Acceder a través de protocolo propio a servidor svnserv.
svn+ssh://	Igual que svn:// pero por túnel SSH.

Datos

- Esta es una gráfica con datos.
- No podía faltar una gráfica con datos en la presentación.



Copia local – working copy

- Es un árbol de directorios y archivos común y corriente en mi computadora, obtenido desde el repositorio.
- Puedo editar, compilar si son fuentes, borrar, etc.
- La copia local es privada: no se aplican mis cambios en el servidor ni los cambios de otros en mi copia a menos que yo lo pida.
- Puedo tener muchas copias del mismo repositorio.

Copia Local

- Luego de hacer cambios, podemos "publicar" (**commit**) los cambios en el repositorio.
- Si otro publicó cambios, podemos fusionarlos en nuestra copia (**merge**).
- En nuestra copia se van a guardar los archivos de configuración bajo el directorio `.svn`

A trabajar...

- Para obtener una copia local del repositorio, hay que hacer un **checkout**:

```
svn checkout svn://192.168.1.42/home/svn/ejemplo
```

- Para publicar nuestros cambios, hacemos un **commit**:

```
svn commit Clase.java -m "Arreglé un bug del setter"
```

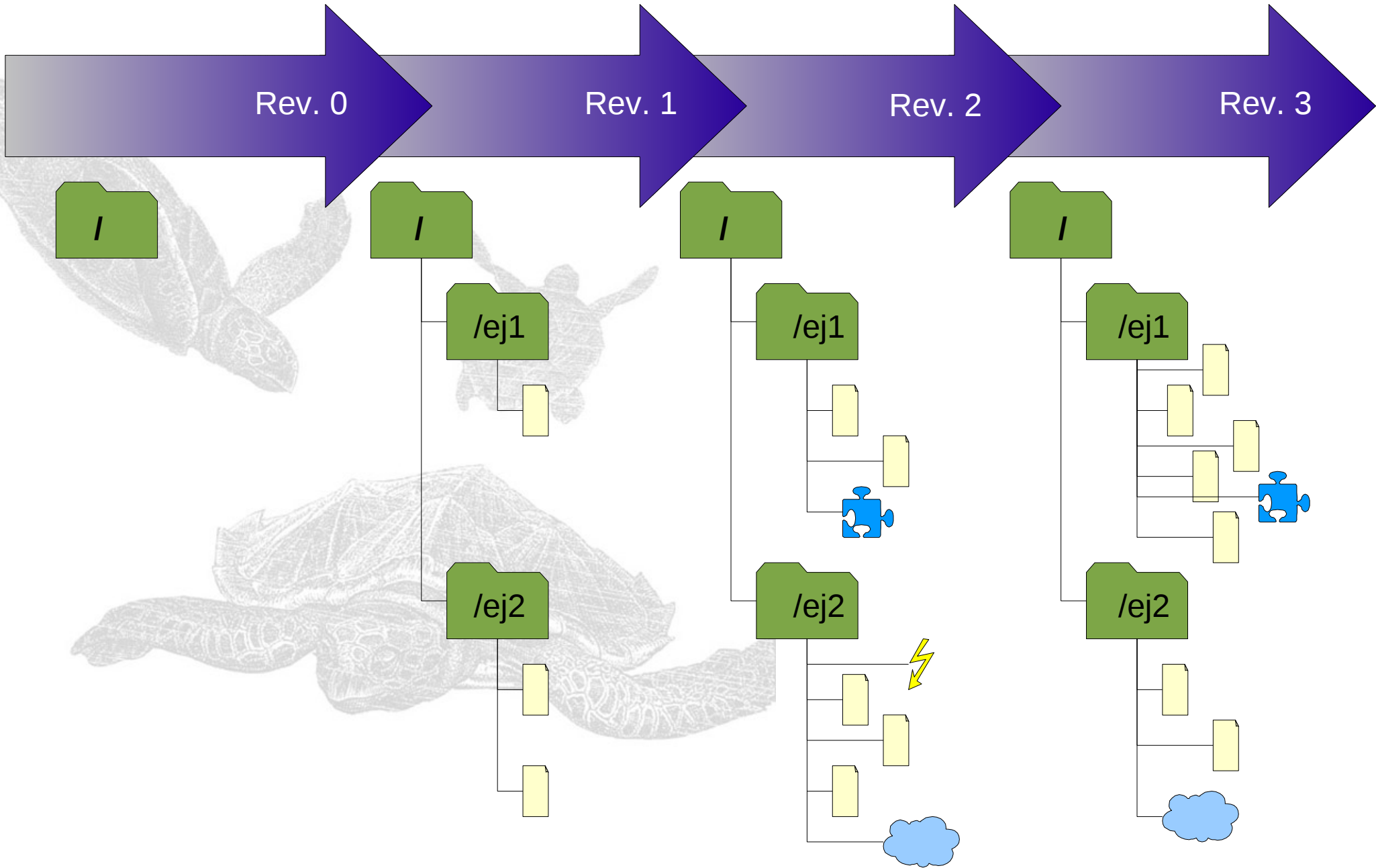
- Para actualizar nuestro repositorio respecto a la versión del servidor, usamos **update**:

```
svn update
```

Revisiones

- **Commit** publica TODOS los cambios de los archivos y directorios como una transacción atómica ("commitea" todo o nada).
- Al aceptar un commit, se crea un nuevo estado del árbol del sistema de archivos, llamado una **revisión**.
- Cada revision tiene asignado un número único, natural, mayor en uno al número de la revisión anterior.

Revisiones



Revisiones

- En Subversion, el número de revisión se aplica a árboles enteros, no a archivos individuales (no así en otros sistemas de control de versiones).
- Cada número de revisión es un estado particular del repositorio tras haber sido commiteado un cambio.
- “La revisión 42 de HolaMundo.java” significa “HolaMundo.java como aparece tras el cuatrigésimo segundo commit del repositorio.”

Revisiones

Nota: Las copias locales no siempre corresponden a una sola revisión en el repositorio, pueden tener archivos de varias revisiones distintas.

Para cada copia local, Subversion guarda esta información en el directorio **.svn**:

- En qué revisión está basada la copia local (llamado **working revision**)
- Una fecha de cuándo se actualizó la copia local por el repositorio

Revisiones

- **svn status --verbose**

Con este comando podemos conocer el estado de nuestra copia local.

- Las copias locales contienen archivos de varias revisiones.

- Los updates y commits son cosas separadas.

- Cada vez que "commiteamos", nuestra copia local termina con una mezcla de revisiones

Resumen

Hasta acá:

- Noción de repositorio central, copia local del cliente, árbol de revisiones.
- Modelo de trabajo colaborativo "copiar-modificar-mergegear".
- Concepto de la forma de trabajo de Subversion.

Esta presentación es libre

Copyright © 2008 – Fernando Briano - <http://picandocodigo.net>

Esta presentación es un trabajo derivado de:

Version Control with Subversion - Copyright © 2002-2008 Ben Collins-Sussman, Brian W. Fitzpatrick, C. Michael Pilato.

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

Licencia del trabajo original:

Creative Commons Attribution License

<http://creativecommons.org/licenses/by/2.0/>

Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

Este trabajo se publica bajo la siguiente licencia:

Creative Commons Attribution License

<http://creativecommons.org/licenses/by/3.0/>

Usted es libre de:

- copiar, distribuir y comunicar públicamente la obra
- hacer obras derivadas

Bajo las condiciones siguientes:

- Reconocimiento. Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).
- Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.
- Alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor
- Nada en esta licencia menoscaba o restringe los derechos morales del autor.