

- a. Abre el formulario principal del proyecto osImageManagerApp en el editor de diseño con doble clic en el fichero Form1.cs en tu vista de árbol del Explorador de soluciones; ver Imagen 9-1.
- b. Usa la ventana de Propiedades (Ver | Ventana Propiedades) para cambiar la propiedad de texto de este formulario de Form1 a osImageManagerApp. Esto modifica el fichero Form1.cs, Form1.Designer.cs, y Form1.resx.

NOTA | Visual Studio detecta cualquier alteración que hagas en los ficheros en tu área de trabajo y los muestra en la ventana Cambios pendientes. Esta lista en última instancia forma el conjunto de cambios que aplicarás en el repositorio; ver paso 9 de este ejercicio.

Debes verificar ahora que tus cambios hacen lo que se supone que han de hacer y no han estropeado el programa de algún modo. Por lo tanto, la próxima parte del ejercicio reconstruye la Solución y realiza una prueba rápida.

7. Arranca la aplicación desde el menú Depurar (Depurar | Iniciar sin depurar), con el propósito de que el Visual Studio incluya tus cambios automáticamente reconstruyendo el programa antes de ejecutarlo.
8. Verifica que has renombrado con éxito el título de la ventana de osImageManagerApp, de "Form1" a "osImageManagerApp" mirando su formulario principal, que se muestra en tu escritorio.

Definitivamente, estás listo para compartir tus cambios con el resto del equipo (ver paso 6b), registrándolos en el repositorio para que en la parte final del ejercicio crees este conjunto de cambios.

9. Realiza las siguientes acciones en la ventana Cambios pendientes (Ver | Otras ventanas), que se muestra al final de la Imagen 9-1:
 - a. Pon un comentario apropiado para tu conjunto de cambios.
 - b. Explora otras maneras en las que podrías describir este conjunto de cambios seleccionando elementos en la herramienta deshacer en la ventana Cambios pendientes.
 - c. Copia el conjunto de cambios en el repositorio de TFS, haz clic en el botón Proteger.
10. Ahora has terminado el ejercicio, así que sal del sistema.

CONSEJO | Encontrarás mucho más fácil trabajar con control de versiones si realizas tu trabajo dando una serie de pequeños pasos. Esto te permitirá regularmente actualizar tu área de trabajo y proteger tus cambios tan pronto como pasen tus pruebas.

Tareas Comunes del Control de Versiones

Además de tu uso diario del control de versiones durante los episodios de programación, algunas veces necesitarás realizar otras tareas, como la creación de nuevas áreas de trabajo, fusión de cambios hechos por otros con los cambios que tú has hecho a un fichero, volver a una versión anterior, y creación de ramas (o aplazamientos), desde la base de código principal del equipo. El resto de este capítulo explica cómo llevar adelante estas tareas.

Uso de Áreas de Trabajo

Los archivos y las carpetas del Proyecto de equipo en el repositorio se copian en tu área de trabajo cuando recibes la versión más reciente de sus archivos. Este área de trabajo mapea un conjunto de directorios sobre tu propio PC, a un correspondiente conjunto de carpetas en el repositorio de TFVC; ver Imagen 8-1 en el Capítulo 8.

Normalmente, el directorio raíz de tu área de trabajo es mapeado a la carpeta raíz de tu Proyecto de equipo en el repositorio, así que cuando seleccionas esta carpeta raíz y preguntas por la última versión, se ponen en tu área de trabajo una copia completa de los archivos y las carpetas de tu equipo. Sin embargo, debes ser consciente de que si seleccionas una carpeta en el repositorio que es más bajo en la jerarquía, solamente sus contenidos se copian en tu área de trabajo; los directorios padre son creados según se necesitan, pero no son rellenados. Por ejemplo, si mapeas `$/OSPACS` al nuevo área de trabajo `c:\trabajo\OSPACS` y preguntas por la última versión de la carpeta de repositorio `osImageManager`, los directorios de Producción y `osImageManager` serán creados dentro de este área de trabajo y el directorio `osImageManager` será rellenado, como lo serán sus diferentes subdirectorios. Sin embargo, tu área de trabajo no contendrá ninguna carpeta de repositorio hermana, como `ospacsWeb`, ni contendrá ningún fichero u otras carpetas presentes en `$/OSPACS` (la carpeta padre).

CONSEJO | Evita crear múltiples espacios de trabajo mapeando la raíz de tu área de trabajo a tu carpeta del Proyecto de equipo, para que cuando diferentes partes de la base de código sean desprotegidas, puedan ser puestas automáticamente dentro de una estructura de directorio existente.

La mayoría del tiempo no necesitas preocuparte demasiado por las áreas de trabajo, así se te propondrá crear una cuando no hay un mapeado existente entre una carpeta del repositorio y un directorio en tu disco duro; ver paso 4 del Ejercicio 9-1. Sin embargo, a veces puedes necesitar crear más de un área de trabajo para la misma persona. Por ejemplo, Luke podría necesitar trabajar en dos ramas diferentes al mismo tiempo, o querer espacios de trabajo diferentes para partes diferentes del producto (como `osImageManager` y `ospacs-Web`). En tal caso, el Ejercicio 9-2 explica cómo podrías crear espacios de trabajo adicionales usando el cuadro de diálogo de Agregar área de trabajo.

ADVERTENCIA | En el caso de múltiples áreas de trabajo propiedad de la misma persona, no se pueden mapear diferentes directorios en un PC a la misma carpeta en el repositorio de TFVC. Igualmente, diferentes carpetas en el repositorio no pueden ser mapeadas al mismo directorio en el PC.

Ejercicio 9-2: Creación de un Área de trabajo Adicional

Este ejercicio crea un área de trabajo adicional para Luke con el propósito de que pueda separar su producción y su trabajo experimental (Muestras) en dos áreas distintas en el DeveloperPC.

1. Entra al DeveloperPC como Luke (Colaborador de OSPACS), arranca Visual Studio, y luego conéctate al Proyecto de equipo OSPACS como hiciste antes.

2. Usa Windows Explorer (o algo similar) para crear el nuevo directorio c:\Pruebas\Muestras, para la nueva área de trabajo
3. Abre el cuadro de diálogo de Administrar áreas de trabajo, (Archivo | Control de código fuente | Áreas de trabajo) y hacer clic en Agregar, para abrir el cuadro de diálogo Agregar área de trabajo; ver Imagen 9-2:
 - a. Teclea "Muestras" como el nombre del área de trabajo y dale un comentario adecuado.
 - b. Haz clic en las diferentes columnas en la fila inferior de la lista "Carpetas de trabajo" para definir tu nueva área de trabajo. Debes poner Estado como Activo, la carpeta de Control de fuente como \$/OSPACS/Muestra, y la carpeta de Local como c:\Pruebas\Muestras.
 - c. Haz clic en Aceptar para crear esta área de trabajo Muestras.
4. Cierra el cuadro de diálogo y sal del sistema.

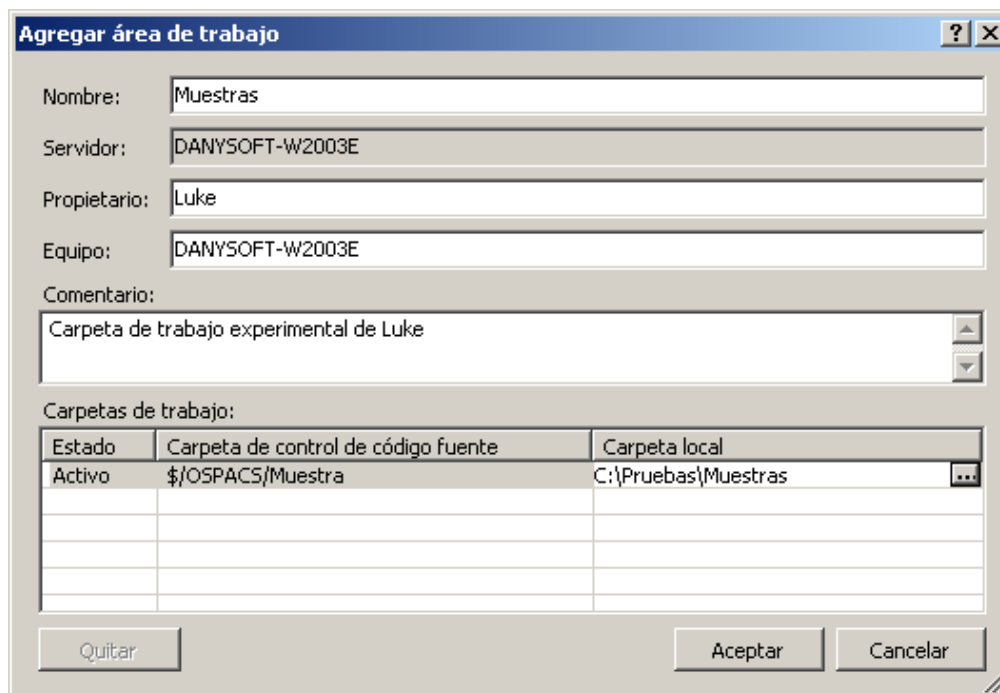


Imagen 9-2: El cuadro de diálogo Agregar área de trabajo (Archivo | Control de código fuente | Áreas de trabajo)

NOTA | Cuando añades o editas un área de trabajo puedes poner el estatus de una Carpeta de Trabajo a "escondido". Esto significa que la carpeta es invisible para ciertas herramientas TFVC, por lo que, por ejemplo, la carpeta local no será actualizada cuando apliques el comando Obtener la última versión para sus padres.

Las áreas de trabajo que están disponibles para ti se muestran en una lista desplegable en la parte alta de la ventana de Control de código fuente; ver Imagen 8-1 en el Capítulo 8. Esta ventana realmente muestra los contenidos del repositorio con los contenidos del área de

trabajo seleccionada puestos en la capa superior. Por lo tanto, una carpeta en el repositorio que no está mapeada a un área de trabajo, se muestra atenuada, y cuando se selecciona, sólo sus contenidos del repositorio se muestran en el panel derecho. Pero cuando una carpeta mapeada del repositorio es seleccionada, sus contenidos del repositorio se incrementan con los contenidos del área de trabajo, junto con la información de estado, así como si es actualizada con respecto a los contenidos actuales del repositorio; el último es "sí."

Fusión de Cambios

Cuando un archivo está previsto para desprotecciones múltiples, más de una persona puede cambiarlo al mismo tiempo. Sin embargo, después de que la primera persona suba el archivo de regreso al repositorio, todos los demás deben fusionar su copia del archivo con la última versión del repositorio, antes de que se permita proteger sus cambios. Veamos cómo funciona repitiendo nuestro episodio de programación, pero esta vez con dos desarrolladores: Sarah, que pone la propiedad de BackColor en el formulario principal; y Luke, que modifica su título de ventana.

NOTA | Un área de trabajo de un miembro del equipo, generalmente mapea una carpeta de repositorio para un directorio local en su propio PC, así pues físicamente aislado de las áreas de trabajo de otras personas. Sin embargo, incluso cuando la gente comparte el mismo PC sus áreas de trabajo deberían ser consideradas privadas.

Ejercicio 9-3: Fusión de Cambios Hechos por Dos Desarrolladores para el Mismo Fichero

En la primera parte del ejercicio, Sarah recibe la versión más reciente de la Solución osImageManager, con los cambios en el título de la ventana del formulario principal hechos por Luke en el paso 6b del Ejercicio 9-1.

1. Entra al DeveloperPC3 como Sarah (colaborador de OSPACS), pon en marcha Visual Studio, y luego conéctate al Proyecto de equipo OSPACS, como se describe en el Ejercicio 5-7 en el Capítulo 5; ver Apéndice A para los detalles sobre este PC y los grupos de seguridad de Sarah.
2. Repite los pasos 2, 3, y 4 del Ejercicio 9-1, pero especifica el área de trabajo, con el propósito de que la última versión de la Solución osImageManager quede puesta en c:\Sarah\OSPACS.
3. Sal del DeveloperPC3.

La próxima parte del ejercicio ve a Luke hacer algunos cambios adicionales en el formulario principal de osImageManagerApp, usando la misma versión de archivos que Sarah recibió del repositorio, y luego devolviéndolos al repositorio para crear otra versión de estos archivos.

4. Entra al DeveloperPC como Luke, pon en marcha Visual Studio, y conéctate al Proyecto de equipo OSPACS.
5. Repite el paso 3 del Ejercicio 9-1 para que tengas actualizada el área de trabajo de Luke con la última versión de la Solución osImageManager, solo para asegurarte de que sea la misma versión que Sarah copió en su área de trabajo.

6. Abre la Solución de Visual Studio osImageManager en el área de trabajo de Luke y cambia el título del formulario principal (Form1), editando su propiedad de Texto; ver los pasos 5 y 6 del Ejercicio 9-1. Esto cambia los archivos de Form1 en área de trabajo de Luke.
7. Genera y prueba la Solución osImageManager en el área de trabajo de Luke y luego protege el conjunto de cambios haciendo clic en el botón Proteger en el ventana de Cambios pendientes; ver los pasos 7, 8, y 9 del Ejercicio 9-1.
8. Sal del DeveloperPC.

En la última parte del ejercicio, Sarah hace algunos cambios para los archivos en su área de trabajo, y luego funde estos cambios con la versión más reciente de los archivos en el repositorio, antes de que se devuelvan al repositorio para crear una nueva versión que contenga ambos conjuntos de cambios.

9. Entra al Developer3 como Sarah, pon en marcha Visual Studio, y conéctate al Proyecto de equipo OSPACS otra vez.
10. Abre la Solución Visual Studio osImageManager en el área de trabajo de Sarah y cambia el título del formulario principal (Form1) editando su propiedad BackColor de una manera similar al del paso 6 del Ejercicio 9-1. Esto cambia el archivo de Form1 en el área de trabajo de Sarah.

NOTA | La herramienta de fusión de VSTS maneja conflictos de protección por defecto, pero puedes especificar una herramienta diferente para manejar los fusionados de tus ficheros cambiando la Configuración de Herramientas de usuario para control de código fuente; ver Imagen 8-5 en el Capítulo 8.

11. Genera y prueba la Solución en el área de trabajo de Sarah, y luego protege el conjunto de cambios haciendo clic en el botón Proteger de Cambios pendientes. Sin embargo, esta vez no tendrá éxito la subida y aparece el diálogo Resolver conflictos.
12. El diálogo Resolver conflictos indica que la copia de Form1.designer.cs en el área de trabajo de Sarah es más viejo que el versión actual en el repositorio, protegido por Luke en el paso 7. Sigue los siguientes pasos:
 - a. Hay solo un fichero con el que resolver el conflicto, así que Form1.designer.cs es el único elemento listado en el diálogo Resolver conflictos. Selecciona este archivo y haz clic en el botón de Resolver, para abrir otro diálogo que sugiere algunas maneras de resolver el conflicto. Selecciona "Combinar cambios en la herramienta de combinación" y luego haga clic en Aceptar para abrir la herramienta de Combinar.
 - b. La herramienta Combinar muestra la diferencia entre el archivo en el área de trabajo de Sarah y el guardado en el repositorio; ver Imagen 9-3. Determinas qué acción tomar modificando el código en la ventana inferior. En este caso, los dos cambios no tienen conflicto así que haz clic en Aceptar para dejar que la herramienta Combinar los combine en el archivo ubicado en el área de trabajo de Sarah y guarde los cambios del fichero.
 - c. El diálogo Resolver conflictos no contiene conflictos adicionales así que haz clic en Cerrar.
13. Deberías ahora poder proteger el conjunto de cambios en el área de trabajo de Sarah como su copia de Form1.designer.cs originada desde la última versión en el repositorio. Por tanto, haz clic en el botón Proteger en la ventana de Cambios pendiente.

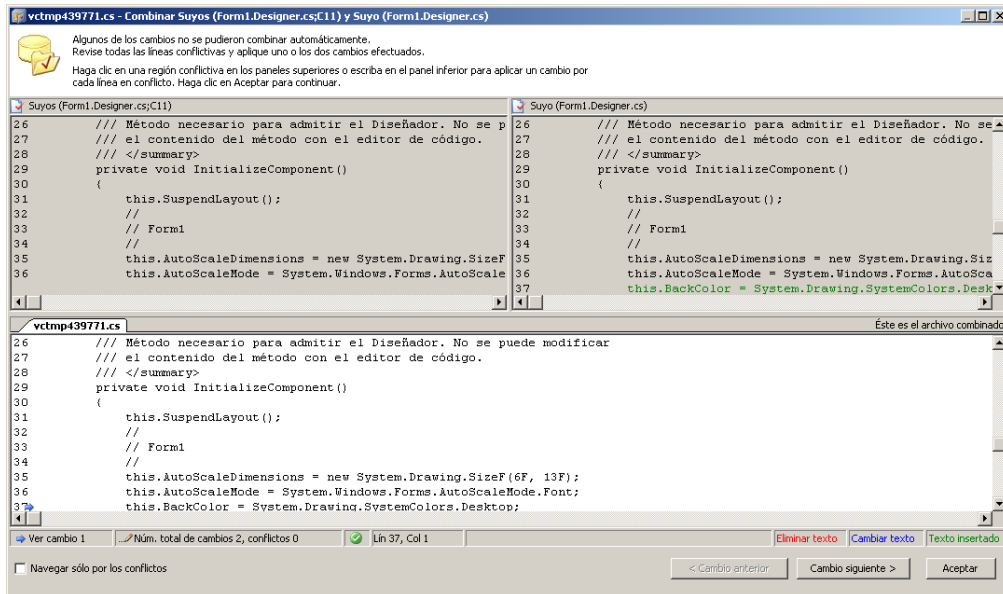


Imagen 9-3: Resolver los conflictos con la herramienta Combinar de VSTS

14. Sigue los mismos pasos dado en el paso 2 del Ejercicio 8-5, en el Capítulo 8, para etiquetar esta última versión de osImageManager en el repositorio como "v1-0-0-0 Iter0 – Poner título y fondo".
15. Has terminado el ejercicio ya, así que sal del sistema.

CONSEJO | Los ficheros gráficos están habitualmente configurados como de “bloqueo estricto”, porque podría ser difícil fundir los cambios que surgieran cuando diferentes personas editan la imagen. Consecuentemente, restringe los derechos de desproteger de cada fichero para solo la gente que realmente los mantendrá.

Volver a Una Versión Previa

Uno de los grandes beneficios que consigues al implementar el control de versiones es la habilidad para volver atrás y reconstruir el producto, según era en cualquier punto de la historia de su desarrollo. A veces sólo quieres volver atrás unas pocas horas para devolver el código al último estado en que funcionaba. Sin embargo, en las otras ocasiones puedes necesitar retroceder meses para reconstruir una versión del producto para un cliente en especial.

Ejercicio 9-4: Volver a Una Versión Previa

Después de terminar este ejercicio, habrás devuelto el código del Proyecto de equipo OSPACS al estado que tenía cuando fue etiquetado en el Ejercicio 8-5 en el Capítulo 8.

1. Entra al DeveloperPC como Luke (colaborador de OSPACS), pon en marcha Visual Studio, y luego conéctate al Proyecto de equipo OSPACS.
2. Copia una versión específica de la Solución Visual Studio osImageManager del repositorio de TFS en el área de trabajo de Luke:
 - a. Selecciona la carpeta osImageManager en el Explorador de control de código fuente y seleccionar Obtener la versión específica del menú contextual (clic derecho | Obtener versión específica) para abrir el cuadro de diálogo Obtener.
 - b. Escoge Etiqueta en la caja de la lista desplegable y haz clic en el botón “...” para abrir el cuadro de diálogo de Buscar etiqueta.
 - c. Haz clic en el Buscar y selecciona la etiqueta que creaste en el Ejercicio 8-5, en el Capítulo 8, para el punto de partida inicial. Haz clic en Cerrar para regresar al cuadro de diálogo Obtener.
 - d. La caja de etiqueta del cuadro de diálogo Obtener ahora contiene la etiqueta del punto de partida inicial y el botón Obtener está activado. Selecciona Obtener para cerrar el diálogo y actualizar el área de trabajo de Luke.
3. Abre la Solución osImageManager (Archivo | Abrir | Proyecto o solución) y vuelve a generarlo (Generar | Generar solución), para verificar que has reconstruido con éxito esta línea base inicial. Cuando ejecutes osImageManagerApp, Form1 debería aparecer como el título de la ventana de su formulario principal.
4. Has terminado el ejercicio ahora, así que sal del sistema.

NOTA | Etiquetar tu Proyecto de equipo en hitos significativos hace más fácil identificar la versión que podrías querer para restaurar a continuación. Asegúrate de aplicar la etiqueta a la carpeta raíz (\$\OSPACS) para que lo apliques a cada cosa en tu Proyecto de equipo.

Creación de una Rama

A veces puedes necesitar volver a una versión más reciente sin cambiar la última versión de los archivos fuente en tu repositorio. Por ejemplo, podrías necesitar aplicar una corrección rápida de un error a la v1.2 de tu producto mientras el resto del equipo continúa trabajando en el lanzamiento de la v1.4. En tales circunstancias, no quieres deshacer toda la base de código porque esto significa que el equipo entonces necesitará implementar todo su trabajo desde la v1.2 a hasta la v1.4. Algunas de las opciones que podrías considerar incluyen lo siguiente:

- **Crear una rama de código para la corrección del error y lanzarla para el cliente** - Creas dos bases de código distintas: la rama v1.2 y el (tronco) principal. Arreglas el defecto en la rama v1.2 y luego lanzas una versión con la corrección del error para el cliente desde esta base de código. Ésta es a menudo tu mejor solución táctica.
- **Crear una implementación local y lanzarla para el cliente** – Regresas a la v1.2 sólo en tu propia área de trabajo, arreglas el defecto, y luego generas una versión con el arreglo del error para el cliente desde los archivos en tu directorio local. Sin embargo, guardas estos cambios en el repositorio creando un aplazamiento llamado "1.2.1", como se describe en el Ejercicio 9-6.
- **Arreglar el problema para el próximo lanzamiento programado para el cliente** – Regresas a la v1.2 en tu propia área de trabajo solo para identificar el problema, y luego aplica estos cambios a la última versión de la base de código, con el propósito de que sea