

# **TortoiseMerge**

**Una herramienta de diferencias/  
fusión para Windows**

**Version 1.14**

**Stefan Küng  
Lübbe Onken  
Simon Large**

---

# **TortoiseMerge: Una herramienta de diferencias/fusión para Windows: Version 1.14**

por Stefan Küng, Lübbe Onken, y Simon Large

Traducción: Fernando P. Najera Cano (yo@FernandoNajera.com), Juan Carlos Fuentes Michel (jcfuentes@gmail.com), Tomás Christie (tommychristie@gmail.com), Ariel Abaca (abacrotto@hotmail.com)

fecha de publicación 2022/09/06 19:53:28 (r29447)

---

---

# Tabla de contenidos

Prefacio .....	vi
1. ¡TortoiseMerge es gratis! .....	vi
2. Reconocimientos .....	vi
1. Introducción .....	1
1.1. Descripción .....	1
1.2. Historia de TortoiseMerge .....	1
2. Conceptos básicos de control de versiones .....	3
2.1. Viendo y fusionando diferencias .....	3
2.2. Editando conflictos .....	3
2.3. Aplicando parches .....	4
3. Usando TortoiseMerge .....	5
3.1. Modos de visión .....	5
3.1.1. Viendo / Fusionando .....	5
3.1.2. Aplicando parches .....	7
3.2. Botones y otros controles .....	8
3.3. Iconos de estado de línea .....	9
3.4. Fusionando / Editando conflictos .....	9
3.5. Abrir ficheros .....	10
3.5.1. Viendo / Fusionando .....	11
3.5.2. Aplicando parches .....	11
3.6. Configuración .....	12
3.6.1. Página de Configuración principal .....	12
3.6.2. Página de Configuración de colores .....	13
A. Keyboard Shortcuts .....	15
A.1. Keyboard Shortcuts .....	15
A.1.1. General Keyboard Shortcuts .....	15
A.1.2. Diff-Mode Keyboard Shortcuts .....	16
A.1.3. Conflict Resolution Mode Keyboard Shortcuts .....	16
B. Automatizando TortoiseMerge .....	17
B.1. Opciones de Línea de Comandos de TortoiseMerge .....	17
Glosario .....	19

---

## Lista de figuras

1.1. Conflicto de ficheros .....	1
2.1. Conflicto de ficheros .....	4
3.1. Vista de un panel .....	5
3.2. Vista de dos paneles .....	6
3.3. Vista de tres paneles .....	7
3.4. Lista de ficheros de parche .....	8
3.5. El diálogo Abrir .....	11
3.6. La página de Configuración principal .....	12
3.7. La página de Configuración de colores .....	13

---

## Lista de tablas

B.1. Lista de las opciones de línea de comandos disponibles .....	17
---	----

---

# Prefacio



## Tortoise Merge

Cuando trabaja en un proyecto, bien sea un proyecto de código abierto donde varias personas revisan su código, o un proyecto comercial donde vende un componente con su código fuente, habrá gente y clientes que pueden encontrar errores o quizás quieran mejorar su proyecto. Para hacerlo, ellos le enviarán parches que tendrá que revisar y decidir si desea aplicarlos.

TortoiseMerge le ayuda con ambas tareas:

- Revisando parches
- Aplicando los parches

TortoiseMerge también le ayuda a poner en orden los ficheros con conflictos, que pueden ser ocasionados si trabaja con un sistema de control de código fuente y otros están trabajando en los mismos ficheros que usted.

## 1. ¡TortoiseMerge es gratis!

TortoiseMerge es gratis. No tiene que pagar por él, y puede utilizarlo de la forma que desee. Se desarrolla bajo la licencia GP (GPL).

TortoiseMerge is an Open Source project. That means you have full read access to the source code of this program. You can browse it on this link <https://osdn.net/projects/tortoisesvn/scm/svn/>. The most recent version (where we're currently working) is located under `/trunk/`, and the released versions are located under `/tags/`.

### ¡Donar!

Even though TortoiseSVN and TortoiseMerge are free, you can support the developers by sending in patches and playing an active role in the development. You can also help to cheer us up during the endless hours we spend in front of our computers.

While working on TortoiseSVN we love to listen to music. And since we spend many hours on the project we need a *lot* of music. Therefore we have set up some wish-lists with our favourite music CDs and DVDs: <https://tortoisesvn.net/donate.html> Please also have a look at the list of people who contributed to the project by sending in patches or translations.

## 2. Reconocimientos

Stefan Küng

por el duro trabajo en TortoiseMerge

Lübbe Onken

por los bonitos iconos, el logotipo, detección de errores y cuidar de la documentación

El proyecto Tigris Style

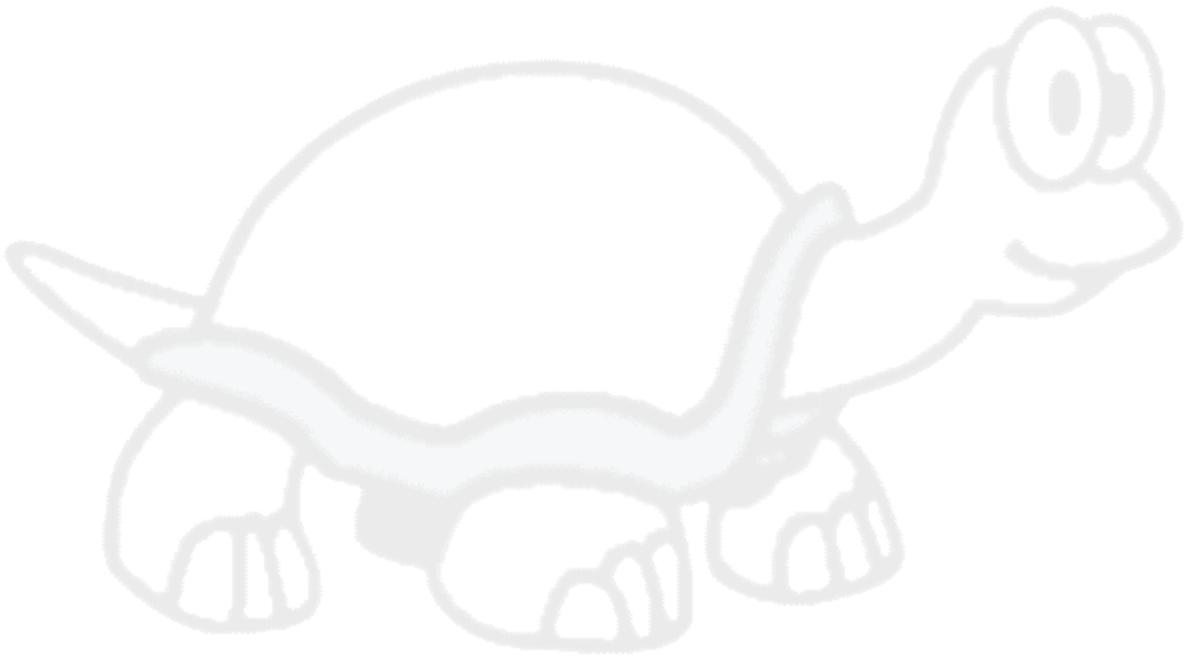
por algunos de los estilos que están siendo reutilizados en esta documentación

Nuestros colaboradores

por los parches, informes de errores y nuevas ideas, y por ayudar a otros respondiendo preguntas de nuestra lista de correo.

Nuestros donantes

por la cantidad de horas de entretenimiento con la música que nos enviaron



---

# Capítulo 1. Introducción

## 1.1. Descripción

TortoiseMerge es una aplicación gratuita/de código abierto. Le permite ver las diferencias entre ficheros de texto, fusionar esos cambios e incluso revisar y aplicar ficheros de diff unificados, a menudo llamados *patches*.

Mientras trabaja con ficheros de texto, por ejemplo ficheros de código en su lenguaje de programación favorito, o ficheros html/xml para su documentación o sitio web, a menudo tendrá situaciones donde desee comparar versiones diferentes de esos ficheros. A veces obtiene una versión diferente de alguien, a veces querrá ver los cambios que ha hecho.

Si está trabajando con un sistema de control de versiones (por ejemplo Subversion) a veces obtendrá *conflictos* cuando actualice su copia de trabajo. Esto ocurre si alguien ha modificado las mismas partes que usted del fichero en el que está trabajando. En esos casos tendrá que resolver esos conflictos manualmente. Aquí es donde TortoiseMerge puede ayudarle.

El siguiente diagrama le muestra las relaciones entre los ficheros involucrados en un conflicto:

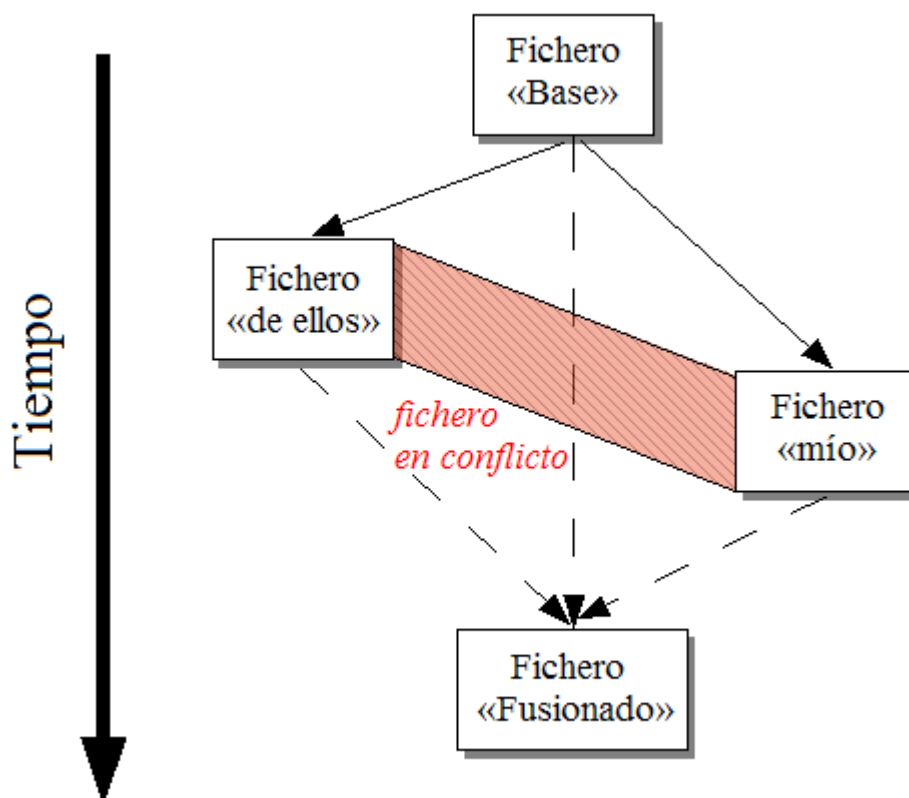


Figura 1.1. Conflicto de ficheros

## 1.2. Historia de TortoiseMerge

While working on *TortoiseSVN* [<https://tortoisesvn.net>] we found that whenever users sent in patches to enhance the project or just to fix bugs, we had a lot of work just applying those patches. And many times those patches were outdated and couldn't be applied with the command line patch tool <https://unxutils.sourceforge.net/> because we were working on those files and had sometimes changed those files already.



Esta era también la razón principal por la que TortoiseSVN no tenía implementada una opción TortoiseSVN → **Crear Parche**: preferíamos que los usuarios nos enviaran los ficheros completos porque así era más fácil aplicar los parches.

Así que gastamos mucho tiempo buscando en internet una herramienta GUI que pudiera aplicar ficheros de parche, pero todo lo que encontramos es que dicha herramienta no existía. Para acortar un poco una larga historia: decidimos que si nadie más podía proporcionarnos dicha herramienta tendríamos que escribirla nosotros. Ahí fue cuando nació TortoiseMerge.

Dado que TortoiseSVN ya utiliza la librería de Subversion, que contiene un buen motor de diferencias, era natural que utilizáramos esa librería en vez del diff de GNU.

Subversion 1.7 introdujo el *svn patch* command which allows to apply a patch file to a working copy. TortoiseMerge uses now the same functions as the Subversion command to avoid different behaviors between the two.

---

# Capítulo 2. Conceptos básicos de control de versiones

TortoiseMerge tiene diferentes usos:

- Viendo las diferencias entre dos ficheros y fusionando los cambios en uno de ellos, o eliminando cambios no deseados.
- Editando conflictos entre cambios locales y cambios en el repositorio de Subversion tras una actualización.
- Applying patch files and reviewing them.

## 2.1. Viendo y fusionando diferencias

En este modo está comparando dos ficheros. El fichero del panel izquierdo se considera el fichero original (a veces referido como *de ellos*) y el fichero en el panel derecho es el fichero modificado (a veces referido como *mío*).

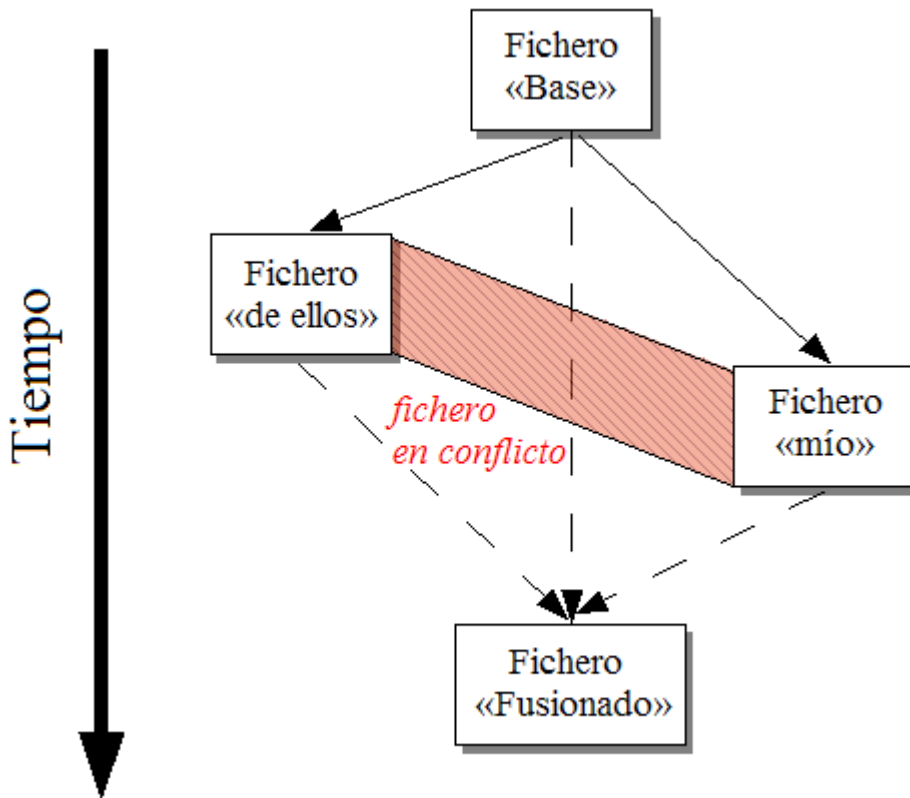
Puede efectuar cambios simples basados-en-línea al fichero en el panel derecho, entre otros:

- Revirtiendo líneas cambiadas a como se muestran en el panel izquierdo.
- Utilizando una combinación de ambos bloques, bien el *de ellos* antes del *mío*, o bien el *mío* antes que el *de ellos*.

También puede editar el fichero de salida tal y como lo haría en un editor de texto. Las líneas que cambie de esta forma se marcarán utilizando un icono de un lápiz. Por favor, tenga en cuenta que si desea hacer alguno de los cambios basados en línea o bloque descritos anteriormente, es mejor que lo haga al principio: una vez que empiece a editar el fichero, TortoiseMerge no podrá mantener la relación de las líneas con los ficheros originales.

## 2.2. Editando conflictos

Esto a veces se denomina fusión a tres bandas, y se muestra en tres paneles. Aunque realmente hay cuatro ficheros involucrados. El fichero que no se muestra es el fichero base común, el último ancestro común de los dos ficheros que ahora están en conflicto. Las relaciones entre los tres ficheros se explican en el diagrama siguiente:



**Figura 2.1. Conflicto de ficheros**

El fichero base representa la versión más antigua del fichero, desde donde *Usted* y *Otros* empezaron a hacer cambios. *Mío* representa el fichero base con todos los cambios que *usted* ha hecho, y *De otros* es el fichero con todos los cambios que *cualquier otro* ha hecho al fichero. El panel izquierdo le muestra los cambios en *de otros* respecto a su fichero base, y el panel derecho le muestra los cambios en *mío* relativos al fichero base. El panel inferior es el fichero resultado que es donde intenta resolver los conflictos.

En la vista de resolución de conflictos, puede elegir utilizar bloques *míos*, o *de ellos*, o ambos. Pero en este caso, los cambios se muestran en el panel inferior.

## 2.3. Aplicando parches

Un fichero de parche es un fichero de diff unificado de Subversion, que contiene la información necesaria para aplicar cambios a un conjunto de ficheros. El parche puede haber sido proporcionado por otro desarrollador para que pueda ver sus cambios y posiblemente confirmarlos en el repositorio. O puede haber sido generado internamente por TortoiseSVN cuando compara dos carpetas. Esto ocurre cuando compara una revisión del repositorio con una carpeta de copia de trabajo, o si compara dos revisiones diferentes del repositorio, o si compara dos rutas de repositorio diferentes.

En cada caso TortoiseMerge mostrará una pequeña ventana mostrando los ficheros incluidos en el parche. Al hacer doble click en uno de estos ficheros, se obtiene el fichero relevante y se aplican los cambios. El panel izquierdo muestra el contenido original del fichero y el panel derecho muestra cómo queda tras aplicar el parche.

Puede editar el fichero en el panel derecho exactamente igual que lo haría en el modo de comparación/edición.

# Capítulo 3. Usando TortoiseMerge

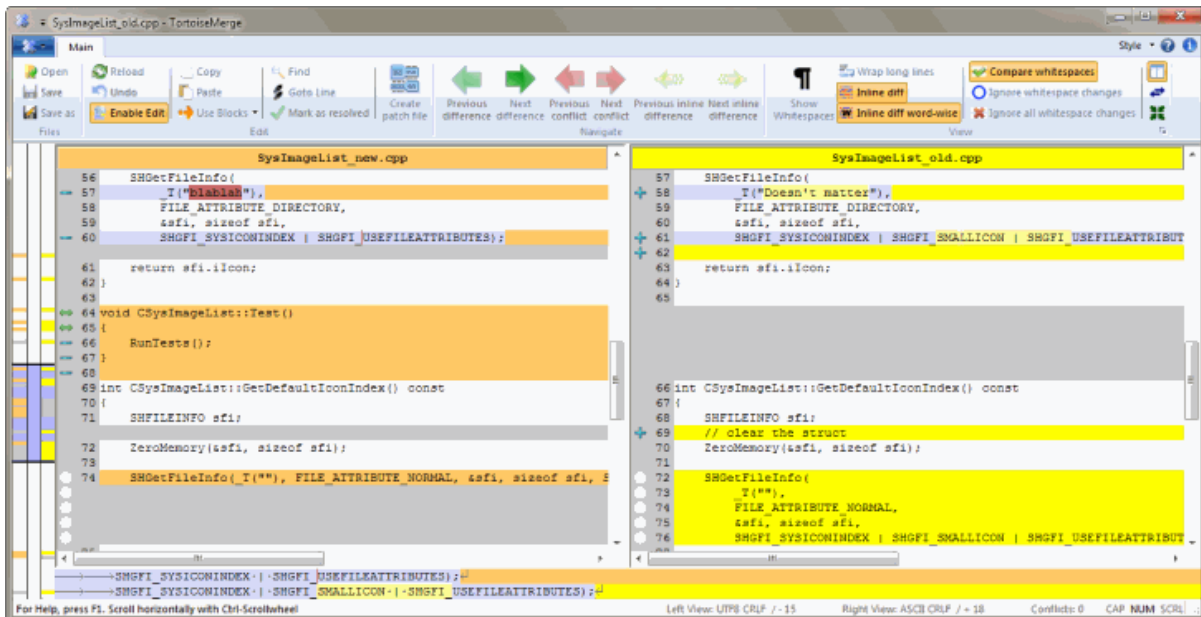
## 3.1. Modos de visión

TortoiseMerge tiene tres modos de vista principales: un panel, dos paneles y tres paneles. La vista de uno o dos paneles se utiliza para ver cambios, y la vista de tres paneles se utiliza para resolver conflictos.

### 3.1.1. Viendo / Fusionando



Figura 3.1. Vista de un panel



**Figura 3.2. Vista de dos paneles**

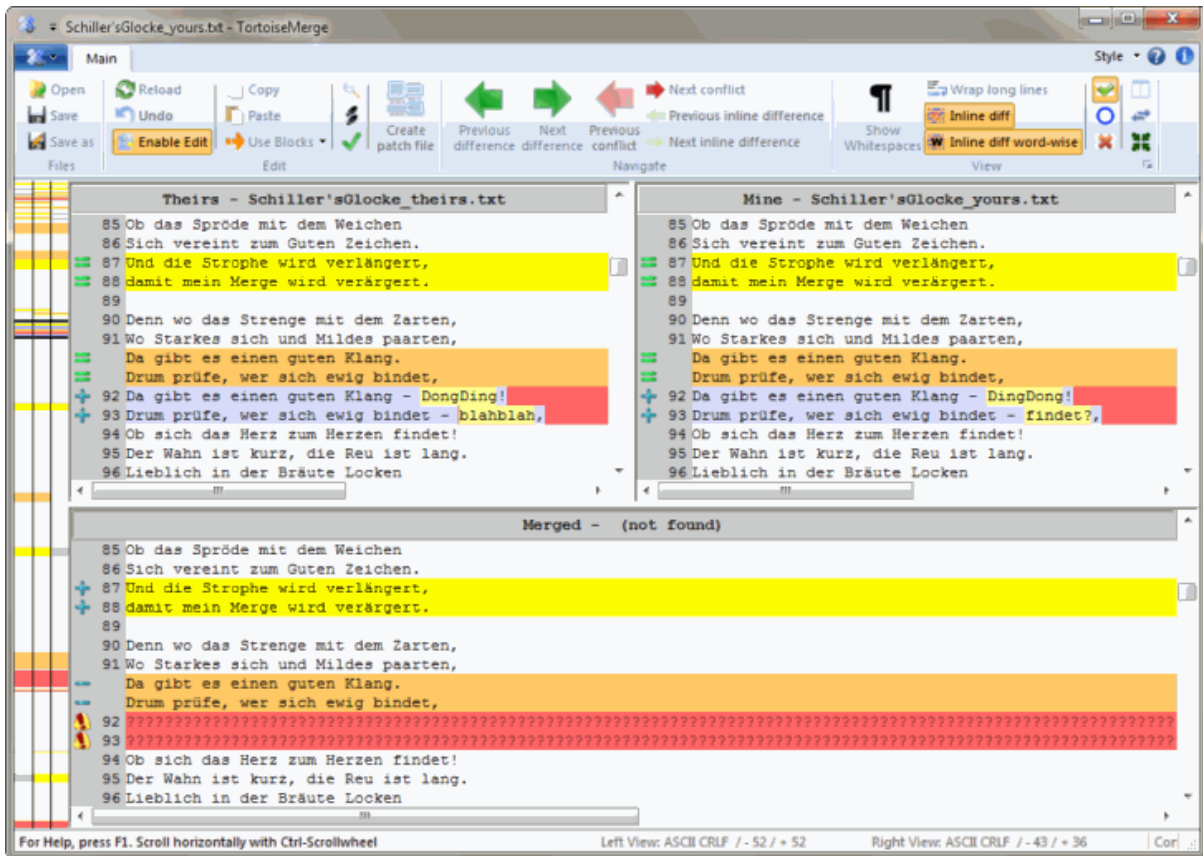
La vista de dos paneles tiene algunas funcionalidades que no están disponibles en la vista de un panel:

- Los cambios dentro de las líneas modificadas se muestran con colores diferentes. Las partes añadidas en la cadena se muestran con un color claro, aunque por supuesto puede configurar los colores utilizados aquí. Las partes eliminadas se indican con una línea vertical marrón oscura en la cadena. Observe la captura de pantalla anterior para ver cómo funciona.
- Reorganizar código a menudo significa un montón de cambios en los espacios en blanco (espacios, tabuladores, nuevas líneas) pero no cambios reales en el código. Por ejemplo, divide una línea muy larga en varias líneas, o a veces junta varias líneas en una única.

Dichos cambios se marcan con un círculo blanco en la parte izquierda de las vistas. Si observa dicho círculo blanco, inmediatamente sabrá que no se ha cambiado realmente nada ahí y no tendrá que seguir comprobando el bloque cambiado.

- Hand editing of the file in the right pane is possible in two-pane view. Such changes are marked with a pencil symbol. Hand editing of the file in the left pane is also possible, only when **Enable Edit** is pressed when the left pane is active.

Si quiere comparar/fusionar tres ficheros, TortoiseMerge le mostrará las diferencias en una vista de tres paneles. Esta vista también se utiliza si necesita resolver ficheros en conflicto.



**Figura 3.3. Vista de tres paneles**

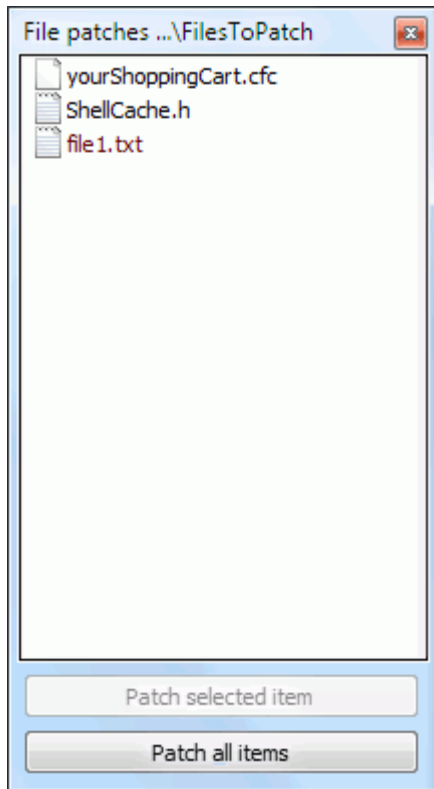
El panel izquierdo le muestra las diferencias entre el fichero *De otros* y el fichero *Base*, mientras que el panel derecho le muestra las diferencias entre el fichero *Mío* y el fichero *Base*. El panel inferior le muestra el resultado de fusionar los ficheros *Base*, *Mío* y el *De otros* con posibles conflictos.

Si pasa el ratón por encima del título del panel, una etiqueta de ayuda le mostrará los nombres de ficheros utilizados para las diferencias en cada panel.

Hand editing of the file in the left pane or right pane is also possible, only when **Enable Edit** is pressed when that pane is active.

### 3.1.2. Aplicando parches

Cuando TortoiseMerge termine de parsear el fichero de parche, le mostrará una pequeña ventana con todos los ficheros que tienen cambios según el fichero de parche.



**Figura 3.4. Lista de ficheros de parche**

Si el nombre del fichero se muestra en negro, el parche se puede aplicar sin problemas. Eso significa que el fichero no está desactualizado según el parche. Sin embargo, si el nombre del fichero se muestra en rojo, entonces el parche no se puede aplicar directamente porque ya ha cambiado ese fichero.

La ventana del archivo parche tiene un menú contextual que le permitirá previsualizar el efecto del parche en el archivo actual (aplicar sin guardar), aplicar y guardar los cambios para el archivo seleccionado, o aplicar y guardar los cambios to *todos* los archivos de la lista. Si hace doble clic se previsualizarán los cambios.

## 3.2. Botones y otros controles

Depending on your setting, you either see a toolbar or a ribbon with many buttons. The buttons all show a tooltip explaining their function when you hover the mouse pointer over them.

En la parte izquierda de la ventana está una barra de lugar. Esto proporciona una rápida referencia visual de dónde están los cambios en el fichero. La barra tiene tres columnas. La columna izquierda se refiere al panel izquierdo, la columna derecha al panel derecho, y la columna central al panel inferior (si está presente). En la vista de un solo panel sólo se utiliza la columna izquierda. La barra de lugar se puede utilizar también como una barra para mover todas las ventanas simultáneamente.

Si haces doble clic sobre una palabra, entonces cada una de las apariciones de esa palabra será resaltada en todo el documento, tanto en los paneles principales como en la barra de localización. Para remover el resaltado haz doble clic de nuevo sobre la palabra.

Si hace clic en el margen izquierdo, o si hace triple clic dentro de una línea, esa línea entera se seleccionará.

Below the bottom window is the status bar. This shows the number of lines added and deleted in *Theirs* and *Mine*, and the number of unresolved conflicts remaining.

The status bar also contains combo box controls which indicate how the files are handled and treated:

#### Codificando

The encoding specifies how the characters in the views are loaded/saved and shown. The most common encoding in English is ASCII (which means the local encoding of the OS language), but you can change this to be UTF8, UTF16LE, UTF16BE, UTF32LE and UTF32BE, both with or without a byte order mark (BOM).

#### Line Endings

The most common line endings on Windows is CRLF, but you can change the line endings to whatever you like. Note that if you change the line endings, then *all* line endings in the whole file will change, even if when loaded the line endings were not all the same.

#### Tabulaciones

The option at the top of the combo box menu indicates whether tabs or spaces are inserted when you press the tab key. The *smart tab char* option if enabled uses an algorithm to determine whether one or the other is best used.

The tab size specifies how many space chars are inserted when editing and pressing the tab char, or how many chars the next word is indented when a tab char is encountered.

### 3.3. Iconos de estado de línea

Las líneas cambiadas se marcan con un icono para indicar qué tipo de cambio ocurrió.



Se añadió una línea.



Se eliminó una línea.



Se deshizo un cambio revirtiendo al contenido original de la línea.



Esta línea sólo contiene cambios en los espacios en blanco. Donde aparezcan líneas consecutivas marcadas, el párrafo puede haberse reformateado, causando que las palabras se cambien a las líneas adyacentes.



Se ha editado la línea a mano, utilizando TortoiseMerge como editor de texto.



Esta línea está en conflicto.



Esta línea está en conflicto, pero el efecto está oculto por la configuración de los espacios en blanco o de los finales de línea.



This line is detected as being moved from / to another location.

### 3.4. Fusionando / Editando conflictos

TortoiseMerge no sólo le *muestra* las diferencias entre los ficheros, sino que también le permite resolver conflictos o aplicar cambios.

If you're in two pane view, then you can only edit the file in the right pane (*Mine*). To apply changes made in the left file (*Theirs*), right click on the changed lines and select Context Menu → Use text block from 'theirs'. Then the changes from the left file are added to the right file.



A veces en realidad querrá utilizar ambos bloques de texto, y el menú contextual también le ofrece **Menú contextual** → **Utilizar ambos bloques de texto (este primero)** y **Menú contextual** → **Utilizar ambos bloques de texto (este el último)**.

También puede editar el fichero de salida tal y como lo haría en un editor de texto. Las líneas que cambie de esta forma se marcarán utilizando un icono de un lápiz. Por favor, tenga en cuenta que si desea hacer alguno de los cambios basados en línea o bloque descritos anteriormente, es mejor que lo haga al principio: una vez que empiece a editar el fichero, TortoiseMerge no podrá mantener la relación de las líneas con los ficheros originales.

If you're in three pane view (sometimes called *merge view*) you can only edit the file in the bottom view (Merged). As in two pane view, you can right click on conflicted lines and either select **Context Menu** → **Use text block from 'theirs'** or **Context Menu** → **Use text block from 'mine'** . In addition, if you want *both* blocks, you can select **Context Menu** → **Use text block from 'mine' before 'theirs'** or **Context Menu** → **Use text block from 'theirs' before 'mine'** . According to the command you've selected, the changes are used in the resulting Merged file.

A veces un fichero se marcará como en conflicto en Subversion, pero cuando lo vea utilizando TortoiseMerge no se muestra ningún conflicto. Esto puede ser debido a su selección de la forma en la que se tratan los espacios en blanco. Si elige ignorar los cambios en el fin de línea o en los espacios en blanco, esas líneas se marcarán utilizando el icono *Conflicto-Ignorado*. Para resolver el conflicto aún necesitará elegir qué versión desea utilizar.

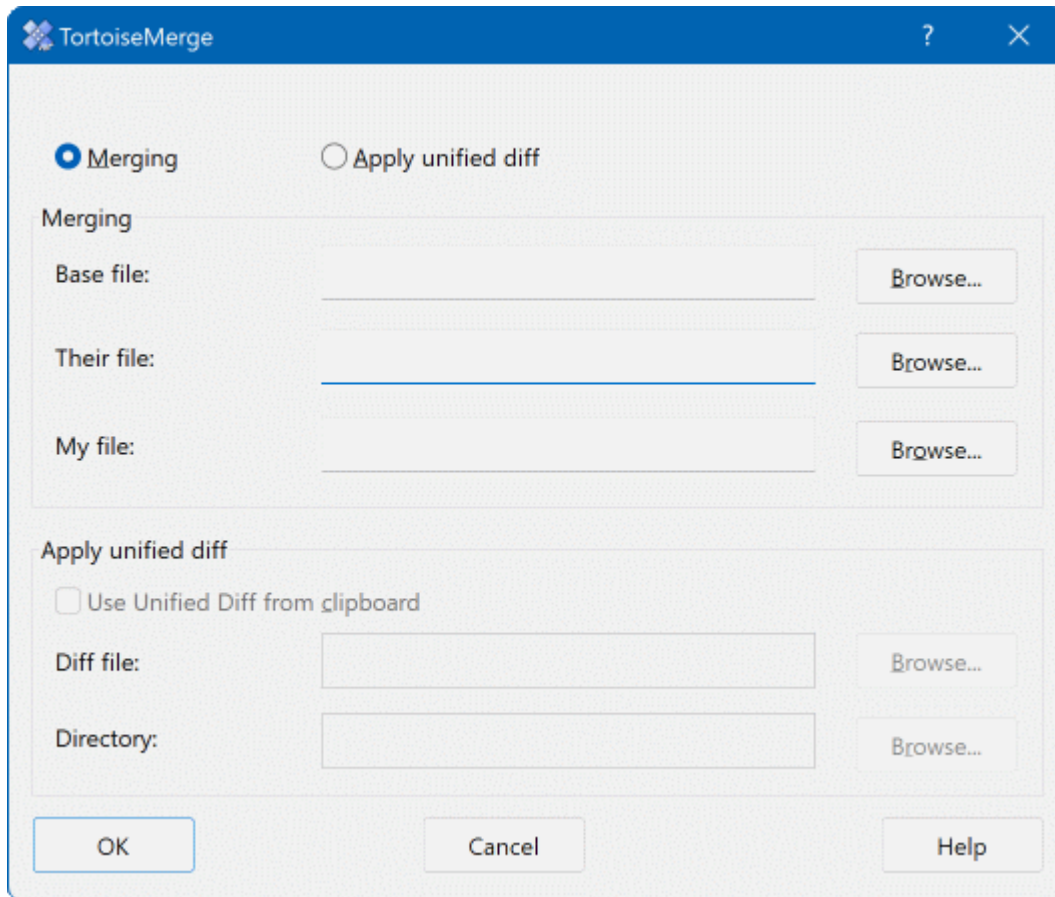


### Importante

Tenga en cuenta que si utiliza TortoiseMerge sobre los mismos ficheros otra vez, cualquier cambio en su copia de trabajo, tanto si se hizo por TortoiseMerge o editándolo a mano, *se descartará* y el fichero aparecerá tal y como lo hizo cuando se inició por primera vez la edición de conflictos.

## 3.5. Abrir ficheros

Cuando inicia TortoiseMerge sin ningún argumento en la línea de comandos tendrá que abrir los ficheros manualmente, utilizando **Archivo** → **Abrir**.



**Figura 3.5. El diálogo Abrir**

Lo primero que tiene que hacer es decidir si sólo quiere comparar/fusionar ficheros, o si quiere aplicar un fichero de parche. Dependiendo de su elección, se activan las casillas de texto y los botones de navegación correspondientes.

### 3.5.1. Viendo / Fusionando

Si quiere comparar/fusionar ficheros necesita establecer al menos dos de las tres posibles rutas para Base, Mío and De otros. Si sólo selecciona dos ficheros, TortoiseMerge le mostrará las diferencias entre esos dos ficheros, en dos paneles o en un único panel.

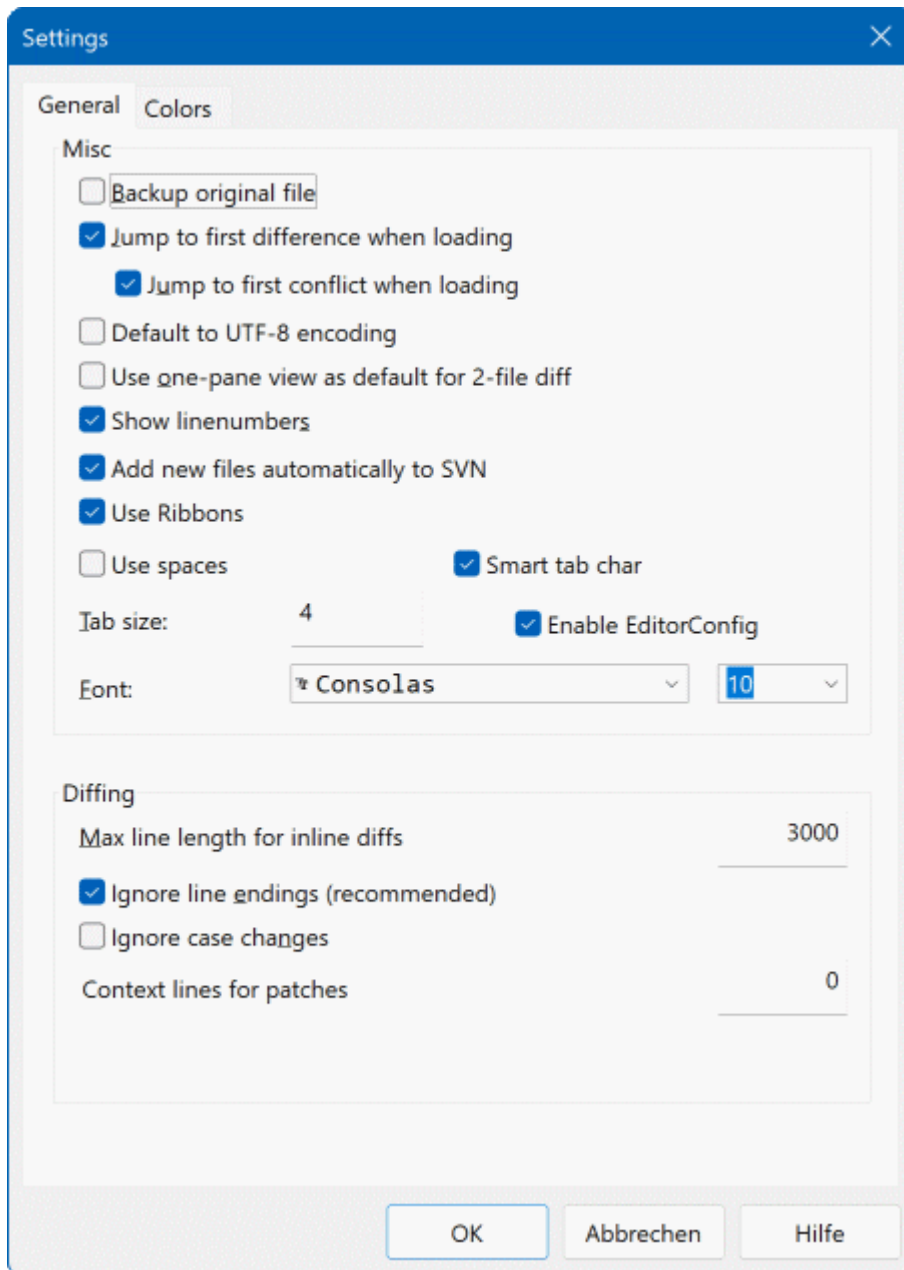
Si quiere comparar/fusionar tres ficheros, TortoiseMerge le mostrará las diferencias en una vista de tres paneles. Esta vista se utiliza normalmente si necesita resolver ficheros en conflicto. El fichero de salida no se nombra en este caso y tendrá que utilizar Archivo → Guardar como... para guardar el resultado.

### 3.5.2. Aplicando parches

Si desea aplicar un fichero de parche, tiene que especificar tanto la ruta al fichero de parche en si mismo como la ruta a la carpeta donde debe aplicarse el fichero de parche.

## 3.6. Configuración

### 3.6.1. Página de Configuración principal



**Figura 3.6. La página de Configuración principal**

Muchas de estas opciones son autoexplicativas, pero hay algunos pocos puntos que necesitan clarificarse.

Realizar copia de seguridad del archivo original renombra el fichero original en la copia de trabajo a nombre .bak antes de guardar la versión modificada.

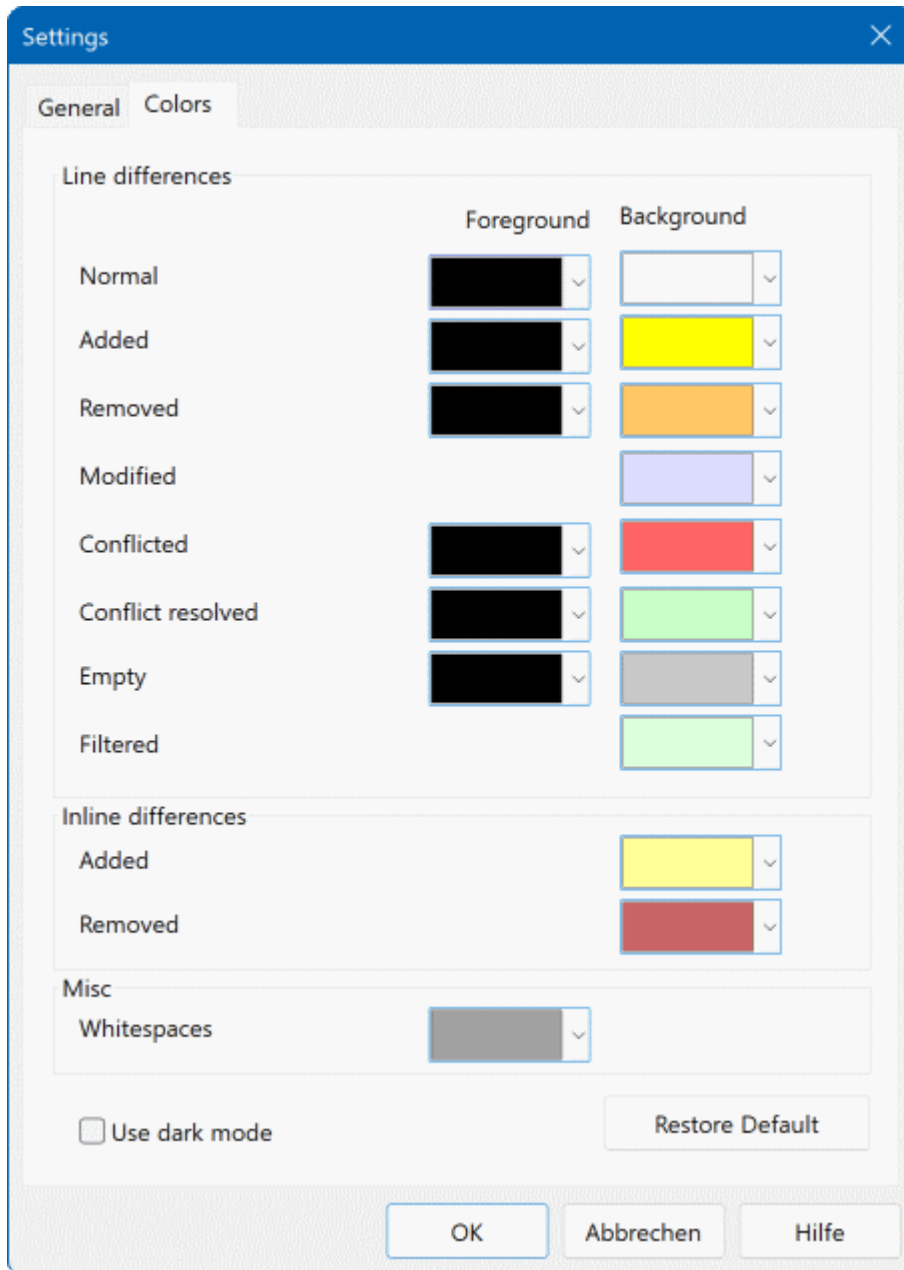
Default to UTF-8 encoding when set, ANSI files are loaded as UTF-8 encoded and saved as such when edited.

Max line length for inline diffs TortoiseMerge can get slow when showing inline diffs for very long lines. Because of that only lines that are shorter than 3000 chars are shown with inline diffs. You can change this value here.

Ignorar finales de línea oculta los cambios que se deben únicamente a la diferencia en el estilo de las terminaciones de las líneas.

Ignorar cambios de mayúsculas/minúsculas oculta los cambios que se deben únicamente a cambiar mayúsculas por minúsculas o viceversa dentro del texto. Esto puede ser útil con aplicaciones como Visual Basic, que cambia las mayúsculas en las variables sin avisar.

### 3.6.2. Página de Configuración de colores



**Figura 3.7. La página de Configuración de colores**

Este diálogo le permite seleccionar la codificación de colores de fondo que se utilizan para remarcar los diferentes cambios en las líneas.

Normal

Todas las líneas que no tienen cambios, o cuyos cambios están ocultos.

Añadido

Líneas que han sido añadidas.

**Borrado**

Líneas que han sido borradas.

**Modificado**

Líneas que tienen relativamente pocos cambios, y se muestran utilizando diferencias en línea. Este color se utiliza para la parte *sin cambios* del contenido de la línea. Las partes modificadas se mostrarán utilizando los colores de añadido y eliminado en-línea que se describen a continuación. Si **Colorear cambios dentro de las líneas** no está seleccionado, este color no se utilizará, y las líneas con cambios siempre se mostrarán como reemplazos.

**En conflicto**

Donde la misma línea se ha cambiado en ambos ficheros.

**Conflicto resuelto**

Cuando la misma línea se ha cambiado en ambos ficheros, y ahora ya ha seleccionado qué versión debería utilizarse.

**Vacío**

Donde las líneas se han añadido en el panel contrario, y no existe dicha línea en este panel.

**Texto añadido en línea**

Cuando se muestran diferencias en-línea, el texto añadido se muestra en este color.

**Texto eliminado en línea**

Cuando se muestran diferencias en-línea, el texto eliminado se muestra en este color.

**Espacios en blanco varios**

Los caracteres usados para indicar espacios en blanco se muestran en un color distinto del texto normal.

**Usar modo oscuro**

Enables a dark mode for the UI. This requires at least Windows 10 1809.

---

# Apéndice A. Keyboard Shortcuts

Lista de atajos y comandos de teclado

## A.1. Keyboard Shortcuts

### A.1.1. General Keyboard Shortcuts

Ctrl-Q, Ctrl-W, Escape  
Sale del programa

Ctrl-C  
Copia el texto seleccionado al portapapeles

Ctrl-X, Shift-Sup  
Cortar el texto seleccionado al portapapeles

Ctrl-V, Shift-Insertar  
Copiar el texto seleccionado desde el portapapeles

Ctrl-Z, Alt-Borrar  
Deshacer las últimas ediciones

Ctrl-F  
Open the Find dialog for text searches and replaces

Ctrl-O  
Abre los ficheros para ver las diferencias/fusionar

Ctrl-S  
Guardar los cambios

Ctrl-Mayús-S  
Guardar como...

F7, Ctrl-Down  
Ir a la siguiente diferencia

Shift-F7, Ctrl-Up  
Ir a la diferencia previa

Ctrl-R  
Recarga el fichero y revierte todos los cambios hechos

Ctrl-T  
Cambia entre mostrar espacios en blanco o no

Ctrl-L  
Alternar entre colapsar las secciones sin cambios o no

Ctrl-P  
Alternar entre el ajuste de línea

Ctrl-G  
Ir a la línea

Ctrl-A  
Seleccionar todo el texto

Ctrl-rueda del ratón  
Mueve el display a la izquierda/derecha

Ctrl-Tab  
Switch between left/right/bottom view

Ctrl-M  
Toggles between marking the selected change

### **A.1.2. Diff-Mode Keyboard Shortcuts**

Ctrl-D  
Cambia entre la vista de un panel y la vista de dos paneles

Ctrl-M  
Toggles between marking the selected change

Ctrl-U  
Cambia las vistas

F12  
Usar bloque izquierdo

### **A.1.3. Conflict Resolution Mode Keyboard Shortcuts**

F8  
Ir al siguiente conflicto

Mayús-F8  
Ir al conflicto previo

Ctrl-F9  
Usar bloque izquierdo

Ctrl-Shift-F9  
Use left block then right block

Ctrl-F10  
Usar el bloque de la derecha

Ctrl-Shift-F10  
Use right block then left block

---

# Apéndice B. Automatizando TortoiseMerge

TortoiseMerge puede iniciarse con parámetros en la línea de comandos para evitar pasar por el diálogo Abrir y seleccionar los ficheros allí. Esto es útil si desea utilizar TortoiseMerge desde otra aplicación.

## B.1. Opciones de Línea de Comandos de TortoiseMerge

Most switches require additional information like a path or some other string. In those cases, append an ':' to the switch and put the string/path after it. Example:

```
/base:"c:\folder\my base file.txt"
```

Comando	Descripción
/?	Muestra un diálogo con los parámetros de la línea de comandos más importantes.
/help	Lo mismo que ?.
/base	Especifica el fichero <i>base</i> utilizado en las diferenciaciones en tres sentidos. Este es el ancestro común de los ficheros que están siendo diferenciados, aunque no se muestra en una ventana diferente. Para las diferenciaciones en dos sentidos, este es el fichero izquierdo.
/basename	El nombre del fichero base. Esto se muestra en el título de la vista en vez de la ruta del fichero. Para diferenciaciones de tres sentidos se muestra en un texto de ayuda para el título de la vista.
/basereflectedname	The name to use for editorconfig templates.
/theirs	Especifica el fichero <i>de ellos</i> utilizado en las diferenciaciones de tres sentidos, mostrado en el panel izquierdo.
/theirsname	El nombre del fichero de ellos. Esto se muestra en el título de la vista en vez de la ruta del fichero.
/theirsreflectedname	The name to use for editorconfig templates.
/mine	Especifica el fichero <i>mío</i> utilizado en las diferenciaciones en tres sentidos, mostrado en el panel derecho. Para las diferenciaciones en dos sentidos, este es el fichero derecho.
/minename	El nombre del fichero <i>mío</i> . Esto se muestra en el título de la vista en vez de la ruta del fichero.
/minereflectedname	The name to use for editorconfig templates.
/merged	Specifies the resulting <i>merged</i> file used in diffs. This is the file path where the result of the merge/conflict resolving is saved.  If this is not set for a three-way diff, then TortoiseMerge will ask the user where to save the result.  If this is not set for a two-way diff, then TortoiseMerge will automatically use the path of the file shown in the right view as the save path.
/mergedname	El nombre del fichero fusionado. Esto se muestra en el título de la vista en vez de la ruta del fichero.
/mergedreflectedname	The name to use for editorconfig templates.



Comando	Descripción
/patchpath	La ruta donde se debería aplicar un parche. Si no establece esta ruta, entonces TortoiseMerge intentará encontrar la ruta por sí mismo, de forma que encaje con las rutas en el fichero de parche, pero eso puede llevarle <i>mucho tiempo</i> .
/saverequired	If specified, forces TortoiseMerge to ask to save the file before exiting, even if the user does not modify the files.
/saverequiredonconflicts	If specified, forces TortoiseMerge to ask to save the file before exiting if there are conflicts found, even if the user does not modify the files.
/patchoriginal	El nombre del fichero original a parchear. Utilizado para el título de la vista.
/patchpatched	El nombre del fichero parcheado resultante. Utilizado para el título de la vista.
/diff	La ruta del fichero de parche/diferencia a aplicar a un directorio.
/oneway	Fuerza a TortoiseMerge a iniciar la vista de un sólo sentido en vez de la vista que el usuario especificó en la configuración.
/reversedpatch	Cambia la vista derecha e izquierda de los dos ficheros especificados para diferenciar.
/createunifieddiff	Crea un fichero de diff unificado (fichero de parche) de dos ficheros especificados con /origfile:"ruta_al_fichero_original" y /modifiedfile:"ruta_al_fichero_modificado". La ruta de destino se establece con /outfile:"ruta_al_fichero_de_parche_resultante". Si /outfile no se establece, se muestra un diálogo de guardar fichero para que el usuario pueda elegir el lugar donde guardar el fichero de parche. Nota: si se establece /createunifieddiff, todos los demás parámetros se ignoran.
/line	Especifica el número de línea donde ir una vez cargados los archivos.
/readonly	Impide que el archivo sea editado. Esto significa que las capacidades de edición de TortoiseMerge están deshabilitadas.
/nosvnresolve	Prevents TortoiseMerge from asking whether to mark a file as resolved in SVN after saving it.

**Tabla B.1. Lista de las opciones de línea de comandos disponibles**

You can also supply simple filenames on the command line for compatibility with other diff programs. In this simplified form the command line is

```
TortoiseMerge BaseFilePath MyFilePath [ TheirFilePath ]
```

If two files are given they will be compared with each other. If three files are given, the first is taken as the BASE file, and the other two are compared with it in a three-way diff.

---

# Glosario

Actualizar	Este comando de Subversion incorpora los últimos cambios del repositorio a su copia de trabajo, fusionando cualquier cambio hecho por otros con los cambios locales en la copia de trabajo.
Añadir	Un comando de Subversion que se utiliza para añadir un archivo o un directorio a su copia de trabajo. Los nuevos ítems se añaden al repositorio cuando confirme.
Autoría	Este comando es sólo para archivos de texto, y anota cada línea para mostrar la revisión del repositorio en la que se cambió por última vez, y el autor que hizo ese cambio. Nuestra implementación GUI se llama TortoiseBlame y también le muestra la fecha y hora de la confirmación y el mensaje de registro cuando mueve el ratón por encima del número de revisión.
Bloqueo	Cuando obtiene un bloqueo de un ítem versionado, lo marca en el repositorio como no confirmable, excepto para la copia de trabajo desde la que se obtuvo el bloqueo.
Cambiar	De la misma forma que “Actualizar-a-la-revisión” cambia la ventana temporal de una copia de trabajo para que quede como estuvo en un punto diferente de su historia, “Cambiar” cambia la ventana espacial de una copia de trabajo para que apunte a una parte diferente del repositorio. Es particularmente útil cuando esté trabajando en troncos y ramas en los que sólo hay unos pocos archivos diferentes. Puede cambiar su copia de trabajo entre los dos y sólo se transferirán los archivos cambiados.
Confirmar	Este comando de Subversion se utiliza para pasar los cambios desde su copia de trabajo local al repositorio, creando una nueva revisión en el repositorio.
Conflicto	Cuando los cambios desde el repositorio se mezclan con los cambios locales, a veces esos cambios ocurren en las mismas líneas. En este caso Subversion no puede decidir automáticamente qué versión utilizar, y se dice que el archivo está en conflicto. Tiene que editar el archivo manualmente y resolver el conflicto antes que pueda confirmar más cambios.
Copia de trabajo	Esta es su “caja de arena” local, el área donde trabaja con los archivos versionados, y normalmente se encuentra en su disco duro local. Puede crear una copia de trabajo al “Obtener” desde un repositorio, y devolver sus cambios al repositorio utilizando “Confirmar”.
Copiar	En un repositorio de Subversion puede crear una copia de un único archivo o un árbol completo. Estos se implementan como “copias baratas” que actúan un poco como un enlace al original en el sentido que casi no ocupan espacio. Haciendo una copia preserva la historia del ítem en la copia, por lo que puede rastrear los cambios que se hicieron antes que ocurriera la copia.
Diff	Abreviatura de “Mostrar diferencias”. Muy útil cuando desea ver exactamente qué cambios se han hecho.
Eliminar	Cuando elimina un ítem versionado (y confirma el cambio), el ítem ya no existirá nunca más en el repositorio después de la revisión confirmada. Pero por supuesto aún existe en las revisiones previas en el repositorio, por lo que aún puede acceder a él. Si es necesario, puede copiar un ítem eliminado y “resucitarlo” por completo con su historia.
Exportar	Este comando produce una copia de una carpeta versionada, igual que una copia de trabajo pero sin los directorios locales <code>.svn</code> .

FSFS	Un sistema de archivos propietario de Subversion que se utiliza como soporte de los repositorios. Se puede utilizar en unidades compartidas de red. Se utiliza por defecto para los repositorios creados a partir de la versión 1.2.
Fusionar	<p>El proceso por el cual los cambios del repositorio se añaden a su copia de trabajo sin interrumpir cualquier cambio que haya realizado ya localmente. A veces estos cambios no se pueden reconciliar automáticamente, y se dice que la copia de trabajo está en conflicto.</p> <p>Las fusiones ocurren automáticamente cuando actualiza su copia de trabajo. También puede fusionar cambios específicos desde otras ramas utilizando el comando Fusionar de TortoiseSVN.</p>
GPO	Objeto de políticas de grupo.
Historial	Muestra la historia de las revisiones de un archivo o carpeta. También se conoce como “Registro”.
Importar	Comando de Subversion para importar una jerarquía de carpetas completa en el repositorio en una única revisión.
Limpieza	Citando el libro de Subversion: “Limpia recursivamente la copia de trabajo, eliminando los bloqueos y continuando las operaciones sin terminar. Si alguna vez obtiene un error copia de trabajo bloqueada, ejecute este comando para eliminar los bloqueos y conseguir que su copia de trabajo vuelva a un estado usable de nuevo.” Tenga en cuenta que en este contexto “bloqueado” se refiere a un bloqueo del sistema de archivos local, no a bloqueos del repositorio.
Obtener	Un comando de Subversion que crea una copia local de trabajo en un directorio vacío al descargar los archivos versionados desde el repositorio.
Parche	Si una copia de trabajo tiene cambios únicamente en archivos de texto, es posible utilizar el comando Diff de Subversion para generar un único archivo con el resumen de esos cambios en el formato de Diff unificado. Un archivo de este tipo a menudo se denomina “Parche”, y puede ser enviado por email a otro (o a una lista de correo) y aplicado en otra copia de trabajo. Alguien sin acceso de confirmación puede hacer cambios y enviar un archivo de parche para que lo aplique un confirmador autorizado. O si no está seguro de un cambio, puede enviar el parche a otros para que lo revisen.
Propiedad	Además de versionar sus directorios y archivos, Subversion le permite añadir metainformación versionada - llamadas “propiedades” para cada uno de sus directorios y archivos versionados. Cada propiedad tiene un nombre y un valor, como un entrada del registro. Subversion tiene algunas propiedades especiales que utiliza internamente, como <code>svn:eol-style</code> . TortoiseSVN tambitsvn:logminsize. Puede a
Propiedad de revisión (revprop)	Al igual que los archivos pueden tener propiedades, las revisiones del repositorio también. Algunas revprops especiales se añaden automáticamente cuando se crea la revisión; son: <code>svn:date</code> <code>svn:author</code> <code>svn:log</code> que representan la fecha/hora de la confirmación, el confirmador y el mensaje de registro, respectivamente. Estas propiedades pueden editarse, pero no están versionadas, por lo que cualquier cambio es permanente y no puede deshacerse.
Rama	Un término de uso frecuente en los sistemas de control de versiones para describir lo que ocurre cuando el desarrollo se parte en un punto en concreto y sigue dos caminos separados. Puede crear una rama desde la línea principal de desarrollo para desarrollar una nueva funcionalidad sin hacer que la línea

principal quede inestable. O puede hacer una rama de una versión estable en la que sólo hará correcciones de errores, mientras los nuevos desarrollos tienen lugar en el inestable tronco. En Subversion una rama se implementa como una “copia barata”.

**Registro** Muestra la historia de las revisiones de un archivo o carpeta. También se conoce como “Historial”.

**Relocalizar** Si su repositorio se mueve, quizás porque lo ha movido a un directorio diferente en su servidor, o el nombre de dominio del servidor ha cambiado, necesitará “relocalizar” su copia de trabajo para que sus URLs de repositorio apunten al nuevo lugar.

Nota: sólo debería utilizar este comando si su copia de trabajo apunta al mismo lugar en el mismo repositorio, pero es el propio repositorio el que se ha movido. En cualquier otra circunstancia probablemente necesita el comando “Cambiar” en su lugar.

**repositorio** Un repositorio es un lugar central donde se almacenan y mantienen los datos. Un repositorio puede ser un lugar donde se encuentran múltiples bases de datos o archivos para distribuirlos en una red, o un repositorio puede ser un lugar directamente accesible por el usuario sin tener que viajar por una red.

**Resolver** Cuando los archivos en una copia de trabajo se quedan en un estado de conflicto tras una fusión, estos conflictos deben arreglarse por un humano utilizando un editor (o quizás TortoiseMerge). Este proceso se denomina “Resolviendo conflictos”. Cuando finalice, marque los archivos con conflictos como resueltos, y eso le permitirá confirmarlos.

**Revertir** Subversion mantiene una copia local “prístina” de cada archivo tal y como estaban cuando actualizó su copia de trabajo por última vez. Si ha hecho cambios y decide que quiere deshacerlos, puede utilizar el comando “revertir” para volver a la copia prístina.

**Revisión** Cada vez que confirma un conjunto de cambios, crea una nueva “revisión” en el repositorio. Cada revisión representa el estado del árbol del repositorio en un cierto punto de su historia. Si desea volver en el tiempo para examinar el repositorio tal y como estaba en la revisión N.

En otro sentido, una revisión puede referirse al conjunto de cambios que se hicieron cuando esa revisión se creó.

**revisión BASE** La revisión base actual de un archivo o una carpeta en su *copia de trabajo*. Esta es la revisión en la que estaba el archivo o la carpeta cuando se hizo la última operación de obtener, actualizar o confirmar. La revisión BASE normalmente no equivale a la revisión HEAD.

**revisión HEAD** La última revisión de un archivo o una carpeta en el *repositorio*.

**SVN** Una abreviatura muy usada para Subversion.

El nombre del protocolo personalizado de Subversion que utiliza el servidor de repositorios “svnserve”.