

Help topic not found.

There is no online Help on the current topic.

What's New in Microsoft Visual SourceSafe Version 5.0?

See Also

Microsoft Visual SourceSafe version 5.0 has many new features, including:

- Better integration with Microsoft Visual Basic, Microsoft Visual C++, Microsoft Access, Microsoft Visual FoxPro.
- New features making web management easier, including site map generation, easy file deployment to web servers, and more!
- A new visual merge window that makes merging conflicting changes easier than ever before.
- A new project difference window so you can not only see differences between a project and its working folder, but also move files from one place to another.
- File editing with a double-click in the Visual SourceSafe Explorer.
- The ability to name and open different Visual SourceSafe databases dynamically.
- Performance enhancements, especially if you use Visual SourceSafe over a slow network or modem.
- Quick customization of your workspace through new tabbed dialog boxes.
- Improved drag and drop and right mouse-button support.
- A cloaking feature to let you hide selected projects from some commands.
- An archive utility so your administrator can archive (and later restore) unneeded portions of your **Visual SourceSafe database** to a backup location.
- Visual SourceSafe for the Macintosh is now available from MetroWerks. There used to be a Microsoft Visual SourceSafe program for the Macintosh, and that product still exists with full PC-compatibility. However, that program is now made and sold by Metrowerks, who will handle support issues as well as questions about using Visual SourceSafe on the Macintosh computer.
- Visual SourceSafe for UNIX is now available from MainSoft, who will handle support issues as well as questions about using Visual SourceSafe on UNIX.

See Also

Command Summary

Visual SourceSafe Explorer

Visual SourceSafe Naming Syntax and Conventions

Using Visual SourceSafe 4.0a with 5.0

Getting Started With Visual SourceSafe

See Also

Microsoft Visual SourceSafe is a project-oriented version control system for all types of files, including text files, graphics files, binary files, sound files, and video files. Using Visual SourceSafe, you can track changes made to a file from the moment it was created. And you can merge changes from two or more different versions of a file into one file that contains them all.

Setting Up Visual SourceSafe

Getting started with Visual SourceSafe begins with setting up and connecting to a **Visual SourceSafe database**. When you use Visual SourceSafe, your files are stored in such a database, which provides both safe storage and important historical information and version tracking.

Generally, your **Visual SourceSafe administrator** installs Visual SourceSafe on a network server for you and then adds your name to the database **user list**. Next, you need to use the NETSETUP.EXE file in the Visual SourceSafe directory on your server to set up the **Visual SourceSafe Explorer** on your computer. Then, you can run the Explorer from your computer and access the server's database where your files are stored.

More and more, developers are accessing SourceSafe's functions from within their development environment. Visual SourceSafe can be easily integrated with Microsoft Visual Basic, Microsoft Visual C++, Microsoft Access, Microsoft FoxPro, and other development tools. If Visual SourceSafe is integrated into your development environment, you do not need to run Visual SourceSafe separately to realize the advantages of source code control.

Note If a Visual SourceSafe database is not yet installed, see your administrator. If you are the administrator, read the Administration chapter of the *Microsoft Visual SourceSafe User's Guide* before proceeding.

Setting Up Projects

The next step is setting up a project. A project is a group of related files, typically all the files required to develop a software component.

Use the Create Project command to create a project or subproject. Use the Add Files command to add files to it. During the Add Files operation, Visual SourceSafe asks you to set a working folder for the project. The working folder is where you work on files and is typically a folder on your local computer.

Working With Files

When you want to modify a file, you check it out of the database. Visual SourceSafe copies the file from the database into your working folder. You can then edit the file. If anyone else attempts to check out the same file for editing, Visual SourceSafe generates a message stating the file is already checked out. This simple checkout protocol ensures that conflicts do not arise among multiple users working on the same file.

After you are done editing the file, you check it into Visual SourceSafe using the Check In command. This copies the modified file from your folder into the Visual SourceSafe's database, making your changes accessible to other users. However, Visual SourceSafe stores all the changes that have been made to the file—the most recent copy is always available, but earlier versions can be retrieved as well. Visual SourceSafe's **reverse delta** technology ensures that all versions of a file are available, but uses a minimum of disk space.

If you want to check something in a file, but don't need to edit it, you can use the Get Latest Version command to get the most recent version of the file into your working folder. You can use SourceSafe's Show History command to conveniently view the file or project's history, and the Show Differences command to determine differences between a file in your local folder and the latest version of that file

stored in the SourceSafe database.

See Also

[Add Files Command](#)

[Check In Command](#)

[Check Out Command](#)

[Create Project Command](#)

[Installing Visual SourceSafe](#)

[Understanding the Working Folder](#)

[Understanding Visual SourceSafe Projects](#)

[Visual SourceSafe Explorer](#)

[Visual SourceSafe Naming Syntax](#)

[Using Visual SourceSafe 4.0a with 5.0](#)

Installing Visual SourceSafe

When you install Visual SourceSafe, you need two components—the **Visual SourceSafe Explorer** interface and the **Visual SourceSafe database**, which holds your project files and **version** information. Both are needed to use Visual SourceSafe.

Visual SourceSafe can be installed in one of three configurations. You can:

- Install Visual SourceSafe on your local computer and create a database there. This makes limited use of Visual SourceSafe's version control features, because you are the only user.
- Install Visual SourceSafe on a network server and have each user run Visual SourceSafe Explorer from the server. This may affect performance because each user is running the Visual SourceSafe executable file over the network.
- Install Visual SourceSafe on a network server and install the appropriate Visual SourceSafe executable files on each user's local computer.

Usually, your **Visual SourceSafe administrator** sets up the Visual SourceSafe database on a server (using the Server option in the Setup program), and expects you to set up the necessary client files on your computer. To do this, connect to the Visual SourceSafe server and find the Visual SourceSafe root directory. Run the NETSETUP.EXE program found there to install the client components on your computer. This requires at least 2 megabytes of disk space on each local computer.

Note If you intend to run Visual SourceSafe together with an integrated development environment (IDE) like Microsoft Visual Basic or Microsoft Visual C++, you must perform the client setup to properly register Visual SourceSafe on your local computer. This client setup must occur after your administrator has finished installing SourceSafe on a network server. If you want to have both the Visual SourceSafe database and the client on the same computer, use the Server setup button to install a SourceSafe database on your local computer rather than a network server. Then run setup again and click the Client button to register your IDE properly.

The *Microsoft Visual SourceSafe User's Guide* contains detailed information on setting up Visual SourceSafe. Chapter 8, "Administration," provides information on Visual SourceSafe database administration responsibilities.

Visual SourceSafe Naming Syntax, Conventions, and Limitations

Visual SourceSafe has the following naming conventions and limitations.

- **User names** and label names can be up to 31 characters long.
- User passwords can be up to 31 characters long.
- Filenames can be up to 255 characters long. Filenames can include spaces on operating systems that support them.
- Project paths can be up to 259 characters long (including the filename).
- Project or file **labels** can be up to 31 characters long.
- Search strings can be up to 63 characters long.
- Comments can be up to 63 characters long for the Check Out command, and 4K characters long for other comments.
- Lines in the Visual SourceSafe initialization files can be up to 511 characters long.
- Files and projects (added together) in the Visual SourceSafe database can total no more than 4 billion.
- Files and projects (added together) in any one project of the Visual SourceSafe database can total no more than 8191.
- The size of a file stored in the Visual SourceSafe database is limited to 2 gigabytes.
- The number of versions of a given file that Visual SourceSafe can store is limited to 32,767.

Most names in Visual SourceSafe cannot begin or end with spaces or tabs. Any characters can be used for names or labels, except the following:

- Dollar sign (\$)
- At sign (@)
- Angle brackets (< >), brackets ([]), braces ({ }), and parentheses (())
- Colon (:) and semicolon (;)
- Equal sign (=)
- Caret sign (^)
- Pipe (vertical bar) (|)
- Asterisk (*)
- Exclamation point (!)
- Forward (/) and backward slash (\)
- Percent sign (%)
- Question mark (?)
- Comma (,)
- Quotation mark (single or double) (' ")
- Tab

Visual SourceSafe supports Universal Naming Convention (UNC) filenames. If your network operating system supports UNC pathnames, you can use them anywhere you specify a path name.

Visual SourceSafe does not support the use of @ characters on Banyan Vines networks. Banyan users can access all SourceSafe functionality by using mapped network drive letters.

Visual SourceSafe Keyboard Guide

See Also

Navigation Keys

The following tables show some of the keys you can use to move around Visual SourceSafe Explorer.

To navigate Visual SourceSafe Explorer

Press	To
TAB	Move between the file list and the project list .
UP ARROW	Move up the file list or project list.
DOWN ARROW	Move down the file list or project list.
HOME	Move to the top of the file list or project list.
END	Move to the bottom of the file list or project list.
F6	Go to next pane.
SHIFT+F6	Go to previous pane.
SHIFT+DOWN ARROW	Move to next file or project, extend selected range.
SHIFT+UP ARROW	Move to previous file or project, extend selected range.

Navigation Keys for the File List

Press	To
HOME, then SHIFT+END	Select all files in the current project. (Or, simply select the project in the project list.)
ENTER	View a file.

Navigation Keys for the Project List

Press	To
ENTER	Show (or hide) subprojects .
NUMPAD *	Expand All projects and subprojects.
NUMPAD +	Expand a project.
NUMPAD -	Collapse a project.

Menu Shortcut Keys for Commands

Each command has a key combination associated with it that enables you to execute a command with keystrokes. Typically, you press the CTRL or ALT key and simultaneously press the underlined letter in the menu command.

Shortcut Keys for Commands

Press	To
ALT+ENTER	Show properties of a file or project.
ALT+X	Close Visual SourceSafe.
ALT+E	Edit a file.
CTRL+*	Globally expand all projects.
CTRL+A	Add files to the current project .
CTRL+C	Copy text.

CTRL+D	Set the working folder for the current project.
CTRL+ENTER	View a file.
CTRL+F	Show differences for the selected file or project.
CTRL+G	Get the selected file or project.
CTRL+H	Show the history of the selected file or project.
CTRL+INS	Copy selected text to Clipboard.
CTRL+K	Check out the selected file(s).
CTRL+L	Select files. (In the Project Difference window, CTRL-L is used for Reconcile All.)
CTRL+O	Check out a file.
CTRL+Q	Cancel a search display or status. Return to normal file display.
CTRL+R	View a file. (In the Project Difference window, CTRL-R is used for Report.)
CTRL+S	Search for a file checked out to a specific user. Use CTRL+Q to return to normal file display.
CTRL+U	Check in the selected file(s).
CTRL+V	Paste text from the Clipboard .
CTRL+W	Perform a Wildcard character search.
CTRL+X	Cut text and place it on the Clipboard.
CTRL+Z	Undo editing in dialog box.
DELETE	Delete the selected file or project.
ENTER	View a file.
F1	Get online Help on a specific item.
F5	Refresh file list.
INSERT	Create Project (if focus is in project pane), or Add Files (if focus is in file pane).
SHIFT+CTRL+L	Select all files.
SHIFT+CTRL+S	Search for all files currently checked out. Use CTRL+Q to return to normal file display.
SHIFT+DELETE	Cut selected text and copy to Clipboard.
SHIFT+INSERT	Paste text on Clipboard to edit box.

See Also

[File Viewer Keyboard Shortcuts](#)

File Viewer Keyboard Shortcuts

The following tables show some of the keyboard shortcuts you can use to move around in any of the SourceSafe file viewers and to carry out certain commands.

Navigation Keys

Press	To
HOME	Go to beginning of line (scroll display).
CTRL+HOME	Go to beginning of file (scroll display).
CTRL+END	Go to end of file (scroll display).
CTRL+LEFT ARROW	Go left one word.
F7	Go to next difference (supported in Visual Difference Viewer only).
SHIFT+F7	Go to previous difference (supported in Visual Difference Viewer only).
F8	Go to next conflict (supported in Visual Difference Viewer only).
SHIFT+F8	Go to previous conflict (supported in Visual Difference Viewer only).
CTRL+RIGHT ARROW	Go right one word.
DOWN ARROW	Go down one line.
END	Go to end of line (scroll display).
LEFT ARROW	Go left one character.
PAGE DOWN	Go down one page.
PAGE UP	Scroll display up one page.
RIGHT ARROW	Go right one character.
SHIFT+HOME	Go to beginning of line (scroll display). If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+CTRL+HOME	Go to beginning of file (scroll display). If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+CTRL+END	Go to end of file (scroll display). If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+CTRL+LEFT ARROW	Go left one word. If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+CTRL+RIGHT ARROW	Go right one word. If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+DOWN ARROW	Scroll display down one page. If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+END	Go to end of line (scroll display). If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+LEFT ARROW	Go left one character. If a selection is active, the

	accelerator key maintains the selection state and modifies the selection.
SHIFT+PAGE DOWN	Go down one page. If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+PAGE UP	Go up one page. If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+RIGHT ARROW	Go right one character. If a selection is active, the accelerator key maintains the selection state and modifies the selection.
SHIFT+UP ARROW	Go up one page. If a selection is active, the accelerator key maintains the selection state and modifies the selection.
UP ARROW	Go up one one line.

Shortcut Keys

Press	To
CTRL+INS	Copy text.
CTRL+C	Copy text.
ALT+G	Go to specified line number.
ALT+S	Find a string in the viewed file.
F1	Get online Help on a specific item.
F3	Find next search string occurrence.
SHIFT+F3	Find previous search string occurrence.
SHIFT+CTRL+O	Select difference text (supported in Visual Difference Viewer only).

Visual SourceSafe: Project-Oriented Version Control

See Also

Microsoft Visual SourceSafe is a version control system for team development of software applications. Version control systems track and store changes to a file so developers can review a file's history, return to earlier versions of a file, and develop programs concurrently. Microsoft Visual SourceSafe does this using **reverse delta** technology, and stores only the changes to a file, not each complete version of the file itself. And, unlike other version control systems, Visual SourceSafe is project-oriented.

Visual SourceSafe Projects

A project is a collection of files you store in Visual SourceSafe. Organizing your code in a Visual SourceSafe project is similar to organizing files in folders. The manner in which you set up a project and the **subprojects** beneath it mirrors the folder structure of an operating system. Rather than start from a root folder (like C:\), however, you start from a **root project**, \$/.

Another difference is that in Visual SourceSafe, one file can simultaneously exist in many projects—it is **shared**. This ability to share a file is what enables Visual SourceSafe to ease tracking modular code

—when you modify a file in one project, the change is automatically propagated to all the projects sharing the file.

Visual SourceSafe's project hierarchy is operating system independent. A **Visual SourceSafe database** of files can be accessed transparently from Windows 95 or Windows NT, and Visual SourceSafe is available from third-party vendors for the Macintosh and UNIX. Visual SourceSafe maintains both long filenames (up to 255 characters) and short filenames for every file (MS-DOS standard format), and translates end-of-line characters.

Visual SourceSafe Version Control

Visual SourceSafe provides a dimension of control not available in an operating system: version control. Visual SourceSafe can maintain multiple versions of a file, including a record of the changes to the file from version to version. Version control addresses:

- Team coordination—making sure, by default, that only one person at a time is modifying a file, to prevent accidental replacement of another user's changes. (Your SourceSafe administrator can change this default to allow multiple simultaneous check outs of a single file.)
- Version tracking—archiving and tracking old versions of source code and other files, which can be retrieved for bug tracking and other purposes.
- Cross-platform development—tracking portability issues involved in maintaining one code base across multiple development platforms.
- Reusable or object-oriented code—tracking which programs use which modules so that code can be reused.

See Also

[Getting Started With Visual SourceSafe](#)

[Installing Visual SourceSafe](#)

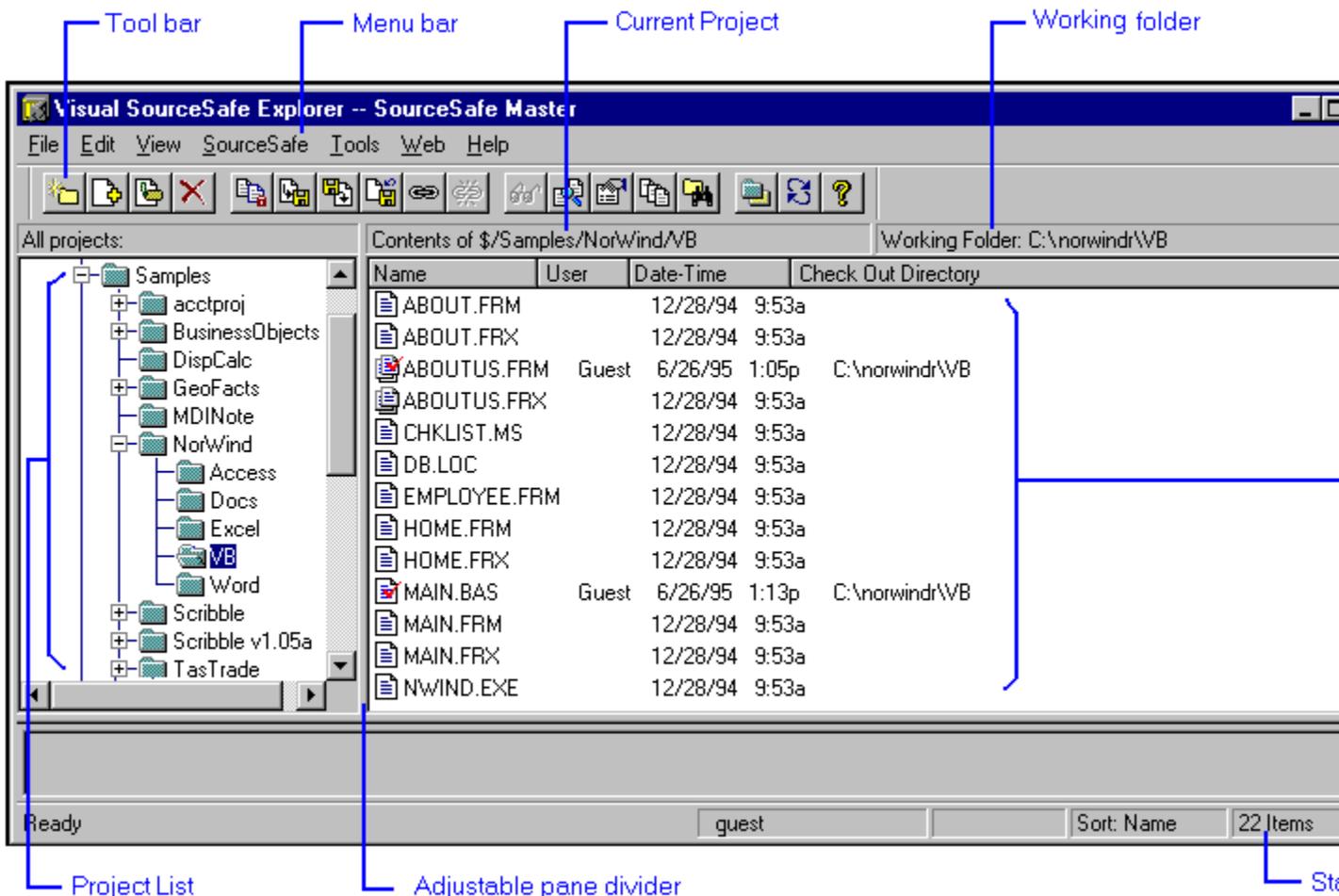
[Understanding Visual SourceSafe Projects](#)

[Using Multiple Checkouts of a File](#)

[Visual SourceSafe Explorer](#)

Visual SourceSafe Explorer

See Also



When you start Microsoft Visual SourceSafe, it displays Visual SourceSafe Explorer. From the Explorer window, you can navigate the **Visual SourceSafe database**—view your **project list**, examine file statistics, and execute commands that act on projects and files. The **project pane** shows the project list, and includes information on special projects, like those you have designated **web site projects**, cloaked projects, or the **currently open project**.

Visual SourceSafe Explorer displays important status information, such as your current **working folder**, search criteria, number of files, and so forth. Some of this information is shown in **file pane** column headings, and other information is shown in the **status bar**, at the bottom of the Visual SourceSafe Explorer window. The title bar of Visual SourceSafe Explorer shows the database name you are connected to (if one has been defined by your **Visual SourceSafe administrator**.)

The menu bar, the horizontal list of menu names, is at the top of Visual SourceSafe Explorer. You click a menu name to display the list of commands associated with the menu.

Just below the menu bar, the toolbar in Visual SourceSafe Explorer provides command buttons so you can quickly access the most common Visual SourceSafe commands. You can position your mouse cursor over each toolbar button and hold it there for a couple of seconds to pop up an explanation of what the button is for.

At the bottom of the default Visual SourceSafe Explorer is the **Results pane**. This pane shows the results of your current Visual SourceSafe operation. As with other Explorer panes, you can resize

this pane. You can also eliminate it entirely by unchecking the Show Results pane checkbox on the Tools Options View tab.

Drag and Drop

Several **drag and drop** operations are enabled with Visual SourceSafe. For example, you can drag a file from the file pane to the project pane to quickly share the file with another project. By holding down the CTRL key while dragging a file to a project, you can share and immediately branch the file.

You can also drag a file from the Windows95 Explorer into the Visual SourceSafe Explorer to add it to a project, or drag a folder from the Windows95 Explorer into the Visual SourceSafe Explorer to create a new project under the project you drop on. Of course, you can select multiple files or folders and do the same thing with several items at once.

If you drag a file you have checked out from the Windows95 Explorer into the Visual SourceSafe project containing that file, you check the file back in. If you don't have the file checked out, an error results. You can also drag your working folder for a project from the Windows95 Explorer into the appropriate Visual SourceSafe project to check in all checked out files in that folder.

Finally, if you drag a Windows95 Explorer shortcut into Visual SourceSafe, Visual SourceSafe displays a dialog box to allow you to choose whether you want to save the shortcut itself, or the file the shortcut points to, in Visual SourceSafe.

The Right Mouse Button

Visual SourceSafe Explorer provides another way of working with the **file list** and **project list**. If you click a file with the right mouse button, Visual SourceSafe displays a shortcut menu of common file commands. If you click a project with the right mouse button, Visual SourceSafe displays a menu containing commands that operate on projects.

Visual SourceSafe Explorer also displays shortcut menus during some drag and drop operations. For example, when dragging items within Visual SourceSafe, a shortcut menu is displayed when you drag a project to another project using the right mouse button, and drop the project over the destination. The menu choices include sharing the project, sharing and branching the project, and moving the project. If you are dragging a selection and you attempt to drop it on an area that cannot receive the drop, your cursor will change from the normal cursor to a .

Similarly, when you drag items from the Windows95 Explorer into Visual SourceSafe, the right mouse button displays a list of options including Add File(s), Check In, Undo Check Out, and Cancel. These options all perform the same functions as they do when selected from the Visual SourceSafe menus, with the exception of Check In and Add, which are only available when you have selected a group of files, some of which are checked out to you from a project, and others are not included in your project. If you select this option, Visual SourceSafe will add the selected files that are not already in your project, and will check in the files already checked out to you.

Note You cannot drag items from one copy of Visual SourceSafe to another unless you have the same **Visual SourceSafe database** open in both copies.

See Also

[Configuring Visual SourceSafe Explorer](#)

[File Pane](#)

[Project Pane](#)

[Results pane](#)

[Status Bar](#)

[Branching Files](#)

Status Bar

See Also

Visual SourceSafe Explorer has a status bar at the bottom that shows the following information. You can turn off the status bar display on the View tab of the SourceSafe Options command on the Tools menu.

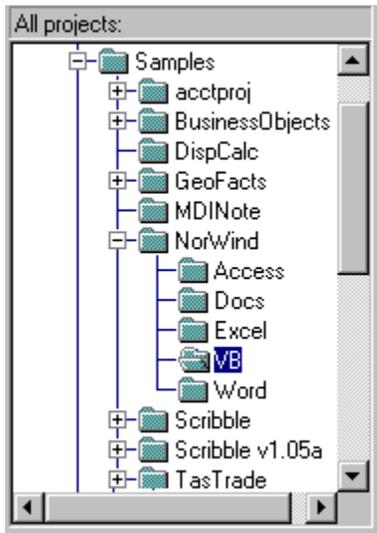
Information	Purpose
Message	Conveys a status message, usually, "Ready," or information on a currently selected command.
Username	Shows the <u>username</u> of the user currently logged on.
Search criteria	Shows the search criteria currently in effect. For example, Visual SourceSafe displays "Search: *.C" if you restrict the <u>file list</u> to .C files.
Sort order	Shows the sort order currently in effect in the file list.
File count	Shows the current number of files displayed in the file list.

See Also

Sort Command

Project Pane

See Also



The project pane is a hierarchical display of **projects**. The **root project** (\$/) is always at the top of the **project list**. As you move through the project list, Visual SourceSafe displays the full path of the currently selected project above the **file list**.

The project list is collapsible and expandable. Click the root project, or the minus sign (-) next to it, to collapse the root project. A plus sign (+) appears next to the folder icon for the root project, indicating that there are **subprojects** that are not displayed. An open folder indicates the current project. A folder with a small globe on it (🌐) indicates a **web site project**, and a veiled image of a folder indicates a **cloaked project**.

To display projects, double-click the root project. Press the ENTER key or double-click any of these projects, and Visual SourceSafe displays the subprojects of those projects, if any.

In addition, you can use other keyboard keys to navigate through the project list. On the numeric keypad, the + key expands a project, the - key collapses a project, and the * key recursively expands a project. If you press the * key on the root project, the project list expands fully, displaying all the projects in the currently open Visual SourceSafe database.

See Also

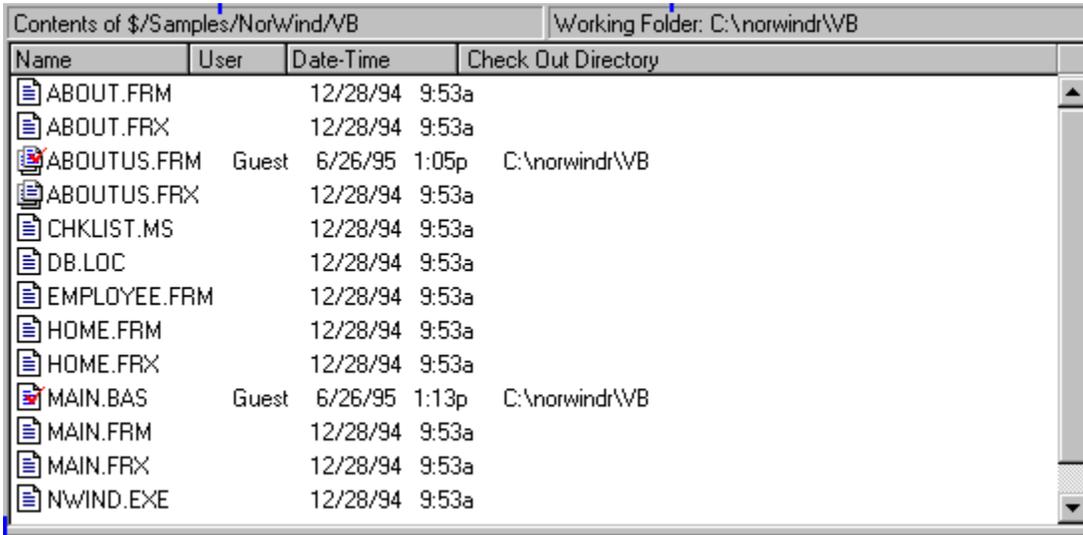
[The Visual SourceSafe Explorer](#)

[Visual SourceSafe Web Features](#)

[Cloaking](#)

File Pane

See Also



The screenshot shows a file pane window titled "Contents of \$/Samples/NorWind\WB" with a "Working Folder: C:\norwind\WB". The file list is as follows:

Name	User	Date-Time	Check Out Directory
ABOUT.FRM		12/28/94 9:53a	
ABOUT.FRX		12/28/94 9:53a	
ABOUTUS.FRM	Guest	6/26/95 1:05p	C:\norwind\WB
ABOUTUS.FRX		12/28/94 9:53a	
CHKLIST.MS		12/28/94 9:53a	
DB.LOC		12/28/94 9:53a	
EMPLOYEE.FRM		12/28/94 9:53a	
HOME.FRM		12/28/94 9:53a	
HOME.FRX		12/28/94 9:53a	
MAIN.BAS	Guest	6/26/95 1:13p	C:\norwind\WB
MAIN.FRM		12/28/94 9:53a	
MAIN.FRX		12/28/94 9:53a	
NWIND.EXE		12/28/94 9:53a	

The file pane contains a heading that displays the project path and the **working folder** for the **current project**. (If no working folder has been assigned, the working folder specification reads "No Working Folder.") It also contains a **file list**, which displays all the files in the current project. The file list has four labeled columns:

Column Heading	Purpose
Filename	For Windows 95 and Windows NT, Visual SourceSafe supports long filenames (up to 255 characters). On other operating systems, you can toggle between the long filename and short filename (filenames truncated to an MS-DOS format name) modes.  File exists only in the current project.  File is shared by two or more projects.  File is currently checked out.  File is checked out to more than one person.  File is pinned to a specific version.  File is exclusively checked out to one person.
User	For a checked-out file , displays the name of the user who has the file checked out.
Date/Time	For a checked-in file , displays the date/time of the last modification. For a checked-out file, displays the date and time that the file was checked out.
Check Out Folder	Shows the folder to which the file is checked out. If you have a file checked out to more than one computer, then the computer name is displayed along with the folder path.

You can sort the items in the file pane by clicking the appropriate heading.

See Also

File List

Visual SourceSafe Explorer

Configuring Visual SourceSafe Explorer

See Also

Visual SourceSafe Explorer is configurable in a number of ways. None of them has an effect on Visual SourceSafe's data or capabilities.

Refreshing the File List

While you are using Visual SourceSafe, other people may change the contents of the **current project**. For instance, they may check out and check in files. To stay up-to-date, Visual SourceSafe automatically refreshes its file list periodically—it re-reads all the information about the current project in the Visual SourceSafe database.

In addition, you can manually refresh the file list at any time. Simply press F5 or use the Refresh File List command (View menu).

To disable automatic file list refreshing, or change the delay between refreshes, use the View tab in the SourceSafe Options dialog box (Tools menu).

Resizing the File and Project Panes

To resize the file and project panes, click the project bar, which separates the **file** and **project panes**, and drag it to the left or right. This changes the amount of space allocated to the **project list** and **file list**, respectively. You can also drag the divider between file pane headers back and forth to resize the amount of space dedicated to each column.

Sorting the File List

You can sort the file list in a particular column by clicking that column's header. For example, to sort the file list by users who have the files checked out, click User above the file list.

Setting View Options

Use the View tab in the SourceSafe Options dialog box (Tools menu) to set many Visual SourceSafe configuration options. One option displays all filenames in lowercase letters. Another determines whether Visual SourceSafe displays the Results pane.

Showing or Hiding the Toolbar

You can choose to show or hide the Visual SourceSafe toolbar. On the Tools menu, click Options, and on the View tab, click Show Toolbar.

Customizing the Toolbar

You can enable or disable ToolTips, brief descriptions of the names of toolbar buttons, by using the Customize Toolbar command (Tools menu). Even if ToolTips are not enabled, the status bar displays information on menu items as you scroll a menu. The Customize Toolbar dialog box also displays buttons that you can add to your toolbar. To add a new button, drag it from the Available Buttons side of the dialog box to the Toolbar Buttons side. To remove a button, move it in the other direction within the dialog box. Re-order the list in the Toolbar Buttons side of the dialog to change the order of Toolbar Buttons.

You can also click in the margin area of the Toolbar to "detach" it from the Visual SourceSafe Explorer, and move it anywhere on your desktop.

When the toolbar looks exactly as you want it, click OK to save your settings.

Changing the Display Font

You can change the font used to display files and projects. Use the Font command (Tools menu) to change the Visual SourceSafe font face and size, or choose boldface or italic.

See Also

[Results pane](#)

[Customize Toolbar Command](#)

[Font Command](#)

[Options Command](#)

[View Options Tab](#)

[Starting Visual SourceSafe from the Command Line](#)

File List

See Also

The file list is a list of all the files in the current project.

Sorting the File List

The file list is usually sorted by filename. You can change the sort order, however, with the Sort command on the View menu. The available sort options are:

- Name – Filename
- Type – File extension
- User – **Username**
- Date – Date user checked out file or last check in date
- Checkout Folder – Folder to which file is checked out

Filename, User, Type, and Checkout Folder sort in ascending order; Date sorts in descending order, so that the most recent files are on top. The current sort order is always displayed in the status bar.

Note You can also sort the file list by clicking the appropriate column header in Visual SourceSafe Explorer.

When the file list is sorted by filename, you can move quickly through it by typing. For example, if you type the letter S, the selection moves to the first filename that begins with the letter S. If you then type O, the selection moves to the first filename starting with SO.

Ordinarily, the file list shows all the files in the current project. However, you can choose the Search command on the View menu to see only files that match certain criteria, such as all files with the *.C extension, or to view only checked-out files. Searches can be recursive, to search all the subprojects under the current project, or global, to search the entire project list.

Selecting Multiple Files

You can simultaneously select multiple files using one of the following methods:

- Hold down the SHIFT key in combination with the arrow keys or click files with the mouse.
- On the Edit menu, click Select to select files using wildcard characters. For example, if you type *.H and click OK, Visual SourceSafe selects all the .H files. Click Select to select files without closing the dialog box.
- On the Edit menu, click Select All to simultaneously select all the files in the list.

File List Symbols

The following symbols are used in the file list:

<u>Symbol</u>	<u>Meaning</u>
---------------	----------------

—	File.
—	Checked out file.
—	File checked out to multiple users.
—	Shared file.
—	File exclusively checked out.
—	File pinned to a specific version.
	Labeled file.

See Also

File Pane

Search Command

Select Command

Sort Command

Visual SourceSafe Explorer

Project List

See Also

The project list is a hierarchical display of **projects**. The **root project** (\$/) is always at the top of the list. Below that are **subprojects**. As you move through the project list using the UP and DOWN ARROW keys, you can see the full path of the **current project** displayed above the **file list**.

The project list does not generally display all the projects in your system, however, because it is collapsible and expandable. A plus sign (+) appears in the folder icon for the root project if there are subprojects that are not displayed. If all of a project's subprojects are expanded, a minus sign (-) is displayed to the left of the project name. If a project has no subprojects, neither a + or - is displayed next to the project name.

Tip If you press the * key on the numeric keypad on the root project, the project list fully expands, displaying all the projects in your system.

Other keys on the numeric keypad you can use to manipulate the project list are + (to expand a project) and - (to collapse a project).

Project List Symbols

The following symbols are used in the project list:

Symbol	Meaning
	Project.
	Current(ly) open project.
	Indicates subprojects exist in this project. Click the icon to expand the project list.
	Click to collapse the project list.
	Cloaked project.
	Web site project.
	Labeled project.

See Also

[Keyboard Guide](#)

[Project Pane](#)

[Understanding the Working Folder](#)

[Visual SourceSafe Explorer](#)

Using Visual SourceSafe Commands

See Also

Visual SourceSafe commands are grouped on menus at the top of **Visual SourceSafe Explorer**. Most commands work on the currently selected project or file. So, if you click a file and choose "Get Latest Version" you get that file; if you click a project and then choose "Get Latest Version" you get that entire project.

► **To carry out a command**

- 1 Select the file, files, or project that you want to act on
- 2 In the menu bar at the top of Visual SourceSafe Explorer, click the menu name to display the menu.
- 3 Click the command you want.
- 4 If the command displays a submenu to the right, click a command on the submenu.

In most cases, a Visual SourceSafe command displays a dialog box in which you can select and implement options. Any options you select affect the current operation only—their settings are not carried over to future operations.

Note You can also use the right mouse button in the SourceSafe Explorer to access several of the most common SourceSafe commands. Select the files or project you want to act on, then click the right mouse button and choose from the list of available commands.

► **To cancel a command**

- To cancel a command before it begins executing, click the Cancel button in the dialog instead of the OK button.
- To cancel a command while it is running, click the Cancel button that appears on the status bar at the bottom of the SourceSafe explorer during long commands, or press ESC.

See Also

Command Dialogs Options Tab

Controlling the Display of Command Dialog Boxes

Understanding the Working Folder

Controlling the Display of Command Dialog Boxes

Use the Command Dialogs tab in the Options dialog box (Tools menu) to indicate which dialog boxes should be displayed each time you execute the related command, and which should not. For instance, if you click the Get Latest Version check box under Projects, Visual SourceSafe presents you with the Get Latest Version dialog box each time you use the Get Latest Version command on a project. If you clear this check box, Visual SourceSafe performs the Get Latest Version command without displaying the dialog box.

Note When you initially install Visual SourceSafe, it is set up by default to display dialog boxes for most project-level operations, but not for most file-level operations.

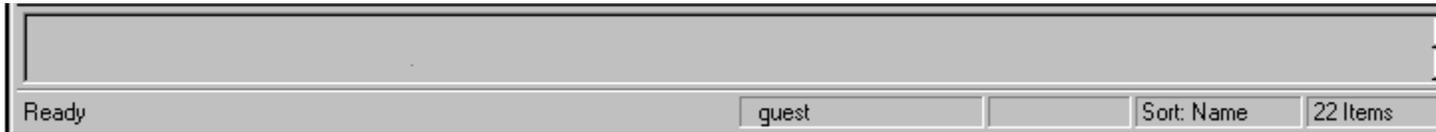
In addition, the Get Latest Version, Check Out, Undo Check Out, Difference, and History dialog boxes contain a Skip This Dialog option. Click this check box to clear the related option on the Command Dialogs tab. Thereafter, when you perform the command, Visual SourceSafe does not display the dialog box. To display the dialog box again, click the related check box on the Command Dialogs tab.

Note Even if you have set Visual SourceSafe not to display a particular dialog box, you can override this setting on a case-by-case basis.

To override the setting, hold down the SHIFT key when you click the command. Visual SourceSafe displays the dialog box for that instance of the command.

Results Pane

See Also



When you carry out a command that generates a status or error message, Visual SourceSafe displays the results of the command in the Results pane. You can resize this pane by dragging the frame around it to the desired position.

If you choose, you can tell Visual SourceSafe not to display the Results pane – clear the Show Results pane check box in the View tab of the SourceSafe Options dialog box (Tools menu). This automatically sets the Results_Windows initialization variable to No in your SS.INI file. Error messages, however, are still displayed.

See Also

Last Results Command

Results Window Initialization Variable

View Options Tab

Understanding Visual SourceSafe Projects

See Also

The organizational group Visual SourceSafe uses for your files is a project. Understanding projects is the key to understanding Visual SourceSafe.

It is helpful to think of a project as similar to a file system folder. Like a folder, a project is a collection of files that you create and maintain. Also, like a folder, a project is hierarchical; that is, you can place a **subproject** under another project, another subproject under that one, and so on.

Important While it is useful to think of a project as analogous to a folder, it is important to remember that it is not quite the same thing. Unlike a folder, every Visual Source project keeps a detailed record of its **history**, can be deleted and then recovered, and can share one file with many other projects.

There is always one **current project**, which Visual SourceSafe Explorer displays above the **file list**. You can change what is displayed by navigating the **project list**. Most commands you carry out act on files and subprojects inside your current project.

When setting up a project, keep these guidelines in mind:

- Mirror your folder tree in your project tree. That is, examine your folder tree to determine which subfolders are under which folders. Then, create the same project organization in Visual SourceSafe using the Create Project command (File menu). For instance, if you store your code in one folder with three subfolders, create a project in SourceSafe with three subprojects.
- Set up your Visual SourceSafe project to contain all the files necessary to build one program. Such files are generally code, but may also include makefiles, libraries, bitmaps, dynamic-link libraries, and subsidiary programs.
- Place only the most recent version of compilable code in the Visual SourceSafe project. Because you can recompile code with the source files stored in Visual SourceSafe, you don't have to create a larger SourceSafe database than necessary by storing .EXEs and .DLLs.
- Keep files checked out only as long as necessary.
- Label each version of a project by using the Label command (File menu) at appropriate points in the project's history, so you can easily return to important milestones.
- Store source files, and not executable files or object files.

Note Because executable files tend to be large, storing them can rapidly consume disk space. However, if you have an object file or executable file that takes time to build, you may want to check it into a project to save rebuilding time in case the file is lost and you want to recover it.

Project Path Syntax

Project path syntax is similar to folder path syntax, but always starts with a dollar sign (\$). The dollar sign symbolizes the **root project** in the **Visual SourceSafe database**. When you specify a project without a forward slash (/), it is assumed that the project is under the current project. You can use a dot (.) to refer to the current project, two dots (..) to refer to its **parent project**, three (...) to the parent project's parent, and so on.

A few examples of project syntax follow:

- `$/` The root project.
- `$/Word` A project under the root.
- `$/Win` A project under the current project.
- `$/Word/me.c` A file in the `$/Word` project.
- `$/../me.c` A file in the current project's parent.

See Also

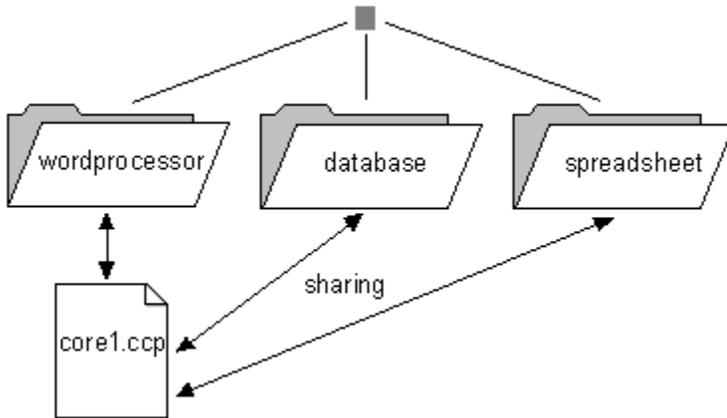
Create Project Command

Project List

Project Pane

Sharing Files

See Also



Visual SourceSafe **projects** are in many ways analogous to operating system folders. However, whereas every file in an operating system exists uniquely in a folder, a Visual SourceSafe file can simultaneously exist in many different projects.

In Visual SourceSafe, you can **share files**. Inside the **Visual SourceSafe database**, there is only one copy, the **master copy**, of the file; each project to which the file belongs simply has a pointer to it. When you change the file in any one project, Visual SourceSafe immediately updates it in all the projects that share it.

Note It is important to understand what happens when you carry out the **Destroy** and **Purge** commands on shared files.

If a file is shared by multiple projects, destroying or purging it in one project does not delete the file from the Visual SourceSafe database, and consequently does not free any disk space.

► **To share a file**

- 1 In the **Visual SourceSafe Explorer**, click the project into which you want to share a file.
- 2 On the SourceSafe menu, click Share.
- 3 In the Share with Project dialog box, select the file(s) from another project that you want to share with your **current project**. Click Share.
- 4 Click Close to close the dialog box.

Note When you add a file called ERROR.C to Project A and to Project B, you have added two unconnected files with the same name. When you add the file to one of the projects, and share it with the another project, the two projects are **linked** to the same file.

A shared file exists equally in all the projects to which it is shared – that is, each project in which the file is shared has an equal pointer to the one master copy of the file.

Pinning a File

Generally, projects that share a file all use the most recent version of the file. However, you can specify that a project share a specific version—the project always uses that version no matter what changes are made elsewhere. In Visual SourceSafe, sharing a particular version of a file is known as *pinning the file*.

Sharing a particular version of a file enables you to control the pace of changes to different facets of your project. For example, you may be writing code that uses a function in an application programming interface (API) at the same time that another user continues to change the API. Use pinning to keep from receiving those changes until you are ready.

- **To share an particular version, or pin, a file**

- 1 In Visual SourceSafe Explorer, click the file you want to pin.
- 2 On the Tools menu, click Show History.
- 3 Click the version of the file you want, and then click Pin.
- 4 Click Close to close the dialog box.

When it is time to share the most recent version of the file, you unpin it.

- **To share the most recent version, or unpin, a file**
 - In the History of File dialog box, click the pinned file, and then click Unpin.

See Also

Branching Files

Links Tab

Destroy Command

Purge Command

Share Command

Getting, Checking Out, Checking In, and Viewing Files

See Also

By far, the most commonly used Visual SourceSafe commands are those that copy files into and out of the **Visual SourceSafe database** during day-to-day use.

Getting Files

When you want access to a file for viewing or compiling, but not for modification, use the Get Latest Version command. This copies the file from the **current project** into your **working folder**. The file Visual SourceSafe creates is **read-only**—any modifications cannot be saved.

- **To get the most recent version of a file**
 - Click the file you want in **Visual SourceSafe Explorer**, and on the SourceSafe menu, click Get Latest Version.
- **To get an earlier version of a file**
 - 1 Click the file you want in Visual SourceSafe Explorer, and on the Tools menu, click Show History.
 - 2 In the History of File dialog box, click the version of the file you want, and then click Get Latest Version.

Viewing Files

You can view the **master copy** of a file without placing a **local copy** of the file in your working folder.

- **To view a file**
 - Click the file you want in **Visual SourceSafe Explorer**, and on the Edit menu, click View File.

Editing Files

You can edit a file in your working folder by double-clicking the file in the file list. In the confirmation dialog, choose Checkout this file, and edit it in your working folder. Visual SourceSafe ensures that the file is checked out, and opens it in the editor associated with the file extension.

Checking Out and Checking In Files

To edit a file, you must check it out of the Visual SourceSafe database. The Check Out command creates a writable copy of the file from the project in your working folder. A file check out is generally exclusive, that is, no one else can check out a file that you have checked out. Visual SourceSafe indicates who has a file checked out in the User column of the **file pane**.

Note Your installation of Visual SourceSafe may be set up to allow **multiple checkouts**. See your **Visual SourceSafe administrator** for more details.

You can complete your check out operation in one of two ways. You can check your updated file into Visual SourceSafe, storing your changes in the current project. Or, you can undo your check out, which cancels your changes, both in Visual SourceSafe and in your working folder—the file returns to the way it was before you checked it out.

- **To check out a file**
 - Click the file you want in Visual SourceSafe Explorer, and on the SourceSafe menu, click Check Out.
 - **To check in a file, saving your changes**
 - Click the file you want in Visual SourceSafe Explorer, and on the SourceSafe menu, click Check In.
 - **To check in a file, undoing your changes**
 - Click the file you want in Visual SourceSafe Explorer, and on the SourceSafe menu, click Undo Check Out.
-

Warning If you choose the Undo Check Out command, you will lose any changes you have made to the local copy of your file(s).

See Also

[Check In Command](#)

[Check Out Command](#)

[Get Latest Version Command](#)

[Undo Check Out Command](#)

[View File Command](#)

Adding Files to the Visual SourceSafe Database

See Also

To add a file to Visual SourceSafe, you specify a file, its drive, and its folder, and that file is copied into the current Visual SourceSafe **project**.

– **To add files to the Visual SourceSafe Database**

- 1 On the File menu, click Add Files to display the Add Files dialog box.
- 2 Select the drive from which you want to copy the file. Use the Drives list box to display a list of available drives.
- 3 In the Folders list box, select the folder from which to copy the file.
- 4 In the List Files of Type list box, select a file type. This is an optional step used to filter the list down to a more manageable size.
- 5 In the File Name list box, select a file and click Add Files to add it to Visual SourceSafe.

Note Files already in the current project are not displayed in the File Name list box.

You can add multiple files at once by holding down the CTRL key, clicking all the names you want, and then clicking Add. To select all displayed files, press the HOME key to move to the beginning of the list, then SHIFT+END to move to the end.

1. Visual SourceSafe displays a dialog box in which you can enter a comment. Select the Apply Same Comment For All check box to have this comment applied to all the files being added; otherwise the dialog box is repeated for each file you have selected.
2. Click OK to add the files with a comment or click Cancel to cancel the Add Files operation. Visual SourceSafe returns to the Add File dialog box. Click Close.

– **To add an entire folder to Visual SourceSafe**

- 1 Choose a folder from the Folders list box and click the Add button.
- 2 Visual SourceSafe displays a dialog box in which you can enter a comment to describe the files or folder you are adding.
You can recursively add subfolders and the files they include by clicking the Recursive check box.
- 3 Click OK to add the folder with a comment or click Cancel to add the folder without a comment. Visual SourceSafe returns to the Add File dialog box. Click Close.

– **To add files with a drag and drop operation**

- 1 Open and display both **Visual SourceSafe Explorer** and Windows 95 Explorer.
- 2 Select one or more files or folders from the file list in Windows 95 Explorer, and **drag** the selected files or folders on the Visual SourceSafe Explorer.
Files are added to the current Visual SourceSafe project. Folders are added as **subprojects** of the current project.

Note You can only drag files or folders from the right side of Windows Explorer.

See Also

Add Files Command

Tracking and Labeling Versions of Files and Projects

See Also

Visual SourceSafe keeps track of every change you make to your files and **projects**. Both file-level changes—lines that you add, delete, or change in a particular file

- and project-level changes
- files and **subprojects** that you add, delete, or rename
- are tracked and stored in the **Visual SourceSafe database** as history logs. This is what gives you the ability to retrieve any version of a file or project.

– **To retrieve a version of a file or project**

- 1 Click the file or project you want in **Visual SourceSafe Explorer**, and on the Tools menu, click Show History.
- 2 In the History of File or History of Project dialog box, click the version you want.
- 3 Click the command button, such as View or Get, that tells Visual SourceSafe what you want to do.
- 4 Click Close to close the dialog box.

– **To label a file or project**

- 1 Click the file or project you want in **Visual SourceSafe Explorer**, and on the File menu, click Label.
- 2 In the Label box, type a user-defined label; optionally, in the Comment box, type a comment.
- 3 Click OK to close the dialog box.

Understanding Versions

Visual SourceSafe utilizes two types of version numbers. Both are useful in different situations.

The first is the internal version number, which is maintained by Visual SourceSafe. Visual SourceSafe gives every version of each file and project a version number, and displays it in the History of File or History of Project dialog box. The version number may occasionally skip a number, but is always a whole number.

Far more useful, however, are user-defined labels. You can associate a label with any version of any file or project. A label can be a string of up to 31 characters. Any of the following are valid labels: 1.0, 2.01b, Final Beta, and Approved for QA. Once you apply these labels (using the Label command), you can recall file versions at this point from the Show History dialog.

Labeling Versions

As you become experienced with Visual SourceSafe, you may find that at a project-level, you refer to user-defined labels, and not internal version numbers. On the other hand, it is rare to label individual files.

Some users, especially those moving from other source code control programs, wonder if you can add multiple labels to one item. In most of these other source code control systems, you label files. If the file SOURCE.C has not changed at all between version 2.00 and 3.00, that means the same version of the file is in both 2.00 and 3.00, and you have to label that version twice. If you can't label the same version twice, there's no way to indicate that this file is in both releases, which is a terrible flaw in those systems!

There are two reasons this reasoning doesn't apply to Visual SourceSafe:

- In SourceSafe, you can click on a file, and label it 2.00; and then click on the file again, and label it 3.00; and then click on the file again, and label it 4.00; and so on. Conceptually, you really are giving multiple labels to the same version of the file, because the file is not changing. Only in a very technical sense are you not, since every label actually creates a new version in SourceSafe.
- In SourceSafe, you usually label projects, not files. If you like, you can label a project with the label 2.00; and then later label the same project 3.00. Even if the files in the project haven't

changed between the two labels, everything works correctly.

Using Labels to Patch Code

In some situations, you may have a labeled version of your software that you want to patch with some new file versions. You can share and branch by label, but then you've got a whole new code base. Instead, you may want to use the Show History command on that project, click on the desired label, and share that to a new project. You will get shared and pinned project files of just the versions you want. Then, to apply the patch, either Unpin the necessary files, or separate them altogether.

Of course, this requires you to religiously use labels on all important milestones in your project.

See Also

[History of File Dialog Box](#)

[History of Project Dialog Box](#)

[Label Command](#)

[Show History Command](#)

Understanding the Working Folder

See Also

The working folder is where you work on a **project**. It is associated with the project, not the files within the project. When you check out or get a file, Visual SourceSafe copies it from your **current project** into your working folder for that project. After you make changes to the file and check it in, Visual SourceSafe copies it from your working folder back into the current project.

By default, Visual SourceSafe assumes that every file in the associated project is in your working folder. (Use the Show Differences (Project) command to see how the contents of your working folder differ from those of the current project.) Visual SourceSafe also assumes that the files you have checked out are writable and that the others are **read-only**. Accordingly, when you get multiple files, SourceSafe does not replace any writable file, but does replace read-only files if they're not up-to-date. If you want to change this behavior, you can change how Visual SourceSafe treats the read-only flag.

Note Each person working on a project should have his or her own working folder. In addition, each person should have a separate working folder for each project he or she works on.

On the other hand, it is acceptable for many users to have working folders of the same name, but on different disk drives or network paths.

Setting the Working Folder

After creating a new project, it is important for you to designate a working folder in which to work on the project's files. Visual SourceSafe prompts you to do this; if you do not, you may not be able to perform some commands.

Note When you set a working folder for a project, that setting is inherited by all **subprojects**, unless you explicitly override the setting for particular subprojects.

You can use an existing folder on your computer's hard disk as your working folder. You can also create a new folder from within Visual SourceSafe or Microsoft Windows.

Your working folder is displayed above the **file pane** of **Visual SourceSafe Explorer**, for example, C:\WORKDIR\MYPROJ.

– **To set or change the working folder**

- 1 Click a project in Visual SourceSafe Explorer, and on the File menu, click Set Working Folder.
- 2 Type a path and name in the Name box, or use the Folders and Drives boxes to find the folder you want.

You can also type a folder name that does not exist, and then click Create Folder to create it.

- 3 Click OK to set the folder.

–Or

Click the Save As Default For Project check box, and then click OK to save the folder as your default working folder for the project. If this check box is checked, you are making a permanent working folder change for this project. If it is not checked, the working folder change is for this session only, and the change is lost when you exit SourceSafe.

The Check Out Folder

Visual SourceSafe tracks the folders to which users check out files, and displays that information in the file pane of Visual SourceSafe Explorer as the check out folder. The check out folder is not always the same as your working folder for a project. When another user checks out a file, the check out folder shows that user's working folder for the file.

See Also

[Check In Command](#)

[Check Out Command](#)

[Get Latest Version Command](#)

[Modifying Files – Read-Only Flag](#)

[Setting the Working Folder in SS.INI](#)

Setting the Working Folder in SS.INI

See Also

When you save a working folder as the default with the Save As Default For Project check box in the Set Working Folder dialog box, Visual SourceSafe stores that information in your SS.INI file. The following is an example of what it might look like.

```
[$/Spelling/American]  
Dir (PC) = C:\DEV\SPELL\ENGLISH
```

The first line is a project header, indicating that variables below the header apply to the specified project. The second line uses the Dir initialization variable to specify your working folder. The platform identifier in parentheses indicates that the setting is machine-specific; if you were working on the same project on a Macintosh, you would want to specify a different working folder, or the syntax of your folder path.

It is possible to set these variables yourself in your SS.INI file, although generally it is easier to let Visual SourceSafe set them for you.

Note The **Visual SourceSafe administrator** may find setting the Dir variable useful for setting defaults. For instance, the administrator could place the two example lines into the SRCSAFE.INI file, which applies to all users, and tell each user to create a C:\DEV\SPELL\ENGLISH working folder on the user's disk drive. Each user would then have an identical folder structure.

See Also

[Dir Initialization Variable](#)

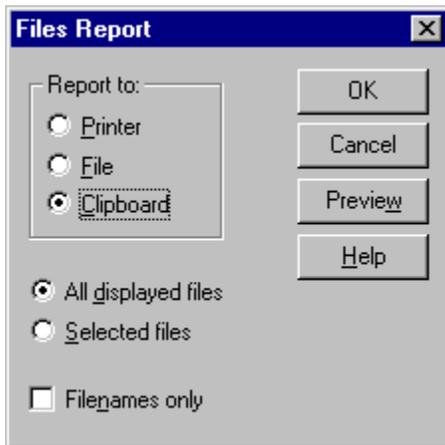
[Set Working Folder Command](#)

[Understanding the Working Folder](#)

Creating Reports

Visual SourceSafe offers a wide variety of reports. For example, you can create a report on:

- The history (Show History command) of a file or project.
- The differences (Show Difference command) in a file or project.
- The **file list** (Files Report command).



– To create a report

- 1 On the Tools menu, click Files Report, or in a dialog box that has the report option, click Report. For example, you can create reports from the Show Difference, Links, Paths, and Find In Files dialog boxes.
- 2 Click one of the following options:
 - **File**—Creates a text file of the contents. In the Report to File dialog box, set the folder, drive, and filename. Visual SourceSafe exports the report as ASCII text, with all formatting controlled by space and tab characters. If you select a file that already exists, you can append it to the end of that file, or replace it.
 - **Printer**—Sends a report to the printer. Set printer options in the Print dialog box.
 - **Clipboard**—Copies the report to the Clipboard. The report is saved as ASCII text, with all formatting controlled by space and tab characters.

– To preview a report

- Click Preview in the Report dialog box.

Common Questions About Visual SourceSafe

The following are some of the questions that people most commonly ask about Visual SourceSafe.

- Where is my file, really?
- Which project is the "home base" of a shared file?
- Why doesn't my initialization variable work?
- Why aren't files different after a check in?
- How do I change my Visual SourceSafe configuration?
- How do I convert my PVCS or Microsoft Delta database to Visual SourceSafe?

Where is my file, really?

See Also

After I add a file to Visual SourceSafe, where does the file go? Is the copy that I add the "real" copy, which Visual SourceSafe is simply pointing to? Or has it created a copy in another folder somewhere?

Answer

The **master copy** of the file does not exist in a folder at all. It exists in a **Visual SourceSafe database** in a **project**. The file you added still exists; but you can delete it from a folder if you like— Visual SourceSafe has the file available for you to work with at any time.

It is important to understand, moreover, that a project is not a folder *and* a folder is not a project. Rather, a project can be associated with folders, such as your **working folder** or a **shadow folder**.

See Also

Add Files Command

Understanding Visual SourceSafe Projects

Understanding the Working Folder

Which project is the "home base" of a shared file?

See Also

After you have **shared a file** among a number of **projects**, you may lose track of which project the file actually started in. Which project is the "home base" of the file; which project is it really in, and which is it shared to?

Answer

The file is in all the projects that share it: no project is "home base." Internally, the file simply exists in the **Visual SourceSafe database**, and every project that shares it has a pointer to the file.

See Also

Sharing Files

Why doesn't my initialization variable work?

Visual SourceSafe has a host of initialization variables and configuration settings you can use to control the way Visual SourceSafe works, and they are all accessible by editing a simple text file. However, it can be frustrating to place a variable into the file, and then discover that it does not seem to be working.

Answer

There are five common causes of this problem.

1 The variable was set while Visual SourceSafe was running.

Visual SourceSafe reads the initialization files when you open a Visual SourceSafe database. If you make changes to your SS.INI file while running Visual SourceSafe, no variable changes take effect until you close Visual SourceSafe and restart it.

2 The variable is under the wrong header.

The order of variables in the initialization file does not matter, but which header a variable is under matters a great deal. Most variables can be set in one of two places—either at the top of the file (before any header), in which case they apply to all projects, or under a project header to apply to a specific project. If you place a variable under a header, such as [Printing], to which it doesn't apply, Visual SourceSafe ignores it.

3 There are backslashes in a project header.

The project header in an initialization file must exactly match the Visual SourceSafe name of the project: for instance, \$/WORD/SPELLING. Even small changes, such as changing forward slashes to backslashes, cause Visual SourceSafe to ignore the variable.

4 You placed quotation marks around the value.

Unless a specific variable specifies otherwise, do not use quotation marks in any variable or value in Visual SourceSafe.

5 You used an operating system environment variable in the value, and tried to set or change it from the Tools Options dialog box..

SourceSafe's expansion of the % sign (commonly used to indicate an environment variable) works differently in the SourceSafe Explorer than it does when editing initialization files directly. If you want to use environment variables in your initialization files, edit the files manually.

Why aren't files different after a check in?

If I change a file, check it in to Visual SourceSafe, and then use the Show Differences command on that file, despite the changes, Visual SourceSafe displays the message, "The files are identical." Why doesn't Visual SourceSafe show me the changes I just made?

Answer

The key is understanding what the Show Differences command does. It compares Visual SourceSafe's most recent version of a file to the file in your **working folder**. If you just checked in the file, the two files are identical.

One way to work around this is to use the Show Differences command before checking in the file – seeing your changes may also help you formulate your check-in comment.

After check in, you can see your changes by comparing the current **master copy** of the file against the previous version of the file. To do this, choose the Show History command from the Tools menu. Then, select the previous version and click the Diff button.

How do I change my Visual SourceSafe configuration?

See Also

Visual SourceSafe can be configured in many ways.

Answer

The short answer is to see your Visual SourceSafe administrator. Also, see **Configuring Visual SourceSafe Explorer** for additional information.

You can also refer to the *Microsoft Visual SourceSafe User's Guide*. It contains information on setting up Visual SourceSafe. Chapter 8, "Administration," provides information on Visual SourceSafe database administration responsibilities.

For changes that affect just your own copy of Visual SourceSafe (and not other user's copies as well), you can:

- Choose the Customize Toolbar command from the Tools menu to change the Toolbar configuration.
- Choose the Font command from the Tools menu to change the font used in the SourceSafe Explorer.
- Choose the Change Password command from the Tools menu to change your logon password.
- Choose the Options command from the Tools menu to change numerous configuration variables in SourceSafe.

See Also

[Change Password Command](#)

[Configuring Visual SourceSafe Explorer](#)

[Customize Toolbar Command](#)

[Font Command](#)

[Installing Visual SourceSafe](#)

[Options Command](#)

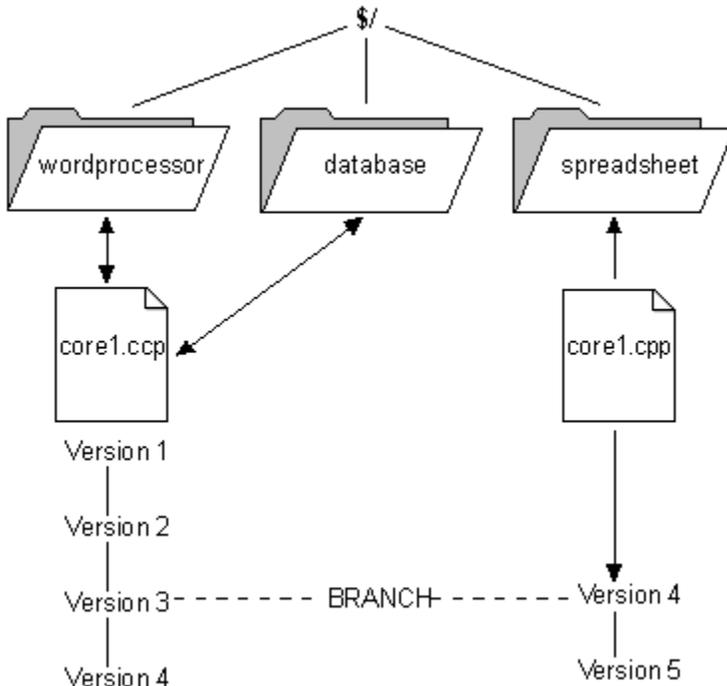
How do I convert my PVCS or Microsoft Delta database to Visual SourceSafe?

Answer

This is a task for your **Visual SourceSafe administrator**. During setup, Visual SourceSafe installs files named PVCS_SS.EXE and DELTA_SS.EXE to a folder named for the operating system you are working on (for example, ALPHA or WIN32). These programs perform the conversion from one database format to that of Visual SourceSafe. Help on these conversion utilities is available from Visual SourceSafe Administrator.

Branching Files

See Also Examples



Branching is the process of taking a file in two directions at once. The different directions are branches, or development paths, that the file or program has taken.

Visual SourceSafe uses project organization to track branches: each direction you take is represented by a project; the project names keep different development paths distinct.

Note In other (non-project-oriented) version control systems, branches are tracked through extended version numbers: for instance, version "2.3.9.2" is the second revision of the ninth branch off of version 2.3 of the file. In Visual SourceSafe, branches are tracked through named projects. The Paths dialog box displays the relationships among the many different development paths that a particular file has taken.

Visual SourceSafe also tracks **shared files**, that is, files used in common by many paths. Visual SourceSafe automates the maintenance of the different branches by making sure that changes to a file in one project are shared appropriately in others.

– To branch a file

- Click the shared file you want in **Visual SourceSafe Explorer**, and on the SourceSafe menu, click Branch.

Visual SourceSafe displays the Branch dialog box.

Basic Branching

Consider a file that is shared among multiple projects. If you change that file in any project, it changes in all projects. You can use the Links tab in the File Properties dialog box to see a list of all projects sharing a file.

Now, select that file in one project and branch it. This breaks the **share link**, making the file in that project independent of all other projects. The changes you make in the file are not reflected elsewhere, and vice versa. A branch has been created: two files (the file in the project, and its counterpart in other projects) have a shared **history** up to a certain point, and divergent histories after

that time.

After you branch a file, the Links tab does not show the link that has been broken. Instead, you use the Paths tab, also in the File Properties dialog box, to view the branch history of the file; the Paths tab includes a diagram that shows both current links, and former links that have been branched.

As an example, suppose you are working on a program called `$/Word` for two different clients. The two programs are virtually identical, but one or two files differ to tailor the program's behavior.

To begin with, you would share the project `$/Word` to a new branch project, say `$/Client2`. (If `$/Word` has subprojects, you could recursively share them.) You now have two project trees, `$/Word` and `$/Client2`, which are identical in every way. And because all their files are shared, all changes will apply equally to both projects, so they will stay identical through time.

Now, you select a few files in `$/Client2` and branch them. When you modify these specific files in `$/Client2`, your changes do not propagate to `$/Word`. All the other files are still shared between the two projects.

The Branch command is quick and easy to use, and a powerful management tool. As you add features to `$/Word`, most of your changes automatically propagate to `$/Client2`, so you only have to make changes once., (This becomes increasingly important as you add clients 3, 4, and 5!) But at any time, you can get `$/Word` to build your baseline product, or `$/Client2` to build the customized version. The only caveat is that if you add a file to one project, you will need to share it to the other(s). If you delete a file in one project, you will need to delete it from the other as well.

Note that this exact technique can also be used when you are supporting one program under multiple platforms, or in any other case where the two versions should follow each other through time, but retain different code in specified ways.

Example 1 – Customized Versions of a Program

Suppose you are working on two programs for two clients. The two programs are virtually identical, but you want to tailor the programs' behavior in one or two files.

To begin, you share one project, `$/WORD`, to a new branch project, `$/CLIENT2`. (If `$/WORD` has subprojects, you recursively share them as well.) You now have two projects that are identical in every way.

Next, you select the few files you want to tailor in `$/CLIENT2` and branch them. Now when you modify these specific files in `$/CLIENT2`, your changes do not propagate to `$/WORD`. All the other files, however, are still shared between the two projects.

Branching is quick and easy, and powerful. But keep in mind that if you subsequently add a file to one project, you have to share it into the other; if you delete a file from one project, you have to delete it in the other as well.

Note This technique can also be used when you are supporting one program under multiple platforms, or in any other case where two versions follow each other through time, but retain different code in specified ways.

Example 2 – Parallel Versions of the Same Program

Suppose after your project reaches version 3.1, development proceeds in two different directions. One team is working toward the next major release, the 4.0 version. The other team is working on a maintenance release, perhaps version 3.2.

You create a new project, labeled `$/PATCH`, to represent the 3.2 version. You share all the files into it. However, you do not want the projects to track each other, because versions 3.2 and 4.0 are going in different directions. Therefore, you use the Branch command at the same time that you share the projects. Accordingly, changes made in one are not automatically reflected in the other.

The projects now proceed in different directions, independent of each other. Later, you may want to merge the version 3.2 changes—bug fixes or minor feature enhancements, for example

—into version 4.0.

Example 3 – Branching After Release

Suppose you have a project that you have never branched. You labeled it version 3.1 for release. Now, you are working towards the 4.0 version, and a customer finds a bug in version 3.1.

Create a branch project from version 3.1 of the project, which is not a separate project, but an earlier moment in the life of the **current project**. In this scenario, you want to share and branch version 3.1 into a new project. Use the Share command for this, and check the Branch After Share check box in the Share With Project dialog box.

See Also

[Branch Command](#)

[Label Command](#)

[Links Properties Tab](#)

[Merge Branches Command](#)

[Merging Files](#)

[Paths Properties Tab](#)

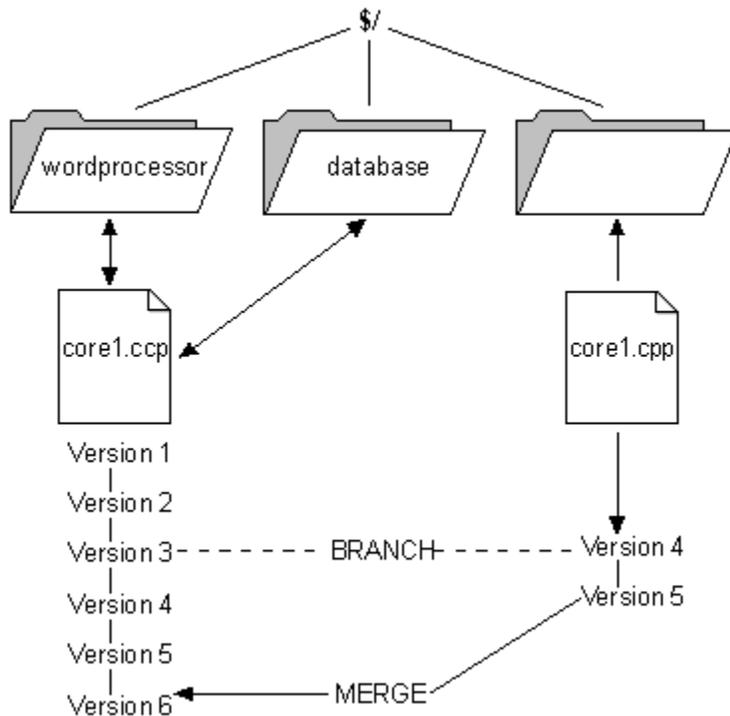
[Share Command](#)

[Sharing Files](#)

[Tracking and Labeling Versions of Files and Projects](#)

Merging Files

See Also



Merging is the process of bringing different development paths of a file back together.

In Visual SourceSafe, a merge can occur in two different circumstances. The first is when you enable **multiple checkouts**. In this case, the first user simply checks in the file. Subsequent users also check in the file, but their changes are merged with all other users' changes, and Visual SourceSafe retains all the modifications.

After a multiple checkout, there are two possible times that changes can be merged in – at Check In, or (if you have the Replace Writable Files option in the Local Files Options tab set to Merge) on a Get Latest Version command. In this case, differences between your local copy of a file and the version saved in the SourceSafe database are merged when you get the latest version.

The second circumstance is after **branching**. In this case, you merge the file branches back into one of the branched projects; this takes the changes made along one line of development, and implements those changes in the other line.

Under any circumstance, what happens is the same. Visual SourceSafe takes the two files with differences and creates a third, resultant file containing all the changes. For example, if you check in a file that causes a merge, the file on your hard drive and the file in the SourceSafe project are combined into a third file containing both sets of changes. Because there are three files involved, this operation is sometimes referred to as a three-way merge.

– **To merge branches**

- 1 Click the file or project you want in **Visual SourceSafe Explorer**, and on the SourceSafe Menu, click Merge Branches.

Visual SourceSafe displays the Merge dialog box.

- 2 In the Project box, click the project that has the version of the file from which you want to merge changes.
- 3 Click Merge.

After a Merge

After a merge, the file in your **working folder** is changed. If the merge was successful, that file represents the complete merge. If there was a problem, Visual SourceSafe opens the Visual Merge window (unless you have opted for the manual merge). The Visual Merge window gives you a point-and-click way of choosing how to resolve each conflict, at the end of which you can save the completely merged file. Visual Merge is the recommended way to resolve your conflicts. For those used to earlier SourceSafe versions, you can choose the manual SourceSafe merge to place identifying markers (six consecutive equal signs) in your file. In any text editor, you can search for and edit them manually.

In addition, after completing a merge, Visual SourceSafe observes the following rules.

- If there were **conflicts**, Visual SourceSafe keeps the file checked out. You must resolve the conflicts and check in the new file to Visual SourceSafe.
- If you use the Merge Branches command to merge a file into a project where you already have the file checked out, you still have the file checked out.
- In all other circumstances, Visual SourceSafe gives you the option of either updating the file automatically after the merge, or leaving the file checked out to your working folder so that you can inspect it before updating.

When the conflicts are resolved, check the file in again. SourceSafe re-checks for any new conflicts, and if all conflicts are resolved, checks in the file.

Problems That Can Occur While Checking In a File

You may have to check in a file more than once. This situation can occur when you:

- Check in a file, Visual SourceSafe merges the changes, and you have to resolve conflicts or make other changes.
- Are resolving the first round of merge conflicts, when someone else checks in the same file. When you check in the file again, another merge takes place, possibly with new conflicts.

If you have a problem when resolving a conflict, and want to start over, look for a file in your working folder with the extension .ORG. This file contains your original (pre-merged) file. Copy the .ORG file over the original filename, and then check in the file. Visual SourceSafe asks if you have successfully resolved the conflict. Answer No, and Visual SourceSafe offers to remerge the file so that you can start over and try again.

If there is a power outage in the middle of a merge, you may find an .MRG file in your working folder. This indicates that Visual SourceSafe did not complete the merge; your original filename is still the original file, and the .MRG file is the (perhaps incomplete) merged file. Delete the .MRG and begin the merge again.

See Also

[Branching Files](#)

[Merge Branches Command](#)

[The Visual Merge](#)

[Merging Options](#)

[Resolving Merge Conflicts](#)

[Using Multiple Checkouts on a File](#)

Resolving Merge Conflicts

See Also

In most merge situations, different changes are made to two copies of a file; the merge creates a new file that contains all the modifications made in both copies. However, when different modifications are made to the same line in both copies of the file, there is no way for Visual SourceSafe to determine which set of changes you want. It therefore reports a conflict and requires you to resolve it.

Conflicts can occur during a check in, get, or merge branches operation. If Visual SourceSafe finds a conflict, it informs you that you have to edit the file and resolve the conflict before updating or checking in the file. You have two methods to choose from for resolving the conflicts.

The first method, Visual Merge, lets you quickly identify and correct any conflicts without having to start a separate editor. Visual Merge is the recommended way to resolve your conflicts. You can set Visual Merge as the default on the General tab of the Tools Options dialog box.

For those used to earlier SourceSafe versions, you can choose the standard SourceSafe merge to place identifying markers (six consecutive equal signs) in your file. (Visual Merge doesn't use these conflict markers.) When you edit the file in your text editor, search for these six equal signs (====) as the sign of a Visual SourceSafe conflict report. After you resolve the conflict, search again—there may be more than one. Only after all conflicts have been successfully resolved should you check the file into Visual SourceSafe.

With SourceSafe's manual merge, a conflict marker might look like the following.

```
<<<<<< SourceSafe version
if (a <= 5)
=====
if (a < 6)
>>>>>> Local version
```

A set of six left-angle brackets (<) precedes the Visual SourceSafe **master copy** version, the version you are merging into. Next are one or more lines from that copy. These are followed by six equal signs, which in turn are followed by the conflicting lines from the **local copy**. Finally, six right-angle brackets (>) indicate the end of the conflict. In short, whatever is between the angle brackets is in conflict; whatever is outside those boundaries is not in conflict.

If you are using the Merge Branches command, there is no SourceSafe or local version in the conflict marker. Instead, there are two **branched files** in two branched projects. The format looks the same, except that "SourceSafe version" and "Local version" are replaced by the project names that each variation took place in.

In resolving the conflict, remember to delete the conflict markers. In addition, remember to delete one of the two sets of changes, or create a new version that encompasses both.

See Also

[The Visual Merge](#)

[Merge Command](#)

[Merging Files](#)

[Merging Options](#)

Merging Options

If you choose not to use the visual merge, and want to customize how Visual SourceSafe displays **merge conflicts** in the manual merge, one way is to set the Mark_Merges initialization variable in your SS.INI file. Using the visual merge means you don't have to worry about **conflict markers**. But, if you use Mark_Merges, Visual SourceSafe:

- Marks all changes, even if they are not conflicts. This gives you the opportunity to see and approve all changes. Of course, you should remove all the change markers before checking in the file or compiling the code.
- Shows you the original lines as well as both modified lines.

The following is an example of what the Visual SourceSafe conflict markers may look like.

```
<<<<<< SourceSafe version
if (a <= 5)
| | | | | Original version
if (a < 5)
=====
if (a < 6)
>>>>>> Local version
```

The format is similar to the manual merge conflict marker, except that Visual SourceSafe displays the original (pre-checkout version) between the **master** and **local** versions, which may be helpful to you in determining how best to clean up the file.

Note When Visual SourceSafe marks a non-conflicting change, the local or the SourceSafe master version is not displayed in the conflict marker. Therefore, neither the right nor left angle brackets are present. This is why the equal signs, which are always present, are used in searching for changes.

Setting Visual Merge Colors

If you do decide to use visual merge, you can customize the colors used to mark additions, deletions, and changes in the Visual Merge dialog box by setting the options on the Difference tab of the Tool Options Dialog box.

Setting Merge Options in the Initialization Files

You can set the following two initialization variables to change the file extensions used by the merge process.

```
Backup_Ext = .ORG
Merge_Ext = .MRG
```

Backup_Ext controls what happens to your original (pre-merged) file after the merge has taken place. By default, Visual SourceSafe deletes the file if there was no conflict; it renames the file .ORG if there was a conflict. When you set this variable, however, Visual SourceSafe always renames the file, so you always have a copy of your original. You can set this to the default .ORG extension, or to a different extension if .ORG is already used by your system.

Merge_Ext controls the extension that Visual SourceSafe uses for its temporary merge file. This file is always deleted after the merge process is completed—the only reason to change it is if the extension .MRG is already used by your system.

See Also

[Mark_Merges Initialization Variable](#)

[Backup_Ext Initialization Variable](#)

[Merge Command](#)

[Merge_Ext Initialization Variable](#)

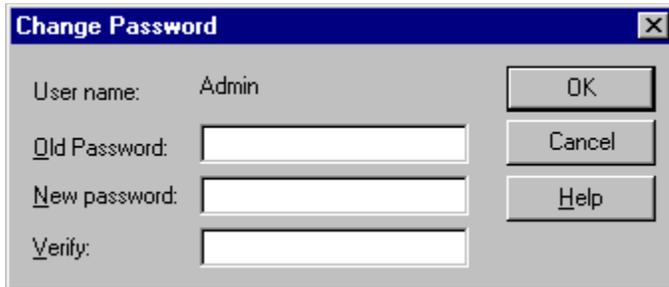
[Merging Files](#)

[Resolving Merge Conflicts](#)

[The Visual Merge](#)

Changing Your Visual SourceSafe Password

See Also



The screenshot shows a standard Windows-style dialog box titled "Change Password". The dialog has a blue title bar with a close button (X) in the top right corner. The main area is light gray and contains four input fields on the left and three buttons on the right. The first input field is labeled "User name:" and contains the text "Admin". The second input field is labeled "Old Password:". The third input field is labeled "New password:". The fourth input field is labeled "Verify:". The buttons on the right are "OK", "Cancel", and "Help".

To change your password

- 1 On the Tools menu, click Change Password.
- 2 Type your old password in the Old Password box.
- 3 Type your new password in the New Password box. As you type, the characters are displayed as asterisks.
Note Your password can be up to 15 characters long.
- 4 Retype the new password in the Verify box and then click OK. This verification makes sure you typed the new password correctly the first time.

See Also

[Change Password Command](#)

[Visual SourceSafe Naming Syntax and Conventions](#)

Keyword Expansion

See Also

Keyword expansion refers to Visual SourceSafe's ability to place certain information directly into your file to create a file header for you.

To do this, you place certain keywords, in comments so that they do not affect your code, into the text of your file. When you add or check in the file, Visual SourceSafe looks for these keywords, and places the relevant information after them.

This topic describes the traditional way Visual SourceSafe expands keywords. With version 5.0, you now can also expand keywords in a different way that makes [using keywords with HTML files](#) much more useful.

SourceSafe Keywords

The following table is a list of all the Visual SourceSafe keywords. Some of the keywords are redundant, either for convenience or for backward compatibility with other version-control systems, such as RCS. Keywords are case-sensitive, so be sure to capitalize them properly:

Keyword	Description
\$Archive: \$	Visual SourceSafe archive file location
\$Author: \$	User who last changed the file
\$Date: \$	Date and time of last check in
\$Header: \$	Logfile, Revision, Date, Author
\$History: \$	File history, Visual SourceSafe format
\$Log: \$	File history, RCS format
\$Logfile: \$	Same as Archive
\$Modtime: \$	Date and time of last modification
\$Revision: \$	Visual SourceSafe version number
\$Workfile: \$	Filename
\$NoKeywords: \$	No keyword expansion for all keywords that follow. (The colon is optional.)
\$JustDate: \$	Date, without the time addendum.

A SourceSafe keyword has the following syntax:

```
$keyword: $
```

For example, you could place the following in a file:

```
$Revision: $
```

Upon check in, Visual SourceSafe could replace it with

```
$Revision: 23 $
```

The next time you check in the file, the 23 is replaced by a 24, and so on. Without running Visual SourceSafe, you can always look at the top of your file to see, for example, what version your copy of the file is.

Enabling Keyword Expansion

Because keyword expansion requires Visual SourceSafe to scan each file for keywords, it can considerably slow the Check In and Add Files commands. For this reason, keyword expansion is by default disabled for all files. Your Visual SourceSafe administrator must indicate which files Visual SourceSafe should scan for keywords in the Administrator program.

– **To make keyword expansion work, use either of the methods described here:**

- 1 In the Visual SourceSafe Administrator, point to Tools and click Options and in the "Expand keywords in files of type" option, enter the file types for the keyword expansion. For example, type *.txt.

- or -

Add the following two lines to the SRCSAFE.INI file on the server:

```
Keyword_Masks = *.TXT  
Expand_Keywords_Locally = YES
```

You could add them to a user's SS.INI file, but it is better to put them in the SRCSAFE.INI file on the server so that keywords are expanded when a user updates or checks in a file.

Note The file may not have updated keywords if the user chooses to keep the file checked out on the update (that is, Keep_Checkedout=YES). Also, the "Keyword_Masks =" line can have any extension or combination of extensions. For example:

```
Keyword_Masks = *.C, *.CCP, *.H, *.HPP, *.BAS
```

- 2 Add keywords to files. Keywords are case sensitive.
- 3 Format the keywords correctly. There are two ways to format keywords in files. The most frequently used method is to align the keywords on the left margin:

```
$Archive: $  
$Author: $  
$Date: $
```

When expanded, it will look like this:

```
$Archive: /KeywordAlignTest/KeyAlign1.txt $  
$Author: Guest $  
$Date: 11/23/95 9:31a $
```

The other way is to format the keywords so that when they are expanded, the values are aligned in a column. Note the use of two colons and the amount of space between the colons and the second dollar sign.

```
$Archive::                               $  
$Author::                                $  
$Date::                                  $
```

When it is expanded, it looks like this:

```
$Archive:: /KeywordAlignTest/KeyAlign1.txt    $  
$Author:: Guest                               $  
$Date:: 11/11/95 5:57p                        $
```

If you don't leave enough space between the colons and the dollar sign, SourceSafe will truncate the value so it would look like this:

```
$Archive:: /KeywordAli$  
$Author:: Guest          $  
$Date:: 11/11/95 5:$
```

- 4 Place comment characters in front of keywords so the keywords are not compiled. SourceSafe sets the following comment characters by default.

```
*.ASM = ";" "  
*.BAS = "rem "  
*.BAT = "rem "  
*.S = ";" "  
<all other file types> = "*" "
```

If you want to use a different comment character in a file that has a specific file extension, add the following lines to the Srcsafe.ini file on the server:

```
[Keyword Comments]
```

```
*.C = "/*", "*/"
```

The comment character is especially important in dealing with \$Log: \$ and \$History: \$ keywords. Unlike the other keywords, \$Log: \$ and \$History: \$ keywords add more information to the file each time keywords are updated – instead of just replacing the previous information with the current information. A comment character, as defined by the SRCSAFE.INI, is added to the beginning of each line created by \$Log: \$ and \$History: \$ keywords. For example, if the following is in a .txt file:

```
*$Log: $  
*$History: $
```

and the default comment characters are in effect, it would look like this after keyword expansion took place.

```
*$Log: /KeywordAlignTest/KeyAlign1.txt $  
*  
* 3    11/23/95 10:47a Guest  
*  
* 2    11/23/95 10:45a Guest  
*  
* 1    11/11/95 5:57p Guest  
*$History: KeyAlign1.txt $  
*  
* ***** Version 3 *****  
* User: Guest      Date: 11/23/95   Time: 10:47a  
* Updated in $/KeywordAlignTest  
*  
* ***** Version 2 *****  
* User: Guest      Date: 11/23/95   Time: 10:45a  
* Updated in $/KeywordAlignTest  
*  
* ***** Version 1 *****  
* User: Guest      Date: 11/11/95   Time: 5:57p  
Updated in $/KeywordAlignTest
```

Effect of Keyword Expansion

When you check in or add a file with keyword expansion enabled, Visual SourceSafe places a modified copy of your file directly into the current project. It then immediately gets the file, which refreshes your **working folder** with the checked-in file.

The Get Latest Version command slows down the updating process. You can avoid it by not setting the Copy Keyword Expanded-Files Into Working Folder option on the Local Files tab of the Options dialog box. When you clear this option, Visual SourceSafe still expands keywords, and places the modified file into the project. However, it does not copy the file into your working folder.

See Also

Columnar Formatting and Binary Files

Formatting the History and Log Keywords

HTML File Keywords

Columnar Formatting and Binary Files

See Also

Under ordinary circumstances, Visual SourceSafe replaces all the space between a **keyword** and the next dollar sign with its information. This can make it difficult to create an attractive columnar display, as the value length may change each revision.

To fix this, you can place two colons instead of one after the keyword. This tells Visual SourceSafe that the dollar sign after the keyword is exactly where you want it: Visual SourceSafe either adds space, or truncates the value, to ensure that the columns stay intact.

In addition to creating attractive columns, this feature is very useful when you want to embed keyword information in binary files (such as executable files), where the exact length of the data is important.

For example:

```
$Revision::          $
```

SourceSafe expands the keyword with the version number of the file, and keeps the extra space you added inside the dollar sign character so that the revision number column maintains its width.

See Also

Keyword Expansion

Formatting the History and Log Keywords

See Also

The History and Log keywords instruct Visual SourceSafe to put history information after the keyword. Visual SourceSafe places the filename between the colon and the dollar sign; in the lines following the keyword, it places the latest entry in the file's history.

To make sure the history is read as comments (to keep code compilable), your **Visual SourceSafe administrator** can define comment characters that Visual SourceSafe places before and after the History and Log lines.

Each file extension is associated with a string that is placed before every line, followed, optionally, by a string to be placed after every line. Both strings are in quotes, separated by commas. If no keyword comments are specified, the following are assumed.

```
;Assumed keyword comments
*.ASM = "; "
*.BAS = "rem "
*.BAT = "rem "
*.S = "; "
<All others> = "* "
```

Suppose your administrator does not have a setting for *.C in your SRCSAFE.INI file. Visual SourceSafe uses its default, which is an asterisk. You could then put the following into the header of your SS.INI file:

```
/*
 * $History: $
 */
```

Expanding this, Visual SourceSafe would place an asterisk before each line, and nothing after. The resulting file, after you add it to Visual SourceSafe, might look like the following.

```
/*
 * $History: IDRANGE.TXT $
 *
 * ***** Version 1 *****
 * User:mariaj Date:8/01/92 Time:11:02a
 * Created
 * Comment: Defined ranges for IDs.
 */
```

See Also

Keyword Expansion

Setting the File Modification Date/Time

When you get or check out a file from Visual SourceSafe into your **working folder**, Visual SourceSafe sets the modification date that the operating system tracks for each file.

Visual SourceSafe uses the following rules to determine the date/time it gives your file.

- If you already have a **local copy** of the file and it's identical to the file you are retrieving, Visual SourceSafe does nothing—the file's time stamp remains unchanged.
- If Visual SourceSafe does retrieve the file (you don't already have it or your copy is not up-to-date), it gives the file the current date/time, the time of the retrieval.

Why Visual SourceSafe Works This Way

The modification date/time is used by many compilers and make systems as a signal of whether a file has changed and therefore has to be recompiled. Visual SourceSafe's treatment of date/time is designed to optimize this process, ensuring that you compile only when you need to.

Consider the following example. You have a file HELP.CPP in Visual SourceSafe and in the working folder. Visual C++ turns this code file into the object module HELP.OBJ. At any given time during the development cycle, both files — HELP.CPP and HELP.OBJ

— exist in the same folder. Each time you build your project, Visual C++ compares the date/time stamp of HELP.CPP with HELP.OBJ to determine if it should compile, using the following rules.

- If HELP.CPP is more recent than HELP.OBJ, the code has changed, and a compile is necessary.
- If HELP.OBJ is more recent than HELP.CPP, the code has already been compiled, and no recompile is necessary.

Now consider the ramifications for Visual SourceSafe. You compile HELP.CPP into HELP.OBJ, giving the latter a new date/time. If you build your project again without changing anything, HELP.CPP is not recompiled and you check in HELP.CPP to Visual SourceSafe.

The next day, you get HELP.CPP again. If HELP.CPP has not changed, Visual SourceSafe recognizes that the **master copy** and your local copy are identical. It does not get the file, and it does not change the file date/time.

If another developer has changed HELP.CPP, Visual SourceSafe recognizes that the file is different. It gets HELP.C, and makes its date/time the current time. When you build your project again, the file is recompiled.

Changing the Date/Time Stamp

There are two ways to modify Visual SourceSafe's file modification time stamp.

In the first, you set the Set Date/Time On Local Files option on the Local Files tab in the Options dialog box (Tools menu). This option has three possible values:

- Current—The default; uses the date/time of retrieval.
- Check In—Uses the date/time of the most recent check-in to Visual SourceSafe.
- Modification—Uses the date/time of the file's most recent modification.

In the second, for the Check In, Check Out, or Get Latest Version command, you specify the date/time in the dialog box that Visual SourceSafe displays when you check in, check out, or get a file or project.

Modifying Files–Read-Only Flag

When a file is not checked out, Visual SourceSafe tags the file as one that can't be modified. The exact method of preventing change depends on the operating system: an MS-DOS or Windows file is made read-only, and a Macintosh file is locked or uses the CKID Resource.

Why Visual SourceSafe Makes Files Read-Only

Visual SourceSafe does not make files read-only for security purposes. Rather, it makes your **local copy** read-only:

- To remind you that the file is not checked out.
- To indicate to Visual SourceSafe whether you have been modifying a file. When a file is read-only, Visual SourceSafe assumes that you have not modified the file. Therefore, when you get the file, Visual SourceSafe replaces your local copy with the more up-to-date **master copy**. But a writable file may be one you have modified, and may therefore contain information that exists nowhere else. By default, Visual SourceSafe never replaces such a file.

Important Although you can manually make a read-only file writable, Visual SourceSafe does not let you check that file in if you haven't first checked it out.

Changing Visual SourceSafe's Use of Read-Only Files

There are two ways to change Visual SourceSafe's default behavior.

- Clear the Use Read-Only Flag For Files That Are Not Checked Out option on the Local Files tab in the Options dialog box (Tools menu). If you do so, Visual SourceSafe never tags files read-only, and therefore can't use the read-only attribute as its signal that a file should not be replaced. (Visual SourceSafe uses the file's actual check out status in this case.)
- Change Visual SourceSafe's use of the read-only flag for a Get Latest Version, Add Files, or Check In operation. You set the Make Writable check box in the Command Options dialog box to make a file writable.

Using Shadow Folders

A shadow folder is a centralized folder on your network server that contains all the files in a project. Shadow folders must be set up by your **Visual SourceSafe administrator**. The shadow folder does not contain the **master copy** of the files (the master copy of every file is in the **Visual SourceSafe database**), and it does not contain the **local copy** of the files in your **working folder**. It contains a copy of the most recently checked-in version of each file in the project. Therefore, it serves as a centralized area to view and compile your source code.

A shadow folder is completely optional. It is generally used in one of two situations.

- To allow users to view but not modify files. There may be users who want to see the files, but who do not have access to Visual SourceSafe.
- To prevent your working folder from keeping its own compilable copy of the project. To make sure that every user has the most recent version of files, it may be desirable for all users to compile source code out of a centralized folder, instead of in personal working folders. In this case, shadow folders are frequently used in conjunction with the Remove local copy after Add or Check In option on the Local Files tab in the Options dialog box (Tools menu).

Note: Shadowed folders do not track changes to their subprojects. For example, if you have shadowed project \$/A which contains two subprojects \$/A/1 and \$/A/2, and you rename \$/A/2 to \$/A/B, the name change is not shadowed. Project level changes need to be tracked manually, or you can use Reconcile All from the Project Difference dialog box occasionally to keep things up to date.

You can also use the Deploy command to write a set of files to a server location (but the project containing the files must be designated a web project first!).

Setting the File Type: Text and Binary Files

See Also

You can use Visual SourceSafe to store many types of files: source code, word processor documents, graphics, executable files, CAD drawings, dialog box and menu descriptions, and so on. Visual SourceSafe is similar to an operating system in this respect: it stores all files equally, without regard to the contents of any file.

Visual SourceSafe does make a distinction, however, between text files (files that contain only characters) and binary files (all others). (For most operations, you can treat text and binary files exactly the same—Visual SourceSafe uses its highly disk-efficient **reverse delta** storage on all files, text and binary.)

Consequently, it is important to understand how Visual SourceSafe determines the type of a file and how Visual SourceSafe treats these two.

How Visual SourceSafe Identifies Files

When you add a file to Visual SourceSafe, Visual SourceSafe automatically assigns it a type: text or binary. The default mechanism for creating this assignment is a simple test: Visual SourceSafe scans the file for NULL characters (bytes with value 0). If it finds such a character, Visual SourceSafe identifies the file as binary.

Although generally accurate, Visual SourceSafe's method may on occasion incorrectly assign the text type to a binary file. Therefore, Visual SourceSafe allows you to explicitly set the file type.

When you add a file, you can set the file type option: Auto-Detect, Binary, or Text. Auto-Detect is the default. After you add a file, it retains the type it was originally given, unless you explicitly change it.

Note Use the General tab in the Properties dialog box (File menu) to view and change the file type setting. You can also use the Binary Files box on the File Types tab in the Options dialog box (Tools menu) to set the file type.

How Visual SourceSafe Treats Binary Files

There are three significant differences regarding how Visual SourceSafe treats binary files:

- Internally, Visual SourceSafe's mechanisms for storing changes differs for text files (which have distinct lines as units of comparison) and binary files (which have no obvious line delimiter). That's why it is important for Visual SourceSafe to correctly identify the type of a file—the wrong type does not result in errors, but it makes Visual SourceSafe's algorithms much less efficient.
- Another area of difference is with respect to the Show Differences command. With a binary file, Visual SourceSafe stores each change as a small record of which bytes moved where: useful for reconstructing earlier versions, but not for display. Therefore, when you use the Show Differences command on a binary file, Visual SourceSafe can tell you if the file has changed; but it cannot display how the file has changed.
- The third difference occurs with end-of-line characters. With a text file, Visual SourceSafe automatically translates end-of-line characters between different operating systems; with a binary file, Visual SourceSafe never alters the contents of the file in any way (except for **keyword expansion**).

See Also

Add Files Command

File Types Options Tab

Show Differences (File) Command

Optimizing Visual SourceSafe Performance

Set the following variables in the SS.INI or SRCSAFE.INI file:

- Diff_Ignore (PC) = c-e-s-w-
Tells Visual SourceSafe not to ignore end-of-line differences, which speeds up the comparison. By default, Visual SourceSafe ignores end-of-line differences, which slows the algorithm significantly.
- CP_OnSelection = No
Disables the selection of a project as you use the mouse or arrow keys to move through the **project list**. If you set this initialization variable to No, you must use the ENTER key or double-click to set your **current project**.

Visual SourceSafe's default installation, while simple to create and use, is not necessarily optimized for performance. Therefore, Visual SourceSafe provides two ways that you can customize your system to improve performance:

- Copy as much as possible from the network to your disk drive. This change is extremely useful for all network users, but is essential for Macintosh users on PC networks. You can use the Visual SourceSafe NETSETUP.EXE program (on the Visual SourceSafe server) to copy the Visual SourceSafe executable files to your local computer.
- Fine-tune your initialization files for performance. Most fine-tunings, described below, involve compromises of various sorts, such as disabling certain kinds of functionality that involve time-consuming overhead. None of these optimizations provides as much benefit as the measures described above.

General Optimizations

You can set the following options on the Local Files tab in the Options dialog box (Tools menu):

- Copy Keyword-Expanded Files Into Working Folder
Useful if you use **keyword expansion**. Unchecking this check box may increase SourceSafe's speed, as it tells SourceSafe not to get a file after check in.
- Compare Files By
Controls how Visual SourceSafe determines if you already have the latest copy of a file in your **working folder**. The default, Checksum, is not as fast as the Time setting. The Time setting sometimes does not behave as you expect when you try to get earlier versions. A solution is to occasionally delete your **local copy** to force a Get operation to occur.
- Temp_Path
Specifies where you store temporary files. The default setting is to store temporary files for your users on the server, but setting this variable in each user's SS.INI file to a local folder can speed up operations like viewing files. For example:
Temp_Path = c:\windows\temp

Ask your Visual SourceSafe administrator to verify the following variable in the SRCSAFE.INI file:

- Lock_Mode = Native
This is the default setting for this SRCSAFE.INI variable. Setting this initialization variable to Native speeds up almost every Visual SourceSafe operation. (This variable must be set by the SourceSafe administrator.)

Features You Can Disable to Improve Performance

Your Visual SourceSafe administrator can disable any of the following to optimize the speed of many Visual SourceSafe commands.

- Shadow Folders
- Journal Files
- Project Security System

Macintosh-Specific Optimizations

Set the following variable in your SS.INI or SRCSAFE.INI file on the Macintosh:

- Text_Types = TEXT
Binary_Masks = *.*
Together, tells Visual SourceSafe to treat all files of type TEXT as text, and all other files as binary without scanning. By default, Visual SourceSafe scans the file contents, which is a slow operation.
- Compare = Time
Compare = CKID (on the Macintosh)
Controls how Visual SourceSafe determines if you already have the latest copy of a file in your **working folder**. The default, Contents, is the slowest and the most accurate. Checksum is faster. Time is the fastest. CKID offers performance and safety for Macintosh users.

The two Compare lines can be used together, and instruct Visual SourceSafe to use CKID on the Macintosh and Time on a personal computer. If you do use the CKID setting, you must also set the initialization variable Use_CKID = Yes.

The Time setting sometimes does not behave as you expect when you try to get earlier versions. A solution is to occasionally delete your **local copy** to force a Get operation to occur.

Writing Batch Files

See Also **Example**

You can use the Visual SourceSafe command line to write batch files and macros, so that many Visual SourceSafe commands can be executed at one time.

When you do so, however, you may also want to change the way the command line interacts with the user. You can disable input so that batch files can continue without stopping for user input, redirect program output to a file, and use Visual SourceSafe's exit code to test for successful batch file completion.

Disabling Input

There are three command-line options that can be used together to ensure that Visual SourceSafe does not prompt the user for input.

By default, Visual SourceSafe prompts you for a comment on many commands, such as Add and Check In. However, you can use the `-C` command-line option to avoid this prompt.

```
-c-      No comment.  
"-cHello"Use the string Hello as comment.  
-c@COMMENT.TXT Use contents of COMMENT.TXT as the comment.
```

In addition to prompting for comments, Visual SourceSafe frequently asks yes or no questions. You can use the `-I` option to avoid these questions.

```
-i-y     Automatically answer Yes to all Yes/No questions.  
-i-n     Automatically answer No to all Yes/No questions.
```

Finally, Visual SourceSafe may prompt for a logon name. Use the `-Y` command-line option to supply it.

```
-yMaryUsername is Mary.  
-yMary,ABCDEF Username is Mary, password is ABCDEF.
```

Redirecting Output

By default, Visual SourceSafe directs all output to the screen. You can use the `-O` command-line option to disable or redirect the output.

```
-o-No output  
-o@RESULTS.TXT Redirect all output into RESULTS.TXT.
```

If this file already exists, the output is appended to the end.

See Also

[-C Command-Line Option](#)

[-I Command Line-Option](#)

[-O Command-Line Option](#)

[-Y Command-Line Option](#)

[Command-Line Exit Codes](#)

[Using the Visual SourceSafe Command Line](#)

Sample Batch File

FoxPro Example

If you use the command line to check out HELP.C and HELP.H and one of the two files is already checked out, Visual SourceSafe checks out the other file. The following batch file prevents this: It first checks the status of all the specified files and only checks them out if none of them is checked out.

This example also demonstrates the use of -O- command-line option to disable output, -Y to specify a username, -C to specify a comment, and the exit code of the Status command to determine if files are checked out.

```
@echo off
ss stat -o- -yMaria %1 %2 %3 %4 %5 %6 %7 %8 %9
if errorlevel 100 goto BAD_FAILURE
if errorlevel 1 goto CHECKED_OUT

rem Exit code 0, no files are checked out.
echo No specified files are checked out: checking all out.
ss che "-cChecked out automatically" -yMaria %1 %2 %3 %4 %5 %6 %7 %8 %9
goto END

rem Exit code 1, something is checked out.
:CHECKED_OUT
echo One or more files are checked out: quitting without checking anything
out.
goto END

rem Exit code 100, there is an error.
:BAD_FAILURE
echo Visual SourceSafe could not run successfully

:END
```

Customizing the SS.INI and SRCSAFE.INI Files

See Also

Visual SourceSafe provides two initialization files that contain settings you can use to customize your Visual SourceSafe environment:

- SS.INI, for individual user variables
- SRCSAFE.INI, for the **Visual SourceSafe database** global variables

Each user has an SS.INI file that can be edited. Only the **Visual SourceSafe administrator** should edit SRCSAFE.INI.

Important Manually editing the initialization files is not recommended, unless you are very familiar with SourceSafe initialization variables and their behavior. Instead, you should set most initialization variables directly in Visual SourceSafe in the Options dialog box on the Tools menu. You can set the SRCSAFE.INI variables in the Visual SourceSafe Administrator Options dialog box. The only exception to this general policy is when you want to use environment variables as part of your initialization variable value. To do this, you must edit the initialization file manually.

When you use the Options dialog box, Visual SourceSafe saves your changes in memory, but does not write the changes to the SS.INI file until you exit Visual SourceSafe or open a different database. Therefore, if you start a second Visual SourceSafe session before exiting the first, your changes are not yet in effect.

You can read and modify initialization files with any text editor. There is a limit of 511 characters on a initialization file line, although this maximum may be restricted by specific variable requirements (such as project path limitations).

Usually, SS.INI is located in the USERS subfolder of the folder in which Visual SourceSafe is installed, for example, SS\USERS\USERNAME. You can move your SS.INI file from this location, providing you tell Visual SourceSafe of the change by editing the USERS.TXT file appropriately (see your Visual SourceSafe administrator).

The SRCSAFE.INI file is usually placed in the main SourceSafe folder on a server.

Note Initialization variables set in SS.INI override settings in SRCSAFE.INI.

The Format

There are three kinds of lines in SS.INI and SRCSAFE.INI.

- Initialization variables, which are the functional lines, take the following format:

```
Initialization variable = value
```

Boolean variables can be set either with "Yes" or "No", "True" or "False."

- Comments begin with a semicolon:

```
; Here I am, JP.
```

Visual SourceSafe ignores all comments.

- Headers are provided in brackets:

```
[$/Word]
```

Most headers specify particular project paths. Other headers, such as [Keyword Comments], are used for specific purposes.

The order of variables inside an initialization file does not matter at all; however, the header under which a variable is placed matters a great deal. If a variable is placed under the wrong header—for instance, if a general-purpose variable is placed under [Keyword Comments]

—it is ignored.

SS.INI on Multiple Platforms

If you are running a multi-platform Visual SourceSafe installation, you may want certain variables to have different values on the different platforms or on different computers. You can therefore place a platform or computer name in parentheses after the variable, to indicate that the variable applies only on that platform. For example:

```
Use_ReadOnly (WIN) = Yes
Comment_Editor (JOE1)= c:\MSOFFICE\WINWORD\WINWORD.EXE
```

The setting in the first example applies only under Microsoft Windows. The setting in the second example applies only to Joe's computer named JOE1. Other possible settings include Win, NT, MAC, and PC—the last is a catch-all for personal computer operating systems.

Every variable in SRCSAFE.INI can be followed by a platform specifier in parentheses. This specifier enables one variable to have different settings on different platforms. Not all variables recognize different computer names, but many do.

Different versions of Visual SourceSafe look for different specifiers. For example, the Visual SourceSafe interface for Windows NT looks for the following:

```
JOE1, NT, Win, PC, UI
```

This looks for a variable set for Joe's computer named JOE1; if it does not find that, it looks for Windows NT; then Win, then PC, then UI, and finally, for a variable with no platform specifier.

However, you can override this setting by setting the environment variable **SSINIENV**. For example:

```
Set SSINIENV = Ragiv, MS-DOS, PC, CL
```

Visual SourceSafe looks for variables specified for Ragiv before looking for MS-DOS, PC, CL, or no-specifier. Now you place the following two lines in SRCSAFE.INI:

```
Data_Path (Ragiv) = R:\SS\DATA
Data_Path (Kim) = R:\SS2\DATA
```

Visual SourceSafe uses the first setting and ignores the second. With a different setting of **SSINIENV**, Visual SourceSafe uses the second string and ignores the first. Hence, by setting the **SSINIENV** variable appropriately, you can control which **Data_Path** variable Visual SourceSafe reads and which DATA folder is used.

You use the /D option in the command line for the icon you use to run Visual SourceSafe as shown in the following example:

```
C:\VSS\WIN32\SSEXP.EXE /DKIM
```

See Also

[Initialization Variable Summary](#)

[Understanding Visual SourceSafe Projects](#)

[Using Project Headers in .INI Files](#)

[Visual SourceSafe Naming Syntax and Conventions](#)

Using Project Headers in .INI Files

See Also

Project headers allow you to specify variables on a project-by-project basis. Because almost every variable can be placed under a project header, you can customize Visual SourceSafe so that different projects act very differently.

If a variable occurs in the SS.INI file before any header, it applies to all projects. If a variable is placed under a project header, it applies only under that project. Consider the following example, in which the Diff_Format variable is set three times:

```
;the following variable applies to all projects
Diff_Format = Visual
;the following variables apply under $/Reports
[$/Reports]
Diff_Format = SS
Dir = C:\DEV
;the following variable applies under $/Reports/JournalLogs
[$/Reports/JournalLogs]
Diff_Format = UNIX
```

The first setting of Diff_Format occurs before any header; consequently, it sets the default for all projects. The second setting of Diff_Format overrides this setting for the project \$/REPORTS. The setting also, by default, is **inherited** by all **subprojects** under that project. However, under \$/REPORTS/JOURNALLOGS, the variable is explicitly set to UNIX; this setting blocks other inheritance, and applies only to JOURNALLOGS and its subproject.

The Dir variable behaves differently, because it specifies a folder path. Inheritance in projects creates a folder tree that mirrors the project list. In the example, for \$/REPORTS/JOURNALLOGS, the Dir variable is not C:\DEV, but rather C:\DEV\JOURNALLOGS.

See Also

Customizing the SS.INI and SRCSAFE.INI Files

Using Multiple Check Outs of Files

See Also

By default, Visual SourceSafe prevents more than one user from checking out the same file at the same time. However, your **Visual SourceSafe administrator** can allow, on a project-by-project basis, more than one person to check out a file simultaneously. This feature is very useful in a team development situation, where team members have different areas of code responsibility and are unlikely to need to change the same code module at the same time.

In a project where multiple checkouts are allowed, SourceSafe keeps track of all users who have the file checked out at any given time, and compares changes made to the file when each user checks in changes. In most cases, users have changed different lines of code in the file, so Visual SourceSafe simply merges the changes into the **master copy** of the file. However, if Visual SourceSafe detects that two users have each modified the same line of the file, Visual SourceSafe displays a message alerting the user checking in changes to resolve conflicting line changes before proceeding. When this happens, no check in occurs—the user receiving the message should edit the **local copy** in his or her **working folder** for the file and resolve the conflicts.

To help you resolve conflicts, Visual SourceSafe offers two methods of fixing the problems. The first method, Visual Merge, lets you quickly identify and correct any conflicts in a visual fashion. Visual Merge is the recommended way to resolve your conflicts. You can set Visual Merge as the default on the General tab of the Tools Options dialog box.

But, for those used to earlier SourceSafe versions, you can choose the standard SourceSafe merge to place identifying markers (six consecutive equal signs) in your file. Use these markers to see how the two versions of the file differ. When the conflicts are resolved, check the file in again. SourceSafe checks for the conflicts it initially found. At this point, Visual SourceSafe displays one of three messages:

- If there are no conflicts, Visual SourceSafe displays the message "<file> has been merged with no conflicts. Check in now?" Answer Yes if you want to proceed with the check in operation.
- If the same conflicts still exist, Visual SourceSafe displays the message "Have the conflicts in <file> been properly resolved?" Answer Yes if you satisfied that the conflicts have been resolved and therefore want to proceed with the check in operation. Answer No if you want to check with another user to be sure.
- If new conflicts have occurred (because another user with the file checked out has made changes while you were fixing the first set of conflicts), Visual SourceSafe displays the message "An automatic merge has occurred and there are conflicts." In this case, you have to start resolving conflicts process again.

See Also

Merging Files

Resolving Merge Conflicts

The Visual Merge

Sample Microsoft FoxPro Integration Code

This sample program provides an example of how to call the SourceSafe command line from FoxPro. You can use the command line to access SourceSafe files. This program creates a menu item that allows you to add and checkout files from SourceSafe.

```
*Create variables defining SourceSafe executable, and
* default project and command switches
PUBLIC ssexec, ssopt, ssprj
ssexec = "h:\ss\sswcl.exe "
ssoutput = "H:\ss\foxnss.txt "
ssopt = " -yguest -o&" + ssoutput + " -exitcode "
ssprj = "$/test/"

* Save old FoxPro menu and add SourceSafe menu pad
PUSH MENU _msysmenu
DEFINE PAD sspad OF _msysmenu PROMPT "\<SourceSafe" ;
    KEY alt+s
ON PAD sspad OF _msysmenu ACTIVATE POPUP sourcesafe
DEFINE POPUP sourcesafe MARGIN RELATIVE
DEFINE BAR 1 OF sourcesafe PROMPT "\<Add"
DEFINE BAR 2 OF sourcesafe PROMPT "\<Checkout"
DEFINE BAR 3 OF sourcesafe PROMPT "\<Disable SourceSafe Menu"
ON SELECTION POPUP sourcesafe ;
    DO callss IN ssmacro.prg WITH PROMPT()

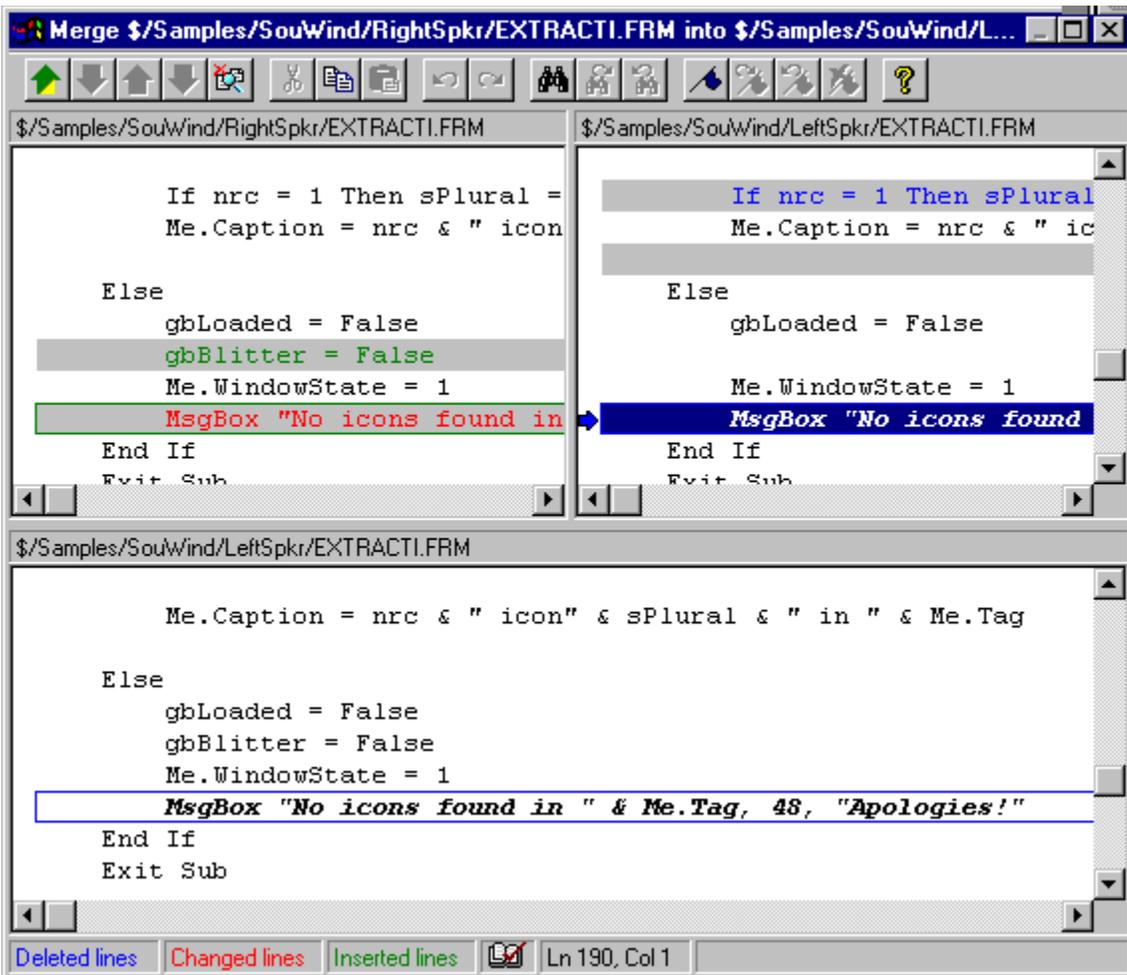
* Procedure to handle SourceSafe Menu options
PROCEDURE callss
PARAMETER mprompt
ssfname =GETFILE("Select file you wish to ")
DO CASE
CASE mprompt = "Add"
    sscmd = "add "
    ssfname =GETFILE(,"Select file you wish to ADD to
SourceSafe","Select" )
    cmdline = ssexec + sscmd + ssfname + CHR(32) + ssopt
    RUN /N &cmdline
CASE mprompt ="Checkout"
    sscmd = "checkout"
    ssfname =GETFILE(,"Select file you wish to CHECKOUT in
SourceSafe","Select" )
CASE mprompt = "Disable SourceSafe Menu"
    POP MENU _msysmenu
ENDCASE
```

The Visual Merge

See Also

If multiple users have checked the same file out, Visual SourceSafe (either when you check in a file, or optionally when you get the latest version) merges together your changes, with other users' changes. The same merging process is used when you choose the Merge Branches command. In either case, a *conflict* can occur, meaning that you and the other user changed the same line in different ways. Visual SourceSafe cannot resolve the conflict: it can only present it to you for a resolution. There are two methods which can be used for viewing and resolving merge conflicts.

The first method is the manual SourceSafe merge where SourceSafe adds conflict markers to the merged file and you edit the markers and conflicts manually with your text editor. This method is only recommended if you are using automated batch files that you do not want interrupted for conflict resolution. The second method, recommended in most cases, is the visual merge. The visual merge provides a graphical dialog box in which you can resolve all the conflicts quickly and with maximum control. You choose which method to use on the General Options Tab of the Tools Options dialog box. Visual merge is chosen by default for all merged files where conflicts are identified.



The Visual Merge dialog box uses a three-pane display. The top two panes show the two files being merged together, with differences between them highlighted. The bottom pane shows the resultant file you are creating. In the above picture, the merge is being performed because the file had been checked out to two users who made conflicting changes; hence, the file on the left is the one already checked into SourceSafe, the file on the right is on your hard drive. In the case where you have performed a Merge

Branches command, the two versions in the upper panes would be from different projects. Single-click on a shaded area in the upper panes to implement that change in the bottom pane, or edit the bottom pane manually.

The line below the bottom pane is called the status line, and shows the color legend, tips as applicable, and a graphic that tells you if you still have unresolved conflicts in the file. The  picture is used if there is a conflict. The

 picture is used when all conflicts have been resolved.

The default presentation of the Visual Merge dialog box follows these rules:

- Lines the same in both of the merged files appear in all three panes.
- Lines added in one file appear in **green** in that file. A blank background is shown in the corresponding place in the other file. Note that the addition could appear in either the left or right pane, depending on where it was added. In the bottom pane, the added line is shown, in black.
- Lines deleted from one file appear as a line shown in **blue**. Once again, note that deleted lines could appear in either or both files. Also note that when you see a line in your deletion color, it means that this line was deleted in the *other file*. In the bottom pane, a gray box is shown, to indicate that something was deleted here.
- Lines changed in one file appear in **red** in that file. The line is shown in black in the other file. In the bottom pane, the changed line is shown in black. Note that this is the only scenario that is not depicted in the picture above.
- A line changed in both files, but changed differently, appears in **red** against a blank background in both files. In the bottom pane, the original line is shown, in black against a gray background. Around the line is a red box to show an unresolved conflict. The red box is removed as soon as you change the conflicting line.

Note You can set your default colors for added, changed, and deleted text by using the Tools Options Difference tab.

Using the Visual Merge Dialog Box

Once the Visual Merge dialog box is displayed, you can begin to resolve any conflicts. You can make changes to the bottom pane, but cannot edit the upper two panes (because they are showing files in the state their authors left them). You can scroll these two upper panes, and they scroll together (and with the bottom pane), so you can easily see the differences between the two files. You can accept a change in one or the other top pane by clicking the changed line in the appropriate pane. You can also click your right mouse button on a change in one of the top panes to see a context menu with the following options:

- **Apply change** This choice is available only if the change is not already accepted. Choosing this option causes the change to be copied to the bottom pane in the appropriate place.
- **Remove change** This choice is available only if the change is already accepted. Choosing this option causes any accepted change to this line to be undone, returning the line to the way it was before the change. You can even return to a line you changed several modifications ago, and choose Undo to return the line to its state when you started this visual merge session.
- **Apply both changes** This choice is available only if the change has some text, and is not already accepted. It appends the selected change to what is already in the bottom pane. So in the case of a conflict, you can easily insert *both* sets of changes into the bottom pane.

The Bottom Pane

All of the File Viewer keyboard shortcut keys are supported in the bottom pane, including all typed characters, all four arrow keys, <Home>, <End>, <PgUp>, <PgDn>, and the standard cut, copy, and paste keys.

If you right-click in one of the changed lines in the bottom pane (marked with a gray background), you can choose from a context menu with the following options:

- Apply change on the left. (This appears only if the change on the left is not currently implemented. Choosing this option is the same as a left-mouse-click on the left-hand pane.) The change is written to the bottom pane.
- Apply change on the right. (This appears only if the change on the right is not currently implemented. Choosing this option is the same as a left-mouse-click on the right-hand pane.) The change is written to the bottom pane.
- Apply both changes. (This appears only if there is text in both the left-hand and right-hand panes; in other words, neither one is empty. It makes the change in the left-hand pane first, then appends the change of the right-hand pane.) Both changes are written to the bottom pane.
- Remove change. (This returns the change to its state in the original file. This choice is available unless no changes have been made to the original file, which happens in two cases – after a conflict, or after you have already chosen this option.)

When you are finished resolving the conflicts, click the Close button (the upper right hand corner of the window with the X on it). You will be prompted to confirm the closure.

- Choose Yes to confirm the changes and save them.
- Choose No to discard your changes in the Visual Merge dialog box and cancel the merge, check in, or get operation that began the merge process.
- Choose Cancel to return to the Visual Merge dialog box.

Note that you are asked to confirm changes even if you didn't made any changes at all, since SourceSafe potentially made a number of changes before opening the Visual Merge dialog box. If you choose Yes and there are unresolved conflicts, SourceSafe warns you that some conflicts were not resolved. If you continue and save now, the original (unchanged) text will be used to resolve those conflicts.

Visual Merge See Also

General Options Tab

Merge Branches Command

Merging Files

Merging Options

Resolving Merge Conflicts

File Viewer Keyboard Shortcuts

Visual Merge Sound

Project Difference Window Keyboard Guide

Each command has a key combination associated with it that enables you to execute a command with keystrokes. Typically, you press the CTRL or ALT key and simultaneously press the underlined letter in the menu command.

Shortcut Keys for Commands

Press	To
ALT+F3	Search for text.
ALT+ENTER	Show properties of a file or project.
ALT+X	Close Visual SourceSafe.
CTRL+*	Globally expand all projects.
CTRL+A	Add files to the current project .
CTRL+C	Copy text.
CTRL+D	Set the working folder for the current project.
CTRL+ENTER	View a file.
CTRL+F	Show differences for the selected file or project.
CTRL+G	Get the selected file or project.
CTRL+H	Show the history of the selected file or project.
CTRL+INS	Copy selected text to Clipboard.
CTRL+K	Check out the selected file(s).
CTRL+L	Reconciles all files.
CTRL+O	Check out a file.
CTRL+Q	Cancel a search display or status. Return to normal file display.
CTRL+R	Report on project differences.
CTRL+S	Search for a file checked out to a specific user. Use CTRL+Q to return to normal file display.
CTRL+U	Check in the selected file(s).
CTRL+V	Paste text from the Clipboard .
CTRL+W	Perform a Wildcard character search.
CTRL+X	Cut text and place it on the Clipboard.
CTRL+Z	Undo editing in dialog box.
DELETE	Delete the selected file or project.
ENTER	View a file.
F1	Get online Help on a specific item.
F3	Find next search string occurrence.
F5	Refresh file list.
INSERT	Create Project (if focus is in project pane), or Add Files (if focus is in file pane).
SHIFT+F3	Find previous search string occurrence.
SHIFT+CTRL+L	Select all files.
SHIFT+CTRL+S	Search for all files currently checked out. Use CTRL+Q to return to normal file display.
SHIFT+DELETE	Cut selected text and copy to Clipboard.

SHIFT+INSERT

Paste text on Clipboard to edit box.

Visual SourceSafe Web Features

Visual SourceSafe has several features designed to support webmasters and web authors who are tasked with organizing and maintaining a world wide web site. Typically, web sites involve a multitude of HTML and graphics files organized into various folders. Because these files change frequently, Visual SourceSafe is an ideal way to keep track of file versions, archive your files, and track changes to solve broken links and other problems.

The basic idea is that you collect a set of HTML, graphics, and supporting files that embody a web site and check them into a Visual SourceSafe project. Then, your Visual SourceSafe administrator, using the Visual SourceSafe Administrator program, designates that project as a web site project. When this is done, a special icon (–) is used in the Visual SourceSafe Explorer to show you the project is a web site project. See the [Administrator Help](#) file for more information on doing this.

From that point on, when the designated project is selected, the commands on the Visual SourceSafe Web menu are enabled. These are:

- [Check hyperlinks](#) in your HTML files to see if they are valid.
- [Create a site map](#) of all the HTML files in your project.
- [Deploy](#) a set of web files to a designated web server.

Visual SourceSafe also supports a new form of [keyword expansion](#) that makes adding a "Last modified on" date much nicer for HTML files.

Cloaking Visual SourceSafe Projects

See Also

Cloaking is a way of designating projects that you generally do not work on, and which you therefore do not necessarily want to keep local copies of. If a project is cloaked, Get, Check In, Check Out, Undo Check Out, and Project Difference commands do not apply to the project when attempted indirectly (as part of a recursive operation). Direct commands (with the project selected in the Explorer) still work normally on cloaked projects.

As an example: suppose that you work on a project called `$/OurApp`. In this project there are subprojects called Code and Tests which you need; and another subproject called Docs, which you never use. You might find yourself recursively getting `$/OurApp`, and then deleting the Docs subdirectory from your hard drive, in order to obtain the configuration you need.

The solution is to cloak `$/OurApp/Docs`. Click on the project and choose Properties, and click the "This project is cloaked for me" checkbox. Thereafter, you can still get this project or its files at any time. However, when you recursively get `$/OurApp`, the Docs project (and its subprojects) will be ignored. In other words, cloaking a project says "Ignore this project when I act on higher-level projects recursively."

When you cloak a project, its subprojects are implicitly cloaked when you use a recursive operation on the cloaked project or on a higher project in the project list.

When you do a recursive Get Latest Version command from `$/A`, the recursion stops at `$/A/B`, and no subprojects are retrieved. So, in that sense, they are auto-cloaked. But if you do a recursive Get Latest Version from `$/A/B`, all subprojects are retrieved (unless they are explicitly cloaked). In other words, a cloak says "when something recursive happens above me, stop here."

Cloaking See Also

General Project Properties Tab

Cloak Initialization Variable

Cloak Command Line Command

Decloak Command Line Command

HTML Keyword Expansion

See Also

Keyword expansion refers to Visual SourceSafe's ability to place certain information directly from the Visual SourceSafe database into your file for you. This can be handy, for example, if you want to place the string "Last modified on: *date*" in your web pages.

To do this, you place certain keywords, in comments so that they do not affect your HTML code, into the text of your file. When you add or check in the file, Visual SourceSafe looks for these keywords, and places the relevant information after them.

SourceSafe Keywords

The following table is a list of all the Visual SourceSafe HTML keywords. Note that these HTML keywords are identical to traditional SourceSafe keywords – except that they have an extra dollar sign in front of them. Keywords are case-sensitive, so be sure to capitalize them properly:

Keyword	Description
\$\$Archive: \$	Visual SourceSafe archive file location
\$\$Author: \$	User who last changed the file
\$\$Date: \$	Date and time of last check in
\$\$Header: \$	Logfile, Revision, Date, Author
\$\$History: \$	File history, Visual SourceSafe format
\$\$Log: \$	File history, RCS format
\$\$Logfile: \$	Same as Archive
\$\$Modtime: \$	Date and time of last modification
\$\$Revision: \$	Visual SourceSafe version number
\$\$Workfile: \$	Filename
\$\$NoKeywords: \$	No keyword expansion for all keywords that follow. (The colon is optional.)
\$\$JustDate: \$	Date, without the time addendum.

A SourceSafe HTML keyword has the following syntax:

```
$$keyword: $
```

For example, you could place the following in a file:

```
Last updated <!--$$JustDate:--!> <!--$-->
```

Upon check in, Visual SourceSafe could replace it with

```
Last updated <!--$$JustDate:-->5/16/96<!--$-->
```

The next time you check in the file, the 5/16/96 is replaced by the current date, and so on. This automates the process of updating your HTML pages each time you modify them.

Enabling Keyword Expansion

Because keyword expansion requires Visual SourceSafe to scan each file for keywords, it can considerably slow the Check In and Add Files commands. For this reason, keyword expansion is by default disabled for all files; you must indicate which files Visual SourceSafe should scan for keywords.

– To enable and view keyword expansion

- Set the Copy Keyword-Expanded Files Into Working Folder option on the Local Files tab of the Options dialog box (Tools menu) to view the effects of keyword expansion.

After a Check In or Add command, Visual SourceSafe copies the keyword-expanded versions of the files to your **working folder**.

Effect of Keyword Expansion

When you check in a file with keyword expansion enabled, Visual SourceSafe places a modified copy of your file directly into the current project. It then immediately gets the file, which refreshes your working folder with the checked-in file. The Get Latest Version command slows down the updating process. You can avoid it by not setting the Copy Keyword Expanded-Files Into Working Folder option on the Local Files tab of the Options dialog box. When you clear this option, Visual SourceSafe still expands keywords, and places the modified file into the project. However, it does not copy the file into your working folder.

See Also

Expanding Keywords

Using Visual SourceSafe 4.0a with Visual SourceSafe 5.0

If you have upgraded from Visual SourceSafe 4.0a to Visual SourceSafe 5.0, it is possible for some of your users to continue using Visual SourceSafe 4.0a. This is because the fundamental database format has not changed. You can check files in and out from 5.0, while other users check the same files in and out from 4.0a, and no corruption will occur.

This is most useful if you have users running DOS, Windows 3.1, or Windows for Workgroups. Because Visual SourceSafe 5.0 does not support those platforms, your 16-bit users will have to continue using 4.0a: and they can do so.

However, you should be aware of one important, obvious limitation, which is that Visual SourceSafe 4.0a does not support the Visual SourceSafe 5.0 feature set. As a few examples:

- Visual SourceSafe 4.0a does not work with files over 64 MB.
- If you have a file with "exclusive checkout," 4.0a users will still be able to check it out.
- If you lock the database in the SourceSafe Admin program, 4.0a users will not be locked out.

Again, these are examples of a general principle, which is that SourceSafe 4.0a will not support any new features of 5.0.

Also, note that if you have not upgraded to 4.0a, then your 4.0 users will not be able to co-exist with Visual SourceSafe 5.0 in the same database.

Visual SourceSafe's Web Site

Visual SourceSafe maintains a [web site](#) where you can get the latest information, free utilities to use with Visual SourceSafe, pricing information, helpful hints, and more. Check it out!

Topics

Project <project> has been destroyed, and cannot be rebuilt.

See Also

Visual SourceSafe may not be able to reproduce the project version you requested. This error has the following cause and solution:

- You permanently destroyed a **subproject** or moved the subproject to another location. For example, there may have been a subproject that you deleted or purged. Later, you attempted to go back to an earlier version of the project. Visual SourceSafe attempts to reconstruct the project exactly as it was in the past, but it cannot, because the subproject cannot be recovered.

Click Yes, and Visual SourceSafe does its best to restore the old project version, but restoration is not guaranteed.

Click No, and the attempt to restore an old project version is halted.

Click Yes All, and Visual SourceSafe does not notify you that other selected files may also be unrecoverable, but continues with the best possible reconstruction.

See Also

Purge Command

Delete Command

File <file> has been destroyed, and cannot be rebuilt.

See Also

Visual SourceSafe may not be able to reproduce the file you requested. This error has the following cause and solution:

- You permanently destroyed a file or moved the file to another location. For example, there may have been a file in a project that you deleted or purged. Later, you attempted to go back to an earlier version of the project. Visual SourceSafe attempts to reconstruct the project exactly as it was in the past, but it cannot, because the file cannot be recovered.

Click Yes, and Visual SourceSafe does its best to restore the old file, but restoration is not guaranteed.

Click No, and the attempt to restore an old file is halted.

Click Yes All, and Visual SourceSafe does not notify you that other selected files may also be unrecoverable, but continues with the best possible reconstruction.

An item with the name <name> already exists.

See Also

Each item in a Visual SourceSafe project must have a unique name. This error has the following cause and solution:

- You have attempted to name a file or an immediate **child project** with the same name as the project. (An item with that name can exist in other projects in the database, including subprojects.)

To resolve this error, give the new item a different, unique filename. Or, you can delete or rename the existing item.

Note File and project names in Visual SourceSafe are not case-sensitive. For example, if you have a file named TEST.C, you cannot create a subproject or file named "Test.c" or "test.c" in the same project.

See Also

Add Files Command

Delete Command

Rename Command

Share Command

File or project not found.

See Also

Visual SourceSafe is looking for a file or project that does not exist. This error has the following cause and solution:

- You mistyped the file or project name, or specified a file or project that does not exist.
- The Visual SourceSafe database is corrupted. .

Try retyping the name or select the name from the list in **Visual SourceSafe Explorer**. If you suspect database corruption, run the ANALYZE.EXE utility.

See Also

Visual SourceSafe Explorer

Version not found.

See Also

The version specified cannot be found. This error has the following causes and solutions:

- You specified a version that does not exist.
In Visual SourceSafe, click the Show History command on the Tools menu, and then choose a version from within the History of File dialog box.
- If you are using Visual SourceSafe from the command line, you may not be using the -V parameter properly.
Search online Help for more information on using the parameter correctly (click the Contents button).

See Also

Show History Command

Tracking Old Versions

-V Command-Line Option

This command only works on files.

This command can operate only on files. This error has the following cause and solution:

- You are attempting to perform a file operation (such as using the View command) on a project. Select the proper command, or select a file that works with the command you've chosen.

This command only works on projects.

This command can operate only on projects. This error has the following cause and solution:

- You are attempting to perform a project operation (such as the Move Project command) on a file.
Select the proper command, or select a project that works with the command you've chosen.

Parent not found.

See Also

This error has the following cause and solution:

- Database corruption is indicated.

Ask your **Visual SourceSafe administrator** to run ANALYZE.EXE to find and correct the problem. If that fails, call Microsoft Technical Support for help with recovering the database.

See Also

Technical Support

<File> is not branched in <project>.

See Also

This error has the following cause and solution:

- The file you are trying to merge is not branched.
When a file is branched, and you want to merge the different version of the file into one combined file, you can use the Merge Branches command. In this case, the file is not branched, so Merge Branches has no effect.

Cannot move a project under itself.

See Also

A project cannot be a project of itself. This error has the following cause and solution:

- You attempted to create an invalid loop in your project hierarchy. For example, moving the project \$/WORD under the project \$/WORD, or, more subtly, moving \$/WORD under \$/WORD/APPS. This creates an endless recursive loop, which causes this error.
Restructure your project hierarchy so that there is no invalid loop.

See Also

Move Project Command

Projects

File not checked out by you.

See Also

Some Visual SourceSafe operations can only be performed on a file that is checked out. This error has the following causes and solutions:

- You attempted to check in or undo a check out on a file that you did not have checked out.
Check out the file before checking it in. If you have made changes to a file, see the Advanced options in the Check Out dialog box, and then under Replace Writable, select Skip to check out the file to you without replacing the changes in your **working folder**.
- You attempted to perform an operation that can only be performed on a file that is checked out.
Check out the file.

See Also

Check In Command

Check Out Command

File <file> does not retain old versions of itself.

See Also

This error has the following cause and solution:

- When you check in a file, Visual SourceSafe retains both the current version of the file, and the information required to rebuild past versions. You have turned off the historical version storage for this file.

If you explicitly turn off this change information in the Properties dialog box or when using the Add Files command, Visual SourceSafe can give you the most recent version, but it cannot rebuild old versions.

See Also

Check In Command

Tracking Old Versions

Invalid date string.

Visual SourceSafe cannot use the date syntax entered. This error has the following cause and solution:

- The date you typed or entered is invalid.
Proper date syntax is Month/Day/Year, with each number separated by a slash mark: for example, 2/29/64 or 11/5/1992.

Invalid time or date string.

Visual SourceSafe cannot use the date or time syntax entered. This error has the following causes and solutions:

- The date you typed or entered is invalid.
Proper date syntax is Month/Day/Year, with each number separated by a slash mark: for example, 2/29/64 or 11/5/1992.
- The time you typed or entered is invalid.
Proper time syntax is Hour:Minute, followed by a one-letter A.M. or P.M. indicator: for example, 9:30A or 5:00p. (The indicator is not case sensitive.) Or, you can use a 24 hour syntax such as 13:45.

Invalid syntax on line <line> of file <file>.

See Also

This error has the following cause and solution:

- The syntax of one or more lines in the initialization file is invalid.
Search Help for information on the proper syntax of the initialization variable.

Note You can use any text editor to change or delete the invalid line.

See Also

[The Initialization File Variables](#)

Initialization variable <variable> set to invalid number.

See Also

The specified initialization variable has been set incorrectly. This error has the following cause and solution:

- An initialization variable that should be set with an Arabic number is not. For example, it may be set to "two" when it should be set to "2."

Search Help for information on the proper syntax of the initialization files.

Note You can use any text editor to change or delete the invalid line.

See Also

[The Initialization File Variables](#)

Initialization variable <variable> must be between <number> and <number>.

See Also

An initialization number is out of the valid range. This error has the following cause and solution:

- An initialization variable has been set to an invalid number.
Search Help for information on valid settings.

Note You can use any text editor to change or delete the invalid line.

See Also

[The Initialization File Variables](#)

Initialization variable <variable> must be set to "Yes" or "No".

See Also

This error has the following cause and solution:

- An initialization variable that should be set to Yes or No is not.
Change the variable so that it is set to Yes (or True) or No (or False).

Note You can use any text editor to change or delete the invalid line.

See Also

[The Initialization File Variables](#)

Initialization variable <variable> set to invalid folder path.

See Also

An initialization variable that should be set to a valid folder path is set to an invalid path. This error has the following cause and solution:

- The path syntax may be incorrect or the path may not exist.
Verify that the path exists or that the syntax is correct and change it as necessary.

Note You can use any text editor to change or delete the invalid path.

See Also

[The Initialization File Variables](#)

Initialization variable <variable> set to invalid value.

See Also

This error has the following cause and solution:

- An initialization variable is set to an invalid value.
Search Help for information on the proper syntax.

Note You can use any text editor to change or delete the invalid path.

See Also

[The Initialization File Variables](#)

Cannot find initialization variable <variable>.

See Also

Visual SourceSafe cannot find an initialization variable it needs to perform an operation. This error has the following cause and solution:

- The specified initialization variable is missing and is required to complete the operation. Search Help or the *Microsoft Visual SourceSafe User's Guide* for information on the variable, determine a proper value for it, and add it to your initialization file.

See Also

[Configuring Visual SourceSafe](#)

Cannot find SS.INI file.

See Also

Visual SourceSafe cannot find your SS.INI file. This error has the following causes and solutions:

- The SS.INI file is not where Visual SourceSafe expects to find it. (See your Visual SourceSafe administrator if you have moved this file.)
- You have inadvertently deleted your SS.INI file.

You can copy an SS.INI file from another user into your Visual SourceSafe folder, then modify it to customize Visual SourceSafe. Your Visual SourceSafe administrator must then set Visual SourceSafe to look in the proper place.

See Also

The Initialization File Variables

Permission denied.

This error has the following cause and solution:

- You are attempting to take an action that you do not have the proper privileges to execute. Talk to your Visual SourceSafe administrator. If you are the **Visual SourceSafe administrator**, run Visual SourceSafe Administrator to check on or change your username, password, and access rights.

User <user> not found.

This error has the following cause and solution:

- Your **username** was not found.
Talk to your **Visual SourceSafe administrator**. If you are the Visual SourceSafe administrator, run Visual SourceSafe Administrator to add your username.

Invalid password.

This error has the following cause and solution:

- Your **username/password** combination is invalid.

Re-enter the password. Because it does not appear on the screen, it is very easy to type it incorrectly. Note that the Visual SourceSafe password is case-insensitive. If you believe you have typed the password correctly, Visual SourceSafe doesn't recognize the password for some reason, so ask your **Visual SourceSafe administrator** to change your username and password. If you have forgotten your password, the administrator can use Administrator to assign you a new one.

Bad username syntax: <syntax>.

This error has the following cause and solution:

- You have entered a **username** with invalid Visual SourceSafe user syntax.
Ask your **SourceSafe administrator** for your correct Visual SourceSafe username.

Invalid SourceSafe syntax: <syntax>.

See Also

The syntax you entered is invalid in Visual SourceSafe. This error has the following cause and solution:

- You have entered a Visual SourceSafe file or **project** path that does not use valid syntax.
In Visual SourceSafe syntax, every path starts with a dollar sign followed by a series of project names, which are separated by slashes and optionally followed by a filename. For example:
- `$/` is the **root project**.
- `$/CODE` is a subproject of `$/`.
- `$/CODE/WIN/TEST.C` is a file in the `$/CODE/WIN` project.

Many parts of this syntax are optional. You can omit the dollar sign in most circumstances. You can also shorten the path by basing it on your current project; for example, if you are in `$/CODE`, you can type `WIN` as a shorthand for `$/CODE/WIN`, `TEST.C` for `$/CODE/TEST.C`, and `..` for the root.

Filenames are any valid operating system filenames; project names can include spaces and can be up to 256 characters long. Project names are not case-sensitive (for example, to Visual SourceSafe, `Code` and `CODE` are the same project).

See Also

SourceSafe Naming Syntax

This command operates on only one item.

For many Visual SourceSafe dialog box lists, you can choose more than one item at once, and then act on all the items together. Some actions, however, cannot operate on more than one item at a time. This error has the following cause and solution:

- You selected more than one item and then applied a command that operates on only one item at a time. For example, you may have tried to use the View command, which works with one file at a time.

Press the space bar on an item to make it the only item selected in the list.

You cannot use wildcards with this command.

Many Visual SourceSafe commands accept **wildcard characters**, such as * and ?. But some commands, such as the View command, can operate on only one item at a time. This error has the following cause and solution:

- The command you have chosen does not accept wildcard characters.
Use an explicit filename, with no wildcard characters.

No items listed to operate on.

This error has the following cause and solution:

- You are attempting to perform an action without first selecting an item to operate on.
Click an item to select it, and then try the action again.

Network not found.

This error has the following cause and solution:

- No network is installed, or Visual SourceSafe cannot find a valid network.
Perhaps there is no network installed, or one or more network connections have failed. Check your network hardware, and make sure that all the necessary network drivers are running. Check with your Visual SourceSafe administrator for additional assistance.

<File> is an invalid or corrupted logfile.

See Also

This error has the following cause and solution:

- Visual SourceSafe cannot find or read your logfile.
Run ANALYZE.EXE with the -F option. If necessary, check with Technical Support for more assistance.

See Also

Technical Support

Corrupted history in file <file> after revision <number>.

This error has the following cause and solution:

- The Visual SourceSafe database has been corrupted.
DDCONV.EXE saves as much of the database as possible, but some information will be irretrievably lost (unless you have backed up the database and can recover the information from there.)

Illegal file or project name(s): <file>".

See Also

This error has the following cause and solution:

- You have entered an illegal file or project name.
Check to make sure that the name was entered correctly. If you are unsure which characters are allowed in a SourceSafe name, check the Visual SourceSafe Naming Syntax and Conventions topic.

See Also

Visual SourceSafe Naming Syntax and Conventions

Illegal version syntax: <number>.

This error has the following cause and solution:

- The version number you entered uses illegal syntax.
Check to make sure that the information was entered correctly.

An item named <item> was already deleted from this project.

See Also

In Visual SourceSafe, you can delete only one instance of a file or project with a given name in a given project. This error has the following cause and solution:

- You deleted a file, created a new file with the same name, and then attempted to delete that file as well.

You can choose Yes to purge the old file, or No to cancel the delete operation. If you purge the original file, you will lose all historical information in it, but you can then continue with the current delete operation. Or, you can choose No, and then rename the new file before deleting it, if you want to keep the file history of the old file.

See Also

Purge Command

Delete Command

Rename Command

Cannot delete, rename, or move the root project.

This error has the following cause and solution:

- The Delete, Rename, and Move Project commands cannot be applied to the **root project** (\$/), which is analogous to the root folder of your hard disk (C:\).
Since the root project cannot be changed in any way, consider restructuring your project.

<file> is already checked out by <user>. Continue?

See Also

This warning has the following cause and solution:

- You attempted an operation on a Visual SourceSafe file that is checked out by another user. Visual SourceSafe gives you the option to continue, which means any differences between the changes you make to the file and the changes made by the other user will need to be merged in the future. Click Yes to continue your operation and create a multiple-checkout situation; answer No to cancel the check out action.

See Also

[Check Out Command](#)

[Destroy Command](#)

[Delete Command](#)

Rename does not move an item to another project.

See Also

The Rename and Move Project commands cannot be implemented at the same time. This error has the following cause and solution:

- The Visual SourceSafe Rename command and Move command are separate, distinct operations; you cannot do both in one step.
First perform one action and then do the other.

See Also

[Move Project Command](#)

[Projects](#)

[Rename Command](#)

File <file> is currently checked out by <user>.

See Also

This error has the following cause and solutions:

- When someone other than you has a file checked out, there are some things you cannot do to the file. For instance, you cannot delete or rename a file that is checked out.

Click Yes to check out the file anyway. (Yes All checks out all selected files even if they are already checked out by another user.) If you choose yes, there may be conflicts between your changes and those of another user, which will have to be resolved and merged later.

Click No to cancel the Check Out command. (No All cancels the command for all selected files.)

If you want to work on a checked-out file, it's best to check the User column of **SourceSafe Explorer** for the name of the user who has it checked out and ask that user to check it in as soon as possible. If the user is unavailable, the **Visual SourceSafe administrator** can undo the check out for you.

See Also

Undo Check Out Command

Cannot rename to <file>; that name already exists.

See Also

The filename specified already exists in the project. This error has the following cause and solution:

- When you rename a file that is shared by many different projects, the name of the file changes in all the projects that are sharing it. If any one of these projects already has a file with the same name, you create an ambiguous and invalid situation, two files in the same project with the same name, and the Rename operation fails.

To correct this, choose another name for the file.

See Also

Rename Command

Shared Files

Project <project> does not exist.

This error has the following cause and solution:

- You are attempting to move a project to a **parent project** that does not exist.
Check the path of the parent project and try again. If you are unsure of Visual SourceSafe project syntax, try selecting the parent from the **project list** in **SourceSafe Explorer**, and then note the project path displayed above the **file list**.

Note To move a file or subproject to a new parent, you must have already created the new parent project. Use the Create Project command to create a new project.

Path <path> too long.

See Also

The path specified is too long. This error has the following cause and solution:

- You typed a Visual SourceSafe path that is too long.
Visual SourceSafe project names are limited to 64 characters. Full project path names can be no longer than 255 characters.

See Also

SourceSafe Naming Syntax

This version of <name> already has a label: overwrite?

Any version of a Visual SourceSafe file or project can have a **label**, which is used to identify that **version**. This error has the following cause and solution:

- One version of a file or project cannot have more than one label. Therefore, when you attempt to label a version that already has a label, Visual SourceSafe displays this message.
Click Yes to apply the new label, and the old label is replaced. Click No to cancel the action, and the old label is left in place.

This label <label> is already used. Remove the old label?

A Visual SourceSafe **label** is used to identify a particular **version** of a file or project. This error has the following cause and solution:

- You are attempting to assign a label to a file or project, and another version of that file or project has the same label.

Click Yes, the new label is applied, and the old label is deleted. Click No, and the old label is not changed. Click Cancel to cancel the labeling operation.

File <file> is already shared with this project.

See Also

This error has the following cause and solution:

- You are attempting to share a file with the **current project**, and that file already exists in the current project.

If you want to share a different version of the file, specify a version in the History of File dialog box and click Pin. If you want to share this file with another project, select the project to share the file into in Visual SourceSafe Explorer and on the SourceSafe menu, click Share. Then, select the file to share from the Share dialog box.

See Also

Show History Command

Shared Files

Tracking Old Versions

-V Command Line-Option

Cannot check out an old version of a file.

See Also

The Check Out command always uses the most recent version of a file. This error has the following cause and solution:

- You are trying to check out an old version of a file.
If you want to use and modify a previous version, take the following steps:
 1. Check out the most recent version of the file.
 2. Delete it from your **working folder** (not the Visual SourceSafe database).
 3. Get the old version that you want to modify.
 4. Make your changes to the old version. (This may require removing its read-only flag manually.)
 5. Check the modified file into Visual SourceSafe.

Note You can also use the Rollback button in the History of File dialog box (on the Tools menu, click Show History). If you roll back to a certain version, all versions subsequent to that version are deleted, and you can check out the file and start modifying it. The disadvantage of this method, however, is that you lose all the subsequent versions, and cannot retrieve them.

See Also

[Check In Command](#)

[Check Out Command](#)

[Get Latest Version Command](#)

[Rollback Command](#)

File <file> is checked out to project <project>, and you are in <project>.

When a file is shared among multiple projects, a file in one project is linked to all other projects that share it. Therefore, when you check out the file from one project, it is also checked out from all the projects that share it. This warning has the following cause and solution:

- You checked out a file from one project, and you are attempting to check the file in or undo the checkout from another project that shares the same file.

This is a valid action—Visual SourceSafe is simply alerting you in case this is not what you intend. To proceed, click Yes; to cancel the operation, click No.

Note You can check in a file or undo a checkout from any project that shares a file.

File <file> was checked out to folder <path>.

When you check out a file, Visual SourceSafe keeps track of the **working folder** you checked it out to. In general, you make your modifications in that folder, and then check in the file or undo the checkout from the same folder when you are done. This warning has the following cause and solution:

- You are attempting to check in or undo a checkout of the file from a folder different from the one you checked it out to.

Your working folder for a project is displayed just above the file pane in **SourceSafe Explorer**, and can be changed with the Set Working Folder command from the File menu (CTRL+D); the **check out folder** of a file is displayed in the **file list**. (The check out folder may be different from your working folder assigned to the project, if someone else has the file checked out, for example.) Although you can check a file in from a folder other than the one you checked it out to, make sure that this is what you intend.

File <file> has not been changed.

This error has the following cause and solution:

- You are attempting to check in a checked out file that you have not changed.
Click Yes, and Visual SourceSafe checks it in anyway. (Clicking Yes All checks in all selected files.) A check in creates a new version of the file, with an optional comment.
Click No, and Visual SourceSafe undoes the check out of the file. (Clicking No All undoes the checkout for all selected files.) No new version of the file is created in this case.

Note To avoid this message on future occasions, you can turn off this notification on the General tab in the SourceSafe Options dialog box on the Tools menu.

A writable copy of <file> already exists.

When you get a Visual SourceSafe file, the file is tagged with the **read-only** flag—you cannot modify the file. When you check out the file for modification, the read-only flag is deleted. When you later check it in, the file is flagged read-only again. This error has the following cause and solution:

- You attempted to use the Get Latest Version command on a file that already exists on your hard disk in writable form. If the **local copy** is writable, you may have it checked out, or are otherwise modifying it; Visual SourceSafe does not replace the file, and displays this error message.

Click Yes or Yes All to replace the file or files with the current version in the Visual SourceSafe database.

Click No or No All to not replace the file or files.

Click Cancel to cancel the Get Latest Version command.

File <file> is not shared with any other projects.

The Branch command is used to break a **share link**. This error has the following cause and solution:

- The file you've chosen to apply the Branch command to has no link to other projects.
Choose another command, or choose a shared file to apply the Branch command.

You are using an old version of <file>, and cannot change it. Rollback to make that version the most recent if you need to change it.

See Also

This error has the following cause and solution:

- You tried to modify an old version of a file.

If you want a project to always use a specific version, and not receive checked in changes made in other projects, you can share a specific version of the file using the Pin command in the History of File dialog box.

If you want to start modifying the file from an old version, you can roll back the file. The Rollback command only affects the current project; it performs an implicit separate operation. Both branching and roll backs are available from the History of File dialog box.

See Also

Rollback Command

Tracking Old Versions

Cannot roll back to the most recent version of <file>.

See Also

The Rollback command returns a file to a previous version of itself. This error has the following cause and solution:

- You have attempted to roll back a file to its most recent version or have not specified a version to roll back to.

You cannot roll back a file to the current version in Visual SourceSafe. If you meant to specify an earlier version, retry the command with the earlier version specified.

See Also

Rollback Command

A deleted link to <file> already exists.

See Also

A deleted file remains part of a project, but it doesn't appear in the **file list**. This error has the following cause and solution:

- You are attempting to share a file from another project with the **current project**. However, the file already exists in the current project. It doesn't appear in the file list because it has been deleted. To make the file appear in the file list, don't share it from another project; rather, recover it in this one using the Recover button on the Deleted Items tab in the Properties dialog box (File menu).

See Also

Recover Command

Delete Command

Shared Files

Folder <path> not found. Create?

The specified **working folder** does not exist. This error has the following cause and solution:

- The folder may have been deleted, or you may have entered its specification incorrectly. Your network connection to a networked location may also have gone down.

Reset the working folder for this project (using the Set Working Folder command on the File menu), or click Yes to create the folder. This tells Visual SourceSafe that this is the local working folder. When you open the project again, Visual SourceSafe automatically goes to this working folder.

If your working folder seems fine, check to see if this project is shadowed to another location, and that your connection to that location is still OK.

<item> specifies a version number, which is illegal for this command.

This command operates only on the most recent version of a file. This error has the following cause and solution:

- You attempted to execute the command on an older version of the file.
Choose another command, or apply the command you've chosen to the most recent version of the file.

This command operates only on projects, not on files (like <file>).

This command operates only on **projects**. This error has the following cause and solution:

- You attempted to execute the command on a file.

In the **SourceSafe Explorer**, select a project in the **project list**, then try the command again.

This command operates only on files, not on projects (like <project>).

This command operates only on files. This error has the following cause and solution:

- You attempted to execute the command on a **project**.
In the **SourceSafe Explorer**, select a file in the **file list**, then try the command again.

<file> is not a text file.

Although most Visual SourceSafe commands work on either text or binary files, this command does not. This error has the following cause and solution:

- You attempted to execute a command, such as View File on the Edit menu or Find In Files on the Tools menu, that operates only on text files.

Binary file types are set either automatically by Visual SourceSafe or manually using the File Types tab in the SourceSafe Options dialog box (Tools menu).

Invalid SourceSafe path <path>.

See Also

This error has the following cause and solution:

- You have entered an invalid Visual SourceSafe project or filename.

Check to make sure that you typed the path correctly, or use **Visual SourceSafe Explorer** to choose the item from the **project list** or **file list**.

If you chose the item from Visual SourceSafe Explorer, it is possible that another user has deleted or renamed the item since you last used Visual SourceSafe. Collapse the **root project** and re-expand it to refresh the **project pane**.

See Also

Delete Command

Rename Command

<item> is not a deleted file or project.

See Also

The Recover and Purge commands operate only on items that have been deleted. This error has the following cause and solution:

- You attempted to use the Recover or Purge command on an item that has not been deleted.
If you are trying to recover a file or project, there is no need, as the file or project already exists in Visual SourceSafe. To view the file, use the View File command.

See Also

Delete Command

Destroy Command

Purge Command

Recover Command

<item> has been deleted.

See Also

This error has the following cause and solution:

- You are trying to act on a file or project that has been deleted in the current project.
Before you can do anything to the file, you must recover it using the Recover command.
If you want to delete the file from the Visual SourceSafe database permanently, on the File menu, click Properties, and then on the Deleted Items tab, click Purge.

See Also

Purge Command

Recover Command

Delete Command

<File> is checked out by <user>; deleting it will cancel the check out. Continue?

This warning has the following cause and solution:

- You are attempting to delete a checked out file.

This warning provides the opportunity to correct from deleting the wrong file. If the filename is correct, click Yes to proceed. Click No to cancel the delete operation.

Destroy cannot be undone; information will be lost permanently!

See Also

This warning has the following cause and solution:

- When you use the Destroy command on a file or project, the action is not reversible.
Click Yes to continue with the Destroy operation. Click No to cancel the operation.

Note If you do not want to receive this message every time you destroy a file or project, you can turn it off on the Warnings tab in the SourceSafe Options dialog box (Tools menu).

See Also

[Destroy Command](#)

[Purge Command](#)

[Delete Command](#)

Delete all specified items?

See Also

This warning has the following cause and solution:

- You are about to delete a file or project.
Click Yes, and Visual SourceSafe deletes the file or project.
Click No, and Visual SourceSafe does not.
Use the Recover command to undo Delete.

Note If you do not want to receive this message every time you delete a file or project, you can turn it off on the Warnings tab in the SourceSafe Options dialog box (Tools menu).

See Also

Recover Command

Purge cannot be undone; information will be lost permanently!

See Also

This warning has the following cause and solution:

- When you purge a file or project, the action is not reversible.

Click Yes to continue with the Purge operation. Click No to cancel the operation.

If you do not want to lose the file or project permanently, use the Delete command instead.

Note If you do not want to receive this message every time you delete a file or project, you can turn it off on the Warnings tab in the SourceSafe Options dialog box (Tools menu).

See Also

[Destroy Command](#)

[Purge Command](#)

[Delete Command](#)

Rollback cannot be undone; some versions will be lost irretrievably!

See Also

This warning has the following cause and solution:

- You are about to roll back a file to an earlier version. The Rollback command is not reversible. To make the old version the most recent version without losing all the versions in between:
 1. Check out the file.
 2. Delete the **local copy**.
 3. Get the version you want to roll back to.
 4. Check that version into Visual SourceSafe.

Note If you do not want to receive this message every time you roll back a file, you can turn it off on the Warnings tab in the SourceSafe Options dialog box (Tools menu).

See Also

Rollback Command

<file> has changed. Undo check out and lose changes?

See Also

This warning has the following cause and solution:

- You are about to undo the check out of a file. If you do so, you discard all the changes made since you checked out the file.

If you want to put your changes into Visual SourceSafe instead of losing them, use the Check In command instead.

Note If you do not want to receive this message every time you undo a checkout on a file, you can turn it off on the Warnings tab in the SourceSafe Options dialog box (Tools menu).

See Also

Check In Command

Undo Checkout Command

This will end your SourceSafe session.

This message has the following cause and solution:

- You are about to exit Visual SourceSafe.

Click OK to exit Visual SourceSafe, or click Cancel to return to Visual SourceSafe.

Note If you do not want to receive this message every time you exit, you can turn it off on the Warnings tab in the SourceSafe Options dialog box (Tools menu).

There are no deleted items in this project.

This error has the following cause and solution:

- There are no items in this project that have been deleted.
Because no files have been deleted from this project, there are no files that you can recover or purge.

The files are identical.

This message has the following cause and solution:

- The files you have selected, the Visual SourceSafe **master copy** and the **local copy**, are identical. You may have selected a file in the **file list** and tried to run the Show Differences command only on that file, in which case the file is being compared to itself and is by definition identical. In this case, before applying the Show Differences command again, specify two distinct files.

No search pattern was given.

This error has the following cause and solution:

- You have not entered a string or pattern that Visual SourceSafe can use to search through files. Enter a string in the Find In Files dialog box and then click OK, or click Cancel to exit the dialog box and return to Visual SourceSafe Explorer.

No matches were found.

This error has the following cause and solution:

- Visual SourceSafe has searched all the selected files, and has found no occurrences of the string you are trying to find.
Check to make sure that you've entered the string correctly.

Always share the latest version of <file>?

See Also

Generally, every project sharing a file always has the latest version of the file. When you check in the file in any project, all the projects are automatically updated. However, you can use the Pin button in the History of File dialog box to share an explicit version of a file. This message has the following cause and solution:

- By clicking the Pin button in the History of File dialog box, you've chosen to share the most recent version of the file.

If you click No, the project shares the pinned version explicitly, and does not receive checked-in changes from other projects.

If you click Yes, the project returns to a sharing state in which it is always using the latest version of the file.

See Also

[Pin Command](#)

[Shared Files](#)

[Tracking Old Versions](#)

[Unpin Command](#)

You must specify a new name.

This error has the following cause and solution:

- You are attempting to rename a file or project without specifying a new name for it.
Either specify a new name and then click OK, or press the ESC key to close the dialog box without renaming the file or project.

Binary files differ.

This message has the following cause and solution:

- The binary files you've selected differ.

The Show Differences command does not display line-by-line differences for binary files.

Therefore, Visual SourceSafe can only tell you whether or not the Visual SourceSafe **master copy** is different from your **local copy**.

The passwords don't match.

This error has the following cause and solution:

- The **password** you typed in the Password box does not match the one typed in the Verify box. If the passwords you type are identical, Visual SourceSafe assumes you typed them correctly, and changes your password. If you type two different passwords, Visual SourceSafe assumes that you mistyped the password; you must re-enter it.

Get the entire project containing this file version?

See Also

This message has the following cause and solution:

- You have selected a file in the History of File dialog box list and then clicked Get.
Click Yes, and Visual SourceSafe gets the entire project as it existed immediately after the selected file was checked in.
Click No, and Visual SourceSafe gets only the file version you selected, but doesn't get any other files in the project.

See Also

[Get Latest Version Command](#)

Running more than one instance of the SourceSafe Explorer or the SourceSafe Administrator is not supported under 16-bit Windows.

This error has the following cause and solution:

- You can run only one version of SourceSafe at a time on a 16-bit platform.
Do not try to run a second instance of Visual SourceSafe. Running multiple instances of Visual SourceSafe is supported under Microsoft Windows NT and Microsoft Windows 95.

File <file> not found.

Visual SourceSafe could not find the file it was looking for. This error has the following cause and solution:

- You may have mistyped a filename in **Visual SourceSafe Explorer** or in an initialization file variable.
- The file may not exist.

Make sure that you've correctly entered the name of the file.

Too many file handles open.

This error has the following cause and solution:

- There are too many files open on your system.
You may want to check whether any of the files can be closed, or you may want to increase the Files count in your operating system's initialization file.

Access to file <file> denied.

At certain times, Visual SourceSafe denies a user access to a file. This error has the following causes and solutions:

- You were denied access (read or write privileges) to a file that Visual SourceSafe was attempting to open.

It is possible that another user has the file momentarily locked; try again.

- You may not have sufficient network privileges to take the action you are attempting.

See your **Visual SourceSafe administrator** for more information on your Visual SourceSafe **access rights**. Also see the Administration chapter of the *Microsoft Visual SourceSafe User's Guide* for a description of the network access rights required for various users in different directories.

Invalid drive: <drive>.

This error has the following cause and solution:

- The drive you are attempting to access is invalid. You may have mistyped the drive specification, either in **Visual SourceSafe Explorer** or in an initialization file.
Check to make sure that you've entered the correct the drive specification.

File <file> already exists.

This error has the following cause and solution:

- You tried to create a file or folder with a name that already exists.
You can give Visual SourceSafe files the same name as many times as you want in different directories, but only once in a given folder. Select a different name or rename the conflicting file or project.

Disk full.

This error has the following cause and solution:

- You tried to save data to a full disk.
Free some disk space by deleting or moving unneeded files.

Invalid filename: <file>.

This error has the following cause and solution:

- You typed a filename with invalid syntax.

Check your typing, or select a different name with valid characters. You may also get this error if you have multiple files selected and try to use a command meant for a single file, such as View.

Error reading from file.

Visual SourceSafe cannot read from a file it has opened. This error has the following cause and solution:

- You may have lost your network connection.
Exit Visual SourceSafe, and then check all network connections before restarting.

Error writing to file.

Visual SourceSafe cannot write data into a file that it has opened. This error has the following cause and solution:

- You may have a full disk.
Exit Visual SourceSafe, and then free disk space by deleting unnecessary files.

Out of memory.

You do not have sufficient computer memory to perform this operation. This error has the following cause and solution:

- You have too many applications, documents, or source files open.
Close any unnecessary applications, documents, or source files that are open.
- You have many device drivers loaded.
Eliminate unnecessary device drivers.

You have <file> checked out; deleting it will cancel the check out.

This warning has the following cause and solution:

- You are attempting to delete a file checked out to you.
This warning provides the opportunity to avoid deleting the wrong file. If the filename is correct, click Yes to proceed. Click No to cancel the delete operation.

Folder not found.

This error has the following cause and solution:

- You specified a operating system folder that could not be found.
Check to make sure that the folder specification is correct. Check capitalization on case-sensitive operating systems, and re-try.

Cannot find Help file <file>.

See Also

This error has the following cause and solution:

- The Help file isn't available.
Try reinstalling Visual SourceSafe to get a copy of the Help file. If the file is still missing, call Technical Support.

See Also

Technical Support

Too many file handles open.

This error has the following cause and solution:

- You have too many files open at once..
Close some open files.

File <file> is locked.

This error has the following cause and solution:

- The file you are trying to access is already being used.
Wait and try your operation again later.

An automatic merge has occurred and there are conflicts. Edit <file> to resolve them.

See Also

Changes you made to the file since you checked it out conflict with other changes made (and checked in) to the source file by another user. This error has the following cause and solution:

- You have to resolve the conflicts between your changes to a file and another user's changes before you can check in the file.

Open the file in your **working folder** and search for six consecutive equal signs (=====). Visual SourceSafe uses these to mark the location of a conflict.

You may prefer to use the Visual Merge feature to make conflict resolution easier. To keep this situation from arising again, your **Visual SourceSafe administrator** can disable multiple check outs in your project.

See Also

Resolving Conflicts

You do not have access rights to <command name>.

This error has the following cause and solution:

- Your Visual SourceSafe access rights are insufficient for the command you are attempting to execute. For example, if you are a read-only user, you can use the Get Latest Version command to view files, but you cannot use the Check Out command to modify them.

If you believe you have received this message in error, contact your **Visual SourceSafe administrator**.

Version <NUMBER> is identical to version <NUMBER> of <file>.

This error has the following cause and solution:

- The two versions you are comparing are identical.
Check to make sure that the two versions are the ones you want to compare.

No additional Help is available on this item.

See Also

- No more information is available on this dialog box. If the cause of the problem is not clear from the message, please call Technical Support for more assistance.

See Also

Technical Support

User <user> already exists.

This error has the following cause and solution:

- You attempted to add a user with a **username** that already exists. You may want to choose another name; for instance, if there is already a Joyce, add JoyceF.
Note that names are not case-sensitive, so if you already have a Joyce, adding JOYCE generates this error.

Cannot delete/rename/edit the Admin user.

This error has the following cause and solution:

- You attempted to delete, rename or change the Admin user. The **Visual SourceSafe administrator** must always exist in your Visual SourceSafe Administrator user list, although you should not use Admin as an actual user.

You can, however, change the Admin user's password, used to **logon** to Visual SourceSafe Administrator.

Cannot check out <file>. It is binary and is already checked out.

This error has the following causes and solutions:

- The file you specified is binary and already checked out. Visual SourceSafe does not allow multiple check outs of binary files, because their changes cannot be merged.
Wait to check out the file until after the other user has checked it back in.

Are you sure you want to delete?

This message has the following cause and solution:

- This is the standard delete warning displayed by Visual SourceSafe.
Click Yes, and Visual SourceSafe deletes the item(s).
Click No, and Visual SourceSafe cancels the delete command.

Filter string exceeds the maximum length of <number>.

This message has the following cause and solution:

- The combined length of the filter values on the File Types tab in the SourceSafe Options dialog box (Tools menu) are too long.

SourceSafe uses the File Types Included in File Group field to create a filter string for the Share and Add Files dialog boxes. This error occurs when the combined length of the file groups are too long. The total buffer length is 511 characters. To correct this error, shorten your file type lists.

Invalid value <value> on File Types tab.

This message has the following cause and solution:

- You have attempted to create a file type using invalid characters.
File Type names cannot contain the following characters: [] () , ; . The value for a file type is a search pattern, and may not contain invalid characters. Also, you cannot leave the "File type included in this group" edit box blank – some file types must be entered.

No application is associated with this file type.

This message has the following cause and solution:

- You have attempted to view a file for which no file association exists.
This error is usually caused by trying to view a binary file. Check that you have selected the correct file, and if you have, associate an editing program with this file extension by using either the Windows95 Explorer or the File Manager.

Have the conflicts in <file> been properly resolved?

This message has the following cause and solution:

- You have attempted to check in a file changed from the current version stored in SourceSafe .
This warning message often is displayed when you have checked out the same file to more than one user or computer. It is displayed so you can be sure that any merge conflicts have been resolved. In some cases, this warning may be displayed even if no merge conflicts occurred, and that is potentially confusing. But, if this is the case, just click Yes and continue with your work.
If there were conflicts, and you did not open the source file to resolve them, click No and do so now. When finished, re-try your Check In operation.

The passwords do not match.

See Also

This error has the following cause and solution:

- The **password** you entered for the user did not match.
When changing a password for a user, you must type the new password twice to guard against typing errors. In this case, the passwords you typed did not match. Try typing them again.

See Also

Technical Support

Out of memory - the rights changes could not be saved.

You do not have sufficient computer memory to perform this operation. The changes you made to the user's rights have not been saved.

This error has the following cause and solution:

- You have too many applications, documents, or source files open.
Close any unnecessary applications, documents, or source files that are open.
- You have many device drivers loaded.
Eliminate unnecessary device drivers.

Path <path> does not exist.

This message has the following cause and solution:

- You have attempted to open a file for which the specified path does not exist.
Check the path and try re-typing it again. Make sure you are on the computer to which the path applies.

You cannot change the Admin name or access privileges.

This message has the following cause and solution:

- You have attempted to change the Admin user's name or access privileges.
The Admin user's name and access rights cannot be changed. Select another user on which to use this command.

Cannot edit the Admin user.

This message has the following cause and solution:

- You attempted to edit the Admin user's access rights.
The Admin user's name and access rights cannot be changed. Select another user on which to use this command.

The editor '<editor>' was not found.

This message has the following cause and solution:

- You have attempted to view or edit a file for which the specified editor could not be found. (Editor associations to file extensions are set on the General Options tab.)
Click Yes if you would like to specify a different editor. Click No to cancel the editing operation.

Editor path is a required field when prompting by editor is selected.

This message has the following cause and solution:

- When you check the Use Editor to Prompt for Comments checkbox, you must specify the editor to use.
Set the path in the Editor field of the Command-Line Options tab of the Tools Options dialog box.

The folder '<folder>' already exists.

Each item in a Visual SourceSafe project must have a unique name. This error has the following cause and solution:

- You have attempted to name a file or an immediate **child project** with the same name as the project. (An item with that name can exist in other projects in the database, including subprojects.)
- You are using both a 16-bit version and a 32-bit version of Visual SourceSafe for the same project and filenames are not being mapped accurately.
To resolve this error, give the new item a different, unique filename. Or, you can delete or rename the existing item.

Note File and project names in Visual SourceSafe are not case-sensitive. For example, if you have a file named TEST.C, you cannot create a subproject or file named "Test.c" or "test.c" in the same project.

There are no items in the list.

This error has the following cause and solution:

- You specified a history operation where no items met the specified criteria.

Examine the check boxes in the history options dialog box, and make sure they are set correctly.

Label comment is invalid when no label specified.

This error has the following cause and solution:

- You specified a comment for a label but no label was supplied.

If you want to label a file or project, and wish to include a comment, you must specify the label as well as the comment.

This command cannot complete without a specified working folder.

This error has the following cause and solution:

- You choose a command that requires a working folder, without having set a working folder.

Choose OK to set a working folder for the current project now, or Cancel to cancel the command. You can also set a working folder at any time by selecting the project and then choosing the Set Working Folder command from the File menu.

No application is associated with this file type.

This error has the following cause and solution:

- You choose to view a file for which a file association doesn't exist. Without this association, SourceSafe doesn't know which editor to use to display the file.

Use File Manager to add an association from this file to the appropriate editor, then try your command again.

If you want to override the File Manager association for a specific file type, you can enter specific associations in your SS.INI file, for example:

```
.C = C:\VC\VC.EXE  
.FRM = C:\WINDOWS\notepad.exe
```

Set <folder> as your personal working folder for the project <project>?

This message has the following cause and solution:

- You need to establish a working folder for all of your projects. This message is giving you the option to create the working folder immediately, rather than having to create it later using the Set Working Folder command on the File menu.

Click Yes to set the working folder for this project. Click No to skip this step for now.

Rollback is not allowed before a pinned version.

This message has the following cause and solution:

- You tried to roll back a file to a version before the one it is pinned to.

A pinned file cannot be rolled back to a version previous to the version it is pinned to. To rollback the file, unpin it first, then do the rollback.

<file> is an Explorer link file.

This message has the following cause and solution:

- You dragged a .LNK file into Visual SourceSafe.

When you drag a .LNK file into Visual SourceSafe, SourceSafe needs to determine if you want the .LNK file itself added to the database, or the file the .LNK file points to. If you want to add the file referred to by the .LNK file, click Yes. Otherwise, click No.

Could not find the file referred to by shortcut <file>.

This message has the following cause and solution:

- The file referred to by this .LNK file could not be found.

Check that the file has not been moved or deleted. Click OK to accept and retry.

File <file> was checked out to <folder>, not to <folder>.

This message has the following cause and solution:

- The file you are checking in was checked out in a different folder.

When dragging a file to check in, you must check it in from the folder where it was checked out. The file you dragged will not be checked in. If you want to check the file in from this folder, use the Visual SourceSafe Check In command from the SourceSafe menu.

<type> is not a valid mask.

This message has the following cause and solution:

- You specified an invalid mask in the Tools Options File Types tab.

You must specify a * or ? character in your file mask, and you cannot have any illegal characters such as (,) [] in the mask.

Retype the file mask using legal characters.

All conflicts have been resolved, would you like to save the file?

This error has the following cause and solution:

- You have successfully resolved all merge conflicts.

Choose Yes to save the file, or choose No to discard your changes.

Some conflicts have not been resolved, would you like to save the file anyway?

This error has the following cause and solution:

- Not all the conflicts in the merged file have been resolved.

Choose Yes to save the unresolved file, or no to discard your changes and exit.

Cannot difference the specified item <item>.

This error has the following cause and solution:

At least one of the items you selected to compare with another item cannot be differenced by Visual SourceSafe.

No deployment path has been setup for <project>.

This error has the following cause and solution:

- You tried to Deploy a project without specifying a deployment path for the project in the Administrator program.

Ask your Visual SourceSafe administrator to set up a deployment path for your web project.

The address <address> is not a valid FTP address.

This error has the following cause and solution:

- You specified an invalid FTP address.

Re-type the address, checking for misspellings and correct addressing.

Unable to open a connection to the FTP host <host>.

This error has the following cause and solution:

- Visual SourceSafe was unable to open a connection to the specified FTP host.

Check your network connections, and check to be sure you entered the correct syntax for your FTP host. This error could also be caused if your FTP host is not available at this time.

Unable to deploy <project> to the FTP host <host>.

This error has the following cause and solution:

- Visual SourceSafe was unable to open a connection to the specified FTP host and deploy your project.

Check your network connections, and check to be sure you entered the correct syntax for your FTP host. This error could also be caused if your FTP host is not available at this time.

No differences were found for project <project>.

This error has the following cause and solution:

- You compared two items with no differences.

This is an informational message to tell you that no differences were found between the compared items.

WARNING: If you do not use a password, others will be able to log into this SourceSafe account. Proceed with no password?

This error has the following cause and solution:

- You attempted to log on without specifying a password.

This is an informational warning to make sure you want to proceed without setting a password on your account. Choose Yes to log on, or No to cancel.

Unable to launch the web browser for page <file>.

This error has the following cause and solution:

- Visual SourceSafe was unable to start your web browser.

Check that your web browser is functioning normally. You may need to re-install the browser to verify your registry settings.

Remove <database> from your Database list?

This error has the following cause and solution:

- You are deleting a Visual SourceSafe database entry from your Open SourceSafe Database list.

Choose Yes to delete the specified entry, or No to leave the database entry on the list.

<database> is already in your database list. Would you like to rename it?

This error has the following cause and solution:

- You are renaming a Visual SourceSafe database entry in your Open SourceSafe Database list. Choose Yes to rename the specified entry, or No to leave the database entry as it is on the list.

The Database <database> does not exist, would you like to search for it?

This error has the following cause and solution:

- You are trying to find a database that does not exist.

Choose Yes to browse for the correct Visual SourceSafe database, or No to cancel the operation.

You must enter a filename with no path for the Site map.

This error has the following cause and solution:

- You tried to specify a file path for the site map file.

The site map file specification does not accept a path. Enter just the filename, such as SITEMAP.HTM.

The Database <database> does not exist. Please select another.

This error has the following cause and solution:

- You are trying to find a database that does not exist.

Choose the Browse button to look for the correct Visual SourceSafe database, or No to cancel the operation.

SourceSafe cannot find a default database. Would you like to select one?

This error has the following cause and solution:

- Visual SourceSafe is trying to find a default database that does not exist.

Choose Yes to browse for the correct Visual SourceSafe database, or No to cancel the operation.

Could not login to <ftp site>.

This error has the following cause and solution:

- You cannot log in to the specified ftp site.

You may be using an invalid address, invalid username, invalid password, or perhaps the site is just not responding. Make sure you have the right ftp address. You can enter the ftp address using a web browser and see if the address is valid.

Could not Deploy <project> to <host>.

This error has the following cause and solution:

- Visual SourceSafe was unable to open a connection to the specified FTP host and deploy your project.

Check your network connections, and check to be sure you entered the correct syntax for your FTP host. This error could also be caused if your FTP host is not available at this time.

This message can also mean that you are out of disk space. It could also mean that you don't have permission to write to the location although you may have read permission.

The URL for project %s was not set properly.

This error has the following cause and solution:

- You set a bad URL path for a web project, or the URL uses a syntax that Visual SourceSafe does not understand (for instance, it might use `www.microsoft.com/...` instead of [http://www.microsoft.com/...](http://www.microsoft.com/))

Use the Administrator program to set a proper URL.

An error occurred while checking hyperlinks for <file>.

This error has the following cause and solution:

- The working folder path, Visual SourceSafe specification, or URL was too long..

Check that you aren't exceeding path limits for folder, specification, or URL. Check that htmlfilt.dll is still present on your system and registered properly (re-running the client installation should fix this problem).

A link in <file> was ignored because it was too long.

This error has the following cause and solution:

- The URL is too long for the internal buffer.

Use a shorter URL.

No HTML files were available in this project to check hyperlinks.

This error has the following cause and solution:

- This informational message tells you that no files were checked for web hyperlinks because no HTML files were found.

If you expected HTML files in this project, check that they were added properly and that you entered the correct project name..

The base URL for this web project, '<project>', has bad URL syntax.

This error has the following cause and solution:

- The URL that you entered for the base URL for a Visual SourceSafe web project has bad URL syntax.

Valid syntax for the URL is shown in these examples:

```
http://www.microsoft.com/default.htm
```

```
http://www.microsoft.com
```

You can change the URL in the Web Projects tab of the Visual SourceSafe Administrator Tools options dialog box, or edit the URL in the SRCSAFE.INI file.

Unable to load the HTML parser.

This error has the following cause and solution:

- You have a corrupted Visual SourceSafe installation. Either the DLL is gone or the registry isn't set to the right DLL.

Re-install Visual SourceSafe.

This command requires internet components which are not installed on your machine.

This error has the following cause and solution:

- You tried to use the Check Web Links command on a RISC platform, but there wasn't a WININET.DLL to distribute when we shipped Visual SourceSafe. You can get this required DLL when a RISC version of the browser exists, or
- You are missing the correct version of WININET.DLL on your Intel or Alpha computer. Check to see if the file has been moved or renamed.

To make sure you have the latest internet components, go to <http://www.microsoft.com/ie/> and download Microsoft Internet Explorer 3.0 or higher. Then restart Visual SourceSafe to enable this command.

Could not display this URL.

This error has the following cause and solution:

- You tried to check an external web link, but there was no web browser we could find in the register to handle a http: path; or, the link syntax was incorrect for ShellExecute (this could happen if you tried to check a virtual root by double clicking on it instead of setting the virtual root).

Install Internet Explorer 3.0 (or later).

Warning: the Rollback command will branch this file, breaking its link to other projects that share it.

This warning has the following cause and solution:

- This warning occurs when you roll back a file that has been branched or is pinned in another project.

Choose Yes to continue, or No to cancel.

Get on Checked Out File Dialog

This message has the following cause and solution:

- You chose to get a file when it was already checked out.

When you get a file, a new copy is written to your working folder. If you already have the file checked out, you may have made changes that will be overwritten. In this situation, Visual SourceSafe gives you some options:

- Choose replace to overwrite the working folder file.
- Choose Merge to merge changes from the working folder file into the Visual SourceSafe database file.
- Choose Leave to leave the working folder file alone (the Get command will be ignored for that file).
- Choose Cancel to cancel the get operation.
- Choose Apply to all items to have your choices in this dialog apply to all other files you currently have checked out.

The Admin user currently has no SourceSafe password.

This message has the following cause and solution:

- You have not assigned a password for the administrator user of the Administrator program.

Assign a password to the Admin user by choosing the Users Change Password command. If you don't do this, any user can log in as Administrator and make changes to your Visual SourceSafe database.

You are about to remove the Admin user's SourceSafe password, which will allow any user to run the Administrator's program.

This message has the following cause and solution:

- You are deleting the Admin user's password.

Leaving the Admin user with no password is a security risk, because then any user can log into the Administrator program and make changes to your Visual SourceSafe database. Are you sure you want to remove the Admin user's SourceSafe password? Choose Yes to continue, or No to cancel the operation.

The URL <URL> does not have proper HTTP URL syntax.

This message has the following cause and solution:

- The URL that you entered has bad URL syntax.

Valid syntax for the URL is shown in these examples:

`http://www.microsoft.com/default.htm`

`http://www.microsoft.com`

The deploy path <path> is not a proper file system or FTP deploy path.

This message has the following cause and solution:

- One of the paths entered for deploying the project is either not a proper operating system file system path or is not a proper FTP URL.

ftp paths can look like one of the following.

```
ftp://user:user@www.microsoft.com/ssafe  
ftp://user@www.microsoft.com/ssafe  
ftp://www.microsoft.com/ssafe
```

They always start with "ftp://". They always end with the ftp address (server name and, optionally, directory path). In between, you may have nothing (as in the bottom example above). Or, you may have a username (as in the middle), or a username and password (as on the top).

Please enter a number between 0 and 32767.

This message has the following cause and solution:

- You have not entered a number of context lines between 0 and 32767.

Context lines are the lines immediately preceding and following a difference. Showing these lines in a difference display helps you understand the context of the change or difference. If you have checked the Show Context box in the Advanced options of the Show Difference command, you need to specify the number of preceding and following lines to show.

Command Summary

- A
- B
- C
- D
- E
- F
- G
- H
- I
- J
- K
- L
- M
- N
- O
- P
- Q
- R
- S
- T
- U
- V
- W
- X
- Y
- Z

A – C

[About Visual SourceSafe](#)

[Add Files](#)

[Books Online](#)

[Branch](#)

[Cancel Search](#)

[Change Password](#)

[Check In](#)

[Check Out](#)

[Check Hyperlinks](#)

[Create Project](#)

[Create Site Map](#)

[Customize Toolbar](#)

D – E

[Delete](#)

[Deploy](#)

[Destroy](#)

[Edit File](#)

[Exit](#)

F

[Files Report](#)

[Find in Files](#)

[Font](#)

G – I

[Get Latest Version](#)

[Help](#)

[Invert Selection](#)

J – L

[Label](#)

[Links](#)

M – O

[Merge Branches](#)

[Microsoft on the Web](#)

[Move Project](#)

[Open SourceSafe Database](#)

[Options](#)

P – Q

[Paths](#)

[Pin](#)

[Properties](#)

[Purge](#)

R

[Recover](#)

[Refresh File List](#)

Rename

Rollback

S

Search

Search for Help on

Select All

Select

Set Working Folder

Share

Show Differences (File)

Show Differences (Project)

Show History

Sort

T – U

Technical Support

Unpin

Undo Check Out

V – Z

View File

[Destroy](#)

[Difference \(on a file\)](#)

[Difference \(on a project\)](#)

[Folder](#)

E – G

[Filetype](#)

[FindinFiles](#)

[Get Latest Version](#)

H – K

[Help](#)

[History](#)

L

[Label](#)

[Locate](#)

[Links](#)

M – O

[Merge](#)

[Move](#)

P – Q

[Password](#)

[Paths](#)

[Pin](#)

[Project](#)

[Properties](#)

[Purge](#)

R

[Recover](#)

[Rename](#)

[Rollback](#)

S

[Share](#)

[Status](#)

T–U

[Undocheckout](#)

[Unpin](#)

V–Z

[View](#)

[Whoami](#)

[WorkFold](#)

[YNC](#)

Common Command-Line Options

-? Command-Line Option

-C Command-Line Option

-G Command-Line Option

-I Command-Line Option

-N Command-Line Option

-O Command-Line Option

-R Command-Line Option

-S Command-Line Option

-V Command-Line Option

-W Command-Line Option

-Y Command-Line Option

Diff_Ignore

Dir

Double_Click

E

Editor

EOL

Expand_Keywords_Locally

Extension_Association

F

File_Types

Filetype

Force_Dir

Force_Prj

G–K

History_Include_Files

IMG_File

IMG_Folder

Keep_Checkedout

L

Long_Filenames

M

Merge_Ext

Merge_Ignore

N–Q

Output

Project

R

Reconcile_Project

Recursive

Refresh_Display

Refresh_Interval

Relevant_Masks

Replace_Writable

OutputWindow

Reuse_Comment

S

Save_Settings

SetTime

Skip

Smart_Mode

Status_Bar

Store_Deltas

T

Temp_Path

Toolbar

Toolbar_Size

ToolTips

U

Update_No_Change

Use_ReadOnly

V

Visual_Merge

W-Z

Warn_Destroy

Warn_Exit

Warn_Multiple_Checkout

Warn_Purge

Warn_Remove

Warn_Rollback

Warn_UnCheckout

Copyright © 1992-1996 Microsoft Corp. All rights reserved.

Microsoft, MS, and MS-DOS are registered trademarks, and Windows and the Windows logo are trademarks of Microsoft Corporation.

Information in this document is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without express written permission of Microsoft Corporation.

The software and/or databases described in this document are furnished under a license agreement or nondisclosure agreement. The software and/or databases may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software except as specifically allowed in the license or nondisclosure agreement. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the purchaser's personal use, without the express written permission of Microsoft Corporation.



Properties Command (File Menu)

See Also

Command Line

Displays the Properties dialog box, which contains basic information about a Visual SourceSafe file or **project**, including the size, name, and type.

Access Rights

You must have the Read **access right** to view the Properties dialog box.

You must have the Add access right to change the type of a file.

You must have the Destroy access right to change the Store Only Latest Version state of a file.

Remarks

Click the tab with the information you want. Tab content differs according to whether a file or project is currently selected in **Visual SourceSafe Explorer**.

Note Click the Report button on any of the tabs to create a report to send to the **Clipboard**, a text file, or a printer.

File Properties Dialog Box Tabs

General

Shows the filename, type, size, latest **version number**, last modification date, comment, and whether the entire file history is stored in the SourceSafe database (or just the latest version). You can change the comment and set whether or not to store only the latest version of the file to save disk space. You can also change the file type here.

Check Out Status

Shows the file's name, who it is checked out by (if anyone), whether it is exclusively checked out, when it was checked out, the current version, the check out path (with the computer name), the project the file is in, and the check out comment. You can change the check out comment.

Links

Shows the current **share links** a file has to other projects.

Paths

Shows which projects have **branched** versions of this file in the past.

Project Properties Dialog Box Tabs

General

Shows the name of the **current project**, the number of files and **subprojects** it contains, the latest version number, when it was last changed, whether the project is cloaked, and the comment for the project. You can change the comment.

Deleted Items

Shows the files that have been deleted from the project since it was added to Visual SourceSafe. Use this dialog box to recover or **purge** a deleted file.

See Also

[Check Out Status Properties Tab](#)

[Deleted Items Properties Tab](#)

[General File Properties Tab](#)

[General Project Properties Tab](#)

[Links Properties Tab](#)

[Paths Properties Tab](#)

Search Command (View Menu)

Command Line

Filters the **file list** in **Visual SourceSafe Explorer** to search for **checked-out files** whose names match a specified pattern or string. Search finds one or more specified files in your **project list** and limits the file list display to those files.

Access Rights

You must have the Read **access right** to use this command.

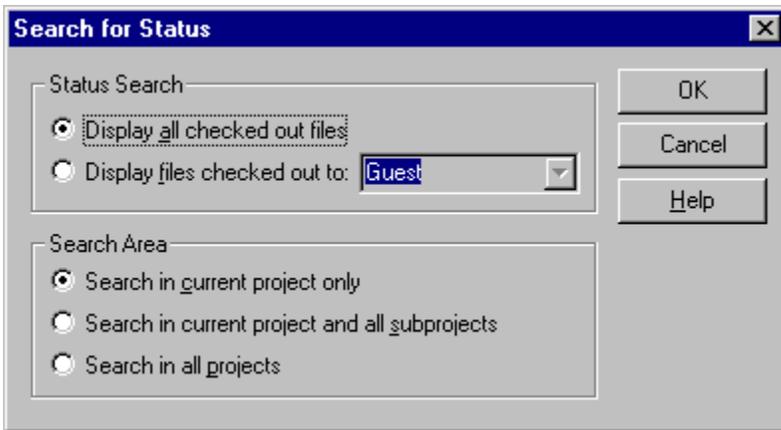
Remarks

A status search can be used to find just the files checked out by a specific user. When a file is shared among multiple projects, the Search command shows you which projects the file is checked out to.

Tip Press SHIFT+CTRL+S to search for all checked-out files in the **current project**. The Search dialog box is not displayed.



Search for Status Dialog Box Options



Display all checked out files

Shows all checked out files, no matter who has them checked out.

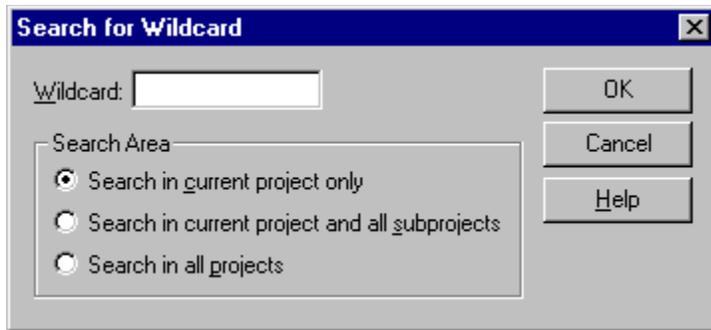
Display files checked out to

Select this button and type a username if you want to see files checked out only by that user, instead of all checked-out files. You can also select user names from the drop-down list. To see all checked out files, select <All Users>.

Search Area

Specify which projects/subprojects you want to be included in the search. You can search the current project only, the current project and all its subprojects, or all projects in the database.

Search for Wildcard Dialog Box Options



Wildcard

Type a filename or string with wildcard characters. For example, **file.***.

Search Area

Specify which projects/subprojects you want to be included in the search. You can search the current project only, the current project and all its subprojects, or all projects in the database.

Note If you clear the Wildcard boxes simultaneously, the search is cancelled. Also, if you enter * or *.* in the Wildcard box, and the Search in all projects box is clear, the search is cancelled.

Remember that the SourceSafe wildcard search is really a filtering operation. You are searching for those files in the current project that match specified criteria. Entering *.* in the Wildcard box effectively removes any filtering criteria; therefore the search is cancelled.

About Visual SourceSafe (Help Menu)

Command Line

Displays a dialog box with information about your copy of Microsoft Visual SourceSafe, including the version number, the platform type, and the copyright, legal, and licensing notices.

Click OK to close the dialog box and return to Microsoft Visual SourceSafe.

Access Rights

Any Visual SourceSafe user can use this command.

Cancel Search Command (View Menu)

See Also

Restores the full file list in Visual SourceSafe Explorer for the current project. This command is used after a search operation, which usually filters the file list to a smaller subset.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

When you execute the Search command, Visual SourceSafe displays the word Search in the **status bar**. Click the Cancel Search command to return to the full **Visual SourceSafe Explorer** file display. The word Search disappears from the status bar.

See Also

Search Command

Exit Command (File Menu)

Exits Visual SourceSafe.

Access Rights

Any Visual SourceSafe user can use this command.

Files Report Command (Tools Menu)

Creates a formatted report of **current project** and file information that you can send to the **Clipboard**, a file, or a printer.

Access Rights

You must have the Read **access right** to use this command.

Remarks

You can prepare a report containing a list of all files in the current project. (The report includes only files and subprojects currently in the project, not deleted files or subprojects.) The Files Report command is useful for large projects for which the **Visual SourceSafe Explorer** may not be able to display all filenames on the screen at once.

Note Dialog box options vary depending on whether you have a file or project selected in Visual SourceSafe Explorer.

Dialog Box Options

All Displayed Files (available only when a file is selected)

Includes all files shown in the **file pane** of Visual SourceSafe Explorer.

Selected Files (available only when a file is selected)

Creates a report for selected files only.

Recursive (available only when a project is selected)

Creates a report that includes information on the subprojects in the current project.

Include Files (available only when a project is selected)

Creates a report that includes filenames within a project.

Filenames Only

Creates a report that includes filenames only. If this box is not checked, the file report includes four columns instead of just one for filename. The four columns are Filename, Username, Date-Time, and Check Out Folder (as in the SourceSafe Explorer). The filename column supports long filenames, which will wrap in the column as necessary.

Report To

- Printer—Sends report to printer.
- File—Sends report to File dialog box.
- Clipboard—Places report on Clipboard.

OK

Confirms selection.

Cancel

Cancels the report.

Preview

Previews the report.

Invert Selection Command (Edit Menu)

Selects all files not currently selected, and deselects currently selected files.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

The Invert Selection command reverses the selection status of each file currently displayed in the **Visual SourceSafe Explorer file list**. The command can be useful when you have performed one operation on all the selected files, and want to perform another operation on the remaining files.

Options Command (Tools Menu)

See Also

Enables you to customize your Visual SourceSafe environment.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

Click the tab with the information you want. Each tab contains groups of options you can set to customize Visual SourceSafe.

Dialog Box Tabs

General

Enables you to set general options applicable to a number of commands in Visual SourceSafe.

Local Files

Configures **working folder** files and how they are treated during the operation of different commands.

View

Sets viewing options, such as whether or not to show the toolbar or **status bar**, and **file list** options.

Difference

Allows you to specify options for viewing differences between files and projects. These options are used in other Visual SourceSafe dialog boxes such as the Project Difference dialog box and the Visual Merge dialog box.

Command Dialogs

Configures the display of intermediate dialog boxes related to several Visual SourceSafe commands. These intermediate dialogs usually present you with additional options.

Warnings

Determines when warning messages are displayed.

File Types

Sets how file type information is determined.

Command Line

Customizes how some command-line commands work.

See Also

[Command Dialogs Options Tab](#)

[Command Line Options Tab](#)

[Difference Options Tab](#)

[File Types Options Tab](#)

[General Options Tab](#)

[Local Files Options Tab](#)

[View Options Tab](#)

[Warnings Options Tab](#)

Select Command (Edit Menu)

See Also

Selects one or more files to perform operations on. The file pane of Visual SourceSafe Explorer must be active for this command to be available.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

The Select command opens a dialog box into which you can type the name of a single file, multiple filenames separated by semi-colons(;), or the names of multiple files using wildcard characters.

Dialog Box Options

Item

Type the name of a single file or the names of multiple files. To select all files in the file list, type *.*; to select all files with the .RTF extension, type *.RTF, and so on. Use the semi-colon (;) as a delimiter between file types you want included in the selection, for example "*.RTF; *.EXE".

Select

Selects the file.

Deselect

Clears the selection of a file.

OK

Confirms the selection.

Cancel

Cancels the operation, closes the dialog box, and returns to Visual SourceSafe Explorer.

See Also

Invert Selection Command

Select All Command

Select All Command (Edit Menu)

See Also

Selects all files currently displayed in the **file pane** of **Visual SourceSafe Explorer** so that you can perform a group operation on all the files.

Access Rights

Any Visual SourceSafe user can use this command.

See Also

Invert Selection Command

Select Command

Refresh File List Command (View Menu)

See Also



Refreshes the **file list** in **Visual SourceSafe Explorer**.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

Refreshes the file list only, not the **project list**. To refresh both lists, collapse the project list to a **parent project**, then double-click that project to expand the project list.

You can refresh the file list automatically. Set the timing with the Automatically Refresh File List option on the View tab in the SourceSafe Options dialog box (Tools menu).

See Also

[View Options Tab](#)

Sort Command (View Menu)

Sorts the **file list** in **Visual SourceSafe Explorer**. Sort order options are by filename, type, user, date, or check out folder.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

The file list in Visual SourceSafe Explorer is usually sorted by filename. You can change the sort order:

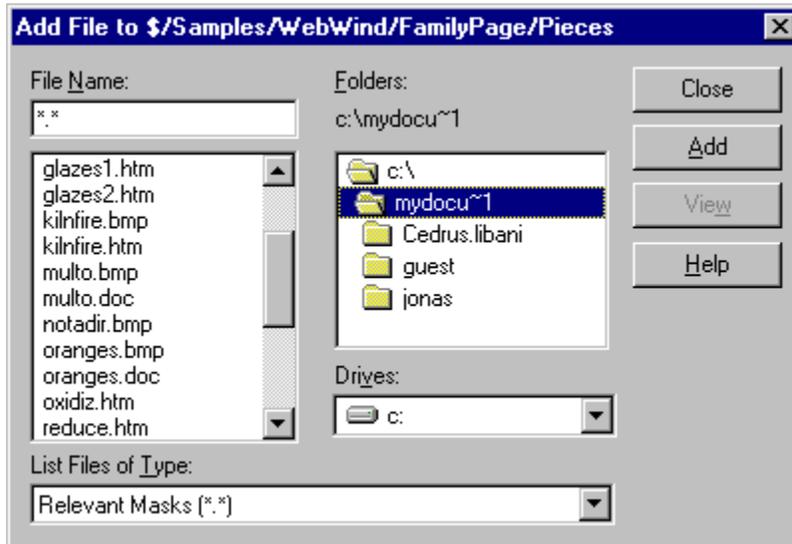
- Name (filename)
- Type (file type extension)
- User (**usernames** of users who have files checked out)
- Date–Time
- **Check Out Folder**

Type, Name, User, and Check Out Folder sort in ascending order; Date–Time sorts in descending order, so that the most recent files are on top. The current sort order is always displayed in the **status bar**.

Add Files Command (File Menu)

See Also

Command Line



Copies a file, or group of files, into the current Visual SourceSafe **project**. In the Add Files dialog box, you can specify a file on any drive and folder, and that file is copied into the current Visual SourceSafe project.

Access Rights

You must have the Add **access right** to use this command.

Remarks

You can use the Purge command to undo an Add Files operation, but it is not recommended because it does not retain a recoverable copy of the file in the **Visual SourceSafe database**. Instead, consider using the Delete command, because it retains a recoverable copy of the file.

Dialog Box Options

File Name

Select or type the name of a file you want to open. This box lists items with the filename extension selected in the List Files of Type box.

List Files of Type

Lists the file type, or tells Visual SourceSafe how to classify a file.

The list of file types available is determined by the File Types Included In File Group filter and the File Groups filter set on the File Types tab in the SourceSafe Options dialog box. All Files (*.*) lists all files in the current folder.

Folders

Specifies the folder that contains the file you want to add.

Drives

Selects the drive that contains the file you want to add.

Close

Closes the dialog box without adding the file to the project.

Add

Adds to the project the selected file or folder containing a group of files. When you click Add, Visual SourceSafe displays a comment box, in which you can enter any comments. Specify the Apply same comment for all check box to use the same comment for all the files you are adding. Click Advanced for more options.

View

Displays the selected file with the View File command.

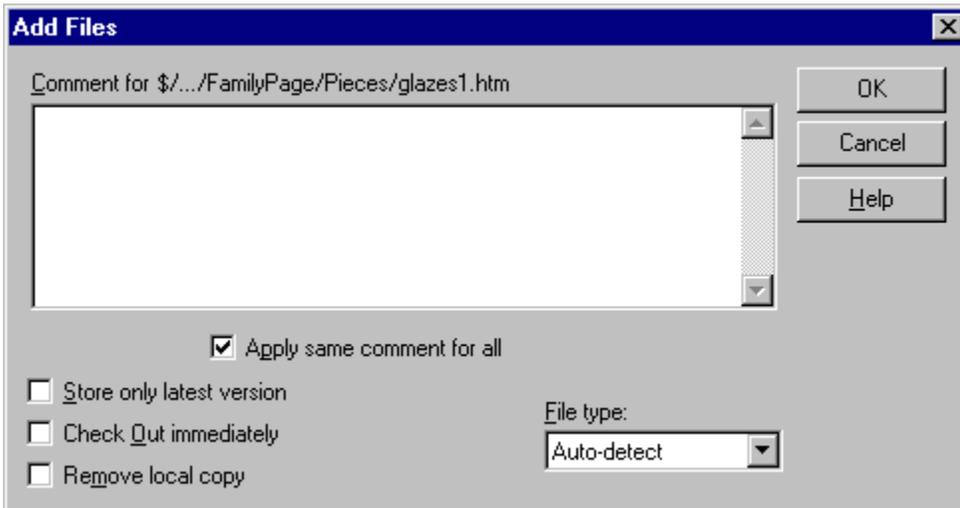
Apply same comment for all

Uses the same comment for all files involved in the Add Files operation. If this check box is clear, you can enter a separate comment for each file. The default is checked (your comment is applied to all files selected for this operation.)

Recursive

Indicates you want files in all subfolders added when you select an entire folder to add. Available only when you select a folder and then click Add.

Advanced Options



Store Only Latest Version

Indicates you do not want to save old versions of the file(s) being added to the Visual SourceSafe database. Visual SourceSafe keeps a file **history** of change records with comments. However, the older versions of the file are not stored, and you cannot recover them.

Note To increase performance and save disk space, use this option when storing executable programs in Visual SourceSafe.

Check Out Immediately

Checks out the file to you immediately after adding it to Visual SourceSafe.

Remove Local Copy

Deletes the file from your working folder after adding it to Visual SourceSafe.

File Type

Select an option from the list box:

- AutoDetect—Visual SourceSafe determines the file type.
- Binary—Visual SourceSafe treats the file as a binary file.
- Text—Visual SourceSafe treats the file as a text file.

Note When you are adding a folder, the File Types field is ignored. However, the files added during a folder Add Files operation are filtered by the Relevant_Masks setting on the File Types tab in the SourceSafe Options dialog box.

See Also

[Adding Files to the Visual SourceSafe Database](#)

[Check Out Command](#)

[Delete Command](#)

[Purge Command](#)

[Setting the File Type: Text and Binary Files](#)

[View File Command](#)

[File Types Options Tab](#)



Branch Command (SourceSafe Menu)

See Also

Command Line

Breaks a **share link** while retaining a copy of the file.

Note In earlier versions of Visual SourceSafe, this command was called Separate.

Access Rights

You must have the Add **access right** to use this command.

Remarks

A single file can be shared by multiple **projects**. Any change made to that file is reflected in all the projects that share it so that, for example, the propagation of bug fixes is handled by Visual SourceSafe.

After you apply the Branch command to the file, two separate **branches** of the file exist in different projects. You can track the different branches on the Paths tab in the Properties dialog box, and bring them back together with the Merge Branches command.

Dialog Box Options

Comment

Type a comment describing the **branching**.

OK

Confirms the Branch command.

Cancel

Cancels the Branch command.

See Also

[Branching Files and Projects](#)

[Merge Branches Command](#)

[Paths Properties Tab](#)

[Sharing Files](#)

Change Password Command (Tools Menu)

See Also [Command Line](#)

Sets the **password** for the current user.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

Use the Change Password command to set your password. After you change it, your old password no longer works, and you have to type the new one each time you **log on** to Visual SourceSafe.

To change your password, you must first know what it is. If you forget your password, ask your **Visual SourceSafe administrator** to change it for you.

Note Keep in mind that the Visual SourceSafe password is not meant to replace or augment your operating system or network operating system password. Only when the operating system does not allow Visual SourceSafe to determine user identity does SourceSafe ask for the SourceSafe password when you log on. Your Visual SourceSafe password can be the same as or different from your operating system password.

Dialog Box Options

Old Password

Type the current password, or leave blank if the password is blank (a zero-length string).

New Password

Type the new password.

A password can consist of any combination of 1 to 15 valid characters. As you type, the characters are displayed as asterisks. Microsoft Visual SourceSafe passwords are not case-sensitive.

Verify

Retype the new password to verify it.

See Also

[Changing Your Visual SourceSafe Password](#)

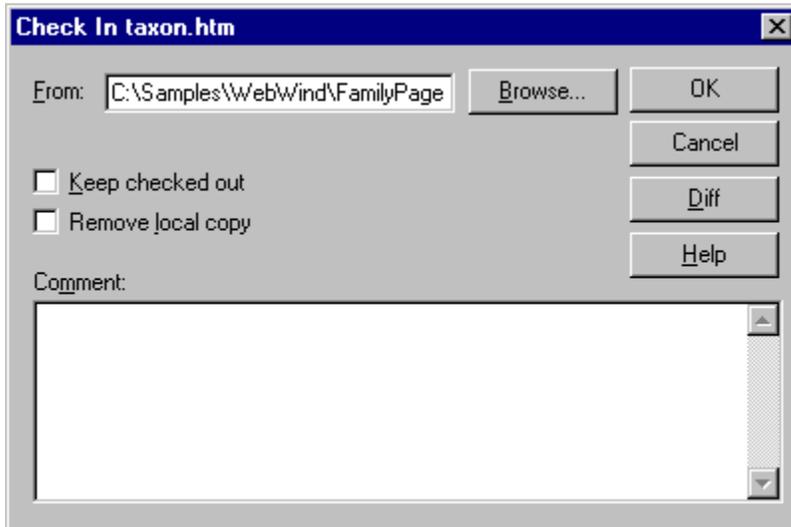
[Visual SourceSafe Naming Syntax and Conventions](#)



Check In Command (SourceSafe Menu)

See Also

Command Line



Updates Visual SourceSafe with changes made to a **checked-out file**, and unlocks the Visual SourceSafe **master copy**.

Note In earlier versions of Visual SourceSafe, this command was called Update.

Access Rights

You must have the Check Out **access right** to use this command.

Remarks

There are two commands you can use to end the check out of a file:

- The Check In command copies your changes into the Visual **SourceSafe database**, creating a new version of the file. When other users use the Get Latest Version command or check out the file, they receive all the modifications you have made.
- The Undo Check Out command voids all your changes and undoes your check out of the file.

When you check in a file, Visual SourceSafe provides a box in which you can enter a comment.

Note To update the master copy of a file while keeping the file checked out, click the Keep Checked Out check box. You can make this the default behavior by selecting the Always Keep Files Checked Out check box on the General tab in the SourceSafe Options dialog box.

You can check in a file that has not changed at all, but still add the unchanged file as a new version to the file **history**. Visual SourceSafe prompts you in this case to see if you want to perform a check-in operation. If you click No, Visual SourceSafe executes the Undo Check Out command, and no new version is added to the file list.

Note To avoid this message on future occasions, you can disable this notification on the General Tab in the SourceSafe Options dialog box on the Tools menu.

Check In from Multiple Check Outs

In a **multiple checkout** situation, when you are the first user to check the file into Visual SourceSafe, your version of the file becomes the latest version. However, if you are a subsequent user to check in the file, Visual SourceSafe performs an **automatic merge**, creating a new file as the latest version in Visual SourceSafe.

Visual SourceSafe informs you that a merge took place. You can check the file in immediately or keep the merged file in your **working folder** and review it before returning it to Visual SourceSafe. If there are **merge conflicts**, you cannot check in the file until you resolve the conflicts.

Dialog Box Options

Comment

Type an optional comment. Comments are limited to 4095 characters.

Keep Checked Out

Updates the master copy with your changes but keeps the file checked out to you.

Remove Local Copy

Removes the local copy from your working folder.

OK

Confirms the selection.

Cancel

Cancels the selection.

Diff

Compares the file you want to check in with the Visual SourceSafe master copy and displays the differences. This feature is useful if you want to enter a meaningful comment when you Check In the file.

Recursive (project-level Check In only)

Indicates you want files in all subfolders added when you select an entire project to check in.

Override Working Folders (project-level Check In only)

Overrides the working folder set for the subprojects involved in a recursive operation. This is useful if you have some shared files checked out in more than one project, and want to check them all in at the same time. Available only when you check the Recursive check box.

See Also

[Check Out Command](#)

[General Options Tab](#)

[Get Latest Version Command](#)

[Getting, Checking In, Checking Out, and Viewing Files](#)

[Tracking and Labeling Versions of Files and Projects](#)

[Undo Check Out Command](#)

[The Visual Merge](#)



Check Out Command (SourceSafe Menu)

[See Also](#)

[Command Line](#)

Check Out taxon.htm

Comment:

I to: Browse...

OK Cancel Help

Don't get local copy

Allow multiple checkouts

Replace writable: Set file time: Skip this dialog

Copies the latest version of one or more selected files from the **current project** into your current **working folder**.

Access Rights

You must have the Check Out **access right** to use this command. You should have a working folder set before using this command.

Remarks

The Check Out command retrieves writable files from Visual SourceSafe and places them in your working folder, so that you can modify them. Visual SourceSafe lists the file as checked out in **Visual SourceSafe Explorer**.

Note There is an important difference between the Check Out and Get Latest Version commands. The Get Latest Version command places a read-only local copy of a file in your working folder. Check Out places a writable copy of the file in your working folder.

To return the file to Visual SourceSafe, you can use the Check In command to confirm your changes and copy your local, modified file into the **Visual SourceSafe database**.

Note You can also use the Undo Check Out command. Undo Check Out does not update Visual SourceSafe. Instead, it copies the most recent version of the file in Visual SourceSafe over your **local copy**, so all the changes made to the local copy since you last checked out the file are lost. In addition, the Undo Check Out command cancels your check out and removes the writable version of the file from your working folder. To display a dialog box when you use the Undo Check Out command, use the Command Dialogs tab in the SourceSafe Options dialog box on the Tools menu.

Generally, only one user can have a file checked out for modification at one time. However, if your **Visual SourceSafe administrator** allows multiple check outs, many users can check out the same file at the same time. When the first user checks the file into Visual SourceSafe, that file becomes the current version. When subsequent users check in their local copies, their changes are merged into the file and a new Visual SourceSafe version is created. Binary format files cannot be checked out by more than one person at a time.

You can also check items out exclusively – that is, check them out so that no one else can check them out while you have the file or project checked out. Your Visual SourceSafe administrator can set an administrative option to disable multiple checkouts for an entire project all the time, or you can clear the Allow Multiple Checkouts advanced option checkbox from the Check Out command dialog to exclusively check the file out on a file-by-file and session-by-session basis.

Dialog Box Options

If you have the file-level Check Out check box selected on the Command Dialogs tab in the Options dialog box (Tools menu), the Check Out command displays the Check Out dialog box, which has the following options. There is an additional Recursive option if the project-level Check Out check box selected as well:

Comment

Type a brief description of the reason for the check out. While the file is checked out, the Check Out Status tab displays the comment. The limit for Check Out comments is 63 characters.

To

Shows the working folder to which Visual SourceSafe copies the file. By default, Visual SourceSafe copies the file with its own filename into your working folder. However, you can enter a different folder and filename, or click Browse to select a different location in your folder tree.

Recursive (project-level Check Out only)

Checks out the project and all its subprojects.

Build Tree—Override Working Folders (project-level Check Out only)

Overrides the working folder set for the subprojects involved in a recursive operation, and builds a project tree that mirrors the project organization on your local computer. Working folder settings for individual subprojects are ignored. Available only when you check the Recursive check box.

Don't Get Local Copy

Tells Visual SourceSafe not to copy the file to your working folder. Visual SourceSafe records that the file is checked out, but does not write a new copy to your working folder.

Advanced Options

You can also set the defaults for the first three of these on the Local Files tab in the SourceSafe Options dialog box (Tools menu).

Allow Multiple Checkouts

Allows multiple users to check out this file/project simultaneously. If another user already has this file checked out, the checkbox is on, and disabled. If multiple checkouts are disabled for this project, the checkbox is off and disabled. When a file or project is exclusively checked out, the Visual SourceSafe Explorer icon for that item has a red box around it.

Replace Writable

Determines how Visual SourceSafe behaves when you try to check out a file that is already marked as writable in your working folder. Select one of the following in the list box:

- **Ask.** Asks you what to do when a writable version of a file is found.
- **Replace.** Replaces the writable file with a read-only version.
- **Skip.** Skips the check out operation for the writable file. An error is displayed in the Results window if it is open.
- **Merge.** Merges the changes between the writable file and the one you are now checking out. If the file cannot be merged (i.e., it is a binary file), the check out operation is skipped for that file, and an error is displayed in the Results window if it is open.
- **Default.** Uses defaults set on the Local Files tab or inherited from a parent project. You cannot set this option; it is used automatically when you have explicitly (in the file list) or implicitly (by

checking Recursive) selected multiple files.

Set File Time

Changes the time assigned to the check out operation. Select one of the following in the list box: current time, last file modification time, last file check-in time, or the default time (set on the Local Files tab).

Skip This Dialog

Skips this dialog box during future check out operations and, by default, uses the settings you've entered. To restore the dialog box, clear the appropriate box on the Command Dialogs tab in the SourceSafe Options dialog box. Pressing the SHIFT key when clicking the Check Out command displays this dialog box, overriding the setting on the Command Dialogs tab.

See Also

[Check In Command](#)

[Command Dialogs Options Tab](#)

[Getting, Checking In, Checking Out, and Viewing Files](#)

[Local Files Options Tab](#)

[Merging Files and Projects](#)

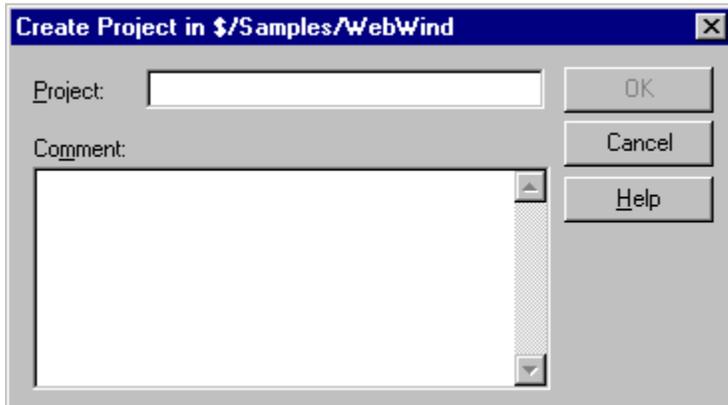
[Undo Check Out Command](#)



Create Project Command (File Menu)

See Also

Command Line



Creates a new **subproject** under the currently selected project.

Access Rights

You must have the Add **access right** in the **parent project** to use this command.

Remarks

Use the Create Project command to create a subproject under the current Visual SourceSafe project. The subproject is empty until you place files or other subprojects in it. All access rights that you have in the parent project **propagate** to the newly created project.

Dialog Box Options

Project

Type a name for the new project.

Comment

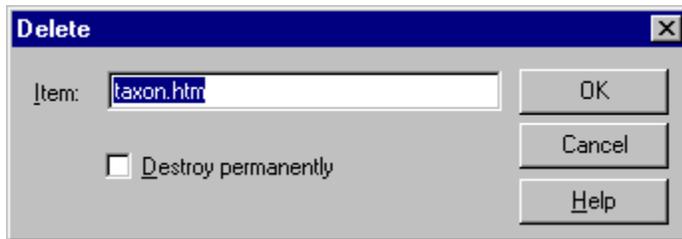
Type a brief description of the project.

See Also

[Visual SourceSafe Naming Syntax and Conventions](#)

Delete Command (File Menu)

See Also [Command Line](#)



Deletes a file or [project](#).

Note In earlier versions of Visual SourceSafe, this command was called Remove.

Access Rights

You must have the Add [access right](#) to use this command.

Remarks

If you don't check the Destroy Permanently check box in the Delete dialog box, the Delete command is reversible—to make recovery possible, Visual SourceSafe stores a deleted file or [subproject](#).

To recover the file, use the Recover command. Another way to retrieve the file (if it hasn't been purged or destroyed) is to return to an earlier version of the project that contains the file.

Note When you delete a project, all its subprojects and the files contained within them are deleted as well. When you recover the project, its subprojects (and the files therein) are recovered.

A deleted file or subproject can be purged or destroyed, which permanently removes the item.

Dialog Box Options

Item

The name of the file or subproject to be deleted.

Destroy Permanently

Clicking this check box permanently removes the file or subproject, which you cannot thereafter recover. This is the same as performing the Purge command on the items. You must have the Destroy access right to enable this check box.

Caution The Destroy command should be used with extreme caution: Before using it, you must be confident that you will never want the file or project, or any of its earlier versions, for any reason.

If there's any chance that you might want to retrieve earlier versions of the project that contained the file, delete the file without destroying it—you can then recover the file later with the Recover command.

See Also

[Destroy Command](#)

[Purge Command](#)

[Recover Command](#)

Destroy Command

See Also **Command Line**

Permanently removes a file or project.

Access Rights

You must have the Destroy **access right** to use this command.

Remarks

When you select the Destroy Permanently check box in the Delete dialog box, the file is permanently removed from the Visual SourceSafe database and cannot be recovered.

To permanently remove a deleted file at a later time, use the Purge command.

Caution The Destroy command should be used with extreme caution: Before using it, you must be confident that you will never want the file or project, or any of its earlier versions, for any reason. If there's any chance that you might want to retrieve earlier versions of the project that contained the file, delete the file without destroying it—you can then recover the file later with the Recover command.

You should use the Destroy command primarily to remove obsolete files and thereby free up disk space. However, it is important to note that Destroy does not always have this effect. If a file is being shared by more than one project, and you destroy it in one project, no disk space is freed, because the file remains in the other projects.

See Also

Delete Command

Purge Command

Recover Command

Sharing Files



Find In Files Command (Tools Menu)

[See Also](#)

[Command Line](#)

Finds all instances of a specified character string within one or more files.

Access Rights

You must have the Read [access right](#) to use this command.

Remarks

Use the Find In Files command to display a list of all occurrences of a character string in the Visual SourceSafe files you specify. You can use the command on a single file, or on an entire project.

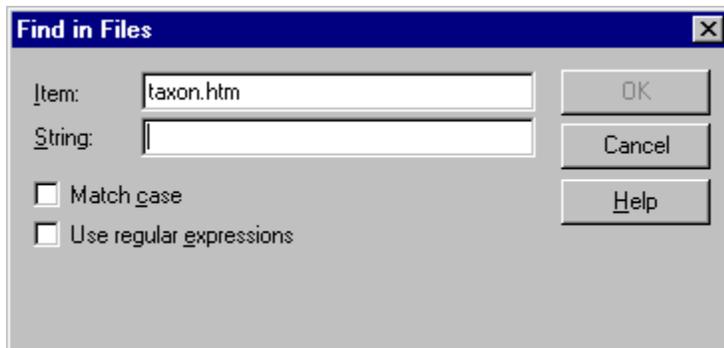
There are three [wildcard characters](#) you can use in a find string, unless Literal Matches is checked:

- A question mark matches any character.
- An asterisk matches any string.
- A backslash makes the character after it literal: *, \?, and \\. This is necessary if you want to search for asterisks, question marks, or backslashes.

Ordinarily, Visual SourceSafe is case sensitive, that is, A does not equal a. To make the Find In Files command case insensitive, click Ignore Case.

The results of the search are shown in a Results window. On top, Visual SourceSafe lists all the files that contained matches. On the bottom, Visual SourceSafe displays the matching strings in the currently selected file. As you scroll through the top list, the bottom display changes to reflect the changing file selection.

Dialog Box Options



Item

Type the filename(s) you want Visual SourceSafe to find.

String

Type the character string you want Visual SourceSafe to search for.

Ignore Case

Conducts a case-insensitive find operation.

Literal Matches

Finds exact matches only. The string you enter cannot contain wildcards.

Recursive (project level only)

Searches through all files in the project and its subprojects.

List All Shared Files Separately (project level only)

In project searches, notes each match found in a shared file as a separate item. (By default, a match in a shared file is listed only for one of the projects that the file exists in.)

See Also

Search Command

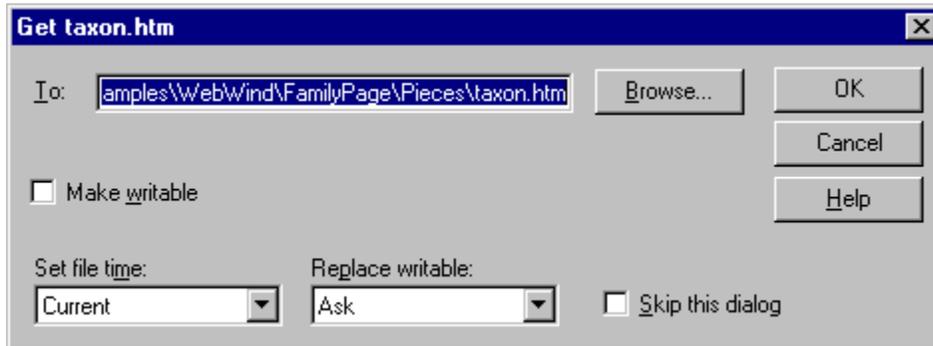
Find in Files Results Dialog Box



Get Latest Version Command (SourceSafe Menu)

[See Also](#)

[Command Line](#)



Retrieves a **read-only** copy of the selected Visual SourceSafe items. To get the most recent version, click Get Latest Version on the Visual SourceSafe menu. To get an older version, click Get in the History of File or History of Project dialog box.

Access Rights

You must have the Read **access right** to use this command. You should have a working folder set before using this command.

Remarks

The Get Latest Version command retrieves the most recent version of a file, a group of files, or an entire **project** from Visual SourceSafe and creates read-only copies for viewing or compiling in your **working folder**.

Note You can choose a different working folder for the current Get operation by typing a new path in the Get Latest Version dialog box. Click Browse to navigate your folders.

Getting a file does not replace the file that you currently have checked out in your working folder, unless you set the Replace advanced option.

Results of a Get Operation

Using the Get Latest Version command has four possible outcomes:

- When the file you are retrieving is not already in your working folder, Visual SourceSafe simply gets the file.
- When the file in your working folder is identical to the file you are retrieving, Visual SourceSafe does not retrieve or modify the file in any way.
- When the file in your working folder is different from the Visual SourceSafe **master copy**, and your local copy is read-only, Visual SourceSafe replaces your **local copy**.
- When the file in your working folder is not read-only, Visual SourceSafe assumes that you have the file checked out, and doesn't replace the file. You can change the way Visual SourceSafe behaves in this case with the Replace Writable option in the Local Files tab of the Options dialog box.

Merging with the Get Latest Version Command

In a **multiple checkout** situation, you can use the Get Latest Version command to retrieve other users' changes before checking in your version of the file. You can tell Visual SourceSafe to merge the changes into your local copy of the file, or you can specify that you want to verify each change before permitting the merge. To do this, set the Replace Checked Out Files option to Merge on the Local Files tab in the SourceSafe Options dialog box.

Dialog Box Options

If you enable the Get Latest Version command dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu), or hold down the SHIFT key when you click the Get Latest Version command, you have the following options:

To

Type the path where you want Visual SourceSafe to place the files.

Recursive (project-level Get only)

Gets a project, all the files in the project, and its subprojects. Each file in a subproject is copied into its own working folder, which can be a folder anywhere on your computer or an accessible remote computer. To override the working folder settings, and create a file structure that mirrors the SourceSafe project structure, use the Build Tree option.

Build Tree—Override Working Folders (project-level Get only)

Overrides the working folder set for the subprojects involved in a recursive operation, and builds a project tree on your local computer that mirrors the project organization. Working folder settings for individual subprojects are ignored. Available only when you click the Recursive check box.

Make Writable

Sets the read-write flag so the file is writable. This is not the default behavior, but may be useful if you want to try a temporary change in the file, or want to rename the file after getting it.

Advanced Options

You can also set the defaults for the first three advanced options on the Local Files tab in the SourceSafe Options dialog box (Tools menu).

Set File Time

Changes the time assigned to the get operation. Select one of the following in the list box: current time, last file modification time, last file check-in time, or the default time (set on the Local Files tab).

Replace Writable

Determines how Visual SourceSafe behaves when you try to get a file that is already marked as writable in your working folder. Select one of the following in the list box:

- **Ask.** Asks you what to do when a writable version of a file is found.
- **Replace.** Replaces the writable file with a read-only version.
- **Skip.** Skips the get operation for the writable file. An error is displayed in the Results window if it is open.
- **Merge.** Merges the changes between the checked out-file and the one you are now getting. If the file cannot be merged (i.e., it is a binary file), the get operation is skipped for that file, and an error is displayed in the Results window if it is open.
- **Default.** Uses defaults set on the Local Files tab or inherited from a parent project.

Skip This Dialog

Skips this dialog box during future get operations, and uses the settings you've entered as the default. To restore the dialog box, clear the appropriate box on the Command Dialogs tab in the SourceSafe Options dialog box. Pressing the SHIFT key when clicking the Get Latest Version command displays this dialog box, overriding the setting on the Command Dialogs tab.

See Also

[Check Out Command](#)

[Getting, Checking In, Checking Out, and Viewing Files](#)

[Local Files Option Tab](#)

Help Menu Commands

See Also **Command Line**

The Help menu commands provide Help on using Visual SourceSafe and Help on specific Visual SourceSafe commands. In addition, you can press F1 at any time to view Help on the currently selected menu item or dialog box.

Help Command	Action
Contents	Runs Help and displays the Visual SourceSafe Table of Contents.
Search For Help On	Displays the Help Search dialog box so you can quickly find the information you want.
<u>Technical Support</u>	Displays information about Microsoft Product Support Services.
Books Online	Displays an online version of the <i>Microsoft Visual SourceSafe User's Guide</i> .
About Microsoft Visual SourceSafe	Displays a dialog box with information about your copy of Microsoft Visual SourceSafe, including the version number and the copyright, legal, and licensing notices.

Access Rights

Any Visual SourceSafe user can use these commands.

Remarks

A quick way to find information is to use Search, an automated index with hundreds of index entries and extensive cross-referencing.

You can also look for Help topics in the Visual SourceSafe Table of Contents.

As you read a topic in Visual SourceSafe Help, you can scroll through the topic with the scroll bar, the arrow keys, or the PAGE UP and PAGE DOWN keys.

Help Menu Commands See Also

Technical Support



Label Command (File Menu)

See Also

Command Line

Assigns a label to the specified version or current version of a file or project.

Access Rights

You must have the Add **access right** to use this command.

Remarks

Every version of a file has a **version number** and a date and time, and can be retrieved by reference to either. To make a version easier to identify, you can assign it a label—any string up to 31 characters in length

—and refer to the version by its label.

With the Label command, a new version of the selected file(s) or project is created, and the label is associated with that version.

Note To change the label of an earlier version, when a file is selected in the **file pane** of **Visual SourceSafe Explorer**, on the Tools menu, click the Show History, and then click Details.

When you edit a label in the History Details dialog box, you are assigning a label to an existing version of the specified item.

If you want to identify a certain version of a project so you can come back to it, do not assign a label to every file in the project, but rather, assign the label to the project itself: Even if you add, delete, and rename files in the future, Visual SourceSafe can accurately restore that version of the project based on the project label.

Dialog Box Options

The screenshot shows a dialog box titled "Label". It has a standard Windows-style title bar with a close button (X). The dialog contains three input fields: "Item:" with the text "\$/Samples/WebWind", "Label:", and "Comment:". To the right of these fields are three buttons: "OK", "Cancel", and "Help".

Item

The name of the file or project.

Label

Type a label up to 31 characters in length.

Comment

Type an optional comment.

See Also

Show History Command

Tracking and Labeling Versions of Files and Projects

Merge Branches Command (SourceSafe Menu)

See Also Command Line



Copies all the changes made in one branch of a file to another branch. Works for files only, not projects.

Access Rights

You must have the Check Out **access right** in the project being merged into, and the Read access right in the project being merged from, to use this command. You should have a working folder set before using this command.

Remarks

Merge copies the changes made in one file into another file (on a separate development path) in Visual SourceSafe. If the merge creates **conflicts**, Visual SourceSafe checks out the file to your **working folder** if it wasn't already checked out. You can then modify the file and resolve any conflicts before checking the file back into Visual SourceSafe.

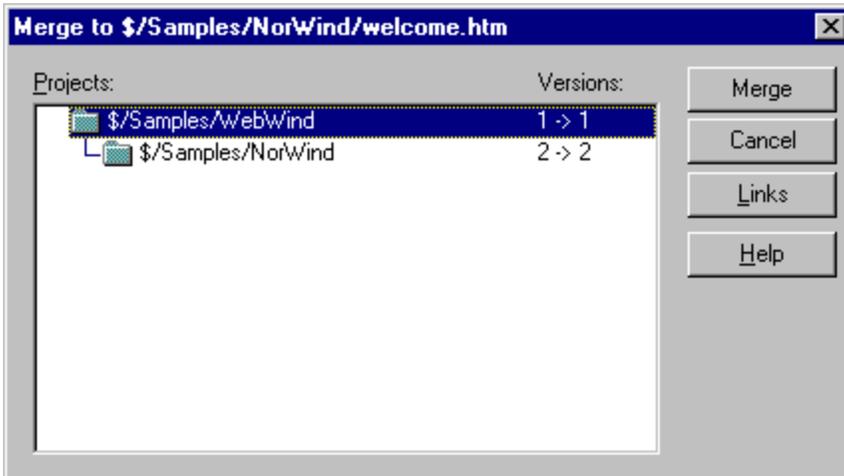
For example, project \$/Newapp is currently shipping version 5. The development team has started working on version 6, when suddenly a big customer calls with a serious bug in version 5. The Newapp project is branched at that point so that the development team can continue working on version 6 while a crack bug team works on fixing the version 5 bug (perhaps their project is named \$/Bugfix). Up to the branch point, the projects are identical, but after the branch the code in each project differs.

Just before version 6 ships, the Bugfix team announces they have solved the bug. You decide to merge their code into the Newapp project so version 6 of your product has the fix. You select a file at a time in the Newapp project and choose the Merge Branches command.

After the merge operation, you still have two independent development paths, but the changes made in \$/Bugfix have been copied to \$/Newapp. Before you choose the Merge Branches command, be sure you have the file selected that you want to merge changes *into*. (The file you are merging changes *from* will be unchanged.)

Note You cannot merge binary files.

Dialog Box Options



Projects

Select the project into which you want to merge your file. The Versions column shows you the version numbers you will be merging.

Merge

Merges branches.

Cancel

Cancel merge of branches.

Links

Shows all of the projects with which the file has **share links**.

See Also

[Branch Command](#)

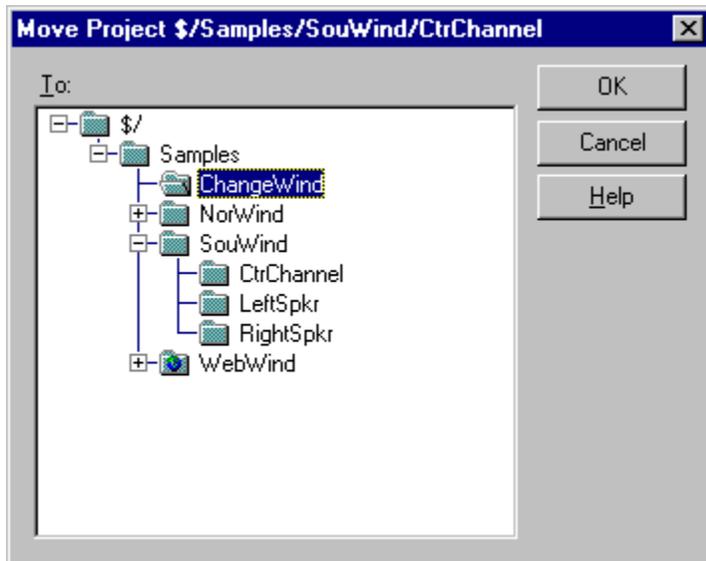
[Merging Files and Projects](#)

[Setting the File Type: Text and Binary Files](#)

[The Visual Merge](#)

Move Project Command (File Menu)

See Also Command Line



Relocates a **subproject** from one **parent project** to another.

Access Rights

You must have the Add **access right** in the parent project being moved to, and the Destroy access right in the parent project being moved from, to use this command.

Remarks

The Move Project command redefines the path to a subproject. For instance, if the project TOOLMANAGER exists under \$/JOE, and you move it to \$/APPS, its new full name is \$/APPS/TOOLMANAGER.

This command does not rename a project, and it does not change its contents or history. It does affect the histories of the old and new parent projects.

After you move a project, you cannot reconstruct an old version of its original parent project. In the preceding example, any attempt to use the Get Latest Version command recursively to retrieve a version of \$/JOE that existed before you moved the TOOLMANAGER project generates a warning that TOOLMANAGER has been moved, and is unavailable.

Note You cannot move individual files with the Move Project command. Instead, to move a file, share the file with the new parent project, and then delete or destroy the file in the original project. The **history** of the file is retained.

Dialog Box Options

To move a project, first select that project in the Visual SourceSafe Explorer. Then select the new project location from this dialog box and choose OK.

To

Select the path of project to which the subproject is to be moved.

OK

Moves the selected project to the new location.

Cancel

Cancels the Move operation.

See Also

Delete Command

Destroy Command

Rename Command (File Menu)

See Also **Command Line**



Renames a file or project.

Access Rights

You must have the Add **access right** to use this command.

Remarks

To use the Rename command, select the file or project you want to rename, then choose the command. You will be able to edit the item name directly in the Visual SourceSafe Explorer window.

When you use the Rename command on a file or project, the item's file **history** is retained. If multiple projects share a file and the file is renamed, it is renamed in all projects that share it. However, if any **share links** have been broken by the Branch command, the branched files are not renamed. The Paths tab in the Properties dialog box shows how the name has changed in different branches of the file.

See Also

- **Paths Properties Tab**
- **Visual SourceSafe Naming Syntax and Conventions**

Share Command (SourceSafe Menu)

See Also Command Line

Makes a file a member of the **current project**, or creates a new branch project.

To share a specific version of a file to your current project, use the Share command (SourceSafe menu) to get the file into the proper project. Then, use the Show History command (Tools menu) on the file and click Pin on the version you want.

Access Rights

You must have the Check Out **access right** in the project you are sharing from, and the Add access right in the project you are sharing to, to use this command.

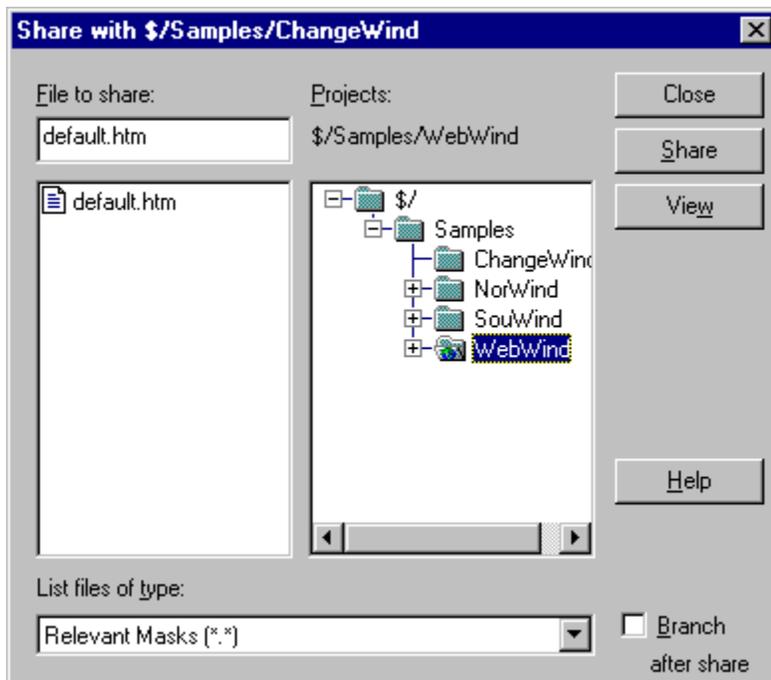
Remarks

To share a file, you create a **share link** between two **projects**. The file is then in two (or more) projects at the same time. When you check in the file to any one of the projects, your changes are automatically checked in to all of them. To display a list of all the projects that share a specific file, use the Links tab in the Properties dialog box.

If you create a branch, changes made in the **branched file** no longer propagate to the originating project, and vice versa. The Links command no longer shows a relationship, but you can use the Paths tab in the Properties dialog box to see the path of the file branch you created.

When you share a project, you create an entirely new project under the current project that is an exact duplicate of the project you specified. All the files in the new project are shared with the corresponding files in the originating project and changes in one are reflected in the other during **check in**.

Dialog Box Options



File To Share

Type the name of file(s) to share or click a filename displayed. Files listed are those that can be shared with your current project: Some files cannot be shared, such as files with the same name as a

file already in your project.

Projects

The name of project being shared from.

List Files of Type

Lists the file type, or tells Visual SourceSafe how to classify a file.

The list of file types available is determined by the File Types Included In File Group filter and the File Groups filter set on the File Types tab in the SourceSafe Options dialog box. All Files (*.*) lists all files in the current folder.

Close

Closes dialog box.

Share

Shares selected file or project.

View

Displays the file with the text editor associated with the file extension of the file. The editor is set with the File Viewer option on the General tab in the SourceSafe Options dialog box.

Branch After Share

Shares the file, then immediately branches it to create separate copies of the file.

See Also

Branching Files and Projects

General Options Tab

Links Properties Tab

Sharing Files

Show History Command

View File Command



Show Differences (File) Command (Tools Menu)

See Also

Command Line

Compares a Visual SourceSafe file and the corresponding file in your **working folder** and shows you the differences, if any.

Note If you want to show differences among projects, select a project in the **project pane** of **Visual SourceSafe Explorer** and click Show Differences.

Access Rights

You must have the Read **access right** to use this command. You should have a working folder set before using this command.

Remarks

Most commonly, you use the Show Differences command immediately before you check in a file to compare your copy of the file to the **master copy** in Visual SourceSafe. This gives you a list of the changes you have made, which can be helpful for generating your comment.

When you use the Show Differences command on an ASCII text file, Visual SourceSafe displays a color-coded list of line differences between the file in Visual SourceSafe and your **local copy**.

For binary files, Visual SourceSafe can tell you if the local copy of the file is different from the master copy, but cannot display the differences.

Note It is also possible to compare two earlier versions of a file to each other or to compare an earlier version to the current version. When you select an item in the History of File dialog box, and then click Diff, Visual SourceSafe displays the differences between the selected version of the file and your current working folder version. If you select two versions, Visual SourceSafe displays the differences between those two versions.

Dialog Box Options

By default Visual SourceSafe uses the Visual Difference dialog box,. If you specify a UNIX display, Visual SourceSafe uses the SourceSafe display options for Ignore Case and Ignore White Space. The SourceSafe display option has line widths calculated to fit on your screen, and no context lines.

Note Enable the display of this dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

Visual

Shows the entire file, with annotations for the changes.

SourceSafe

Shows a simple list of lines that were added, deleted, and changed, optimized for readability.

UNIX

Shows differences in the format used by the UNIX Diff utility.

Ignore White Space

Shows differences without regard for spacing characters, such as tabs.

Ignore Case

Shows differences without regard for upper- and lowercase.

OK

Displays the Difference For File dialog box.

Report

Sends a report to the Clipboard, a text file, or a printer.

Advanced Options**Ignore OS Differences**

Specifies that Visual SourceSafe ignore operating system line-break character differences. This is useful if you are comparing files across a network where one computer is a Macintosh, for example, and the other is operating with Microsoft Windows NT.

Show Context Lines

Includes the specified number of context lines in the difference display. Context lines are the lines in your file surrounding the line identified as different by Visual SourceSafe. For example, setting Show Context Lines to three means you'll see three context lines above and three context lines below the difference line.

Save Settings

Saves your selected settings for future use.

Skip This Dialog

Skips the File Difference dialog box during future Show Differences operations, using the settings you've entered as the default. To restore the dialog box, click the appropriate check box on the Command Dialogs tab in the SourceSafe Options dialog box.

See Also

Command Dialogs Options Tab

Show Differences (Project) Command

Setting the File Type: Text and Binary Files

Differences for File Dialog Box



Show Differences (Project) Command (Tools Menu)

See Also

Command Line

Compares a Visual SourceSafe project and the corresponding project in your **working folder** and shows you the differences, if any. You can also see individual line differences in the differing files, and reconcile differences between your working folder and project.

Note If you want to show differences between a file in your project and that file in your working folder, select a file in the **file pane** of **Visual SourceSafe Explorer**, and then click Show Differences

Access Rights

You must have the Read **access right** to use this command.

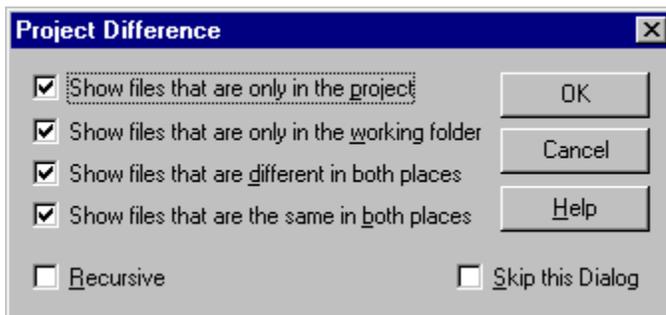
Remarks

The Show Differences command on a project generates a two-paned dialog box that lists the differences between a Visual SourceSafe project and your working folder. Files in these lists are color-coded to show you changed files, files in your working folder but not in your project, and vice-versa. Before the lists of files are displayed, a command dialog comes up that allows you to filter the file list if you'd like, so you can limit reporting to the files that interest you.

A common use of this command is to find the files another user has deleted from the project, so you can delete them from your working folder. It can also be useful if you take a project home for a while, to see where you have made your project-level changes.

Once you have examined the files in the two panes, you can reconcile differences between your working folder and your SourceSafe project in several ways. For example, you can get or check in all files that are different in your working directory, or add to your project all files not already in the project.

Project Difference Dialog Box Options



Note Enable the display of this dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu). The options in this dialog box are not mutually exclusive – they can be combined to filter the file lists in each pane. Each one of the checkboxes below potentially adds more files to the lists in each pane.

Show files that are only in the project

Includes, in the left pane, project files that are not currently in the working folder. This option is checked by default.

Show files that are only in the working folder

Includes, in the right pane, working folder files that are not currently in the Visual SourceSafe project. This option is checked by default.

Show files that are different in both places

Includes all files that are in both the working folder and Visual SourceSafe project, but are different from each other. Most often for these files, you need to get the file to update the version in your working folder. This option is checked by default.

Show files that are the same in both places

Includes files that are in both the working folder and the Visual SourceSafe project, and are the same in both places.

Recursive

Scans the contents of the current project; and also looks in **subprojects** under the **current project**, and shows all differences between the entire project list and your local folder list.

Skip this dialog

Skips this dialog box during future Show Differences operations, and uses the settings you've entered as the default. To restore the dialog box, check the Project Difference checkbox on the Command Dialogs tab in the SourceSafe Options dialog box.

OK

Displays the Difference for Project dialog box.

Cancel

Cancels the Show Differences command.

See Also

[Command Dialogs Options Tab](#)

[Differences for Project Dialog Box](#)

[File Types Options Tab](#)

[Show Differences \(File\) Command](#)

[Difference \(Project\) Command Line](#)



Show History Command (Tools Menu)

See Also **Command Line**

Shows the **history** of a file or project in Visual SourceSafe.

Access Rights

You must have the Read **access right** to use this command.

Remarks

Every time you check in a file or project to Visual SourceSafe, the file or project is considered a new version. Use the Show History command to display a list of the past versions of a file or project.

The versions are listed starting with the most recent and going back to the creation of the file or project. Visual SourceSafe provides information about the version number, which can be used with other commands (such as Get Latest Version) to indicate a specific version.

By default, the history of a project includes the histories of all its files, but does not include **subprojects**.

You can compare one version of a file with another by selecting the two versions simultaneously (hold down the CTRL key and click on each version), then pressing the Diff button.

Dialog Box Options

Note Enable the display of this dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

Include files (project history only)

Includes files in project history.

Recursive (project history only)

Includes subprojects and all files within them.

Labels Only

Includes only **labels** in project history.

From

Type a time, date, version, or label. Use the prefix D to indicate a date and/or time, and the prefix L to indicate a label. For example "D02/29/95;12:15" or "Lbeta1". Dates can be entered in 08.15.97, 08-15-97, or 08/15/97 format, but not as August 15, 1997. Note that if you have a project selected to see the history on, you cannot search for file labels — you must select the file and use the Show History command to find a specific label.

To

Type time, date, version, or label. Use the prefix D to indicate a date and/or time, and the prefix L to indicate a label. For example "D02/29/95;12:15" or "LBeta1". Dates can be entered in 08.15.95, 08-15-95, or 08/15/95 format, but not as August 15, 1995.

User

Shows a history of files worked on by the specified user.

OK

Displays History of Project or History of File dialog box, depending on whether a file or project is selected in Visual SourceSafe Explorer. In these dialog boxes, click any of the items in the list, and then click Details to display the History Details dialog box. From within the History Details dialog box, you can click the Next and Previous buttons to move up and down in history display. You can also edit the comment and **label** associated with the version you are viewing.

Skip This Dialog

Skips this dialog box during future Show History operations, and uses the settings you've entered as the default. To restore the dialog box, click the appropriate check box on the Command Dialogs tab of the Options dialog box.

See Also

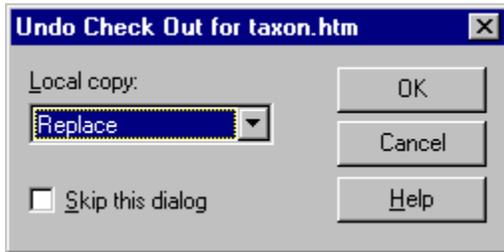
Command Dialogs Options Tab

History of File Dialog Box

History of Project Dialog Box

Undo Check Out Command (SourceSafe Menu)

See Also Command Line



Cancels a Check Out operation, undoing all changes. You must have a working folder set for this command to work properly.

Access Rights

You must have the Check Out **access right** to use this command.

Remarks

The Undo Check Out command undoes all your changes—the file is no longer checked out by you, but instead of updating Visual SourceSafe with your edited copy, Undo Check Out retains the version Visual SourceSafe previously had and copies it over the file in your **working folder**.

You can use the Undo Check Out command only on files that you currently have checked out, not on files checked out by other users.

Note Your Visual SourceSafe administrator can undo the check out of another user. See Chapter 8, "Administration," of the *Visual SourceSafe User's Guide* for more information.

Dialog Box Options

Note Enable the display of this dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

Local Copy

- Replace—Replaces the file in your working folder
- Visual SourceSafe performs the Get Latest Version command on the file. This keeps your working folder synchronized with the contents of the project, because all your changes are ignored.
- Delete—Deletes the file in your working folder.
 - Leave—Leaves the file alone.
 - Default—Uses defaults set on the Local Files tab or inherited from a parent project. This option cannot be set; it is used automatically when you explicitly (in the file list) or implicitly (by checking Recursive) select multiple files.

Skip This Dialog

Skips this dialog box during future Undo Check Out operations, and uses the settings you've entered as the default. To restore the dialog box, click the appropriate check box on the Command Dialogs tab of the Options dialog box.

Recursive (project-level Undo Check Out only)

Undoes the check out on all files in the selected project, and all files in all subprojects of the selected project.

Override Working Folders (project-level Undo Check Out only)

Overrides the working folder set for the subprojects involved in a recursive operation. This is useful if you have files checked out in more than one project, and want to undo the check out of all of them at the same time. Available only when you click the Recursive check box.

See Also

Set Working Folder Command



View File Command (Edit Menu)

Command Line

Shows the text of the **master copy** of the file in Visual SourceSafe. To view an earlier version of a file, use the View button in the History of File dialog box (Tools menu). To view a file before sharing it to your current project, use the View button in the Share dialog box.

Access Rights

You must have the Read **access right** to use this command.

Remarks

The View File command opens and displays a Visual SourceSafe master copy of a file using the editor associated with that file. It does not change anything in either the **Visual SourceSafe database** or your **working folder**.

The associated editor is selected by one of the following means, in the following order:

- 1 Visual SourceSafe looks in its initialization files for a line with the extension of the file being viewed.
- 2 Visual SourceSafe uses Microsoft Windows Explorer (the registration database and WIN.INI) to associate an application with the file extension.
- 3 Visual SourceSafe looks for your entry in the File Viewer option on the General tab in the SourceSafe Options dialog box.
- 4 If Visual SourceSafe cannot determine a file association after using these methods, SourceSafe displays the file using its own editor.

Important With the View command, any changes you make to the file are lost as soon as you close the application. If you want to make permanent changes to the file, use the Edit File command on the Edit menu.

Pin Command

See Also [Command Line](#)

Pins a specific version of a file into the [current project](#).

Access Rights

You must have the Add [access right](#) in the project you are pinning from to use this command.

Remarks

To pin a file to a specific version of the file, select the file in the [file pane](#) of [Visual SourceSafe Explorer](#), on the Tools menu, click Show History, click the options you want in the File History Options dialog box (if it is enabled). In the History of File dialog box, click the file version you want. Then click Pin. The file is pinned to that version in the current project.

If the file to be pinned is currently pinned to another version, click the file and click Unpin to again share the file normally. Then re-pin it to the version you want.

If the file to be pinned is shared in another project, the file can be pinned in the current project and changes made to the file in the other (shared) project will not affect the current project's file.

Note Pinned files cannot be checked out in the project in which they are pinned.

To return to using the latest version of the file, in the History of File dialog box, click the pinned file, and then click Unpin.

See Also

[History of File Dialog Box](#)

[Share Command](#)

[Show History Command](#)

[Unpin Command](#)

Unpin Command

See Also **Command Line**

Cancels a Pin command; changes a file pinned to a specific version to the latest version of the file.

Access Rights

You must have the Add **access right** in the project you are unpinning to use this command.

Remarks

To return a pinned file in your project to the current version, on the Tools menu, click Show History, click the file in the History of File dialog box, and then click Unpin. Unpin appears only when you select a file that is pinned.

See Also

[History of File Dialog Box](#)

[Pin Command](#)

[Share Command](#)

[Show History Command](#)

Font Command (Tools Menu)



Changes the font used to display the project and file lists in **Visual SourceSafe Explorer**.

Rights

Any Visual SourceSafe user can use this command.

Remarks

You set the font name, style, and size for your Visual SourceSafe Explorer display in the Font dialog box. The Sample box displays a preview of the currently selected font. You can also select a script if your localized version of Visual SourceSafe supports multiple scripts.

The font setting does not affect the font used in the file difference viewer. To set those options, use the Difference Options tab.

Customize Toolbar Command (Tools Menu)

Adds or deletes buttons on your Visual SourceSafe toolbar.

Rights

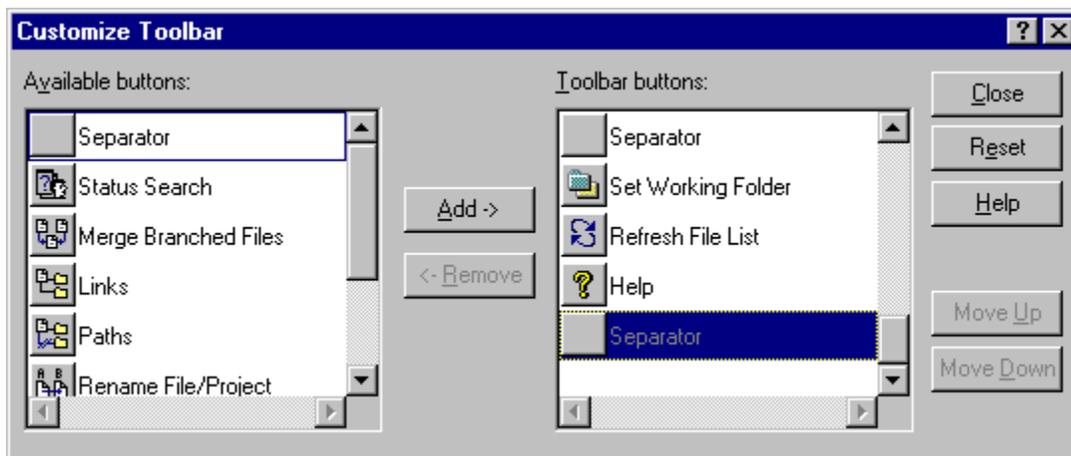
Any Visual SourceSafe user can use this command.

Remarks

To add a button to the toolbar, on the Tools menu, click Customize Toolbar. From the Customize Toolbar dialog box, select the toolbar button you want from the Available buttons area, then click Add to move it to the Toolbar buttons area. To delete a button, select the toolbar button you want from the Toolbar buttons area, then click Remove to move it to the Available buttons area.

After a button is on the Toolbar, you can drag it to a new location in the dialog box to move it to a new position on the Toolbar.

Dialog Box Options



Add

Moves a selected button from the Available button list to the Toolbar buttons list (that is, adds it to the toolbar).

Remove

Moves a selected button from the Toolbar button list to the Available buttons list (that is, removes it from the toolbar).

Reset

Resets the toolbar to its default configuration.

Close

Closes the Customize Toolbar dialog box.

Move Up

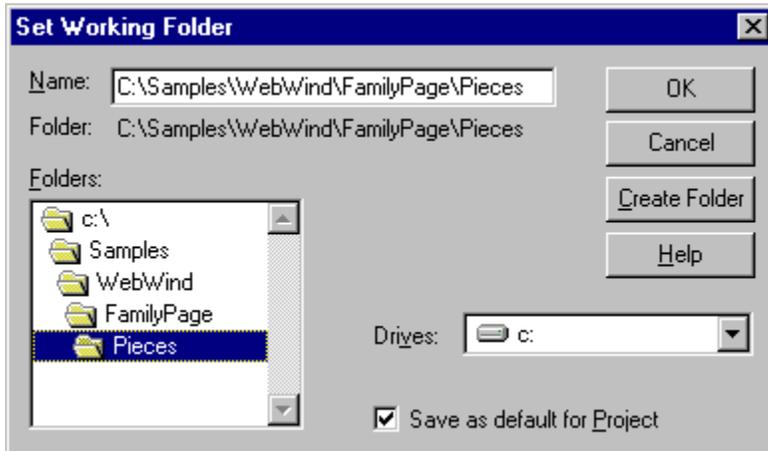
Moves a button up in the Toolbar Buttons list, moving it closer to the left in the Visual SourceSafe Explorer toolbar.

Move Down

Moves a button down in the Toolbar Buttons list, moving it closer to the right in the Visual SourceSafe Explorer toolbar.

Set Working Folder Command (File Menu)

See Also



Sets your **working folder**.

Rights

Any Visual SourceSafe user can use this command.

Remarks

It is important for you to designate, after creating a new project, a working folder in which to work on the project's files. If you do not, SourceSafe prompts you for a working folder whenever you use a command that requires one to be set.

You can use an existing folder on your computer's hard disk as your working folder. You can also create a new folder from within Visual SourceSafe or Windows.

If you just click OK, this command temporarily sets the working folder for the project: The setting is saved after you close Visual SourceSafe. If you turn off the Save as Default for Project check box, the folder is not saved as your default working folder for the **current project**.

Dialog Box Options

Name

Name and path of working folder.

Folder

Your current working folder.

Folders

Specifies a folder for the working folder.

Drives

Selects the drive that contains the working folder.

OK

Confirms selection.

Cancel

Cancels the selection, closes the dialog box, and returns to **Visual SourceSafe Explorer**.

Create Folder

Creates a new folder for the working folder. Type a new folder name in the Name box, then click Create Dir to create the folder.

Save as Default for Project

Saves the folder you specify as your default working folder for the project.

See Also

[Understanding the Working Folder](#)

Microsoft on the Web Command (Help Menu)

Opens the Microsoft Internet Explorer (or your configured web browser) to one of several Microsoft web sites.

From this command, you can choose to access any of the following web sites:

Menu Item	Web Site
Microsoft Visual SourceSafe	The home page for Microsoft Visual SourceSafe.
Free Stuff	Free Visual SourceSafe utility programs that make using Visual SourceSafe even easier.
Frequently Asked Questions	Answers to frequently asked questions about Visual SourceSafe.
Online Support	The Microsoft Knowledge Base, containing answers to many common support questions.
For Developers Only	The For Developers Only web site.
Send Feedback...	Lets you send the Visual SourceSafe team comments, criticism, or suggestions for new features.
Web Directory	A Microsoft web directory service.
Search the Web...	A web search engine.
Web Tutorial	A tutorial on using the world-wide web.
Microsoft Home Page	The Microsoft home page on the world-wide web.

Access Rights

Any Visual SourceSafe user can use this command.

Edit File Command (Edit Menu)

See Also

Allows you to edit a source-controlled file by ensuring that it is checked out and then opening the file in the associated editor.

Access Rights

You must have write permission to use this command.

Remarks

When you select Edit File, Visual SourceSafe first checks to be sure you have the file checked out. If you do not, the file is checked out to you, unless something prevents this, such as the file being already checked out by another user (when your Visual SourceSafe administrator has not enabled multiple checkouts for that project).

Once the file is checked out to you, Visual SourceSafe opens the file in the associated editor. The associated editor is selected by one of the following means, in the following order:

- 1 Visual SourceSafe looks in its initialization files for a line with the extension of the file being viewed.
- 2 Visual SourceSafe uses Microsoft Windows Explorer (the registration database and WIN.INI) to associate an application with the file extension.
- 3 Visual SourceSafe looks for your entry in the File Viewer option on the General tab in the SourceSafe Options dialog box.
- 4 If Visual SourceSafe cannot determine a file association after using these methods, it displays the file using its own editor.

Note If no editor can be found for a specific file type, the file will remain checked out to you in your working directory, so you can open the file normally with the editor of your choice.

You can also set the double-click behavior of Visual SourceSafe on the General tab of the Tools Options dialog box. Setting the Double-click on a file option to Edit File will allow you to double-click on a file in Visual SourceSafe to edit it; however, setting the option to Ask is recommended (so you don't accidentally write over a local file during the check out operation).

See Also

[View Options Tab](#)

Check Hyperlinks (Web Menu)

See Also

Allows you to check web links in your HTML files.

Access Rights

You must have the Write **access right** to use this command.

Remarks

You can use this command to check web links in your HTML files, either the files in your working folder or in a Visual SourceSafe project. Visual SourceSafe only checks the links that jump to other files in your project – links to external pages are not checked.

Links checked are those referred to with HTML <A HREF>, , and <OPTION VALUE> tags, as well as within .MAP files (for hotspots within an image). Links to other web addresses created using code, such as VBScript or Java, are not checked.

When the link check is complete, Visual SourceSafe displays a dialog box with three list boxes. The top box lists all files in which errors were found. The middle box lists the internal links (jumps to other files within the web project) which appear to be invalid in the selected file in the top box, and the bottom box lists the external links that were found but not checked.

See Also

[Check HyperLinks Dialog Box](#)

[Visual SourceSafe Web Features](#)

[Create Site Map Command](#)

[Deploy Command](#)

Deploy Command (Web Menu)

See Also **Command Line**

Allows you to deploy a set of web files to a web server, or multiple web servers. This command works only on projects, not files.

Access Rights

You must have the Destroy **access right** to use this command.

Remarks

When you have completed development and testing of your Web pages, the Deploy command is used to copy your files to the live Web server. The location of this server is a folder or FTP site specified by your Visual SourceSafe administrator. When you choose the Deploy command, the contents of your current project are copied to that location.

In many respects, Deploy acts like the Get command: it copies files from a Visual SourceSafe project, to a local folder. However, Get makes a copy for your personal use; Deploy makes the project public, potentially to the entire world! For this reason, the Deploy command requires Destroy-level privileges in the selected project.

Your Visual SourceSafe administrator must set a deployment path in the Web Sites tab of the Administrator Tools Options menu to enable this command. You can deploy only an entire project marked as a web site project, not single files. The deployment folder can be a comma-delimited set of places you want your files copied to.

When used, the Deploy command overwrites existing files with the same name in the deployment folder, even if you have the Replace Writable files checkbox of the Tools Options Local Files tab set to Skip.

Note that this command is useful in many instances other than web site deployment; for example, if you have a set of product specification .DOC files that you want to publish to a public server.

See Also

[Deploy Dialog Box](#)

[Visual SourceSafe Web Features](#)

[Create Site Map Command](#)

[Check Hyperlinks Command](#)

Create Site Map (Web Menu)

See Also

Allows you to automatically create a map of the HTML files in a web project.

Access Rights

You must have the Write **access right** to use this command.

Remarks

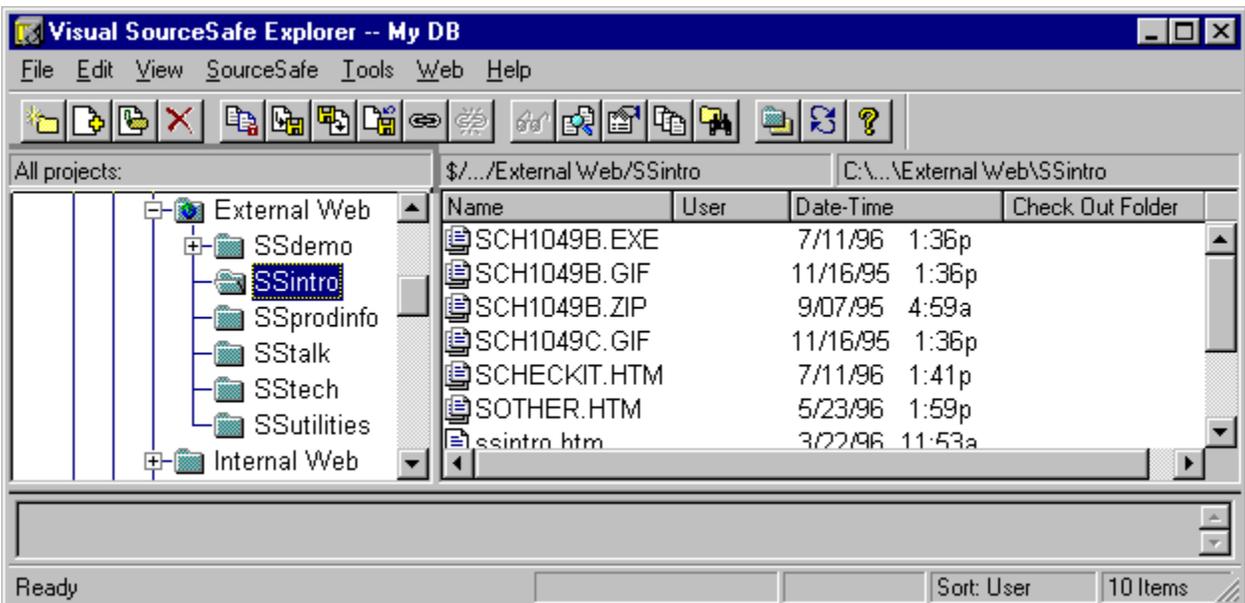
Site maps are becoming more popular as methods of navigating through a web site. This command lets you easily generate a list of the HTML files in your web project, and creates the list as an HTML file with links to each file. With some minor customization, you can then add the resulting file to your web site and provide your users with a classic method of navigation!

Each time you run this command, a totally new file is created. That is, this command does not compare new material to old and just make the changes. So, if there is some customization you have added to the resulting site map file, you should be prepared to add it again each time you run this command, or to use technology such as server-side include files to store it in a separate place. Also, the HTML generated by this command is fairly rudimentary – HTML 2.0 tags. If you want a more customized page, you can cut and paste from the HTML provided by this command, or add HTML tags to the file to make it fit into your web site better.

The Create Site Map command works by getting the titles of your HTML files from their <TITLE> tag. If you have an HTML file without a <TITLE> tag, Visual SourceSafe will use its filename. To build a good site map, you should use descriptive, unique strings in the <TITLE> tags of your files. Category names in the site map file come from the Visual SourceSafe project name. Therefore, it is important for you to use good, descriptive project names in web site projects.

Once the HTML site map file is created, you can add it into your project by using the Add Files command. If you add the file to your web site project, be sure to check it out before running the Create Site Map command.

As an example, consider an External Web project that looks like this:



Running Create Site Map on the External Web project would create an HTML file that displayed like this in a web browser:

Using this HTML:

```
sitemap1.htm - Notepad
File Edit Search Help
<!-- Site Map for project: $/Juniper/Docs/External Web -->
<!-- Automatically generated by Visual SourceSafe (http://www.microsoft.com/ssafe) -->
<!-- User: wendy Date: 08/20/96 Time: 0 -->
<HEAD>
<TITLE>Site Map</TITLE>
</HEAD>
<BODY BGCOLOR=#FFFFFF TOPMARGIN=8>
<H1>Site Map</H1>
<HTML>

<H2>External Web</H2>
<A HREF="vssheader.htm">Microsoft Visual SourceSafe</A><P>
<A HREF="SSfeed.htm">Microsoft Visual SourceSafe Questionnaire</A><P>
<A HREF="Sqa.htm">Microsoft Visual SourceSafe Questions and Answers</A><P>
<A HREF="SiteMap.htm">Microsoft Visual SourceSafe Site Map</A><P>
<A HREF="webfeed.htm">Microsoft Visual SourceSafe Web Feedback</A><P>
<A HREF="SSMAC.HTM">Visual SourceSafe Feedback</A>
```

Of course, at this point, you can customize the HTML to make your site map look like anything you want — use a different background color or pattern, change indentation levels, delete or add items, etc. One way you can customize the site map is to change the graphics files used for the folder and file icon. For more information on doing this, see the [IMG_Folder](#) and [IMG_File](#) initialization variables.

See Also

[Create Site Map Dialog Box](#)

[Visual SourceSafe Web Features](#)

[Check Hyperlinks Command](#)

[Deploy Command](#)

Open SourceSafe Database Command (File Menu)

See Also

Allows you to switch between Visual SourceSafe databases via the Visual SourceSafe Explorer.

Access Rights

You must have the Read **access right** to use this command.

Remarks

Use this command if you regularly need to access different **Visual SourceSafe databases**. You will, of course, need access permissions on the server storing the database and in the database you are trying to access. When you use this command, the current Visual SourceSafe database is closed and the newly specified one is opened.

You can give each database an easy-to-remember name, and each time you connect to a new database it is added to the list in the Open SourceSafe Database dialog box.

Note: If you used Visual SourceSafe 4.0 with multiple databases, you may have used the old initialization variable method of switching between databases. In most cases, each separate database will have its own SRCSAFE.INI file, and you can use the Browse button once to find each database and add it to your database list, and then use the Open SourceSafe Database command from then on.

If you don't have a SRCSAFE.INI for one database, create a SRCSAFE.INI file for that database. Put this SRCSAFE.INI file in the directory immediately above the DATA directory. See the Using the Command Line to Open Databases topic for more information.

Open SourceSafe Database See Also

[Open SourceSafe Database Dialog Box](#)

[Using the Command Line to Open Databases](#)

Using the Command Line to Open Databases

If you wish to maintain separate Visual SourceSafe databases, you can easily switch between them by using the Open SourceSafe Database command on the File menu. You can also use separate instances of the Explorer for different databases. In Windows, you can create separate icons. However, in some circumstances, you may wish to use the command line to open new databases. You can configure the command line of the icon to specify the appropriate database.

The /s Option

For example, you might want to work with databases on two computers. One database is on computer ACTGCOMP, in share directory ACTGVSS. The other database is on computer MFGCOMP, in share directory MFGVSS.

You can create new icons to run Visual SourceSafe. The command line for each icon must include the path to the Visual SourceSafe executable program on your computer and the path to the SRCSAFE.INI file for the database you want to work with. Use the **/s** option to specify the share directory that contains the SRCSAFE.INI file.

Create an icon and give it the following command line to use the database on computer ACTGCOMP:

```
C:\VSS\WIN32\SSEXP.EXE /S\\ACTGCOMP\ACTGVSS
```

Then set up an icon for the database on computer MFGCOMP, with the following command line:

```
C:\VSS\WIN32\SSEXP.EXE /S\\MFGCOMP\MFGVSS
```

You can use a UNC path, as shown in these examples, or enclose in quotation marks a path that uses a mapped drive letter:

```
C:\VSS\WIN32\SSEXP.EXE /S"R:\MFGVSS"
```

The /d Option

It is possible to specify more than one data directory in a single SRCSAFE.INI file. To do this, you use multiple Data_Path lines, each of which contains a name in parentheses and points to a data directory, for example:

```
Data_Path (Sam) = c:\sambal\samsdata  
Data_Path (Kim) = d:\kimchi\kimsdata
```

Use the **/d** option to select from among these lines. This option specifies the name in parentheses in a Data_Path line. In order for you to use the **/d** option, there must be an entry in the SRCSAFE.INI file that contains a value in parentheses and points to a data directory. In conjunction with a SRCSAFE.INI file that contains the Data_Path line shown, the following command line example causes Visual SourceSafe to open the database in C:\kimchi\kimsdata:

```
C:\VSS\WIN32\SSEXP.EXE /DKIM
```

You can use the **/d** option in conjunction with the **/s** option to specify particular databases in specific share directories on a network.

Note The **/d** option exists solely for compatibility with earlier versions of Visual SourceSafe. Its use is not recommended.

From Visual Basic or Visual C++

If you are using Visual Basic or Visual C++, see the Visual Basic or Visual C++ documentation for information about the Run Visual SourceSafe command and the Prompt for Connect option.

Adding a Database

The simplest way to create a new database is to re-install Visual SourceSafe in the new location. You can also create a new database by following these steps:

1. Copy the Visual SourceSafe DATA directory, for example from R:\VSS\DATA to R:\DATA2.
2. Create a SRCSAFE.INI file at the root level of the R: drive, containing only the following two lines:

```
#include VSS\SRCSAFE.INI
Data_Path = DATA2
```

You now have two valid SRCSAFE.INI files, functionally identical (through the #include statement) except that they point to different data directories.

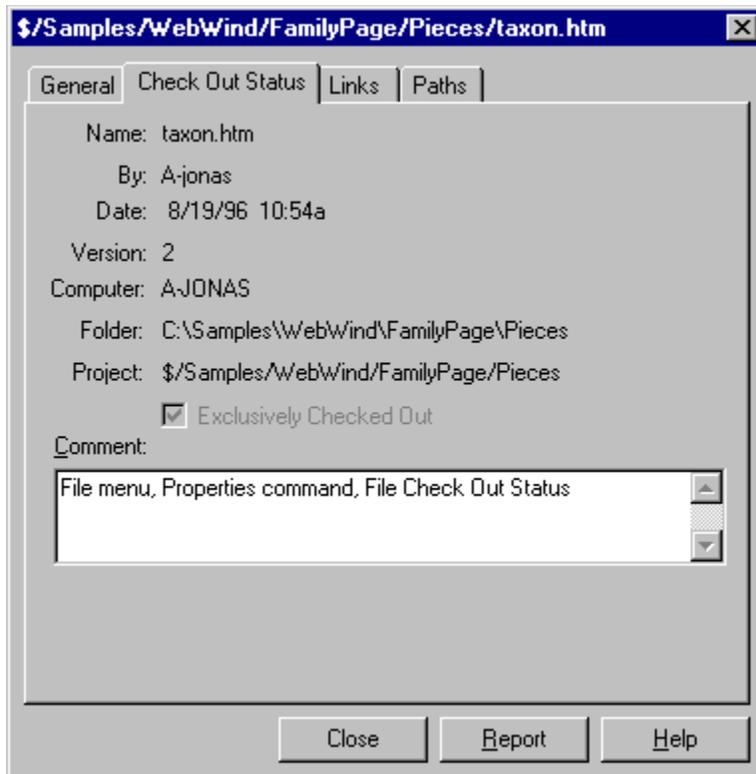
Ideally, the SRCSAFE.INI file for a given database should be at the same level as the data folder. That is, for example, if the data folder is \\myserver\ourshare\vss\data, then the .INI file should be \\myserver\ourshare\vss\srcsafe.ini.

Note The location of the SRCSAFE.INI file is particularly important if you use the Run SourceSafe command in Visual Basic or Visual C++. The Run SourceSafe command looks for a SRCSAFE.INI file only in a folder that contains a data folder. Because the SRCSAFE.INI file in that folder points to only one data folder at a time, only one database is found by this command. If you have two (or more) data folders together inside the folder with the SRCSAFE.INI file, SourceSafe can find only one of those databases when you use the Run SourceSafe command.

To put a database on a different computer, you can copy a data folder and a SRCSAFE.INI file, as described above; or you can install a new copy of Visual SourceSafe on the other computer, which creates a database.

Check Out Status Properties Tab

See Also [Command Line](#)



Displays check out information on a Visual SourceSafe file.

– **To display the Check Out Status tab**

- 1 Select a file in the file pane of Visual SourceSafe Explorer.
- 2 On the File menu, choose Properties, and then choose the Check Out Status tab.

Access Rights

You must have the Read access right to use this tab.

Remarks

If more than one user has a file checked out, clicking Check Out Status first displays a list of all check outs. Double-click a particular check out to see details on that check out. You can also check in a file, or undo the check out of a file.

If the selected file has not been checked out, when you choose Check Out Status Visual SourceSafe tells you the file has not been checked out, and does not display the tab options.

Tab Options

Name

Name of the selected file.

By

User who checked out the selected file.

Date

Date and time the selected file was checked out.

Version

Version number of the selected file.

Computer

Name of the computer from which this file was checked out.

Folder

Working folder the checked out file is in.

Project

Name of the project the file is in.

Exclusively Checked Out

If checked, this box indicates that the file or project is checked out exclusively to one user.

Comment

Comment that was entered for the file. You can edit or add a new comment in this box, up to 64 characters.

Close

Closes the Properties dialog box and returns to Visual SourceSafe Explorer.

Report

Sends a report to the Clipboard, a file or a printer.

See Also

[Check In Command](#)

[Check Out Command](#)

[Undo Check Out Command](#)

Login Dialog Box

Logs you on to Visual SourceSafe. The Login dialog box is displayed when you start Visual SourceSafe.

– **To log on to Visual SourceSafe**

- 1 In the User Name box, type your username.
- 2 Type your Visual SourceSafe password (if you have one).
- 3 Select the Visual SourceSafe database you want to log in to, and press ENTER to log on to Visual SourceSafe. (You can Browse for a different database.)

Access Rights

Any Visual SourceSafe user can use this dialog box.

Remarks

If you are unsure of your Visual SourceSafe username or password, see your Visual SourceSafe administrator.

Select Dialog Box

Many of Visual SourceSafe's most common commands—such as Get Latest Version, Check Out, and Check In

—can work on more than one file at once. For them to do so, however, you must select multiple items before choosing the menu command you want to execute.

— **To select multiple items with the Select dialog box**

1 Click in the file pane of Visual SourceSafe Explorer, and then on the Edit menu, choose Select. Visual SourceSafe displays the Select dialog box.

2 In the Item box, type a filename using wildcard characters. For example, type *.* to select all items.

Choose Select or press ALT+S.

3 To add additional items to the selection list, separate items with a semicolon in the list.

4 Choose Close to confirm the selection and close the dialog box.

— **To select items with the mouse**

- Choose an item in Visual SourceSafe Explorer.

—Or

Hold down the CTRL key as you choose each item you want; to select a block of items, choose the first item, hold down the SHIFT key, and choose the last item.

— **To select items with the keyboard**

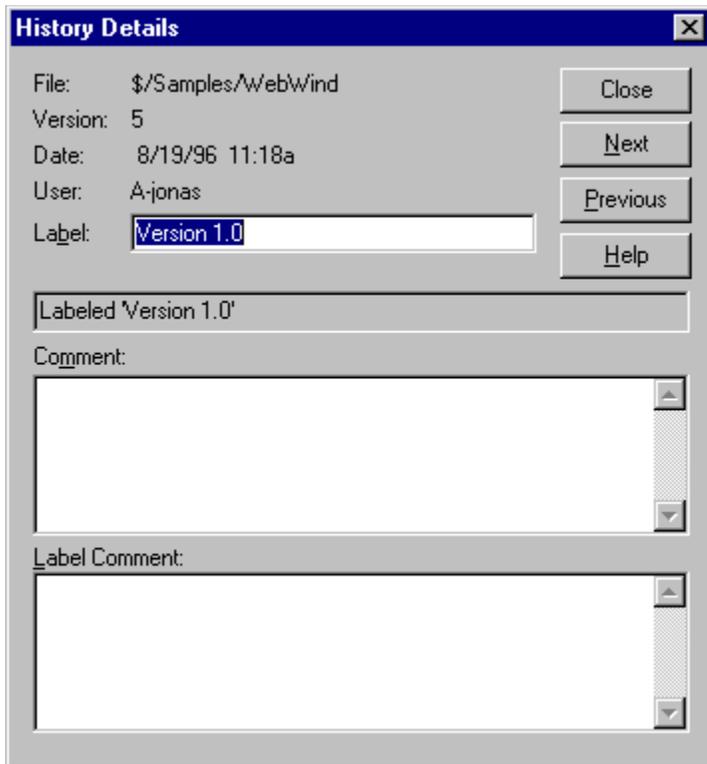
- Press SHIFT+SPACEBAR to select each item you want. Use the arrow keys to move to each.

— **To deselect items with the Invert Selection command**

- On the Edit menu, choose Invert Selection to change selected items to deselected items, and vice versa.

History Details Dialog Box

See Also



Displays details of the history of a file or project.

– **To display the History Details dialog box for a file**

1 Choose a file in the file pane of Visual SourceSafe Explorer.

2 On the Tools menu, choose Show History

Make selections in the File History Options dialog box if it is enabled, and choose OK.

Note Enable the display of the File History Options dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

3 In the History of File dialog box, choose Details.

– **To display the History Details dialog box for a project**

1 Choose a project in the project pane of Visual SourceSafe Explorer.

2 On the Tools menu, choose Show History.

Make selections in the Project History Options dialog box if it is enabled, and choose OK.

Note Enable the display of the Project History Options dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

3 In the History of Project dialog box, choose Details.

Access Rights

Any Visual SourceSafe user can use this dialog box.

Dialog Box Options

File

Name of the selected file.

Version

Version number of the selected file.

Date

Date and time the selected file was created or last checked out.

User

User who created the version.

Label

Shows the label, if any, for the file or project. You can also change the label.

Comment

Adds a new comment or changes an existing comment for a specific version of a file or project. Every version of every file or project in Visual SourceSafe is associated with a user-defined comment. The new comment replaces the old comment, which cannot be recovered.

Label Comment

Shows the comment entered at the time a label was applied.

Next

Shows information on the next item in the History list.

Previous

Shows information on the previous item in the History list.

Close

Closes the History Details dialog box.

See Also

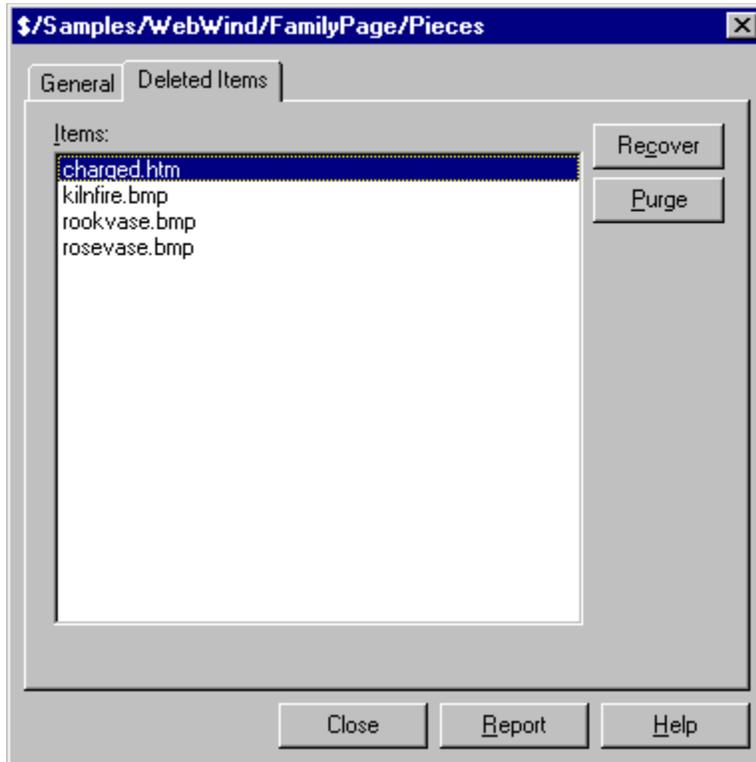
[History of File Dialog Box](#)

[History of Project Dialog Box](#)

[Show History Command](#)

Deleted Items Properties Tab

See Also



Displays a list of files and subprojects that have been removed from the current project.

– **To display the Deleted Items tab**

- 1 Select a project in the project pane of Visual SourceSafe Explorer.
- 2 On the File menu, choose Properties, and then choose the Deleted Items tab.

Access Rights

You must have the Read access right to use this tab.

Remarks

When you use the Delete command to delete a file or project from Visual SourceSafe, the deleted item is removed from Visual SourceSafe Explorer. However, Visual SourceSafe continues to store the item, so that you can retrieve it if you want to build an earlier version of the project. After you recover or purge an item, however, it no longer appears in the list in the Deleted Items tab.

Tab Options

Items

The list of all removed files and subprojects in the current project.

Recover

Recovers (undeletes) an item, and places it in the file list or project list.

Purge

Permanently deletes an item, so that it can no longer be recovered.

Close

Closes the Deleted Items tab and returns to Visual SourceSafe Explorer.

Report

Sends a report to the Clipboard, a text file, or a printer.

See Also

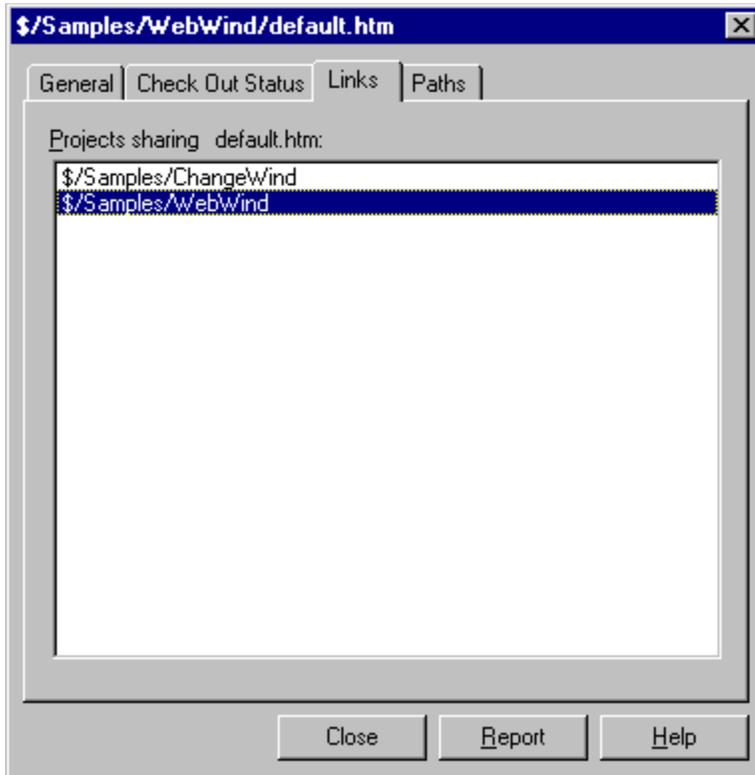
[Delete Command](#)

[Purge Command](#)

[Recover Command](#)

Links Properties Tab

See Also Command Line



Shows all the projects sharing a specified file.

– To display the Links tab

- 1 Choose a file in the file pane of Visual SourceSafe Explorer.
- 2 On the File menu, choose Properties, and then choose the Links tab.

Note To display this tab from within Visual Basic, after adding a Visual Basic project to Visual SourceSafe, on the Visual Basic Add-Ins menu, choose SourceSafe, and then choose SourceSafe Properties.

Access Rights

You must have the Read access right to use this tab.

Remarks

Use the Share command to create a share link between files. A file that is shared by multiple projects is the same in all projects that share it.

Tab Options

Projects Sharing <file>

The list of projects sharing the selected file. When you check in a file, all the projects shown in this list automatically receive the checked-in version of the file. If a file is deleted from a project, or if a project is pinned to a specific version of the file (instead of the latest version), that is also indicated in the list window. Deleted items are noted with a text string, and pinned items are noted with the version number following the filename, for example, \$/CODE/HEADER.H; 4.

Close

Closes the Properties dialog box and returns to Visual SourceSafe Explorer.

Report

Displays the Links Report dialog box, in which you can preview a report, and send it to the Clipboard, a text file, or a printer.

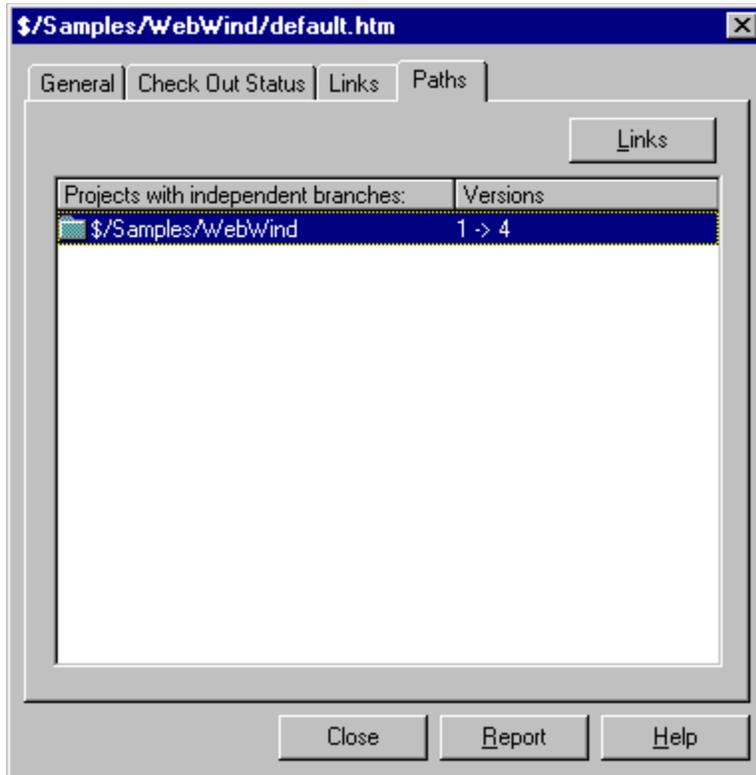
See Also

[Share Command](#)

[Sharing Files](#)

Paths Properties Tab

[See Also](#) [Command Line](#)



Shows all branches of the specified file.

– To display the Paths tab

- 1 Choose a file in the [file pane](#) of [Visual SourceSafe Explorer](#).
- 2 On the File menu, choose Properties, and then choose the Paths tab.

Note To display this tab from within Visual Basic, after adding a Visual Basic project to Visual SourceSafe, on the Visual Basic Add-Ins menu, choose SourceSafe, and then choose SourceSafe Properties.

Access Rights

You must have the Read [access right](#) to use this tab.

Remarks

When you use the Branch command to branch a file, the share link is broken and an independent file branch is created. Once a file is branched, modifications made to that file in one project made to a file in one project are not reflected in the other projects. Nonetheless, it is sometimes valuable to track these relationships.

For example, suppose you find a bug in a file, and fix it. All the projects sharing that file are immediately updated with the new version. However, projects that used to share that file, and then branched from it, are not updated; use the Paths tab to find those projects and then manually update them.

When you view paths, you view the entire branch history of the file, including all the times the Branch command was used to move it to different projects on different development paths. Changes made to the file, however, are not shown – use the Show History command to view these.

Using the Paths command might result in a display like the following.

```
Projects with independent branches
$/A      1->5
--$/B    6->7
--$/D    4->4
```

This display means that there are 3 branches of this file (because there are 3 lines); the first branch has moved from version 1 of the file to version 5, the second branch has the first as its parent (meaning that versions 1 through 5 of the file are the same in this branch), began its life at version 6 and was checked in once since then, moving it to version 7 currently. The third branch began its life at version 4 and has not changed since then.

You can select each branch and choose the Links button to see where that particular file is shared. So, how do you use this information? Assume a bug was introduced in this file at version 5 of the first branch. You know, from looking at this display, that the third branch is unaffected by this bug (because it is still at version 4), but that the second branch needs to have the fix manually inserted, because that branch came from the first (its parent), but began at version 6. Of course, once you have fixed the file in each of these branches, any other projects that share the file will automatically include the fix.

Tab Options

Projects with independent branches

Lists the projects the selected file has been shared in.

Versions

Lists the file version number that was branched in the corresponding project.

Links

Displays the Links tab of the Files Properties dialog box.

Close

Closes the Properties dialog box and returns to Visual SourceSafe Explorer.

Report

Sends a report to the Clipboard, a text file, or a printer.

See Also

[Branch Command](#)

[Branching Files and Projects](#)

[Links Properties Tab](#)

[Share Command](#)

[Show History Command](#)

Purge Command

See Also [Command Line](#)

Permanently removes previously deleted files and projects from the [Visual SourceSafe database](#).

– **To use the Purge command**

- 1 Choose a project in the [project pane](#) of [Visual SourceSafe Explorer](#).
- 2 On the File menu, choose Properties, and then choose the Deleted Items tab.
- 3 Choose Purge. You receive a warning message, because you can't undo a Purge command.

Access Rights

You must have the Destroy [access right](#) to use this command.

Remarks

When you use the Delete command to delete files or projects from Visual SourceSafe, the deleted items are removed from Visual SourceSafe Explorer. However, Visual SourceSafe continues to store the items so that you can recover them – and, for example, rebuild earlier versions of the project that required them.

To permanently delete these files from the Visual SourceSafe database – and free up the disk space they are using

– use the Purge command.

When you purge a previously deleted file from a project, the file is permanently removed from the project, and cannot be recovered. Consequently, if you use the Get Latest Version command to get any version of the project that contained that file, the file can't be found. For this reason, the Purge command should be used with extreme caution: You must be confident that you will never want this file or project, or any of its earlier versions, for any reason.

Note The purpose of purging is to get rid of obsolete files and thereby free up disk space.

However, the Purge command may not always have that effect. If a file is being shared by more than one project, and you purge it in one project, no disk space is freed up, because the file is still being used. Only if you purge the file in all the projects that are using it is the file permanently deleted from your Visual SourceSafe database.

See Also

[Delete Command](#)

[Deleted Items Tab](#)

[Destroy Command](#)

[Recover Command](#)

[Share Command](#)

Recover Command

See Also **Command Line**

Replaces a file or project that had been deleted.

– **To use the Recover command**

- 1 Choose a project in the project pane of Visual SourceSafe Explorer.
- 2 On the File menu, choose Properties, and then choose the Deleted Items tab.
- 3 Choose Recover.

Access Rights

You must have the Add access right to use this command.

Remarks

When you use the Delete command to delete a file or project from Visual SourceSafe, the deleted items are removed from Visual SourceSafe Explorer. However, Visual SourceSafe continues to store them, so that it can rebuild old versions of the project that required them.

The Recover command reverses the Delete command and restores the file. After the file has been recovered, it is once again part of its parent project.

If the file being recovered is shared by other projects, it may have been modified while in those projects. Therefore, the file you recover may not be exactly the file you deleted.

When you delete a project, all the subprojects and files under it become unavailable. However, when you recover the project, its subprojects are also recovered.

See Also

[Delete Command](#)

[Deleted Items Tab](#)

[Purge Command](#)

[Set Working Folder Command](#)

Rollback Command

See Also [Command Line](#)

Eliminates all changes since an earlier version of a file.

– **To use the Rollback command**

- 1 Choose a file in the [file pane](#) of [Visual SourceSafe Explorer](#).
- 2 On the Tools menu, choose Show History.
- 3 Make selections in the File History Options dialog box, if it is enabled.
Note Enable the display of the File History Options dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).
- 4 Choose a previous version of the file, and then choose Rollback. When you rollback a file, you will lose all changes made to that file after the version you roll back to. For example, if you click on version 5 and then choose Rollback, all changes since version 5 are lost.

Access Rights

You must have the Destroy [access right](#) to use this command.

Remarks

Rollback returns a file to a previous [version](#), erasing all the versions and changes since then; all changes made after the version rolled back to are lost. The Rollback command cannot be reversed.

If the file you roll back is being [shared](#) by more than one project, the rollback affects only the project you specify (the [current project](#), by default). To accomplish this, Rollback performs an implicit [branch](#), that is, it branches the file in the specified project away from the same file in all other projects, and creates a new development path. To bring the disparate development paths back together, if you make changes to the specified project, you have to [merge](#) the files.

There are workarounds for undoing a rollback. For a shared file, you can delete the resulting file and re-share it from another project. For an unshared file, if you do not want to lose all the intervening steps, one method is to check out the file, get an earlier version, then check in the file. The earlier version you get then becomes the most recent version, but no information is lost.

See Also

[Check In Command](#)

[Check Out Command](#)

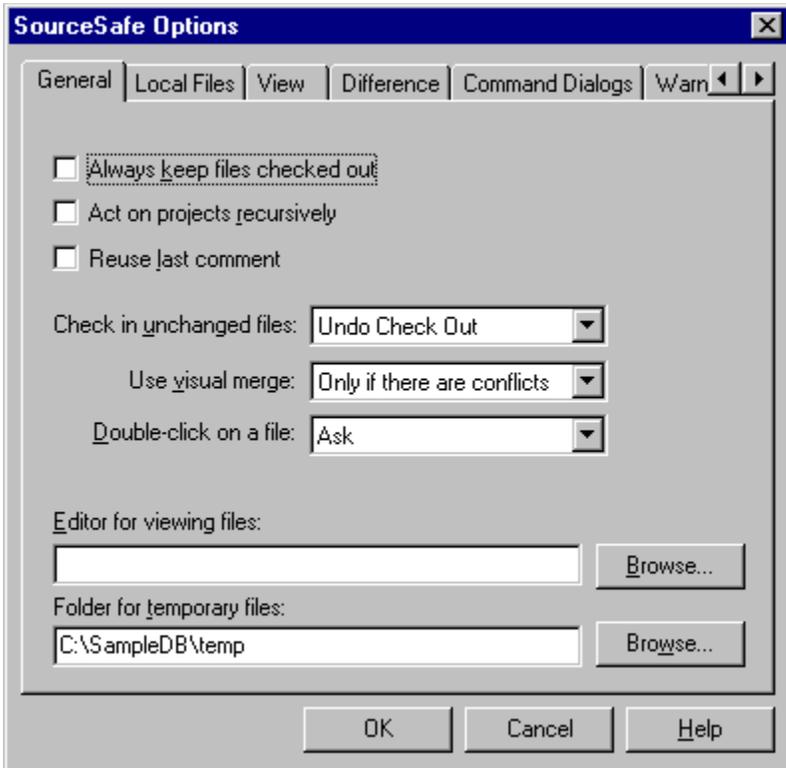
[Get Latest Version Command](#)

[Merge Branches Command](#)

[Tracking and Labeling Versions of Files and Projects](#)

General Options Tab

See Also



Sets and changes general configuration options.

- **To display the General tab from Visual SourceSafe**
 - On the Tools menu, choose Options, and then choose the General tab.

Access Rights

Any Visual SourceSafe user can use this tab.

Tab Options

Always Keep Files Checked Out

Keeps a file checked out to your working folder, even if you check it in. With this option, a file is copied to the Visual SourceSafe database, but left checked out to you, so that you can update the database with your changes, but continue working on the file.

Act On Projects Recursively

Performs all actions on a selected project recursively, that is, the command is applied to the subprojects (and their subprojects, and so forth) as well as to the selected project. This is useful if you have a project with many subprojects that you want to check out and check in as a unit.

Reuse Last Comment

Uses the last comment for the next command. For example, if you check in a file with the comment "Added hyperlink jumps to our Help menu," and then check another file in, the comment "Added hyperlink jumps to our Help menu" is automatically entered in the comment box. If you modify the comment, Visual SourceSafe uses the changed comment for the next command.

Check In Unchanged Files

When you check in a file, and you have made no changes to that file, Visual SourceSafe looks at this option to determine what to do. Choose Check In to have SourceSafe automatically update the Visual SourceSafe database even if Visual SourceSafe detects no changes to the file being checked in. This also increments the file's version number. Choose Undo Check Out to have SourceSafe automatically undo the check out of any unchanged file you try to check in. Choose Ask to have SourceSafe display a dialog box asking what you want to do for each unchanged file you try to check in. Choose Default to have SourceSafe depend on the specific initialization file settings for each project to determine what to do with unchanged files.

Use visual merge

Choose Yes to always use SourceSafe's Visual Merge feature when merging files after a branch or multiple checkout. Choose No to use SourceSafe's manual merging feature (used in previous versions of SourceSafe). Choose "Only if there are conflicts" to use SourceSafe's Visual Merge feature when there are conflicting changes in the two files to be merged, and use SourceSafe's automatic merge otherwise.

Double-click on a file

This option controls what command Visual SourceSafe executes when you double-click on a file in the Visual SourceSafe Explorer. Choose Ask (the default) to always ask whether to view a file or edit the file on a double-click selection from the Visual SourceSafe Explorer. Choose View File to always view the Visual SourceSafe database version of this file (without being able to edit it), or choose Edit File to always edit the file (open the file, with checked-out status, in your working folder). Holding the Shift key down while double-clicking a file will override this setting to Ask.

Editor For Viewing Files

Sets the editor used to display files when you select the View command. Choose Browse to display the Open dialog box and select an editor.

Folder For Temporary Files

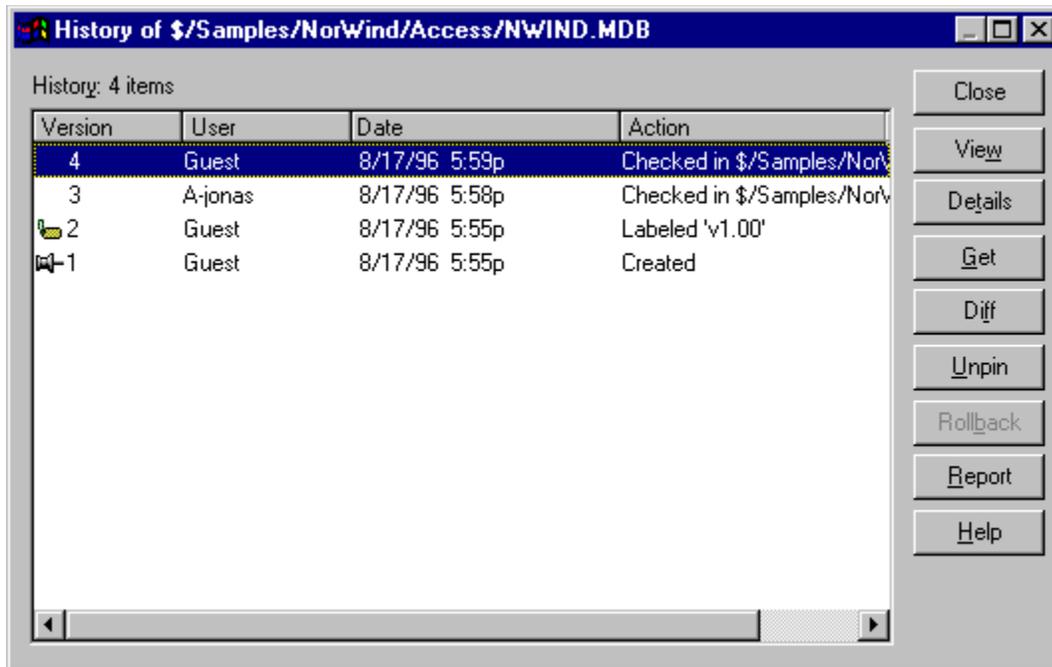
Sets a folder in which to store temporary files used by Visual SourceSafe. The default is the TEMP subfolder under the folder in which SRCSAFE.INI is stored. Choose Browse to display the Select Temporary Folder dialog box.

See Also

[Visual Merge](#)

History of File Dialog Box

See Also



Shows historical information for the selected file.

– To display the History of File dialog box

- 1 Choose a file in the file pane of Visual SourceSafe Explorer.
- 2 On the Tools menu, choose Show History.

Makes selections in the File History Options dialog box if it is enabled, and choose OK.

Note Enable the display of the File History Options dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu). The dialog box gives you the option to restrict the history to labels only, to a specified user, or to a range of dates.

Access Rights

You must have the Read access right to use this dialog box.

Remarks

The History of File dialog box displays the history of a selected file, including all significant events, such as the creation date, check in dates, and current version number. Events are listed with the most current event at the top of the display. In Visual SourceSafe, all access to old versions of files is through this dialog box.

Several commands are available from this dialog box. You can perform a command on a version by first selecting an event (or multiple events) and then clicking a command button.

Dialog Box Options

Close

Closes the dialog box and returns to Visual SourceSafe Explorer.

View

Opens an editor, in which you can view the selected file as it is stored in the Visual SourceSafe database. The editor can be set on the General tab in the SourceSafe Options dialog box.

Details

Displays comments associated with an event, along with more detailed event information, such as label information, version information, check-out comments, and so forth.

Get

Performs the Get Latest Version command on the selected version.

Diff

Performs the Show Differences command on the selected file. You can compare the selected version of a file with the current version, to any other version, or with a file in your working folder. To compare two different versions of a file, hold down the Ctrl key while selecting the two versions of interest. Then choose the Diff button.

Pin/Unpin

Performs the Pin or Unpin command on the selected file. (This button toggles between Pin and Unpin depending on the pinned state of the selected file.)

Rollback

Performs the Rollback command on the selected file or subproject. The Rollback command discards all versions of a file after the selected version, and returns the current version of the file to the selected version.

Report

Sends the file history information to the Clipboard, a file, or a printer.

See Also

[Command Dialogs Options Tab](#)

[General Options Tab](#)

[Get Latest Version Command](#)

[History Details Dialog Box](#)

[Rollback Command](#)

[Share Command](#)

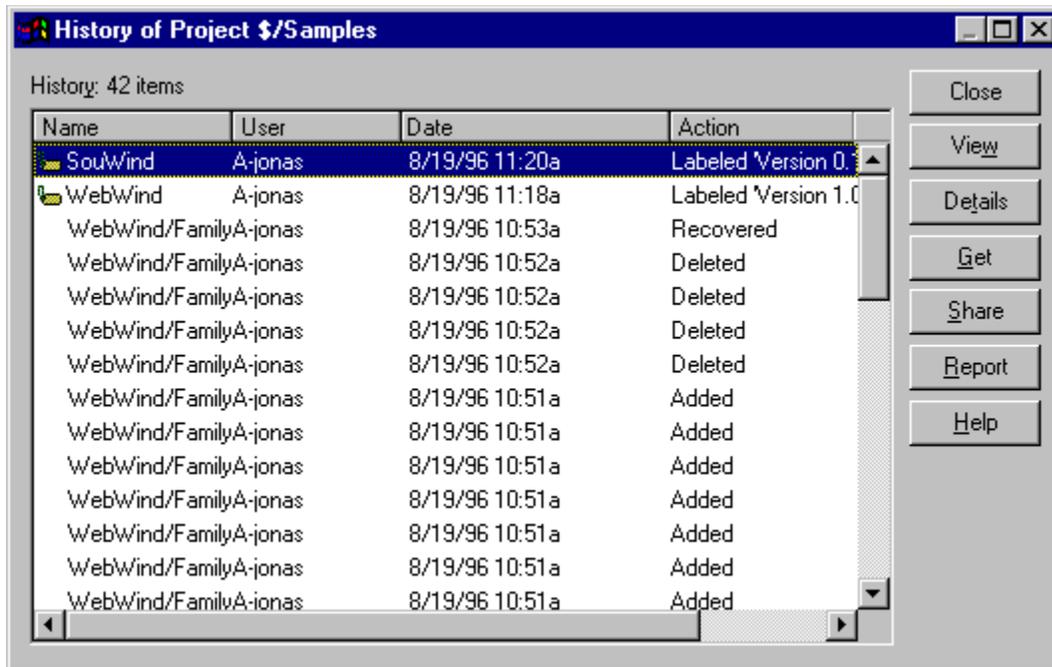
[Show Differences \(File\) Command](#)

[Show History Command](#)

[View Command](#)

History of Project Dialog Box

See Also



Shows historical information for the selected project.

– To display the History of Project dialog box

- 1 Choose a project in the project pane of Visual SourceSafe Explorer.
- 2 On the Tools menu, choose Show History.

Make selections in the Project History Options dialog box if it is enabled, and choose OK.

Note Enable the display of the Project History Options dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu). The dialog box gives you the option to restrict the history to labels only, to include files in the project history list, to make the history list recursive down the project hierarchy, or to restrict the history list to a specified user, or to a range of dates.

- 3 Choose OK.

Access Rights

You must have the Read access right to use this dialog box.

Remarks

The History of Project dialog box displays the history of a selected project, including all significant events, such as the deletion of files or subprojects, addition of files or subprojects, labeling and renaming of items, and check ins. For each significant event, the dialog box shows the affected file or subproject, the user who performed the action, the date and time of the event, and the description of the event. Events are listed with the most current event at the top of the display.

Several commands are available from this dialog box. You can perform a command on a file or subproject by first selecting an event and then clicking a command button.

Dialog Box Options

Close

Closes the dialog box and returns to Visual SourceSafe Explorer.

View

Displays a dialog box containing a list of all files (with their version numbers) as the project existed at the selected point in the project's history.

Details

Shows the comments associated with an event, along with more detailed event information, such as label information, version information, and so forth.

Get

Performs the Get command on the selected file or subproject version. If you select a file, Visual SourceSafe displays a dialog box asking you to clarify whether you want to get just the selected file, or to get (that is, re-create) the entire project as it existed at the point in time the selected file version existed.

Share

Performs the Share command on the selected file or project. With this command, you can share the current or an old version of the project files with another project. This lets you create a branch from an old version of your project. You may want to do this if you need to fix some bugs for a customer while continuing development on the next version of your product.

Report

Sends the project history information to the Clipboard, a file, or a printer.

See Also

[General Options Tab](#)

[Get Latest Version Command](#)

[History Details Dialog Box](#)

[Share Command](#)

[Show History Command](#)

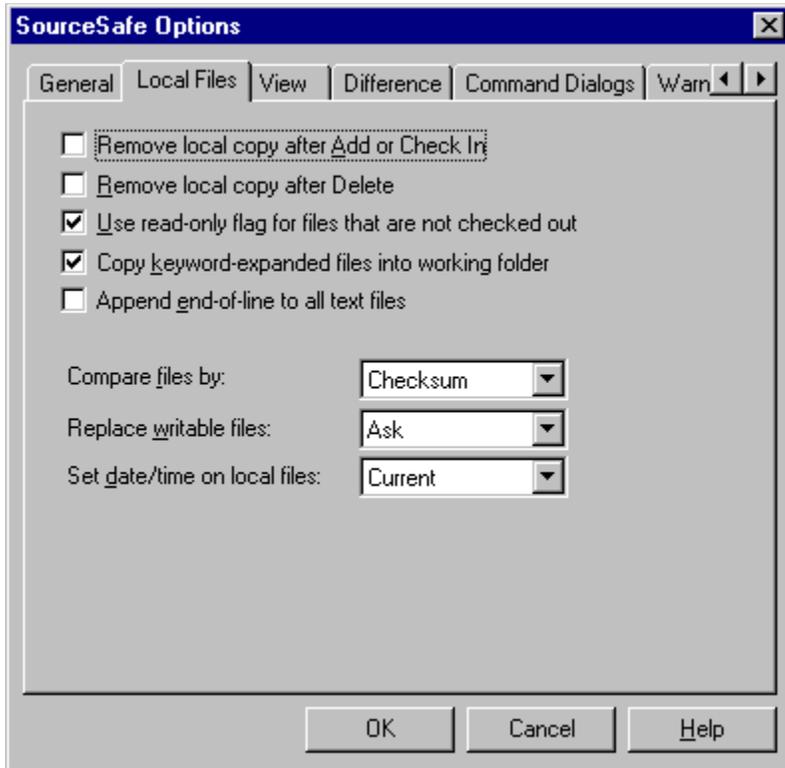
[Show Differences \(Project\) Command](#)

[Rollback Command](#)

[View Command](#)

Local Files Options Tab

See Also



Sets or changes working folder options.

– **To display the Local Files tab**

- On the Tools menu, choose Options, and then choose the Local Files tab.

Access Rights

Any Visual SourceSafe user can use this tab.

Tab Options

Remove Local Copy After Add or Check In

Automatically deletes the working folder copy of files you add or check in. This helps keep your disk usage to a minimum, and ensures that you always get the most recent version when you start work on a file.

Remove Local Copy After Delete

Automatically deletes the working folder copy of a file you delete with the Delete command. This helps keep disk clutter to a minimum. And, as long as you haven't destroyed or purged the file in the Visual SourceSafe database, you can always recover the file later.

Use Read-Only Flag for Files That Are Not Checked Out

Sets the read-only flag on files in your working folder. Visual SourceSafe assumes that files in your working folder that you don't have checked out are read-only. Visual SourceSafe sets the read-only flag when you perform a Get Latest Version, Add, Check In, or Undo Check Out command on a file, and uses it as a signal that you have not been modifying the file (so Visual SourceSafe can replace

it).

Copy Keyword-Expanded Files Into Working Folder

Copies modified files directly back into your working folder. When you check in a file and have keyword expansion enabled, Visual SourceSafe may replace certain keywords in your file with version information. The modified file is replaced directly into Visual SourceSafe. If you do not set this option, the file is not automatically copied back; this improves performance, but the copy in your working folder may not be completely up-to-date.

Append End-Of-Line To All Text Files

Appends an end-of-line character whenever Visual SourceSafe retrieves a text file that does not already end with one. Some compilers require that all the files they compile end in an end-of-line character. If you are working in such an environment, you may want to set this option.

Compare Files By

Sets how Visual SourceSafe determines if your local copy is up-to-date. The following options are in order of increasing speed:

- **Contents**—Compares the full contents of your local file to Visual SourceSafe's copy. This is the slowest method, but is most likely to work.
- **Checksum**—Compares by a checksum stored by Visual SourceSafe.
- **Time**—Compares the date/time flags. The exact method of date comparison depends on the setting of the Set Date/Time On Local Files option: If set to Check In, Visual SourceSafe replaces the local copy if the time of the last check in (or add) is more recent than the local copy's modification date. If set to Current (the default) or Modification, Visual SourceSafe replaces the local file if Visual SourceSafe file's modification time is different from the local file's modification time.

Replace Writable Files

When you get or check out a file, Visual SourceSafe copies the file into your working folder. If there is already a writable copy of the file in your working folder, replacing the file may erase your work.

- **Ask**—Gives you an option each time.
- **Replace**—Replaces (overwrites) the file.
- **Skip**—Does not replace the file, and generates an error in the Results pane if you have it displayed.
- **Merge**—Merges the changes between the checked out-file and the one you are now getting or checking out. If the file cannot be merged (i.e. it is a binary file), an error is displayed in the Results pane if you have it displayed. Your Visual SourceSafe Administrator must enable multiple checkouts on the files involved for this to work properly.

Set Date/Time on Local Files

This option controls the date/time that SourceSafe gives files in your working folder. The option you choose here is the time that you will see when you view files in the Windows Explorer

- **Current**—Sets the date and time of the file to the current date and time by default, whenever Visual SourceSafe gets a file. This signals the compiler that the file has just changed, and needs to be recompiled.
- **Modification**—Sets the local copy to the date and time the file was last modified.
- **Check In**—Sets the local copy to the date and time the file was last checked in.

See Also

[Add Files Command](#)

[Check In Command](#)

[Delete Command](#)

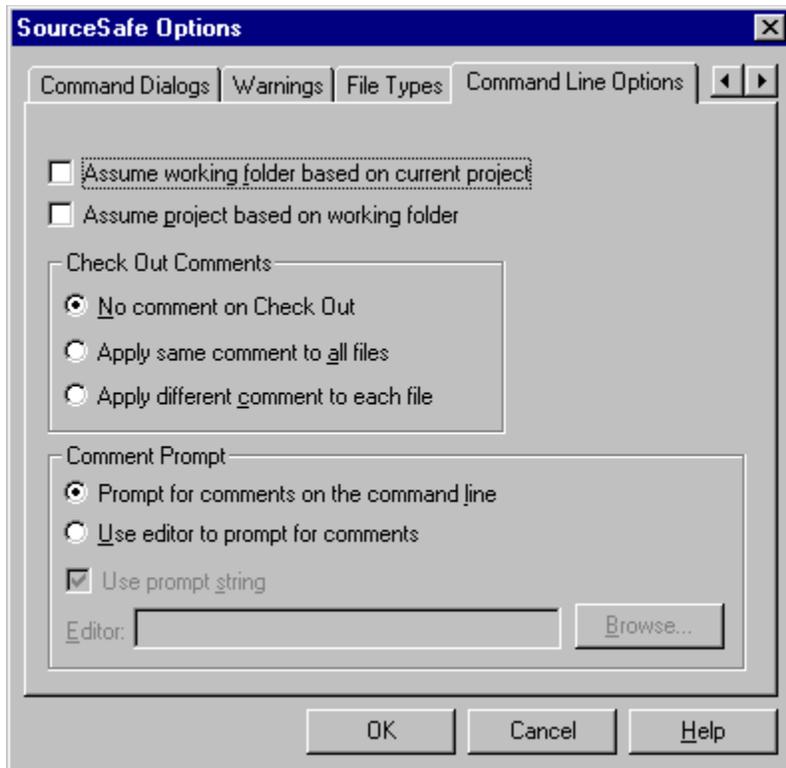
[Destroy Command](#)

[Purge Command](#)

[Recover Command](#)

[The Visual Merge](#)

Command Line Options Tab



Controls options that affect Visual SourceSafe's command line. These options don't affect the behavior of Visual SourceSafe Explorer.

– To display the Command Line tab

- On the Tools menu, choose Options, and then choose Command-Line Options.

Access Rights

Any Visual SourceSafe user can use this tab.

Tab Options

The first two options affect how Visual SourceSafe interacts with your working folder during command-line operations. By default in the Explorer, Visual SourceSafe expects a working folder to be set for each project, and will check out files to that folder only. By default on the command line, Visual SourceSafe assumes your current project and current folder are what you want to be working with, and ignores any working folders set for a specific project. To change current projects, you need to use the CP command. You can modify this behavior by checking one or the other of the following two options:

Assume Working Folder Based On Current Project

Determines how the working folder you've set is used:

- If this option is selected, a file is put into your working folder for the project, even if that is not the current folder when you execute a command on a file from the command line.
- If this option is not selected, subprojects go into their own respective working folders instead of subfolders under the main folder.

Assume Project Based On Working Folder

Automatically sets the current project, based on the current folder. If set, SourceSafe looks at your current folder whenever you execute any Visual SourceSafe command. If this is the working folder for a Visual SourceSafe project, Visual SourceSafe automatically changes to that project before executing the command. Hence, you can change folders to move around your projects, without ever executing the ss Cp command. If you set this option, it is highly recommended that you do not give two projects the same working folder, because Visual SourceSafe won't know which project to switch into.

If not set, SourceSafe does not change to your current project when you execute any Visual SourceSafe command.

No Comment On Check Out

Does not prompt for a comment on check out.

Apply Same Comment To All Files

Prompts you for one comment only (up to 64 characters), which Visual SourceSafe applies to all files involved in that check out operation.

Apply Different Comment To Each File

Prompts you for a separate comment for each file involved in the check out operation.

Prompt For Comments On The Command Line

Asks for comments on the command line.

Use Editor To Prompt For Comments

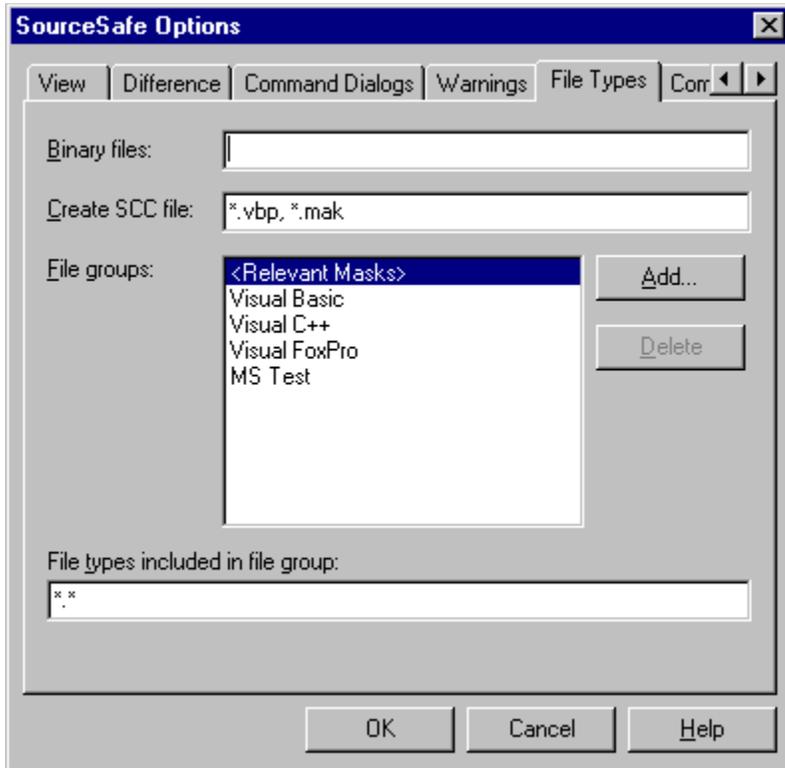
Lets you specifies an editor with which to enter your command-line comments. Once you have selected this option, you can enter the path for the selected editor in the Editor box. You can browse your file system for this information using the Browse button. If you leave the Prompt String box blank, no prompt is displayed in your editor.

Use prompt string

Specifies that a prompt string asking for a comment is inserted into your specified editor when it opens. Disabled when Prompt for comments on the command line is selected. The prompt string is not saved with your comment (you don't have to delete it), unless you change the string in your editor.

File Types Options Tab

See Also



Sets options relating to the types of files you store in Visual SourceSafe.

- **To display the File Types tab**
 - On the Tools menu, choose Options, and then choose File Types.

Access Rights

Any Visual SourceSafe user can use this tab.

Remarks

Many of the Visual SourceSafe dialog boxes contain file lists to let you select multiple files for a specific operation. Use this tab to filter these file lists so that they are more meaningful to the type of work you do. For example, if you are a C++ developer, you may want to store .CPP, .H, .RC, and .BMP files in Visual SourceSafe. If you are a Visual Basic developer, on the other hand, you may want to store .VBP and .FRM files instead. You can use this tab to restrict the file lists to display just these file types rather than all the files in a specific folder.

Tab Options

Binary Files

Overrides Visual SourceSafe's automatic detection of text and binary files, so that you can indicate that certain file types are always binary. For example, you can enter *.OBJ, *.XLS, *.FRX in this box to specify that all object files, Microsoft Excel worksheets, and Visual Basic .FRX files should always be treated as binary files.

Create SCC File

This box is useful only for Visual Basic users. When you work with a Visual Basic project in SourceSafe, a special file is created in your working folder called MSSCCPRJ.SCC. This file can never be added to SourceSafe: it stays in your working folder and stores important information used by the SourceSafe integration in Visual Basic.

Create SCC File dictates when SourceSafe creates this file. Its default setting (*.VBP, *.MAK) means that whenever you get or add a file with an extension of .VBP (Visual Basic 4.0 project file) or .MAK (used by older versions of Visual Basic), SourceSafe creates the MSSCCPRJ.SCC file. This file is very important because without it, SourceSafe integration inside Visual Basic will not function properly.

On the other hand, Visual C++ and other development environments do not use this file at all. You may not want this useless file appearing in your Visual C++ folder. To prevent the appearance of the MSSCCPRJ.SCC file in your Visual C++ folder, remove the *.MAK from this field. (You should make this change only if you do not use Visual Basic 3.0 .MAK files!)

File Groups

Together with the "File types included in file group" option, this option groups file types into logical groups. These groups are displayed in the List Files of Type options in many Visual SourceSafe dialog boxes, so you can specify groups that filter the file list efficiently for your work.

Use the Add button to add a new group of file extensions to the list, and the Delete button to delete the currently selected group. The following example would create file groups including .FRM files, .BAS files, but explicitly not the HELP.FRM file:

```
*.frm; *.bas; !HELP.FRM
```

The <Relevant Masks> file group is a special case. This group cannot be deleted, and indicates which files you would typically place into a project. This does not prevent you from placing other files into the project; however, it does have several potentially useful effects.

- 1** If you add an entire folder into Visual SourceSafe in one command (by typing `ss add *` on the Visual SourceSafe command line, or dragging a folder from File Manager into Visual SourceSafe), only files matching the relevant masks are added. Other files can then be added individually.
- 2** When you perform the Visual SourceSafe Show Differences command on a project level to compare the current project to your working folder, files that do not match the relevant masks are not displayed as differences.
- 3** The relevant masks are used as the default option under Add Files of Type in the Add Files dialog box; therefore, by default, only these files are displayed.

File Types Included In File Group

Shows the list of file types (extensions) included in the currently selected file group under File Groups. To modify a specific group, select the group under File Groups, and then add or delete extensions.

Add

Adds a new group of file extensions to the File Groups list.

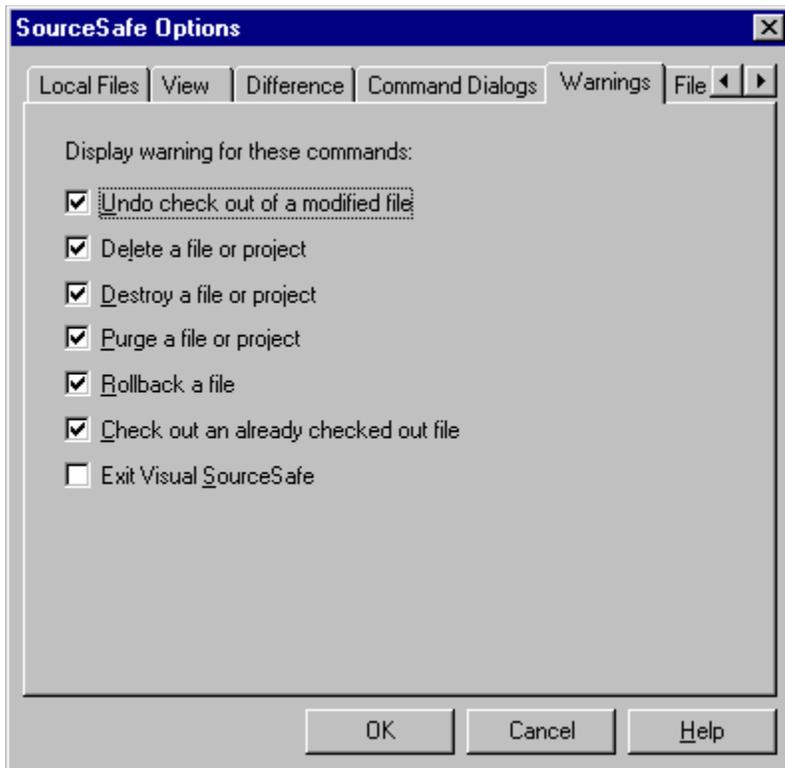
Delete

Deletes the currently selected File Group.

See Also

[Setting the File Type: Text and Binary Files](#)

Warnings Options Tab



Specifies whether warnings are displayed before actions that might result in accidental loss or deletion of data.

– To display the Warnings tab

- On the Tools menu, choose Options, and then choose Warnings.

Access Rights

Any Visual SourceSafe user can use this tab.

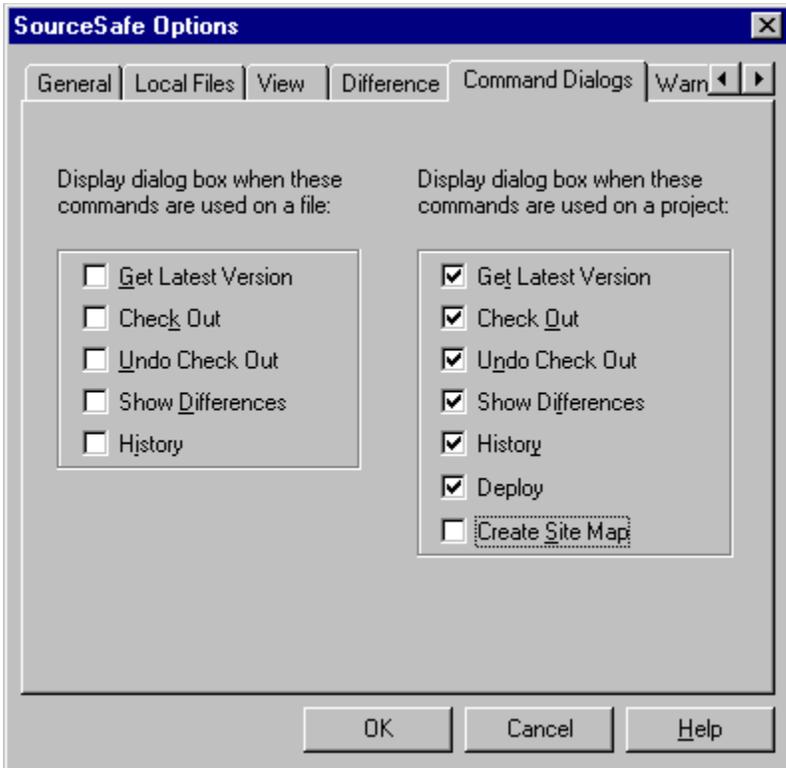
Remarks

When you delete a file from the Visual SourceSafe database, Visual SourceSafe asks "Are you sure you want to delete?" Some users find the warning helpful, because it prevents accidental deletion of data. Others find it annoying, because it requires an extra keystroke. This and other Visual SourceSafe warnings are therefore entirely configurable.

You can choose whether you want to receive warnings before Visual SourceSafe executes the following:

- Undo check out of a modified file
- Delete a file or project
- Destroy a file or project
- Purge a file or project
- Rollback a file
- Check out an already checked out file
- Exit Visual SourceSafe

Command Dialogs Options Tab



Specifies whether command dialogs are displayed with commands.

– To display the Command Dialogs tab

- On the Tools menu, choose Options, and then choose Command Dialogs.

Access Rights

Any Visual SourceSafe user can use this tab.

Remarks

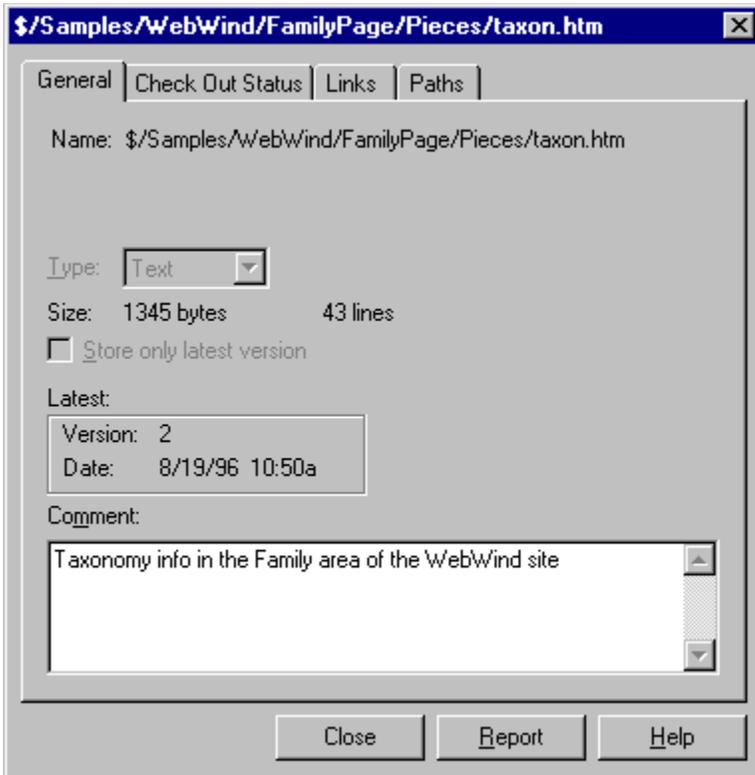
If you never use the options provided by, for example, the Get Latest Version dialog box, you can turn it off so that Visual SourceSafe no longer displays it when you execute the Get Latest Version command. You can specify behavior depending on whether a project or a file is selected with the following five commands.

- Get Latest Version
- Check Out
- Undo Check Out
- Show Differences
- Show History

You can also specify command dialogs for the project-level Deploy and Create Site Map commands.

You can override any setting you make in this dialog by holding down the SHIFT key while you select one of the above commands. In this case, the command dialog will be displayed even if you do not have the corresponding box checked.

General File Properties Tab



Displays general properties for the selected file.

- **To display the General File Properties tab**
 - 1 Choose a file in the file pane of Visual SourceSafe Explorer.
 - 2 On the File menu, choose Properties, and then choose General.

Note To display this tab from within Visual Basic, after adding a Visual Basic project to Visual SourceSafe, on the Visual Basic Add-Ins menu, choose SourceSafe, and then choose SourceSafe Properties.

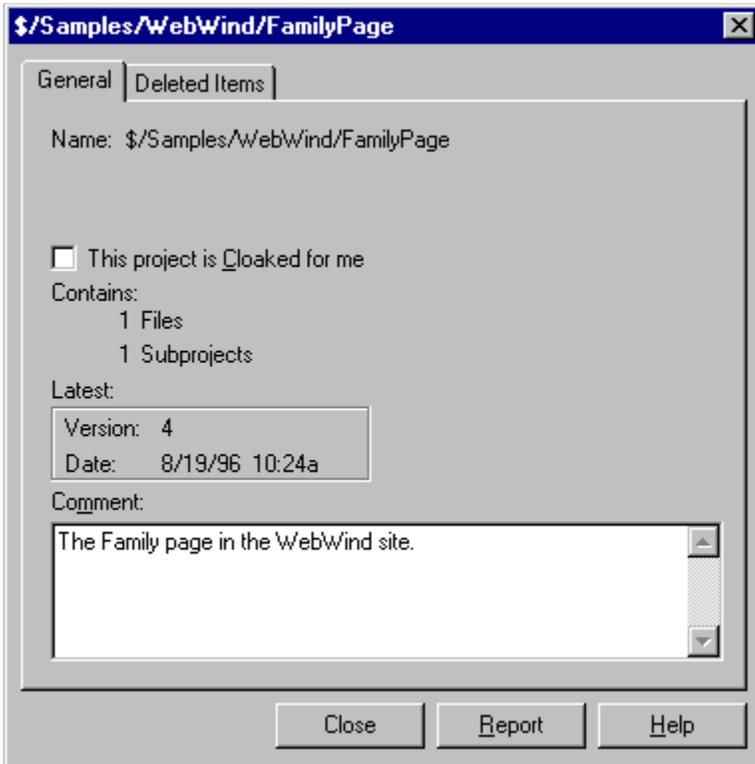
Access Rights

You must have the Read access right to use this tab.

Remarks

If you have a file selected, the General File Properties tab shows you the filename, file type (text or binary), size, whether all versions or just the latest version are stored in the Visual SourceSafe database, the version number and version date, and any comment applied to the file. It also shows the latest label applied to the file, if there is one. You can change the comment, file type, and whether just the latest version is stored from this dialog. If you change a file from storing just the latest version to storing all versions, the change is effective from the time you make it, and is not retroactive.

General Project Properties Tab



Shows properties for the selected project.

To display the General Project Properties tab

- 1 Choose a project in the project pane of Visual SourceSafe Explorer.
- 2 On the File menu, choose Properties, and then choose General.

Note To display this tab from within Visual Basic, after adding a Visual Basic project to Visual SourceSafe, on the Visual Basic Add-Ins menu, choose SourceSafe, and then choose SourceSafe Properties.

Access Rights

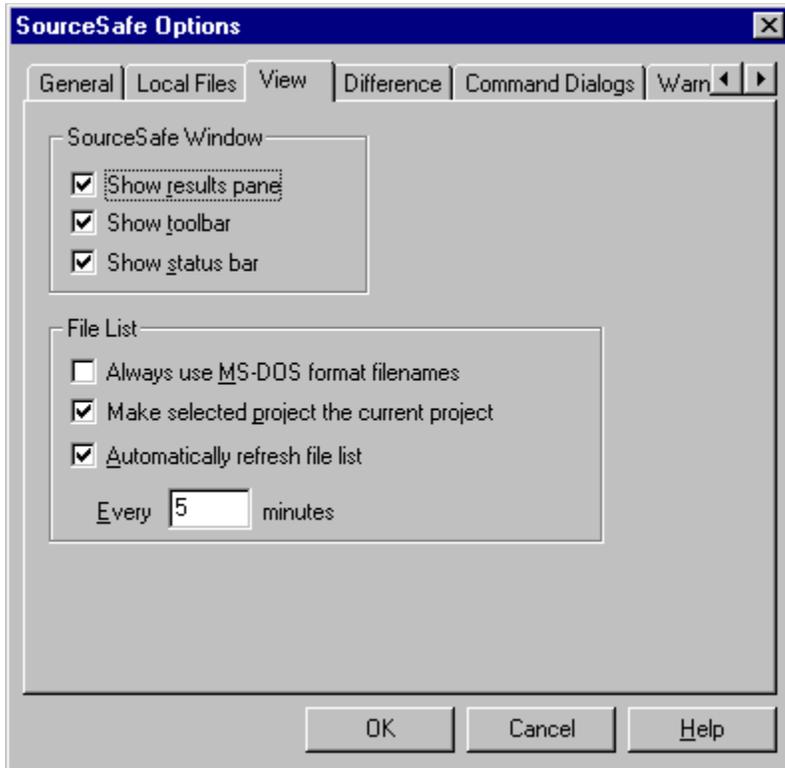
You must have the Read access right to use this tab.

Remarks

If you have a project selected, the General Project Properties tab shows you the project name, any folder this project is shadowed to, the number of files and subprojects in the project (including the number of deleted files and subprojects, if any), the version number and date of that version, and the comment for the project. It also shows the latest label applied to the project, if there is one. From this tab, you can change the project comment.

This tab also contains the checkbox titled "This project is cloaked for me". If this box is checked for this project, a Get, Check Out, Check In, Undo Check Out, or Project Show Difference commands that include this project recursively will not be applied to this project. This feature is meant for people who prefer to have an entire project's structure on their local computer, but do not want to take up space with files they don't work on. Cloaking is described more fully in the topic Cloaking Visual SourceSafe Projects.

View Options Tab



Sets Visual SourceSafe Explorer view options.

- **To display the View tab**
 - On the Tools menu, choose Options, and then choose View.

Access Rights

Any Visual SourceSafe user can use this tab.

Tab Options

Show Results Pane

Shows or hides the Results pane in Visual SourceSafe Explorer. The Results pane, at the bottom of the Visual SourceSafe Explorer, shows you the results of your Visual SourceSafe commands.

Show Toolbar

Shows or hides the toolbar in Visual SourceSafe Explorer. The toolbar contains a set of buttons you can choose to carry out commonly used commands.

Show Status Bar

Shows or hides the status bar in Visual SourceSafe Explorer. The status bar shows you status information about Visual SourceSafe, such as the current file pane sort order, the current username, and the number of items in the file list.

Always Use MS-DOS Format Filenames

Truncates the long filenames supported on 32-bit platforms. This option keeps the filename column of the display narrow, and is also useful if you have computers with several different operating systems using your Visual SourceSafe database.

Make Selected Project the Current Project

This option is normally selected. However, if your network suffers from poor performance, you may want to be able to scroll the project list without Visual SourceSafe updating the current project as you do so. In this case, turn off this option. If it is turned off, you must press the ENTER key on a project (or double-click with the mouse) to make it the current project.

Automatically Refresh File List

Controls whether Visual SourceSafe automatically refreshes its file list. Refreshing keeps Visual SourceSafe Explorer up-to-date, but it also causes a momentary delay (and a network load). If you select this option, enter the refresh interval in minutes in the box below. If you do not select this option, you can manually refresh the file list using the Refresh File List command on the Visual SourceSafe View menu.

Add File Group Dialog Box

See Also

Adds a new file group to the File Types tab in the SourceSafe Options dialog box (Tools menu).

- **To display the Add File Group dialog box**
 - On the Tools menu, choose Options, and then choose File Types. Choose the Add button.

Dialog Box Options

Name

Type the name of the file group as you would like to identify it in the file group list on the File Types tab.

OK

Adds the file group.

Cancel

Cancel the Add File Group command.

See Also

[File Types Options Tab](#)

Checkouts Dialog Box

Shows the various locations to which you have a file checked out.

– **To display the Checkouts dialog box**

- On the SourceSafe menu, choose File Properties and select the Check Out Status tab.

Dialog Box Options

Check Outs

Lists the locations that the file is checked out to.

Details

Shows check out status for the selected file.

Check In

Checks in the selected file.

Undo Check Out

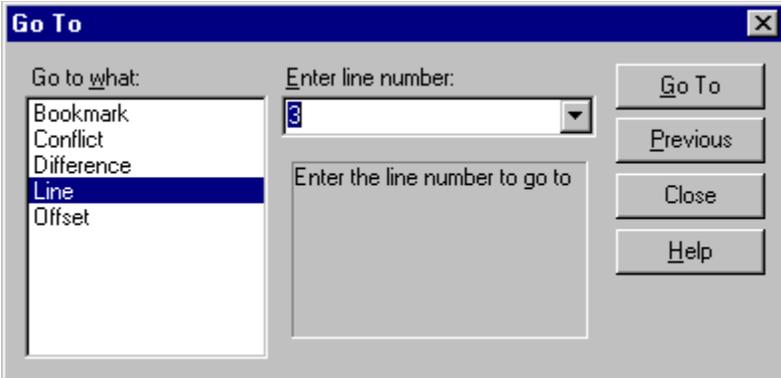
Undoes the check out of the selected file.

Close

Closes the dialog box, and cancels the command.

GoTo Dialog Box

Jumps to a specified position in a file.



To display the GoTo dialog box

- 1 Select a file in the file pane of Visual SourceSafe Explorer.
- 2 On the Tools menu, choose Show Differences, and in the File Difference dialog box choose OK.
Note Enable the display of the File Difference dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).
- 3 In the Difference for File dialog box, choose Report.
- 4 In the File Difference Report dialog box, choose Preview.
- 5 Press ALT+G.

Dialog Box Options

Go to what

Select the type of item you want to go to. Different options are available depending on the operation you are performing, for example, you can go to a bookmark, conflict, difference, line or offset (the number of bytes from the beginning of the file). You may want to use File Offset if you are receiving an error during loading after a certain byte, or if you have a file with a fixed byte length and need to find a specific byte in the file.

Enter position number

Type the position to which you want to go. (Position is relative to the type specified under Go to What.)

Go To

Goes to the specified location.

Previous

Go to the previous item of type specified in the Go to What field.

Close

Cancels the GoTo command.

History Report Dialog Box

See Also

Sends a report of historical information on a file or project to a printer, a file, or the Clipboard.

– **To display the History Report dialog box**

- 1 Select a file or project in Visual SourceSafe Explorer, and on the Tools menu, choose Show History.
- 2 Make selections in the File History Options or Project History Options dialog box (if enabled), and choose OK.

Note Enable the display of the File History Options dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

- 3 In the History of File or History of Project dialog box, choose Report.

Dialog Box Options

Include Details

Includes all historical details of the selected file(s) in the report. Details include creation date, user who modified the file for each check in, comments applied to the file, and time/date information for each event.

Include Differences

This option is useful when you are comparing two different versions of a file or project. The report includes differences between the selected files or projects in the report.

Report To

Sends the history report to a printer, a file, or the Clipboard.

OK

Prepares the specified report.

Cancel

Cancels the report.

Preview

Shows a preview of the specified report, so that you can determine if it is what you wanted before committing it to the printer queue or a file.

See Also

[Show History Command](#)

Print Dialog Box

See Also

Sets printing options.

– **To display the Print dialog box**

- 1 Select a file or project in Visual SourceSafe Explorer, and on the Tools menu, choose Show History.
- 2 Make selections in the File History Options or Project History Options dialog box if it is enabled, and choose OK.
Note Enable the display of the History Options dialog boxes on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).
- 3 In the History of File or History of Project dialog box, choose Report.
- 4 Under Report To, choose Printer, and then choose OK.

Dialog Box Options

Printer

Printer selected in the Print Setup dialog box. To display the Print Setup dialog box, choose Setup.

Copies

Type the number of report copies to print.

Print To File

Sends the report to a PostScript file, and prints the report.

Setup

Displays the Windows Print Setup dialog box.

See Also

[Creating Reports](#)

[Files Report Command \(Tools Menu\)](#)

[History Report Dialog Box](#)

[Show History Command \(Tools Menu\)](#)

Report to File Dialog Box

See Also

Designates a file in which to save a report.

– **To display the Report to File dialog box**

- On Tools menu, choose Files Report. Under Report To, choose File, and then choose OK.

Dialog Box Options

File Name

Name and path of the file to write the report to.

Folders

Specifies a path to the file.

Drives

Lets you select from the available drives.

List Files of Type

Filters the file list to display only files of the specified type.

Append

Appends the report to the end of an existing file.

OK

Confirms selection.

Cancel

Cancels the selections, closes the dialog box, and returns to [Visual SourceSafe Explorer](#).

Network

Displays the Windows Connect Network Drive dialog box.

See Also

[Creating Reports](#)

[Files Report Command \(Tools Menu\)](#)

Share From Project Dialog Box

See Also

Shares a project with the current project, and pins it to a past version.

– **To display the Share From Project dialog box**

- 1 In the project pane of Visual SourceSafe Explorer, choose a project into which you want to share another project.
- 2 On the Tools menu, choose Show History. Make selections in the Project History Options dialog box if it is enabled, and choose OK.
Note Enable the display of the Project History Options dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).
- 3 In the History of Project dialog box, choose Share.

When you use this dialog box, what you are going to do is create a brand new project, which is identical to an old version of this project. For instance, suppose you could create a project that is exactly identical to \$/A/B as it existed last Monday at midnight. Your new project is named, let's say, \$/A/C. C is identical to B--has all the same subprojects and files--but the files in C represent old versions of the files in B. There are two ways that this can happen. The files can be pinned to the old versions (but still shared with their counterparts in B), or they can be rolled back to those versions (no longer shared with B); you choose between these two methods based on whether you choose the Branch after share checkbox.

To do this as described in the example above, you would choose B (the place you want to share from; note that this is the reverse of the normal share direction), go into the History dialog box, and choose the Share button. Then, specify the location and name of the new project, whether you want to branch, whether you want to do it recursively, and so on.

Dialog Box Options

Projects

Shows the project list so that you can browse for the project to share with the current project.

Branch After Share

Branches immediately after sharing, rolling back the project you just created to the specified version. With this option, you can easily create a separate version of the project as it was in a former version.

OK

Shares the selected project into the current project and pins it to the selected version. If the selected project is the current version, the shared project is pinned at that version.

Cancel

Cancels the Share From Project command.

See Also

[Pin Command](#)

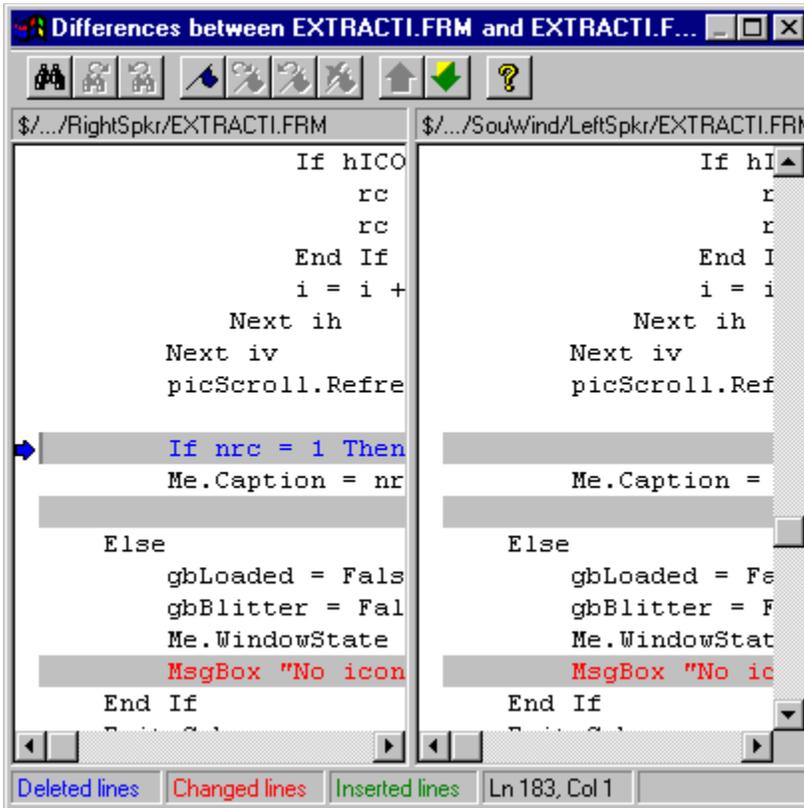
[Rollback Command](#)

[Share Command](#)

[Sharing Files](#)

Differences For File Dialog Box

See Also



Shows differences between a working folder version of the selected file and the Visual SourceSafe database version of the file.

– To display the Differences For File dialog box

- 1 Select a file in the file pane of Visual SourceSafe Explorer.
- 2 On the Tools menu, choose Show Differences.
- 3 If the files compared are the same, a message tells you the files are identical. Visual SourceSafe cannot display differences for binary files; it can only tell you whether or not binary files differ. If text files differ, Visual SourceSafe displays the File Difference dialog box.
- 4 Make selections in the File Difference dialog box if it's enabled, and choose OK.

Note Enable the display of the File Difference dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

Remarks

The Difference For File dialog box displays two files side-by-side. It uses three colors to show deleted, modified, and changed lines. You can change the colors by setting the options on the Difference tab of the Tools Options dialog box. You cannot make changes to the files from this dialog box—it is for display of differences only.

Note To compare a current Visual SourceSafe file to an older version of the same file, choose the file in the file pane of Visual SourceSafe Explorer, and on the Tools menu, choose Show History. Then choose Diff.

Dialog Box Options

This dialog has a toolbar containing all the commands you'll need for finding and marking differences:



Find

Lets you search for a string in the file. The shortcut key is Alt+F3.



Find Next

Lets you search for the next matching occurrence of a string in the file. The shortcut key is F3.



Find Previous

Lets you search for a previous matching occurrence of a string in the file. The shortcut key is Shift+F3.



Toggle Bookmark

Lets you turn a bookmark on or off in a file. The shortcut key is Ctrl+F2.



Next Bookmark

Lets you go to the next bookmark in the file. The shortcut key is F2.



Previous Bookmark

Lets you go to the previous bookmark in the file. The shortcut key is Shift+F2.



Clear All Bookmarks

Lets you clear all bookmarks in the file.



Previous Difference

Lets you find the previous difference in the file. The shortcut key is Shift+F7.



Next Difference

Lets you search for the next difference in the file. The shortcut key is F7.

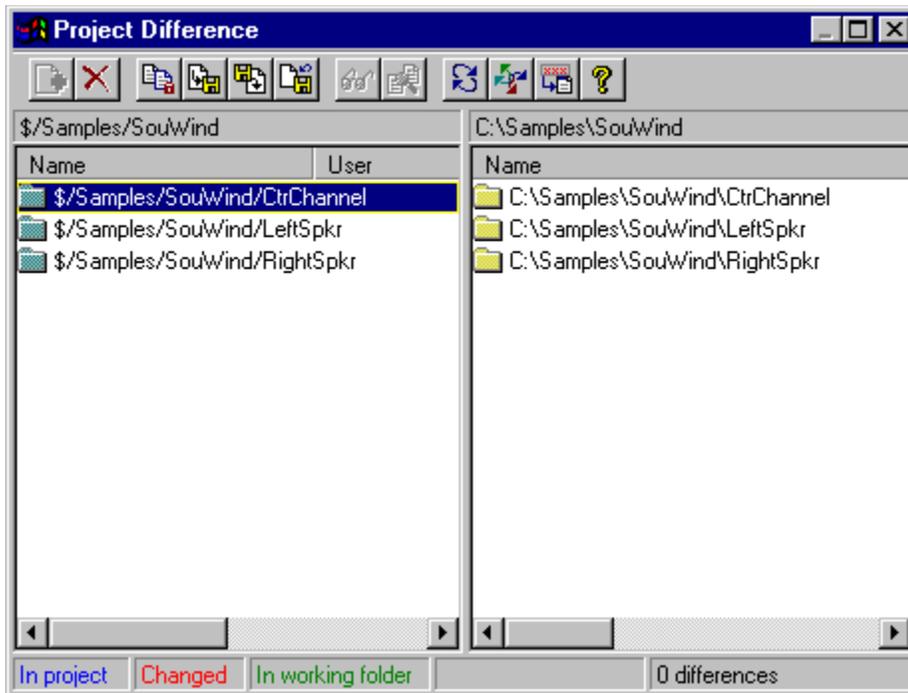
See Also

[Show Differences \(File\) Command](#)

[Show History Command](#)

Difference for Project Dialog Box

See Also



Shows the differences between files in the selected Visual SourceSafe database project and the working folder set for that project. You can also see what is different within a file, and reconcile differences if desired.

– To display the Differences For Project dialog box

- 1 Select a project in the project pane of Visual SourceSafe Explorer.
- 2 On the Tools menu, choose Show Differences. Choose options in the Project Difference dialog box if it's enabled, and choose OK.

Note To enable the display of the Project Difference dialog box, check the Difference checkbox in the Project column on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

Dialog Box Options

This powerful dialog is composed of two panes, one showing the files in your current project (on the left) and the other showing files in the working folder assigned to the current project (on the right). This dialog also has its own toolbar, with buttons for adding files, deleting files, getting files, checking files out and in, undoing file checkouts, viewing files, reconciling the differences between the project and working folder, and more.

Files in both panes are sorted alphabetically, by file name or folder name (if there are subfolders). Remember that the list of files in the right and left panes is filtered by the choices you made in the Show Differences command dialog. For example, if you choose to show only files that are different in both places, you will not see files that are the same in both your working folder and your current project. You can size the columns in the left pane by dragging the column bar with your left mouse button. Pressing F5 will refresh your dialog box with current information (but may be slow for large projects).

There are four types of files which may appear in the lists.

- Files that are the same in project and your working folder appear in black letters against a white

background by default. There is no action you need to take for these files.

- Files that are in the project, but not in your working folder, appear in blue against a gray background by default. Such a file generally represents a file that another user added to the project (in which case you may wish to Get it), or a file that you deleted from your working folder (in which case you may wish to delete it from Visual SourceSafe).
- Files that are in your working folder, but not in the project, appear in green against a gray background by default. Such a file generally represents a file that you created (in which case you may wish to Add it to Visual SourceSafe), or which another user deleted from the project (in which case you may wish to delete your local copy).
- Files that are in both places, but *different*, appear in red against a gray background by default. Such a file generally represents a file which you have checked out (which you may wish to check in), or a file which another user has checked in (which you may wish to Get).

Note You can set your default colors for added, changed, and deleted text by using the Difference tab of the Tools Options dialog box.

The above is a complete list of the four types of files, but not a complete list of the possibilities. For instance, if another user renames a file in Visual SourceSafe you will see a file you do not actually have, in the File List for that project in Visual SourceSafe Explorer; and in your working folder for the project there will be a file that is not present in the Visual SourceSafe database. So you would want to get Visual SourceSafe's file, and delete your local one, to reconcile the differences.

File icons in the left pane of dialog box represent the same things they do in the main Explorer window – a checkmark next to a file means the file is checked out, etc.; but file icons in the right pane represent conditions specific to you

– for example, a checkmark in the right pane means the file is checked out to you specifically, in your working folder. Double-clicking a file views it; double-clicking a folder does nothing.

Toolbar commands in this dialog generally work in the same way as the standard Visual SourceSafe commands, but some work slightly differently, as follows:

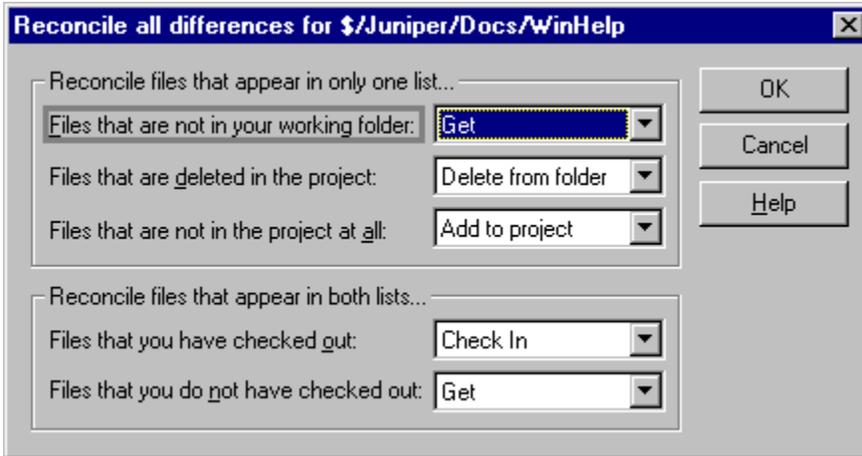
-  The Add command moves a file or directory from the right pane to the left pane. In other words, it adds a file that is in your working folder to the Visual SourceSafe project. The keyboard shortcut for this command is the Windows95 application key + A.
-  The Delete command works normally if a file in the left pane is selected. If a file in the right pane is selected, you can use the Delete command to delete the local file or directory from your working folder. The keyboard shortcut for this command is the Delete key.
-  The Get Latest Version command works normally, getting a file or project from the left pane to the right pane. The keyboard shortcut for this command is the Windows95 application key + G.
-  The Check Out command works normally, reconciling the differences between a file that is different between panes and is not checked out in the right pane, by checking out the file and overwriting the file in your local working folder.. The keyboard shortcut for this command is the Windows95 application key + O.
-  The Check In command works normally, reconciling a file that is in both panes, but different, and is checked out in both panes. With this command, you save the latest changes to the file in your working folder. The keyboard shortcut for this command is the Windows95 application key + I.
-  The Undo Check Out command works normally, reconciling a file that is different between panes and is checked out in both panes by undoing the check out on the file. With this command, you lose the latest changes to the file in your working folder. The keyboard shortcut for this command is the Windows95 application key + U.

-  The View command works on a file selected in the left pane just like the Explorer's View command
 - it displays the file in the Visual SourceSafe database. The View command works on a file selected in the right pane by opening the appropriate viewer directly on the file in the folder. The keyboard shortcut for this command is the Windows95 application key + V.
-  The Difference command compares two selected files. If only one file is selected, Difference compares it between the Visual SourceSafe project and the folder. The keyboard shortcut for this command is the Windows95 application key + S.
-  The Refresh File List command brings the file list up to date.
-  The Reconcile All command reconciles differences found in the compared projects. Reconciliation can mean several different things, depending on the settings you choose in the dialog boxes that are displayed after you choose this command. See the following section for more information.
-  The Report command generates the standard Report dialog box. The keyboard shortcut for this command is the F5 key.

Note If you use a mouse, you can access many of the preceding commands with the right mouse button. Without a mouse, you can access the toolbar commands with the keyboard accelerator keys.

Reconcile All Options Dialog Box

The Reconcile All button displays a dialog box with the following options. These options specify what you want Visual SourceSafe to do when it encounters specific situations.



Reconcile files that appear in only one list...

Files that are not in your working folder. This specifies what you want to happen when Visual SourceSafe finds files that are not in your working folder, but are in the project. Choose Get Latest Version (the default) to get files missing from your working folder. Choose Ignore to take no action with these files.

Files that are deleted in the project. This specifies what you want to happen when Visual SourceSafe finds files that are in your working folder, but have been deleted in the project. Choose Delete from folder (the default) to delete these files from your working folder. Choose Recover deleted file to recover the deleted file and place a copy in your working folder. Choose Add to project to add these files back to your project. Choose Ignore to take no action with these files.

Files that are not in the project at all. This specifies what you want to happen when Visual SourceSafe finds files that are not in your project, but are in your working folder. Choose Add to Project (the default) to add these files to your Visual SourceSafe project. Choose Ignore to take no action with these files.

Reconcile files that appear in both lists...

Files that you have checked out. This specifies what you want to happen when Visual SourceSafe finds files that are checked out to you, and are different in your working folder. Choose Check In (the default) to check these files into your Visual SourceSafe project. Choose Undo Check Out to undo the check out on these files. Choose Get Latest Version to overwrite your checked out file with a read-only copy (losing any changes you may have made). Choose Ignore to take no action with these files.

Files that you do not have checked out. This specifies what you want to happen when Visual SourceSafe finds files that are not checked out to you, but are different in your working folder. Choose Get Latest Version (the default) to get the latest version of these files. Choose Ignore to take no action with these files.

See Also

[Show Differences \(Project\) Command](#)

[Keyboard Accelerator Keys](#)

[Reconcile All Dialog Box](#)

Report Dialog Box

Creates a formatted report of selected information that you can send to the Clipboard, a file, or a printer.

Dialog Box Options

Report To

- Printer—Sends report to printer.
- File—Sends report to File dialog box.
- Clipboard—Places report on Clipboard.

OK

Confirms the report selection.

Cancel

Cancels the report.

Preview

Shows a preview of the specified report, so that you can determine if it is what you wanted before committing it to the printer queue or a file.

Preview Dialog Box

See Also

Shows a preview of the specified report, so that you can determine if it is what you wanted before committing it to the printer queue or a file. You can use the viewer accelerator keys in this dialog box to navigate in the file.

Dialog Box Options

Close

Closes the report preview.

Preview Dialog Box See Also

[File Viewer Keyboard Shortcuts](#)

Find Dialog Box

Allows you to enter a string to search for in the current file.

– **To display the Find dialog box**

- In the SourceSafe file viewer, press ALT+S or choose on the Find toolbar button. (You use the SourceSafe file viewer when you view a file for which you have not established another file association.)

Dialog Box Options

Find What

Enter the string you want to find.

Match Whole Word Only

Check this box to match the specified string only if the string is a whole word, not just part of a word.

Match Case

Check this box to make the search case-sensitive.

Regular Expression

If this box is not checked, the * and ? characters are interpreted literally. Therefore, A*B matches only one string, A*B.

If this box is checked, the * and ? characters are interpreted as wildcards. Therefore, A?B matches ABB or AQB, and A*B matches ADSFFDFDSB, within a line.

Set Bookmarks on All

Sets bookmarks on all matches in the file.

Direction

Specifies whether the search continues up or down in the file.

Find Next

Confirms the search selection.

Cancel

Cancels the search.

Browse Folder Dialog Box

Dialog Box Options

Name

Name and path of the folder.

Folder

Your current folder.

Folders

Lists the folder hierarchy.

Drives

Selects the drive that contains the folder.

OK

Confirms the folder selection.

Cancel

Cancels the selection, closes the dialog box, and returns to [Visual SourceSafe Explorer](#).

Create Dir

Creates a new folder as specified. Type a new folder name in the Name box, then choose Create Dir to create the folder.

Find In Files Results Dialog Box

This dialog box shows the results of the Find in Files command you issued.

Dialog Box Options

File

Shows the name of the file(s) containing matching strings.

Matches

Shows the number of matches found in the file.

Matches

Lists the matches found, in context.

Close

Closes the Find in Files results dialog box.

Report

Allows you to print a report of the match information to the Clipboard, a printer, or a file.

Links

Shows any links the file has to other projects.

Share Dialog Box

See Also

Allows you to specify a project to share with the current project.

– **To display the Share dialog box**

- 1 In the project pane of Visual SourceSafe Explorer, choose a project into which you want to share another project.
- 2 On the SourceSafe menu, choose Share.
- 3 In the Share with Project dialog box, select a project to share and choose Share.

Dialog Box Options

New Name

Enter the name of the project you are sharing with the current project.

Recursive

Check to make the Share operation recursive (include all subprojects and their files).

Comment

Enter a comment for this operation.

OK

Returns to the Share with Project dialog box, which shows the selected project to share with the selected files.

Cancel

Cancel the Share command.

See Also

[Share with Project Dialog Box](#)

[Share Command](#)

[Sharing Files](#)

Comment Dialog Box

Allows you to specify a comment for an Add Files, Share, or Merge Branches command.

Dialog Box Options

Comment

Enter a comment for this operation.

OK

Enters the selected comment(s) in the Visual SourceSafe database.

Cancel

Cancels the Add Files, Share, or Merge Branches command.

The following options are available only during an Add Files operation.

Use this comment for all selected files

Applies the same comment to all files involved in the Add Files operation. If this check box is clear, you can enter a separate comment for each file. The default is checked (your comment is applied to all files selected for this operation).

Recursive

Indicates you want files in all subfolders added when you select an entire folder to add. Available only when you select a folder and then choose Add.

Advanced Options

Store Only Latest Version

Indicates you do not want to save old versions of the file(s) being added to the Visual SourceSafe database. Visual SourceSafe keeps a file history of change records with comments. However, the older versions of the file are not stored, and you cannot recover them.

Note To increase performance and save disk space, use this option when storing executable programs in Visual SourceSafe.

Check Out Immediately

Checks out the file to you immediately after adding it to Visual SourceSafe.

Remove Local Copy

Deletes the file from your working folder after adding it to Visual SourceSafe.

File Type

Select an option from the list box:

- AutoDetect—Visual SourceSafe determines the file type.
- Binary—Visual SourceSafe treats the file as a binary file.
- Text—Visual SourceSafe treats the file as a text file.

Note When adding a folder, Visual SourceSafe ignores the File Types field. However, Visual

SourceSafe filters the files added during a folder Add Files operation by the Relevant_Masks setting on the File Types tab in the SourceSafe Options dialog box.

View/Edit Dialog Box

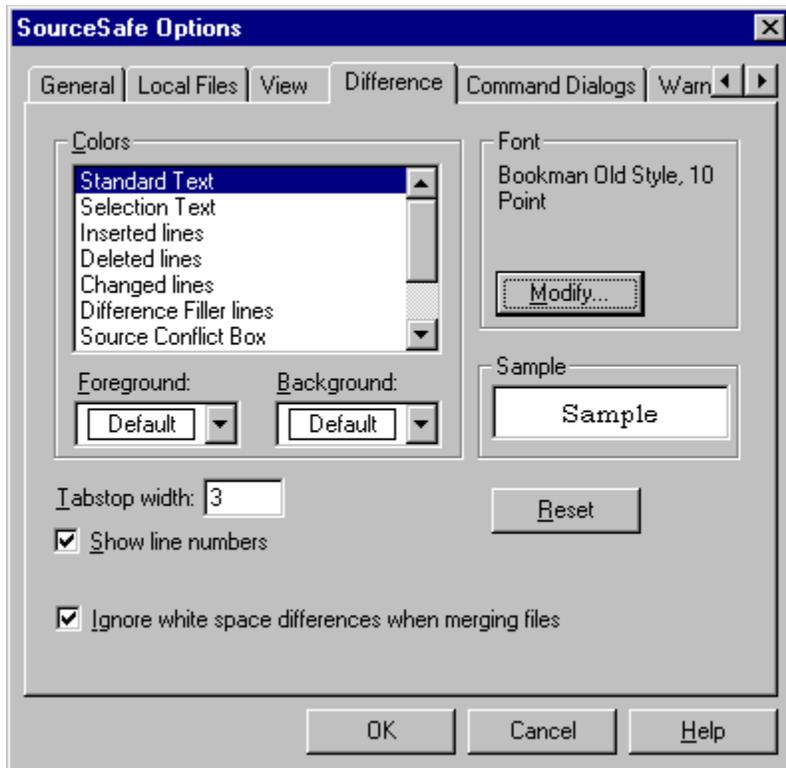
This dialog box allows you to specify whether you want to view the Visual SourceSafe database version of the file you double-clicked, or would prefer to edit this checked-out file in your working folder.

You will see this dialog box only if you have specified that it should be presented when a file is double-clicked in the Visual SourceSafe Explorer. You can change this specification from the Tools options General tab. Or, if you hold the SHIFT key down while double-clicking a file in the Explorer, this dialog will always be presented regardless of the setting on the Tools Options General tab.

In this dialog, select your choice (or choose Cancel to cancel the operation) and choose OK.

Difference Options Tab

See Also



Allows you to specify options for viewing differences between files or projects. These options are used in other Visual SourceSafe dialog boxes such as the Project Difference dialog box and the Visual Merge dialog box.

Dialog Box Options

Reset

Resets all values on this tab to their defaults.

Colors

This area of the tab contains a list of line and text types. Select each type and you will see the corresponding color used for that line or text type in the Sample area.

Note: The color of a bookmark in the visual viewers is the same as the Selection Text background color.

Foreground

This control is a drop down list box used to select the foreground color.

Background

This control is a drop down list box used to select the background color.

Font

Shows the selected font used in the difference viewer windows. Choose Modify to change the font.

Tabstop width

Shows the equivalent number of spaces each tab stop equals.

Show line numbers

Shows line numbers in the visual difference viewers.

Ignore white space differences when merging file

Check this box to ignore differences in white space (tabs, space characters, etc.) when merging two files.

OK

Confirms your selections.

Cancel

Cancels your selections.

Difference Options See Also

Font Dialog Box

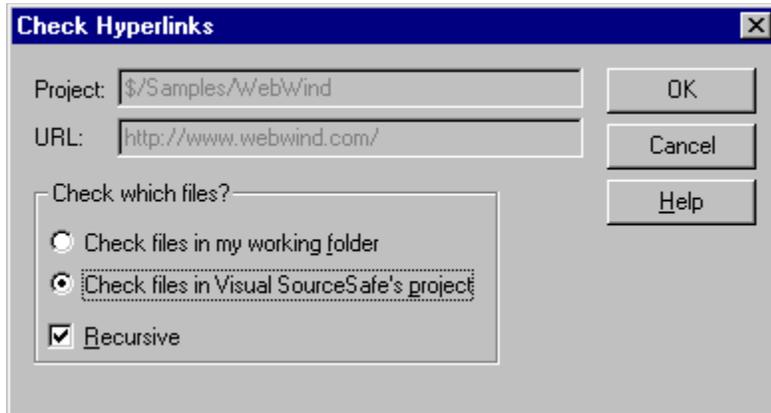
Visual_Diff_ChangeClr Initialization Variable

Visual_Diff_DeleteClr Initialization Variable

Visual_Diff_InsertClr Initialization Variable

Check Hyperlinks Dialog

See Also



Allows you to specify options for checking a set of HTML files for invalid links to other files.

Dialog Box Options

Project

Specifies the project that will be checked for invalid HTML links.

URL

Shows the URL represented by this project. Your Visual SourceSafe administrator sets this URL for each web site project using the Administrator program.

Check which files?

This area is used to specify which files you want to check. You can choose to check the files in your working folder or those in your Visual SourceSafe project. If you have made changes since checking your files out, choose Check files in my working folder. If you are checking a project with subprojects, select the Recursive box to include those subprojects.

There are basically two scenarios. First, all your HTML files are checked in, you are about to deploy, and you want to make sure nothing is broken. In that case, check the files in your project. Second, you are about to check your files in, but want to make sure you don't break anything before you do: in that case, check the files in your working folder.

OK

Confirms your selections.

Cancel

Cancels your selections.

Check Hyperlinks See Also

[Visual SourceSafe Web Features](#)

[Check Hyperlinks Command](#)

Hyperlinks Errors Dialog

See Also

Allows you to see the HTML links identified as errors, and optionally mark them as OK.

Dialog Box Options

File

Lists the files in which Visual SourceSafe identified HTML link errors. Also shows the number of errors found in each file.

Internal links which appear invalid

Lists the errors found in the file selected in the File list box.

External Links (not checked)

Lists the HTML recognized as an external link. External links are not checked.

Close

Closes this dialog box.

Check Hyperlinks See Also

[Visual SourceSafe Web Features](#)

[Check Hyperlinks Command](#)

[Check Hyperlinks Dialog Box](#)

Create Site Map Dialog

See Also

Allows you to create a map of your web site, using HTML files checked into your Visual SourceSafe project as targets for HTML jumps.

Dialog Box Options

To

Specifies the file containing the resulting HTML site map code. Choose Browse to search the file system for a suitable location.

Check in to Visual SourceSafe

Check this box to have the resulting file checked into the Visual SourceSafe project.

OK

Confirms your selections.

Cancel

Cancels your selections.

Create Site Map See Also

Visual SourceSafe Web Features

Create Site Map Command

Open SourceSafe Database Dialog

See Also

Allows you to open different Visual SourceSafe databases and store their locations.

Dialog Box Options

Available databases

Specifies the databases you have already visited, giving friendly name and path information for each. If the database you want is not shown in the list, you can choose the Browse button to look for it. Each database is associated with a SRCSAFE.INI file. The Browse command searches, by default, for those SRCSAFE.INI files. Once you find the one you want and choose OK in the Browse dialog with a specific SRCSAFE.INI file selected, you have the opportunity to give the database a friendly name, easily remembered so you can re-connect later.

Username

Specifies the username with which you want to log into the new database. If you change this to a new username with a different password, you will be prompted to enter the new password.

Open this database next time I run Visual SourceSafe

Check this box to designate this database as your startup database.

Remove

Removes a selected database from the list. You cannot remove the current database from the list.

OK

Confirms your selections.

Cancel

Cancels your selections.

Open SourceSafe Database See Also
Open SourceSafe Database Command

Browse for Visual SourceSafe Database Dialog

See Also

Allows you to name a Visual SourceSafe database.

Dialog Box Options

Database path

Specifies the path to the selected database. You cannot change this value here.

Database name

Specifies the friendly name you want to give this database. This name is displayed in the Open SourceSafe Database dialog box.

OK

Confirms your selections.

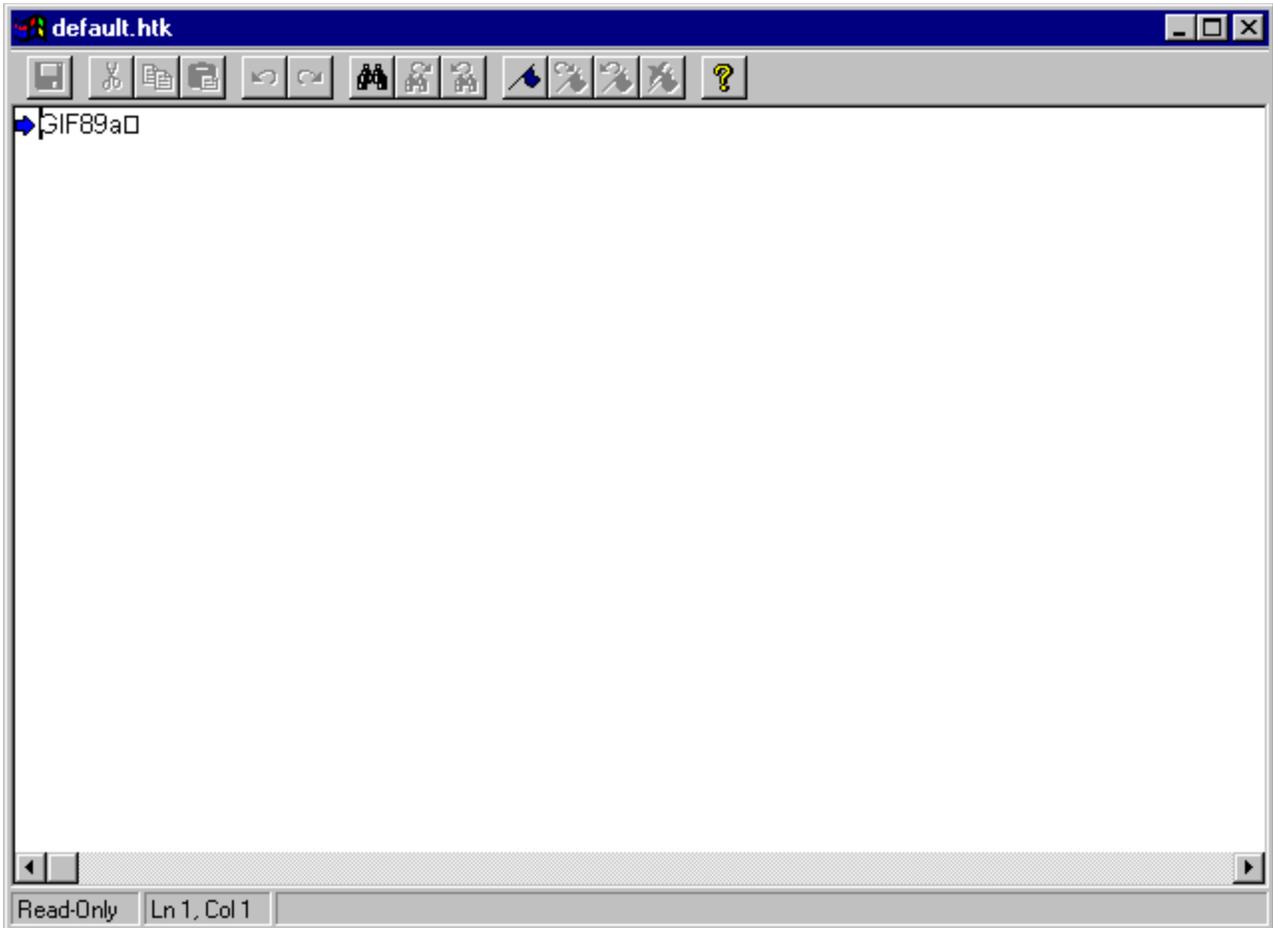
Cancel

Cancels your selections.

Browse for Visual SourceSafe Database See Also

Open SourceSafe Database Dialog

Visual SourceSafe Viewer Dialog Box



Allows you to edit or view a file without a file association.

Dialog Box Options

This dialog box includes a toolbar with the following buttons:



Save

Saves your edited file to disk.



Cut

Cuts the selection out of the file and places it on the Clipboard. The shortcut key is Ctrl+X.



Copy

Copies the selection and places it on the Clipboard. The shortcut key is Ctrl+C.



Paste

Pastes the Clipboard contents at the cursor location. The shortcut key is Ctrl+V.



Undo

Undoes your last editing action. The shortcut key is Ctrl+Z.



Redo

Redoes your last undone editing action. The shortcut key is Ctrl+A.

– **Find**

Lets you search for a string in the file. The shortcut key is Alt+F3.

– **Find Next**

Lets you search for the next matching occurrence of a string in the file. The shortcut key is F3.

– **Find Previous**

Lets you search for a previous matching occurrence of a string in the file. The shortcut key is Shift+F3.

– **Toggle Bookmark**

Lets you turn a bookmark on or off in a file. The shortcut key is Ctrl+F2.

– **Next Bookmark**

Lets you go to the next bookmark in the file. The shortcut key is F2.

– **Previous Bookmark**

Lets you go to the previous bookmark in the file. The shortcut key is Shift+F2.

– **Clear All Bookmarks**

Lets you clear all bookmarks in the file.

Deploy Dialog Box

Allows you to set options for your web site project deployment.

Dialog Box Options

If you enable the Deploy command dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu), or hold down the SHIFT key when you choose the Deploy command, you have the following options:

To

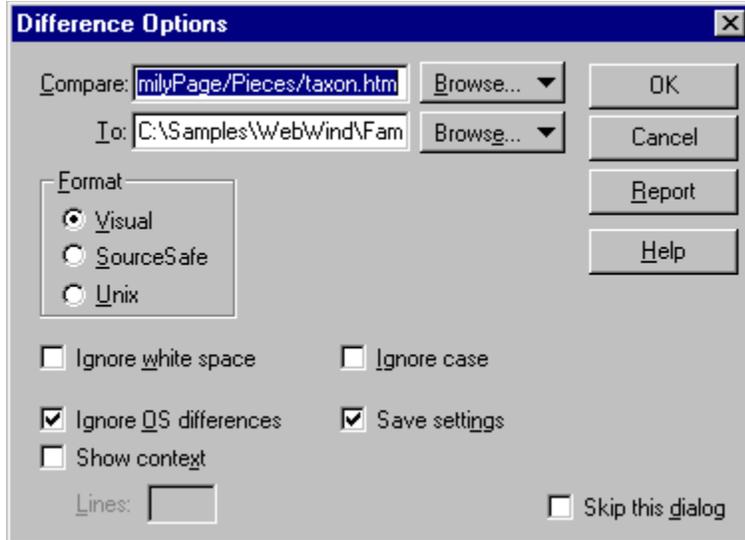
Type the path where you want Visual SourceSafe to place the files. You can Browse for the appropriate location.

Make Writable

Sets the read-write flag so the file is writable. This is not the default behavior, but may be useful if you want to try a temporary change in the file, or want to rename the file after getting it.

Difference Options Dialog Box

Allows you to specify the files you want to compare.



Dialog Box Options

Compare

Enter the name of the file you want to compare. Use the Browse button if necessary to find the file. You can choose to browse through files in your Visual SourceSafe projects, or in your Windows folders on your hard disk. To compare to a specific version of a file, you must add a semicolon and version number to the filename. For example:

```
HELP.C; 5
```

To

Enter the name of the file you want to compare the first file to. Use the Browse button if necessary to find the file. You can choose to browse through files in your Visual SourceSafe projects, or in your Windows folders on your hard disk.

Format

Choose Visual to use the Visual SourceSafe Visual viewer format (a side by side format), choose SourceSafe to use the old-style SourceSafe format (showing only the lines added, deleted, or changed), or choose Unix to use the Unix format (showing only the changed lines, marking the old and new versions).

Ignore White Space

Check this box to have Visual SourceSafe ignore differences in white space; that is, lines that differ only in space, tab, or other white space characters.

Ignore Case

Check this box to have Visual SourceSafe ignore differences in case.

OK

Confirms your selections.

Cancel

Cancels your selections.

Report

Allows you to print a report of the error information to the Clipboard, a printer, or a file.

Advanced Options

Choose this button to specify advanced File Difference options.

Ignore OS Differences

Check this box to have Visual SourceSafe ignore differences in the file caused by development on differing operating systems (such as end of line characters).

Save Settings

Check this box to save the settings you have chosen for use in subsequent Show Difference operations.

Show Context

Check this box to show contextual information. (Essentially, this means that besides showing just a changed line, Visual SourceSafe will also show the lines surrounding the changed line to provide some context to the information.) This option does not apply when you are using the visual format, only to the SourceSafe and Unix formats.

Lines

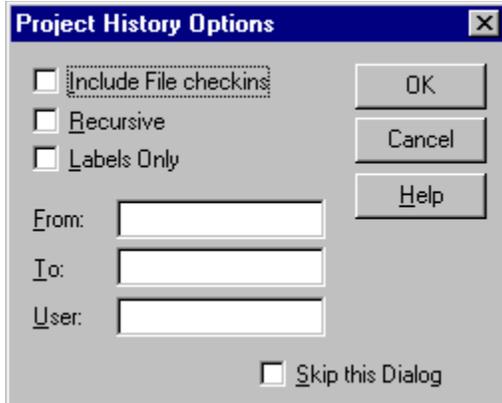
If Show Context is checked, this box lets you specify the number of lines of contextual (surrounding) information you wish to see.

Skip This Dialog

Skips this dialog box during future file difference operations, and uses the settings you've entered as the default. To restore the dialog box, clear the appropriate box on the Command Dialogs tab in the SourceSafe Options dialog box. Pressing the SHIFT key when clicking the Show Difference command always displays this dialog box, overriding the setting on the Command Dialogs tab.

File/Project History Options Dialog Box

Allows you to specify options for presenting the project history.



Dialog Box Options

Include file checkins (project history only)

Includes files in project history.

Recursive (project history only)

Includes subprojects and all files within them.

Labels only

Includes only **labels** in project history if you have selected the Include file checkins option.. You can use this checkbox with the From and To fields to find all projects with the designated label, but it will not search through the files in those projects for designated labels.

From

Restricts the history report to begin at the date specified. Type a date, date;time, version, or label. Use the prefix D to indicate a date and/or time, and the prefix L to indicate a label. For example "D02/29/95;12:15" or "Lbeta1". Dates can be entered in 08.15.95, 08-15-95, or 08/15/95 format, but not as August 15, 1995.

To

Restricts the history report to end at the date specified. Type a date, date;time, version, or label. Use the prefix D to indicate a date and/or time, and the prefix L to indicate a label. For example "D02/29/95;12:15" or "Lbeta1". Dates can be entered in 08.15.95, 08-15-95, or 08/15/95 format, but not as August 15, 1995.

User

Shows a history of files worked on by the specified user.

OK

Displays History of Project dialog box. In this dialog box, choose any of the items in the list, and then choose Details to display the History Details dialog box. From within the History Details dialog box, you can choose the Next and Previous buttons to move up and down in history display. You can also edit the comment and **label** associated with the version you are viewing.

Skip This Dialog

Skips this dialog box during future Show History operations, and uses the settings you've entered as the default. To restore the dialog box, choose the appropriate check box on the Command Dialogs tab of the Options dialog box.

Open Dialog Box

Dialog Box Options

File Name

The name of the file you are looking for. You can specify wildcards here, such as *.EXE to look for more than one file matching the wildcard.

List files of type

Specify the type of files to include in the File name list box.

Folders

Specifies a folder.

Drives

Selects the drive that contains the folder.

OK

Confirms the file selection.

Cancel

Cancels the selection, closes the dialog box, and returns to [Visual SourceSafe Explorer](#).

Network

Allows you to connect a network drive to search.

Differences For File Dialog Box

See Also

Shows differences between a working folder version of the selected file and the Visual SourceSafe database version of the file.

– **To display the Differences For File dialog box**

- 1 Make sure that you have the SourceSafe or Unix button selected in the Difference Options dialog box.
- 2 Select a file in the file pane of Visual SourceSafe Explorer.
- 3 On the Tools menu, choose Show Differences.
- 4 If the files compared are the same, a message tells you the files are identical. Visual SourceSafe cannot display differences for binary files; it can only tell you whether or not binary files differ. If text files differ, Visual SourceSafe displays the File Difference dialog box.
- 5 Make selections in the File Difference dialog box if it's enabled, and choose OK.

Note Enable the display of the Difference Options dialog box on the Command Dialogs tab in the SourceSafe Options dialog box (Tools menu).

Remarks

This dialog box is used only when you have not selected the visual difference comparison. (For visual differences, make sure you select the Visual option from the Difference Options dialog box.) By default, Visual SourceSafe uses the Visual Difference dialog box. The visual difference option shows each file side by side, and lets you easily make changes from one to the other. The SourceSafe difference shows a simple list of lines that were added, deleted, and changed, optimized for readability. The Unix option shows differences in the format used by the UNIX Diff utility.

If you specify a UNIX display, Visual SourceSafe uses the SourceSafe display options for Ignore Case and Ignore White Space. The SourceSafe display option has line widths calculated to fit on your screen, and no context lines.

Note To compare a current Visual SourceSafe file to an older version of the same file, choose the file in the file pane of Visual SourceSafe Explorer, and on the Tools menu, choose Show History. Then choose Diff.

Check Hyperlinks Results Dialog Box

This dialog is shown after Visual SourceSafe has scanned your Web files for broken hyperlinks. Visual SourceSafe differentiates between two kinds of hyperlinks: internal, and external.

Internal links are within your web site: that is, one file refers to another file in the same Visual SourceSafe project tree. Visual SourceSafe checks all internal links, and displays any potential problems. Problems are *not* determined by actually searching for files on your Web site; instead, Visual SourceSafe searches for files inside your project tree. For instance, suppose that the file `$/My/Web/FU.ASP` contains the line:

```
<A HREF="ORDER.HTM">
```

Visual SourceSafe will make sure that `$/My/Web/ORDER.HTM` exists. If that file exists, the hyperlink is valid, and is not displayed. If the file does not exist, this hyperlink is listed as broken.

External hyperlinks are jumps to sites on the World Wide Web that are not in your project tree. Visual SourceSafe does not check these jumps for validity in any way: it simply lists them all in the bottom list.

Find File Dialog Box

Allows you to browse the Visual SourceSafe database for a file to use in a comparison with another file (using the Show Differences command). Tab or use the mouse to choose between the project and file panes of this dialog, and then select the file you wish to compare another to.

Add (Command Line)

See Also

Adds new files into the Visual SourceSafe database.

Syntax

ss Add *local files* [-B] [-C] [-D-] [-H] [-I-] [-K] [-N] [-O] [-R] [-W] [-Y] [-?]

Examples

Adds file HELLO.C to the current project:

```
ss Add C:\HELLO.C
```

Adds files TEST.C and MY LONG FILENAME.H to the current project:

```
ss Add TEST.C "My long filename.H"
```

Adds all files in current folder to the current project:

```
ss Add *
```

Adds all files in current folder and all subfolders to the current project:

```
ss Add -R *
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all added files.
-D-	Do not track old versions of this file.
-R	Recursively add an entire folder tree, creating a project list in Visual SourceSafe.
-K	Check out the file immediately after adding it.
-B	Assume the file is binary. Use -B- to assume the file is text.
-W	Leave the <u>local copy</u> of the file writable.
-O	Paginate or redirect the command output.
-I-	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Add access right to use this command.

Remarks

You can specify a file on any drive and folder, and add that file to the current Visual SourceSafe project. To add a folder recursively, that is, to add all of its subfolders and files, specify a folder instead of a filename and use the -R option.

Note This command supports Universal Naming Convention (UNC) names: it's not necessary to specify a drive. You can type, for example, \\COMPUTER\SHARE\FILE.TXT to add the FILE.TXT file from \\COMPUTER\SHARE.

When you specify *.* as the item to add, or use the -R option, the Add command first checks the

Relevant Masks variable in the Visual SourceSafe initialization files.

The -D- option indicates that you don't want to store anything but the latest version of the file in the Visual SourceSafe database. When you add a file with this option, Visual SourceSafe keeps a file history of change records with comments. However, the change record that tracks the actual changes to your file is not stored; consequently, Visual SourceSafe can retrieve the latest version, but no earlier versions of this file. This option is frequently used with binary files to increase speed and save disk space.

The -B option indicates that all the files you are adding are binary files in cases where you don't want to use the Visual SourceSafe AutoDetect feature; -B- specifies text files. Both override Visual SourceSafe's default AutoDetect mode.

See Also

[-C Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[Add Files Command](#)

[Initialization Variable Summary](#)

[Setting the File Type: Text and Binary Files](#)

[Using the Visual SourceSafe Command Line](#)

Checkout (Command Line)

See Also

Copies a file from the **current project** to the current **folder**, for the purpose of editing.

Syntax

ss Checkout *Visual SourceSafe files* [-C] [-F] [-G] [-H] [-I] [-N] [-O] [-R] [-V] [-Y] [-?]

Examples

Checks out file TEST.C from project \$/TESTDATA:

```
ss Checkout $/TESTDATA/TEST.C
```

Checks out files TEST.C and TEST.H:

```
ss Checkout TEST.C TEST.H
```

Checks out all files in the TESTDATA project and all subfolders:

```
ss Checkout -R $/TESTDATA
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all checked-out files.
-G	Set options for the retrieved local file.
-R	Recursively check out the specified items.
-O	Paginate or redirect the command output.
-V	Specify which file version to check out.
-F	Display files only; do not display projects.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Check Out **access right** to use this command.

Remarks

No more than one user can check out binary files at a time.

Check out comments are requested for all files included in the operation, by default. You can use the -C option to type just one comment, which will be applied to all files included in the operation.

See Also

[-C Command-Line Option](#)

[-G Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-V Command-Line Option](#)

[-Y Command-Line Option](#)

[Check Out Command](#)

[Using the Visual SourceSafe Command Line](#)

[UndoCheckOut \(Command Line\)](#)

Comment (Command Line)

See Also

Changes the previously entered comment for a specific version of a file or project. Available only from the command line. (To change the comment from the Explorer, use the File Properties dialog box after selecting the appropriate file or project.)

Syntax

ss Comment *Visual SourceSafe items* [-C] [-H] [-I] [-N] [-O] [-V] [-Y] [-?]

Examples

Changes the comment for version 5 of TEST.C:

```
ss Comment TEST.C -V5
```

Changes the comment for the most recent version of TEST.C:

```
ss Comment TEST.C
```

Changes the comment for the project TESTDATA:

```
ss Comment $/TESTDATA
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all files being changed.
-O	Paginate or redirect the command output.
-V	Specify the file version to change the comment of.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Add **access right** to use this command.

Remarks

Every version of every file or project in Visual SourceSafe is associated with a user-defined comment, which you can change with the Comment command. After you type the command, you are prompted for the new comment (or you can specify it on the command line using the -C option).

The new comment replaces the old comment, which cannot be recovered.

See Also

[-C Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-V Command-Line Option](#)

[-Y Command-Line Option](#)

[Using the Visual SourceSafe Command Line](#)

CP (Command Line)

See Also

Sets the **current project** path.

Syntax

ss Cp *project* [-H] [-I] [-Y] [-?]

Examples

Makes \$/WORD the current project:

```
ss Cp $/WORD
```

Moves up to the parent of the current project:

```
ss Cp ..
```

Moves up to the **root project**:

```
ss Cp $/
```

The following table describes the command-line options available with this command.

Option	Description
-I	Ignore: Do not ask for input under any circumstances.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

The CP command sets the current project. Once you set the current project on the command line, most commands assume the current project if you simply specify a filename. For example, if you follow `ss Cp $/WORD` with `ss Get TEST.C`, Visual SourceSafe retrieves `$/WORD/TEST.C`.

Note Use the Project command to find the current project from the command line.

On the Visual SourceSafe command-line, when you get or check out a file, the file goes into the current folder. But if the Force_Dir initialization variable is set to Yes, the file goes into your working folder for the project, even if that is not the current folder.

See Also

[-I Command-Line Option](#)

[-Y Command-Line Option](#)

[Force_Dir Initialization Variable](#)

[Project \(Command Line\)](#)

[Using the Visual SourceSafe Command Line](#)

Create (Command Line)

See Also

Creates a new **subproject**.

Syntax

ss Create *projects* [-C] [-H] [-I-] [-O] [-S] [-Y] [-?]

Examples

Creates subproject WORD under the **root project** \$/:

```
ss Create $/WORD
```

Creates subproject WORD under the **current project**:

```
ss Create WORD
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all created projects.
-S	Turn SmartMode on, so that Visual SourceSafe switches to the new project immediately after creating it.
-O	Paginate or redirect the command output.
-I-	Ignore: Do not ask for input under any circumstances.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Add **access right** in the **parent project** to use this command.

See Also

Create Project Command

-C Command-Line Option

-I Command-Line Option

-N Command-Line Option

-O Command-Line Option

-S Command-Line Option

-Y Command-Line Option

Using the Visual SourceSafe Command Line

Destroy (Command Line)

See Also

Permanently removes a file or project.

Syntax

ss Destroy *Visual SourceSafe items* [-H] [-I] [-N] [-O] [-S] [-Y] [-?]

Examples

Destroys the file TEST.C in the project \$/PRJ:

```
ss Destroy $/PRJ/TEST.C
```

Destroys the project \$/PRJ:

```
ss Destroy $/PRJ
```

The following table describes the command-line options available with this command.

<u>Option</u>	<u>Description</u>
-S	Turn SmartMode on, so that Visual SourceSafe deletes your <u>local copies</u> of the files after destroying the <u>master copies</u> .
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Destroy **access right** to use this command.

Use the Delete command if you don't want the file or project to be removed permanently.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-S Command-Line Option](#)

[-Y Command-Line Option](#)

[Destroy Command](#)

[Using the Visual SourceSafe Command Line](#)

Difference (File) (Command Line)

See Also

Shows the differences between the lines in a Visual SourceSafe **master copy** and the corresponding **local copy** in your **working folder**, or a previous version.

Note For binary files, Visual SourceSafe can tell you if two files are different, but not how they are different. Visual SourceSafe stores **reverse deltas** for all files, but in the case of a binary file, the binary record stored cannot be displayed to the user. Only ASCII text files can be compared line-by-line.

Syntax

ss Diff *Visual SourceSafe files* [-B] [-D] [-H] [-I] [-I-] [-N] [-O] [-V] [-Y] [-?]

Examples

Compares the Visual SourceSafe file HELP.C to the local file HELP.C:

```
ss Diff HELP.C
```

Compares the version 2 Visual SourceSafe file HELP.C to the local copy of HELP.C:

```
ss Diff HELP.C -V2
```

Compares versions 2 and 4 of the Visual SourceSafe file HELP.C:

```
ss Diff HELP.C -V4~2
```

Compares file HELP.C in the Visual SourceSafe database against the file BAR.C on your local computer:

```
SS Diff $/HELP.C C:\BAR.C
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-V	Specify the old file version(s) to compare.
-I [C E S W]	Specify which informational messages are ignored. The -IS option means ignore small matches – prevents a large section of changes from looking like there are matches when there really aren't.
-I-	Ignore: Do not ask for input under any circumstances.
-D <S U V> [width]	Specify the format of the difference display.
-DX [- number of lines]	Specify the number of displayed context lines.
-B	Create a brief difference report.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to use this command.

Remarks

The -B option provides a brief difference report. It checks to see if the specified files are different and is faster than computing actual line differences. This can be useful in batch files and scripts that react differently when a file has been modified.

The -D option controls how Visual SourceSafe displays the differences. A file difference can be displayed in one of three manners:

V (visual)—shows the entire file, with annotations for the changes. This display shows differences side-by-side.

S (SourceSafe)—displays a list of lines that were added, deleted, changed, and optimized for readability; for example, an inserted line might be displayed as follows:

```
27     Ins: hello, world
```

U (UNIX)—display gives similar information in the standard format used by the UNIX diff utility:

```
26a27
> hello, world
```

Immediately after the V, S, or U in the -D option, you can specify a number indicating the line width. If you specify more characters than your screen permits, long lines wrap, which may affect the readability of the display.

Finally, X controls how many context lines Visual SourceSafe displays around each change. Seeing these lines may help you understand why a change was made. X by itself shows three lines of context; X- shows no context; X followed by a number specifies how many lines to show.

By default, Visual SourceSafe uses the Visual SourceSafe display, with line width calculated to fit on your screen, and no context lines. If you specify a UNIX display, the other defaults are the same; the visual display, by default, shows all lines (infinite context).

The following tables demonstrates some uses of the -D option.

Example	Description
-DV35	Visual display, 35-character width
-DX	Three context lines
-DU70X5	UNIX-style display, 70-character width, five context lines

The -I option indicates which of four factors Visual SourceSafe should ignore when comparing two files.

- (C)ase: The letter A is considered the same as the letter a, etc., and a change of one to the other is not reported.
- (W)hite space: Any change in spacing or tabbing is ignored; only the insertion or deletion of functional characters is reported.
- (E)nd-of-line character: Every line is not reported as a change simply because of differences on a different operating system.
- (S)mall changes: Small changes are ignored.

The default setting, -IE, ignores end-of-line characters and reports any other difference as a change.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-V Command-Line Option](#)

[-Y Command-Line Option](#)

[Difference \(Project\) Command](#)

[Show Differences \(File\) Command](#)

[Using the Visual SourceSafe Command Line](#)

Difference (Project) (Command Line)

See Also

Compares the contents of a project with the current folder.

Syntax

ss Diff *Visual SourceSafe projects* [-B] [-F] [-H] [-I] [-N] [-O] [-R] [-V] [-Y] [-?]

Examples

Compares the Visual SourceSafe project WORD to the current folder:

```
ss Diff $/WORD
```

Compares the current project with the current folder:

```
ss Diff
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-V	Compare an earlier file or project version.
-R	Recursively compare an entire project list to a local folder tree.
-B	Create a brief difference report.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-F	Reconcile project differences using the user-specified reconciliation behavior stored in the Reconcile_Project initialization variable.
-, -H	Request online Help on a command.

Access Rights

You must have Read **access right** to use this command.

Remarks

The project-level difference compares the contents of your current project to the contents of the current folder assigned to that project, and reports on the differences found. These differences could be that a file is in one place but not the other, or that a file in both places is different. The -R option looks not only in the contents of the current project, it also looks in **subprojects** under the current project, and displays all differences between the entire project and your working folder.

The -B option creates a brief difference report: It does not check file contents at all, but simply looks for files that exist in one place and not the other. The -F option reconciles the project differences using the decision rules stored in the Reconcile_Project initialization variable. See the **Reconcile_Project Initialization Variable** for more information.

The Relevant_Masks initialization variable affects this command by filtering the list of files being compared between projects.

See Also

-I Command-Line Option

-N Command-Line Option

-O Command-Line Option

-R Command-Line Option

-V Command-Line Option

-Y Command-Line Option

Difference (Project) Command

Relevant_Masks Initialization Variable

Using the Visual SourceSafe Command Line

Reconcile_Project Initialization Variable

Directory (Command Line)

See Also

Shows a list of all the files and projects in the specified project, or in the **current project**. Available only from the command line.

Syntax

ss Dir *project path* [-C] [-D] [-E] [-F-] [-H] [-I-] [-N] [-O] [-R] [-V] [-Y] [-?]

Examples

Shows a directory of the project \$/WORD:

```
ss Dir $/WORD
```

Shows a directory of the current project:

```
ss Dir
```

Equivalent to the MS-DOS **dir** command:

```
ss ls
```

The following table describes the command-line options available with this command.

Option	Description
-D	Display deleted files instead of nondeleted files.
-O	Paginate or redirect the command output.
-R	Recursively display an entire project list.
-E	Display extended display, including check out information.
-V	Display a directory of an earlier version of a project.
-C	Display files and projects in columns.
-F-	Display projects only; do not display files.
-I-	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have Read **access right** to use this command.

Remarks

The Directory command in Visual SourceSafe functions very much like the MS-DOS **dir** command, and displays all the files and **subprojects** in a project. If a project is sharing a specific version of a file (rather than the most recent version), the **version number** of the file is shown. The Directory command sets the operating system exit code for use in batch files and scripts.

The -R option displays a recursive folder, that is, shows subprojects, the files and subprojects inside them, and so on.

The -D option displays all files and subprojects that have been deleted. This can be useful, for instance, for obtaining a list of files that you want to recover or purge.

The -E option displays check out information for any **checked-out files**. The Dir -E is similar to the Status command on the command line, with one notable difference: Status shows only checked out files; Dir -E shows all files, but notes the check out status of the ones that are checked out.

The -F- option shows projects only, and does not show files—`ss Dir $/ -F -R` creates a complete list of all the projects in the **Visual SourceSafe database** without listing any files.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-V Command-Line Option](#)

[-Y Command-Line Option](#)

[Using the Visual SourceSafe Command Line](#)

Filetype (Command Line)

See Also

Displays the type (text or binary) of a file. Available only from the command line. To find this information in the Explorer, use the File Properties command.

Syntax

ss Filetype *Visual SourceSafe items* [-B] [-H] [-I-] [-N] [-O] [-R] [-Y] [-?]

Examples

Tells whether HELP.C is text or binary:

```
ss Filetype HELP.C
```

Changes HELP.C to a binary file:

```
ss Filetype HELP.C -B
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-R	Recursively work on an entire project list.
-B	Change the type of a file. -B changes the file type to binary; -B- changes it to text.
-I-	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to check the type of a file, and the Add access right to change it.

Remarks

The Filetype command can be used to determine, or change, the type of one or more files. Every file in Visual SourceSafe is classified as either binary or text. If you do not specify a -B option, Visual SourceSafe simply displays the file type.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-Y Command-Line Option](#)

[Setting the File Type: Text and Binary Files](#)

[Using the Visual SourceSafe Command Line](#)

Get (Command Line)

See Also

Retrieves a **read-only** copy of the specified Visual SourceSafe file(s).

Syntax

```
ss Get Visual SourceSafe items [-G] [-H] [-I] [-N] [-O] [-R] [-V] [-W] [-Y] [-?]
```

Examples

Retrieves a read-only copy of the file TEST.C in the **current project**:

```
ss Get TEST.C
```

Retrieves read-only copies of all the files in the current project:

```
ss Get *
```

Retrieves read-only copies of all the files in the project \$/WORD, and all its **subprojects**:

```
ss Get $/WORD -R
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-R	Recursively get an entire project list.
-V	Get an old file or project version.
-G	Set options for the retrieved local copy .
-W	Make the retrieved local copy writable.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to use this command.

Remarks

The Get Latest Version command places files into your **working folder**. Used recursively on a project (the -R option), Visual SourceSafe gets all the files in the project and subprojects, and places them into subfolders of the same name. If you set the SS.INI variable Force_Dir to Yes, subprojects go into their respective working folders instead of into subfolders of the same name.

By default, Get Latest Version retrieves the most recent version of the files or projects you specify. However, you can get earlier versions as well by using the -V option. (Note that "ss Get -V*" does not work.)

Note Getting an earlier version of a project gets the files that were in the project at a particular time, as they were at that time. For instance, if you use the Get Latest Version command to retrieve version 7 of the project \$/WORD, you may get TEST.C version 3, HELP.C version 10, and TEST.H version 5.

See Also

[-G Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-V Command-Line Option](#)

[-W Command-Line Option](#)

[Force_Dir Initialization Variable](#)

[-Y Command-Line Option](#)

[Get Latest Version Command](#)

[Using the Visual SourceSafe Command Line](#)

Help (Command Line)

See Also

Displays Help on using Visual SourceSafe in general, or on a specific command.

Syntax

ss Help [Command] [-H] [-I] [-Y] [-?]

Examples

Shows general help:

```
ss Help
```

Shows help on the Share command:

```
ss Help Share
```

The following table describes the command-line options available with this command.

Option	Description
-I-	Ignore: Do not ask for input under any circumstances.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

Help for Visual SourceSafe command-line commands is displayed in a Microsoft Windows Help file. You must be running Microsoft Windows for Help to be displayed – there is no separate command-line help.

See Also

[-I Command-Line Option](#)

[-Y Command-Line Option](#)

[Help Command](#)

[Using the Visual SourceSafe Command Line](#)

History (Command Line)

See Also

Shows the **history** of a file or project in Visual SourceSafe.

Syntax

ss History *Visual SourceSafe items* [-B] [-D] [-F-] [-H] [-I-] [-L] [-N] [-O] [-R] [-U[*username*]] [-V] [-Y] [-#] [-?]

Examples

Displays the history of the file TEST.C:

```
ss History TEST.C
```

Displays the history of the project \$/WORD:

```
ss History $/WORD
```

Displays the last three items in the history list for the current project, or file TEST.C, respectively:

```
ss history . -#3  
ss history TEST.C -#3
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-V	Display only versions that fall within specified range.
-R	Display the history of an entire project list.
-L	Display only versions that have labels .
-F-	Do not display individual file updates in the project history.
-B	Display a brief history.
-U	Display only changes made by a particular user.
-D	Display line-by-line file changes.
-#<#>	Display the specified number of history entries for the selected file or project.
-I-	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have Read **access right** to use this command.

Remarks

The History command has a number of options that can be used to customize its display. For example, -L shows only the versions that have labels; because these versions tend to be the major release milestones, the -L option can help generate a general overview of the history of an item.

The -B option shows a brief history—only enough information about a version to fit on one line.

The -U option shows only changes that were made by a particular user (the current user if none is specified on the command line).

The -D option shows all the changes actually made with each file update.

The -V option specifies a range: -V5 starts with version 5, and goes back to version 1. A tilde (~) is used to specify both sides of a range, so -V5~3 gives the history of versions 5, 4, and 3. The range must be specified beginning with the later version, as in the example.

The -R option indicates that you want to see all the changes made to every subproject under the project, and the files and subprojects under those, and so on.

By default, the history of a project includes the histories of all its files, but does not include subprojects.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-V Command-Line Option](#)

[-Y Command-Line Option](#)

[Show History Command](#)

[Using the Visual SourceSafe Command Line](#)

Label (Command Line)

See Also

Assigns a **label** to the specified version or **current version** of a file or project.

Syntax

ss Label *Visual SourceSafe items* [-C] [-H] [-I] [-L*label*] [-N] [-O] [-V] [-Y] [-?]

Examples

Assigns a label to the current version of the **current project**:

```
ss Label
```

Assigns a label to the current version of the project \$/WORD:

```
ss Label $/WORD
```

Assigns a label to version 5 of the file HELP.C:

```
ss Label HELP.C -V5
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all labels.
-V	Label an existing file or project version.
-L	Specify the new label on the command line (instead of being prompted).
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Add **access right** to use this command.

Remarks

The Label command creates a new version of the specified project or file, and assigns your label to that version. However, if you use the -V option to refer to a specific version, the label is simply applied to that version.

If you assign a label to a version that already has a label, or if you assign a label to an item that already uses that label for a different version, the old label is replaced. In either case, Visual SourceSafe issues a warning before replacing the old label.

If you specify multiple items on the command line, Visual SourceSafe prompts you for a different label for each item. The -C option asks for only one label, which Visual SourceSafe applies to all the items you have listed. The -L option bypasses the label prompt entirely by specifying the label on the command line, for example, -L2.00b.

See Also

[-C Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-V Command-Line Option](#)

[-Y Command-Line Option](#)

[Label Command](#)

[Labeling SourceSafe Projects and Files](#)

[Using the Visual SourceSafe Command Line](#)

Links (Command Line)

See Also

Shows the projects sharing a specified file.

Syntax

ss Links *Visual SourceSafe files* [-H] [-I] [-N] [-O] [-Y] [-?]

Examples

Shows all projects that are sharing the file TEST.C:

```
ss Links TEST.C
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-I-	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to use this command.

Remarks

The Links command displays all the current **share links** from a specified file to all other projects that share it. In contrast, the Paths command displays links that existed previously but were branched.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-Y Command-Line Option](#)

[Links Properties Tab](#)

[Paths Properties Tab](#)

[Sharing Files](#)

[Using the Visual SourceSafe Command Line](#)

Locate (Command Line)

See Also

Searches through Visual SourceSafe projects for file or project names. Available only from the command line. In the Explorer, you can use the Search Wildcard Search command to get equivalent information.

Syntax

ss Locate *filename* [-H] [-I] [-N] [-O] [-Y] [-?]

Examples

Looks for files called TEST.C:

```
ss Locate TEST.C
```

Looks for filenames that match the wildcard mask *.old:

```
ss Locate *.OLD
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

This command locates only files in projects for which you have the Read **access right**.

Remarks

The Locate command searches through the entire Visual SourceSafe project hierarchy for a filename you specify, and displays the project path to any matches it finds. In many respects, the results are similar to the results on the Links tab. For instance, if several projects share the file TEST.C, both commands give you a list of all those projects. However, if a project contains a file that is unrelated to the file TEST.C in your project, but has the same name, Locate finds the file; Links ignores it.

You can also use Locate to find a file when you do not know any of the projects it is in, whereas the Links tab requires that you start with one specific file.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-Y Command-Line Option](#)

[Links Properties Tab](#)

[Using the Visual SourceSafe Command Line](#)

Merge (Command Line)

See Also

Copies all the changes made in one **branch** to another branch.

Syntax

ss Merge *Visual SourceSafe files* [-C] [-H] [-I] [-N] [-O] [-Y] [-?]

Examples

Merges the file TEST.C in \$/WORD into TEST.C in the **current project**:

```
ss Merge $/WORD/TEST.C
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all merged files that are checked in.
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Check Out **access right** in the project being merged into, and the Read access right in the project being merged from, to use this command.

Remarks

You cannot perform a merge on a binary file.

See Also

[-C Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-Y Command-Line Option](#)

[Merge Branches Command](#)

[Merging Files and Projects](#)

[Using the Visual SourceSafe Command Line](#)

Move (Command Line)

See Also

Relocates a **subproject** from one parent project to another.

Syntax

ss Move <project to move> <new project path> [-H] [-I] [-Y] [-?]

Examples

Moves the project \$/WORD from \$/JOE to \$/APPS, creating \$/APPS/WORD:

```
ss Move $/JOE/WORD $/APPS
```

The following table describes the command-line options available with this command.

Option	Description
-I-	Ignore: Do not ask for input under any circumstances.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Add access right in the parent project being moved to, and the destroy access right in the parent project being moved from, to use this command.

See Also

[-I Command-Line Option](#)

[-Y Command-Line Option](#)

[Using the Visual SourceSafe Command Line](#)

[Move Project Command](#)

Password (Command Line)

See Also

Sets the password for the current user.

Syntax

ss Password [-H] [-?]

Examples

Triggers the password prompt:

```
ss Password
```

The following table describes the command-line options available with this command.

Option	Description
-?, -H	Request online Help on a command.

Access Rights

Any Visual SourceSafe user can use this command. Visual SourceSafe will prompt for the user name and password.

See Also

[-Y Command-Line Option](#)

[Using the Visual SourceSafe Command Line](#)

[Change Password Command](#)

Paths (Command Line)

See Also

Shows all share links that have been branched.

Syntax

ss Paths *Visual SourceSafe files* [-I] [-N] [-O] [-Y]

Examples

Shows all current and broken shared file links to the file TEST.C:

```
ss Paths TEST.C
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Read access right to use this command.

Remarks

The display generated by Paths on the Visual SourceSafe command line is formatted differently from the display generated by the Paths tab in the Properties dialog box (File menu). Both contain the same information, but users who switch between the two should take extra care to avoid confusion.

In the SourceSafe command line program, the Paths command might show a display like the following.

```
TEST.C
$/A
  TEST.C (Branched at version 5)
  $/B
  $/C
  TEST.C (Branched at version 4)
  $/D
```

The above display shows that the file began in project \$/A, one branch was created at version 5 in project \$/B, which is also shared in \$/C; and another branch, also directly off of the file in \$/A, was created at version 4 in \$/D.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-Y Command-Line Option](#)

[Paths Properties Tab](#)

[Using the Visual SourceSafe Command Line](#)

Project (Command Line)

See Also

Displays the **current project** path.

Syntax

ss Project [-H] [-I] [-O] [-Y] [-?]

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-I-	Ignore: Do not ask for input under any circumstances.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

Any Visual SourceSafe user can use this command.

Remarks

The Project command displays the current project path. It tells you what project you are in, for example, \$/WORD; if you type `ss Get TEST.C`, Visual SourceSafe looks for \$/WORD/TEST.C.

Use the CP command to change your current project.

See Also

[-I Command-Line Option](#)

[-Y Command-Line Option](#)

[CP \(Command Line\)](#)

[Using the Visual SourceSafe Command Line](#)

Purge (Command Line)

See Also

Permanently removes previously deleted files and projects from the Visual SourceSafe database.

Note Purged items cannot be recovered.

Syntax

ss Purge *Visual SourceSafe items* [-H] [-I] [-N] [-O] [-Y] [-?]

Examples

Purges the deleted file TEST.C:

```
ss Purge TEST.C
```

Purges the deleted project \$/PRJ:

```
ss Purge $/PRJ
```

The following table describes the command-line options available with this command.

Option	Description
-I	Ignore: Do not ask for input under any circumstances.
-O	Paginate or redirect the command output.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Destroy access right to use this command.

Remarks

When you use the Delete command to delete a file or project in Visual SourceSafe, the deleted item is removed from Visual SourceSafe Explorer, but Visual SourceSafe is still storing it internally. You can use the Dir -D command on the Visual SourceSafe command line to show a list of the files that are deleted from the current project. Then, you can purge them if you want to destroy them permanently, or rebuild earlier versions of the project that required them with the Recover command.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-Y Command-Line Option](#)

[Delete Command](#)

[Destroy Command](#)

[Purge Command](#)

[Recover Command](#)

[Using the Visual SourceSafe Command Line](#)

Recover (Command Line)

See Also

Recovers files and projects that have been deleted.

Syntax

ss Recover *Visual SourceSafe items* [-G] [-H] [-I] [-N] [-O] [-S] [-W] [-Y] [-?]

Examples

Undeletes the Removed file TEST.C:

```
ss Recover TEST.C
```

Undeletes the Removed project \$/WORD:

```
ss Recover $/WORD
```

The following table describes the command-line options available with this command.

<u>Option</u>	<u>Description</u>
-S	Turn SmartMode on, making a recovered project the <u>current project</u> .
-G	Set options for the retrieved <u>local copy</u> .
-W	Make the retrieved local copy writable.
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Add **access right** to use this command.

See Also

[-G Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-S Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[Recover Command](#)

[Delete Command](#)

[Purge Command](#)

[Using the Visual SourceSafe Command Line](#)

Delete (Command Line)

See Also

Removes files and projects from **Visual SourceSafe Explorer**, and marks them as deleted; the items still exist, however, and can be recovered using the Recover command.

Syntax

ss Delete *Visual SourceSafe items* [-H] [-I] [-N] [-O] [-S] [-Y] [-?]

Examples

Deletes the file TEST.C from the project \$/PRJ:

```
ss Delete $/PRJ/TEST.C
```

Deletes the project \$/PRJ:

```
ss Delete $/PRJ
```

Deletes all the files in the current project:

```
ss Delete *.*
```

The following table describes the command-line options available with this command.

Option	Description
-S	Turn SmartMode on, deleting the local copy of the file after deleting the Visual SourceSafe master copy .
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Add **access right** to use this command.

Remarks

Once you have deleted a file from a project, you cannot delete another file with the same name from the same project: If you delete file TRASH.C from a project, then add a new file called TRASH.C to the project, that is okay. Trying to delete TRASH.C again is not allowed, however.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-S Command-Line Option](#)

[-Y Command-Line Option](#)

[Delete Command](#)

[Recover Command](#)

[Purge Command](#)

[Using the Visual SourceSafe Command Line](#)

Rename (Command Line)

See Also

Renames a file or project.

Syntax

ss Rename <Visual SourceSafe item> <new name> [-H] [-I] [-N] [-O] [-S] [-Y] [-?]

Examples

Changes the Visual SourceSafe file named TEST.C to MISC.C:

```
ss Rename TEST.C MISC.C
```

Changes the Visual SourceSafe project named WORD to EDITOR. (The new project is \$/APPS/EDITOR.):

```
ss Rename $/APPS/WORD EDITOR
```

The following table describes the command-line options available with this command.

<u>Option</u>	<u>Description</u>
-S	Turn SmartMode on, renaming the local copy after renaming the Visual SourceSafe master copy .
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Add **access right** to use this command.

See Also

Rename Command

-I Command-Line Option

-N Command-Line Option

-S Command-Line Option

-Y Command-Line Option

Using the Visual SourceSafe Command Line

Rollback (Command Line)

See Also

Undoes all changes since an earlier version of a file.

Syntax

ss Rollback *Visual SourceSafe file* [-G] [-H] [-I] [-N] [-O] [-V] [-W] [-Y] [-?]

Examples

Rolls back TEST.C to version 10:

```
ss Rollback TEST.C -V10
```

Rolls back TEST.C to version 2.01b:

```
ss Rollback TEST.C -VL2.01b
```

The following table describes the command-line options available with this command.

Option	Description
-V	Specify which file version to Roll back to; required unless you are sharing an earlier version of a file.
-G	Set Get options for the retrieved local copy .
-W	Make the retrieved local copy writable.
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Destroy **access right** to use this command.

Remarks

Rollback performs a Get operation when it is done, unless the Delete_Local initialization variable in SS.INI is set to Yes, or you use the -G- command-line option to prevent it.

See Also

[-G Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-V Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[Rollback Command](#)

[Using the Visual SourceSafe Command Line](#)

FindinFiles (Command Line)

See Also

Shows all instances of a specified string in one or more files.

Syntax

ss FindinFiles *string* [*Visual SourceSafe items*] [-H] [-I] [-IC] [-N] [-O] [-R] [-V] [-Y] [-?]

Examples

Looks for the string "giGlobal" in all .C files:

```
ss FindinFiles giGlobal *.C
```

Looks for the string "Hello world" in the **current project**:

```
ss FindinFiles "Hello world"
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-R	Recursively scan through an entire project list.
-V	Look through an earlier version of all specified files or projects.
-IC	Ignore case when looking.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to use this command.

Remarks

The FindinFiles command displays a list of all occurrences of a character string in the Visual SourceSafe files specified (in the same way the Grep command does in UNIX systems). You can look through a file, or through a project (which scans all the files in that project); if you don't specify Visual SourceSafe items, it is assumed you are looking through the files in the current project.

The Visual SourceSafe FindinFiles command supports two **wildcard characters** – * and ?.

Ordinarily, the Visual SourceSafe FindinFiles command is case-sensitive and looks for exact matches: the letter "A" does not match the letter "a". The -IC option makes the search case-insensitive, so "A" matches "a".

FindinFiles returns an exit code of 1 if there were 1 or more matches, 0 if there were no matches and 100 if an error occurred.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-V Command-Line Option](#)

[-Y Command-Line Option](#)

[Finds In Files Command](#)

[Using the Visual SourceSafe Command Line](#)

Branch (Command Line)

See Also

Breaks a share link and creates a branch of a file in a project.

Syntax

ss Branch *Visual SourceSafe file* [-C] [-H] [-I] [-N] [-O] [-Y] [-?]

Examples

Branches HELP.C in the current project away from its counterpart in other projects:

```
ss Branch HELP.C
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all branched files.
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Add access right to use this command.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-Y Command-Line Option](#)

[Branch Command \(SourceSafe Menu\)](#)

[Using the Visual SourceSafe Command Line](#)

Share (Command Line)

See Also

Makes the specified file or project a member of the **current project**.

Syntax

ss Share *Visual SourceSafe files* [-C] [-E] [-G] [-H] [-I] [-N] [-O] [-P] [-R] [-V] [-W] [-Y] [-?]

Examples

Makes \$/WORD/TEST.C a member of the current project:

```
ss Share $/WORD/TEST.C
```

Makes version 6 of TEST.C a member of the current project (note the command verb, Pin):

```
ss Pin TEST.C -V6
```

Creates a branch from the project \$/WORD:

```
ss Share $/WORD -R -E
```

The following table describes the command-line options available with this command.

Option	Description
-V	Share an earlier version of a file (or project).
-G	Set Get options for the retrieved <u>local copy</u> .
-W	Make the retrieved local copy writable.
-R	Share an entire project list recursively.
-P	Specify a new name for the shared project.
-C	Use the same comment for all shared projects (Available only when sharing projects.).
-O	Paginate or redirect the command output.
-E	Branch after sharing.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Check Out **access right** in the project you are sharing from, and the Add access right in the project you are sharing to, to use this command.

Remarks

To share a file, first set the current project path to the project that you want to share the file into with the CP command. Then execute the Share command, specifying the existing file(s) that you want to bring into your current project.

If you use the -E option, the file becomes a member of your current project, but it is branched: changes made in this project do not propagate to the originating project, and vice versa. The Links tab in the Properties dialog box (File menu) no longer shows a relationship, but you can use the Paths tab in the Properties dialog box to see the path of the file branch you have created.

Sharing an Explicit Version of a File or Project

To share an explicit version of a file, use the Pin command. With files, the -V option on Share is supported, but is not recommended. It specifies a file (that may or may not already be in the current project) and shares a specific version of it. When a project is sharing a specific version, it always uses that version no matter what changes are made in other projects. (If you are sharing a specific version, the `ss Dir` command displays this information.) The advantage of sharing a specific version of a file is that it allows you to control, in your project, the pace of changes being made by other users in other projects.

With projects, the -V option specifies an earlier project version to share. This creates a new project based on an existing one, but the new project is based on a specific version of the original project, rather than the current version. If you do not branch, the new project still shares all its files with the original, but it shares earlier versions of those files. If you branch, the new files are not shared with the old; it is as if you had rolled back all the files in the new project.

Sharing a Project

When you specify a project to share, instead of sharing a file from another project, you create an entirely new project under the current project that is an exact duplicate of the project you specified.

The -R option recursively shares a project and creates a new project list that shares all its files with the project list you specify.

If you use the -E option with a project, the new project does not share its files with the original project; all the files are branched.

The -P option specifies a new name for the newly created project. By default, the new project has the same name as the project you are sharing from. This default does not work, however, if you want the new project to be under the same parent as the old: Use the -P option to create a new branch with a new name. For example, if you share the project `$/TESTDATA` into your project `$/GIZMO/TESTDATA` without using the -P option, you create a situation where the subproject has the same name as the parent. With -P, you can avoid this as follows: `SS SHARE -PSHARDATA $/TESTDATA`

See Also

[-G Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-R Command-Line Option](#)

[-V Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[Pin \(Command Line\)](#)

[Share Command](#)

[Sharing Files](#)

[Using the Visual SourceSafe Command Line](#)

Status (Command Line)

See Also

Shows check out information on files.

Syntax

ss Status [*Visual SourceSafe items*] [-H] [-I] [-N] [-O] [-P[*project*]] [-R] [-U[*username*]] [-Y] [-?]

Examples

Lists all **checked-out files** in the current project:

```
ss Status
```

Indicates whether TEST.C is checked out by user Joyce:

```
ss Status TEST.C -UJoyce
```

Shows every file in the system that is checked out by the current user:

```
ss Status $/ -R -U
```

Shows files in the \$/TESTDATA project that are checked out in \$/TESTDATA. Does not show files shared into \$/TESTDATA but checked out in other projects:

```
ss Status -P$/TESTDATA $/TESTDATA
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-R	Display recursive status through an entire project list.
-U	Display only files checked out to a certain user.
-P	Filters the list of checked-out files to include only files checked out to the current or specified project.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to use this command.

Remarks

If you have the same file checked out to multiple computers, the Status command shows the computer name as well as the check out path. If you checked out the file using UNC path format, the Status command uses this format in its return.

The -U option lists files checked out by the specified user. If you use the -U option with no username, the current user (you) is assumed. If you do not use the -U option at all, all checked out files are displayed.

When a file is shared among multiple projects, if it is checked out from one project, it is checked out from all projects. For this reason, the Status Search command may actually show you files that are checked out to multiple projects, because they are shared. The -P option restricts the display to files that were checked out from a certain project. By default, this is the project on which you are performing the Status Search.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-Y Command-Line Option](#)

[Status Search Command](#)

[Using the Visual SourceSafe Command Line](#)

Undocheckout (Command Line)

See Also

Cancels a check out, voiding all changes.

Syntax

ss Undocheckout *Visual SourceSafe files* [-G] [-H] [-I] [-N] [-O] [-P[*project!*]] [-R] [-W] [-Y] [-?]

Examples

Cancels your check out of the file TEST.C:

```
ss Undocheckout TEST.C
```

Cancels all check outs made from the **current project**:

```
ss Undocheckout * -P
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-G	Set Get options for the retrieved local copy .
-W	Make the retrieved local copy writable.
-R	Undo Check Out of an entire project list recursively.
-P	Undo Check Out only of files checked out to the current or specified project.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Check Out **access right** to use this command.

Remarks

If you specify the -P option, the command acts only on items that are checked out from the specified project. If you do not specify a particular project after the -P, the **current project** is assumed.

Undocheckout performs a Get operation when it is done, unless the Delete_Local initialization variable in the SS.INI file is set to Yes, or you use the -G- option.

Undocheckout ignores files not checked out to you.

See Also

[-G Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[CheckIn \(Command Line\)](#)

[CheckOut \(Command Line\)](#)

[Undo Check Out Command](#)

[Using the Visual SourceSafe Command Line](#)

Checkin (Command Line)

See Also

Updates Visual SourceSafe with changes made to a **checked-out file**, and unlocks the Visual SourceSafe **master copy**. You can also use the command name Update (for backward compatibility).

Syntax

ss Checkin *Visual SourceSafe files* [-C] [-H] [-I] [-K] [-N] [-O] [-P[*project!*]] [-R] [-W] [-Y] [-?]

Examples

Checks in your changes to the file TEST.C:

```
ss Checkin TEST.C
```

Checks in all files that you have checked out from the current project:

```
ss Checkin * -P
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all checked-in files.
-K	Keep the file checked out after checking it in.
-G	Set options for the retrieved local file.
-W	Make the retrieved local copy writable.
-R	Check in an entire project list recursively.
-P	Check in only files checked out from the current or specified project.
-O	Paginate or redirect the command output.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Check Out **access right** to use this command.

Remarks

If you check in many files at once, Visual SourceSafe asks for a comment for each file individually: If you want them all to have the same comment, use the -C option.

It is possible to check in a file and create a new version while leaving the file checked out. To keep a file checked out, use the -K option. You can also make this behavior the default by setting the Keep_Checkedout initialization variable in the SS.INI file. Use the -K- option to indicate that you do not want to keep the file checked out.

Sometimes you check in a file that has not changed and create a new version. Visual SourceSafe displays a message asking if you actually want to perform a Check In; if you say No, Visual SourceSafe performs an Undo Check Out, so that the file is unlocked, but no new version is logged. You can avoid this prompt by setting the Update_No_Change initialization variable in your SS.INI file.

The -P option acts only on items that are checked out from the specified project. If you do not specify a particular project after the -P, the current project is assumed.

See Also

[-C Command-Line Option](#)

[-I Command-Line Option](#)

[-G Command-Line Option](#)

[-N Command-Line Option](#)

[-R Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[Check In Command](#)

[CheckOut \(Command Line\)](#)

[UndoCheckOut \(Command Line\)](#)

[Keep_Checkedout Initialization Variable](#)

[Update_No_Change Initialization Variable](#)

[Using the Visual SourceSafe Command Line](#)

View (Command Line)

See Also

Displays the text of a file.

Syntax

ss View *Visual SourceSafe files* [-H] [-I] [-N] [-O] [-V] [-Y] [-?]

Examples

Shows the contents of the Visual SourceSafe file TEST.C:

```
ss View TEST.C
```

Retrieves version 5 of TEST.C into a file called TEST.OLD (text files only):

```
ss View TEST.C -V5 -O TEST.OLD
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-V	View an earlier version of a file.
-I	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to use this command.

Remarks

The View command displays the text of a Visual SourceSafe file on the screen in much the same manner as the **type** command in MS-DOS. It does not perform a Get operation on the file or change anything in either Visual SourceSafe or your **working folder**.

By default, Visual SourceSafe displays the file by scrolling it on the command line. If you want the text displayed differently, set the Editor initialization variable to specify the text editor in which you want to view the text of the file. You can also use the Extension_Association initialization variable to set the editor associated with different file extensions.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-V Command-Line Option](#)

[-Y Command-Line Option](#)

[Editor Initialization Variable](#)

[Extension Association](#)

[Using the Visual SourceSafe Command Line](#)

Whoami (Command Line)

See Also

Shows your current Visual SourceSafe **username**. Available only from the command line. To see who you are logged in as in the Explorer, check the status bar.

Syntax

ss Whoami [-H] [-I] [-Y] [-?]

The following table describes the command-line options available with this command.

Option	Description
-I-	Ignore: Do not ask for input under any circumstances.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

Any Visual SourceSafe user can use this command.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-Y Command-Line Option](#)

[Using the Visual SourceSafe Command Line](#)

Pin (Command Line)

See Also

Pins a specific version of a file in the current project.

Syntax

ss Pin *Visual SourceSafe items* [-G] [-H] [-N] [-V] [-W] [-Y] [-?]

Examples

Pins the file JUSTIFY.CPP in \$/WORD/CODE to version 2:

```
SS Pin $/WORD/CODE/JUSTIFY.CPP -V2
```

Pins all the files in the \$/EXCEL/PIVOTTBL project to the label "beta4." In this example, "beta4" is assumed to be a project label:

```
SS Pin $/EXCEL/PIVOTTBL/*.* -Vbeta4
```

Pins all header files in the SPECS project to version 2.

```
SS Pin *.H -V2
```

The following table describes the command-line options available with this command.

Option	Description
-V	Version number you want to pin to the current project.
-G	Set Get options for the retrieved <u>local copy</u> .
-W	Make the retrieved local copy writable.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Check Out access right in the project you are pinning from to use this command.

Remarks

The Pin command is similar to the Share command with the -V switch, but the Pin command is the preferred method of locking a file at a certain version. However, Pin operates directly on files in projects specified on the command line. Without the -V option, Pin does not share files with the current project as the Share command does. Essentially, Pin freezes a file at a specified version number.

For Visual SourceSafe items, specify files in the current project or a project path containing a wildcard specification or filename. Pin only operates on the files in the current project or project specified.

If the file to be pinned is currently pinned to another version, the Pin command unpins the file from it and pins it to the version specified on the command line. If the file is already pinned to the specified version, no action is taken.

The command pins one version of the file in the current folder or working folder, depending on the state of the initialization variable Force_Dir: Setting Force_Dir = Yes pins files to the working folder. Setting Force_Dir = No pins files to the current folder. This variable can also be set in the Command Line tab of the Options dialog box (Tools menu).

The -G and -W options control the file attributes of the retrieved file.

See Also

[-G Command-Line Option](#)

[-N Command-Line Option](#)

[-V Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[Pin Command](#)

[Using the Visual SourceSafe Command Line](#)

Unpin (Command Line)

See Also

Undoes a Pin operation.

Syntax

ss Unpin *Visual SourceSafe items* [-G] [-H] [-N] [-W] [-Y] [-?]

Examples

Cancels the pinned status of MYFILE.CPP and MYFILE.H:

```
ss Unpin MYFILE.CPP MYFILE.H
```

Cancels the pinned status of all .DOC files in the current project:

```
ss Unpin *.DOC
```

Cancels the pinned status of all files in the SPECS project:

```
ss Unpin $/CEDAR/SPECS/*.*
```

The following table describes the command-line options available with this command.

Option	Description
-G	Set Get options for the retrieved local copy .
-W	Make the retrieved local copy writable.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Check Out **access right** in the project you are unpinning to use this command.

Remarks

The Unpin command searches the files specified. For files pinned to a specific version, the Unpin command returns them to sharing the latest version in the project.

The Unpin command ignores files that are not pinned.

See Also

[-G Command-Line Option](#)

[-N Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[Pin \(Command Line\)](#)

[Share Command](#)

[Unpin Command](#)

[Using the Visual SourceSafe Command Line](#)

Properties (Command Line)

See Also

Shows basic information about a Visual SourceSafe file or project.

Syntax

ss Properties *Visual SourceSafe Items* [-B] [-F-] [-H] [-I-] [-L] [-N] [-O] [-R] [-Y] [-?]

Examples

Shows property information on the file TEST.C, and changes its file type to binary:

```
ss Properties -B TEST.C
```

The following table describes the command-line options available with this command.

Option	Description
-B	Change the type of a file. -B changes the file type to binary; -B- changes it to text.
-F-	Disable the display of file properties. The -B, -B-, -L, and -L- options only operate on displayed files.
-O	Paginate or redirect the command output.
-R	Show information on an entire project list recursively. Ignored if a filename or file mask is specified on the command line.
-L	Store only the most recent version (not past versions). -L stores only the most recent version during check in. -L- stores delta information on each check in.
-I-	Ignore: Do not ask for input under any circumstances.
-N	Change between long and short filename mode.
-Y	Specify a username and/or password.
-?, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to view an item's properties.

You must have the Add access right to change the type, text or binary, of a file.

You must have the Destroy access right to use the -L option.

Remarks

The Properties command displays information about whether a file is text or binary, and additionally enables you to store only the latest version or all versions of a file.

If the Properties command is executed without the -B or -L option, it displays information about the file or project without changing the attributes of the file. With either the -B or -L option, it changes the attributes of the file first, then displays the results of the change.

See Also

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-Y Command-Line Option](#)

[Properties Command](#)

[Using the Visual SourceSafe Command Line](#)

Using the Visual SourceSafe Command Line

See Also

You can use the Visual SourceSafe command line to run batch files and macros. In addition, you can use the command line to perform all **Visual SourceSafe Explorer** commands.

Carrying Out Commands on the Command Line

You can start a command shell from Windows 95 or Windows NT and type SourceSafe commands from there. To do this, the Visual SourceSafe folder must be on your path (set with the PATH environment variable).

You can also use the Run command (File menu) from the Run command (Start menu) to run a specific command. For example, you can type `ss Get *.*` to get all files in the **current project**.

Command-Line Syntax

Command-line commands have the following general syntax:

ss *command* [*one or more items*] [*one or more options*]

What an item is depends on the command. For instance, for the Get command, an item is a Visual SourceSafe file or project; for the Add command, an item is an operating system file.

In almost all cases where you can specify an item to act on, you can specify more than one item. For instance, any of the following are valid ways of executing the Get command:

```
ss Get HELLO.C
```

```
ss Get HELLO.C HELLO.H TEST.H makefile
```

```
ss Get *.C *.H makefile linkfile
```

```
ss Get *.*
```

The command line is case-insensitive, that is, the command `ss Get HELLO.C` can also be typed as `ss get Hello.c`.

Note You do not have to type an entire command name for the Visual SourceSafe command line to recognize it; you need only type enough of the name for Visual SourceSafe to be able to distinguish it from every other command. In most cases, typing the first two or three characters is sufficient.

Command-Line Options

In addition to the command and the items, you can specify options, or parameters. Options are preceded by a hyphen and modify the command. A few options are specific to a command, and are described in that command's command-line Help topic. There is also a set of general options applicable to many commands. They are described in the following table.

Option	Description
-C	Use one (or no) comment for all specified items. (The Dir command uses this option in a different way.)
-O	Redirect or paginate the output of the command.
-V	Operate on a specific version of an item.
-R	Perform a command recursively.
-S	Turn SmartMode on or off for the command.
-G	Change the way the Get command works on any file retrieved to the local drive.

- W Indicate whether **local copies** are **read only**.
- N Change between long and short filename mode.
- I Ignore: Do not ask for input under any circumstances.
- Y Specify **username** and **password**.
- ?, -H Request online Help with a command: The -H command-line option obtains the same result.

Use quotation marks around the entire parameter whenever an item or option contains the space character as part of a string of text. For example, use "-CThis is the comment" rather than -C"This is the comment". Otherwise, Visual SourceSafe treats the space as a separator between different parameters, and the command either fails or yields unexpected results.

See Also

[-C Command-Line Option](#)

[-G Command-Line Option](#)

[-H Command-Line Option](#)

[-I Command-Line Option](#)

[-N Command-Line Option](#)

[-O Command-Line Option](#)

[-R Command-Line Option](#)

[-S Command-Line Option](#)

[-V Command-Line Option](#)

[-W Command-Line Option](#)

[-Y Command-Line Option](#)

[-? Command-Line Option](#)

[Command Line Exit Codes](#)

[Dir \(Command-Line\)](#)

[Starting Visual SourceSafe from the Command Line](#)

[Entering Comments on the Command Line](#)

[Using Alternate Commands on the Command Line](#)

Entering Comments on the Command Line

See Also

When the command line prompts you for a comment, type your comment, and then press the ENTER key. Comments for the Check Out command can be up to 63 characters in length; comments for all other commands can be up to 4K. Visual SourceSafe wraps the text to the next line as you type.

If you want to force a carriage return without ending the comment, type a backslash (\) and then press ENTER. The backslash is not part of the comment. Keep in mind that, after inserting a carriage return, you cannot press BACKSPACE to return to previous paragraphs—you must cancel the operation and start over.

You can use the -C command-line option for many items as well as for one item. This enables you to type one comment and apply it to all the files.

See Also

-C Command-Line Option

-C Command-Line Option

See Also

Use the -C option for any command that requires a comment. If you do not specify the -C option, Visual SourceSafe asks for a comment for each item you specify on the command line. If you do specify the option, all items receive the same comment.

The following table lists ways to use the -C option.

Option Example	Description
-C	Ask for one comment, and then use it for all items.
-C-	Do not use a comment.
-C <i>text</i>	Use the string <i>text</i> as the comment for all items.
-C@FILE.TXT	Use the text file FILE.TXT as the comment for all items.
-C?	Return to Visual SourceSafe's default; ask for a comment for each individual file.

To set any of the option variations listed in the table as a default for the Check Out command, use the Checkout_Comment initialization variable in SS.INI. If you have set a default in SS.INI, you can replace it for a particular command with the option -C?, and Visual SourceSafe asks for a comment for each individual file (Visual SourceSafe's default).

You can set the Comment_Editor variable in SS.INI to invoke an executable editor of your choice as an alternative to the prompt-driven method of comment entry Visual SourceSafe uses.

See Also

Checkout_Comment Initialization Variable

Comment_Editor Initialization Variable

-O Command-Line Option

The -O option controls the output from commands that might display large amounts of information. If you do not specify a -O option, the information scrolls by on the screen too quickly to be read.

The following table lists ways to use the -O option.

Option Example	Description
-O	Format output in pages, waiting for a keystroke each page.
-O-	Do not provide output other than errors.
-O@FILE.TXT	Send all output to the file FILE.TXT. This is the command-line method for exporting reports, such as the History report, to a filename or printer. If the file named already exists, output is appended to that file.
"-O&FILE.TXT"	Output standard errors to the file FILE.TXT, in addition to the command output. Note the quotation marks required on NT platforms, and optional on others. If the file named already exists, output is appended to that file.
-O&-	Do not provide output.

-V Command-Line Option

The -V option specifies earlier versions of files or projects. For example, you may want to view a file as it looked last week, or a project exactly as it was last month.

The following table lists ways to use the -V option.

Option Example	Description
-V14	Display version 14.
-Vd2-29-92	Display the version dated 2-29-92.
"-VL Final Beta"	Display the version with the label Final Beta.

The -V option displays all files and projects identified by the **version number** specified on the command line. You can also specify a version number for one particular item, by following that item with a semicolon and then the version number.

The following table shows how to specify a particular item.

Option Example	Description
HELP.C;14	Display version 14 of HELP.C.
SHOW.PRG;d2-29-92	Display SHOW.PRG as it appeared on 2-29-92.
"\$/CW;L Beta 1"	Display the version of the project \$/CW with the label Beta 1.

Occasionally, you will want to reference a time with the date in a Visual SourceSafe command. To do this, reference the time along with the date using the -vd switch.

For example, to list the history of the \$/TEST project from 9 a.m. to 3 p.m. on 3/3/95, issue the command:

```
ss history $/test -vd3/03/95;3:00p~3/03/95;9:00a
```

In the above command "a" or "p", refers to a.m. or p.m. respectively. And the ~ is used to indicate you want the history between the dates and times specified. Note that the later date and time must be specified first.

-R Command-Line Option

The -R option is used with Visual SourceSafe commands that operate on projects; it makes the commands recursive to subprojects. Normally, commands act only on the project and the files in it: If you get a project, you get all the files that are in that project. Subprojects, and the files and subprojects inside them, however, are ignored. When you use the -R option to make a command recursive, the command acts on the entire project hierarchy. For example, if you type `SS Get projectname -R`, folders are created for that project's subprojects, and the subprojects inside of them, and so on.

-G Command-Line Option

See Also

The Get command copies files from a Visual SourceSafe project into a folder. Visual SourceSafe gets files in response to the Get command and also in response to many other commands such as Check Out, Undo Check Out, and Share.

For any of these commands, the -G option can be used to change the nature of the Get operation. Many of these options provide command-specific overrides that you can set in your SS.INI file.

The following table lists ways to use the -G option.

Option	Example	Description
-G-		Don't perform a Get operation.
-GL	<i>path</i>	Copy a file to the specified folder, not to the current or working folder .
-GTC		Give the local copy the current date and time.
-GTM		Give the local copy the date and time that the file was last modified, not the current date and time.
-GTU		Give the local copy the date and time that the file was last updated, not the current date and time.
-GCC		Determine whether the local copy of the file is up-to-date by comparing the file's contents.
-GCK		Use a checksum to determine whether the local copy of the file is up-to-date.
-GCD		Use a date and time comparison to determine whether the local copy of the file is up-to-date.
-GR		Specify the carriage return as the end-of-line character in text files.
-GN		Specify the linefeed as the end-of-line character in text files.
-GRN		Specify the carriage return–linefeed pair as the end-of-line character in text files. This is the default on the PC.
-GF		Turn on the Force_Dir initialization variable for this command.
-GF-		Turn off the Force_Dir initialization variable for this command.
-GWA		Display a dialog asking the user to choose between replacing, skipping, or merging writable files on Get Latest Version and Check Out operations.
-GWR		Replace writable files on certain operations (Get and Check Out).
-GWS		Skip writable files on certain operations (Get and Check Out).
-GWM		Merge writable files on certain operations (Get and Check Out).

-G Command Line Option See Also
Force_Dir Initialization Variable

-S Command-Line Option

See Also

The `Smart_Mode` initialization variable in `SS.INI` puts Visual SourceSafe in SmartMode, in which it follows many actions with logical next steps. With SmartMode on:

- After you create a project, SmartMode sets it as your current project.
- When you delete or destroy a file, SourceSafe automatically deletes the corresponding local file.
- When you rename a file, SourceSafe automatically renames the corresponding local file.

This command-line option overrides the `Smart_Mode` variable for a particular command.

The following table lists ways to use the `-S` option.

Option	Example	Description
-S		Turn on SmartMode for this command.
-S-		Turn off SmartMode for this command.

See Also

Smart_Mode Initialization Variable

-W Command-Line Option

See Also

When you do not have a file checked out, your **local copy** is locked, or made **read-only** to remind you that you should not change it. The Use_ReadOnly variable in SS.INI changes this default behavior. The -W variable changes it for a specific command.

The following table lists ways to use the -W option.

Option Example	Description
-W	Make local files writable.
-W-	Make local files read-only.

See Also

Use ReadOnly Initialization Variable

-N Command-Line Option

On operating systems that support long filenames, the -N option changes between long and short filename mode one command at a time.

The following table lists ways to use the -N option.

Option Example	Description
-NL	Use long filename mode.
-NS	Use short filename mode.

-I Command-Line Option

There are a number of circumstances in which Visual SourceSafe commands ask for input from the user – warnings containing yes or no questions, for example. This is not wanted when you write scripts or macros that execute the Visual SourceSafe command line from inside other programs. In these cases, you can use the -I- option to ensure that Visual SourceSafe never asks for user input.

When using -I, it is almost always a good idea to use the -C option to avoid the comment prompt. The following table lists ways to use the -I option.

Option Example	Description
-I-Y	Instruct Visual SourceSafe to answer Yes to all Yes or No questions.
-I-N	Instruct Visual SourceSafe to answer No to all Yes or No questions.

-Y Command-Line Option

The -Y option specifies a **username** in case you occasionally want to execute a command as a user other than yourself.

The following table lists ways to use the -Y option.

Option Example	Description
-YMary	Specify a username.
-YMary,Birthmark	Specify a username and <u>password</u> .

-? Command-Line Option

Use the `-?` option with any command to get Help on that command. You can also use the `-H` option to obtain online Help.

Option Example	Description
<code>ss Share -?</code>	Displays Help on the Share command.

Command-Line Exit Codes

See Also

Whenever you run Visual SourceSafe's command line, Visual SourceSafe sets the exit code to indicate its running status. You can use this operating system variable in batch files, scripts, and macros to react based on what happened when you ran Visual SourceSafe.

The following table lists the Visual SourceSafe exit codes.

Exit Code	Description
100	Indicates a serious error (Visual SourceSafe could not find its data files, or a file you want to check out is already checked out).
1	Indicates a less serious error. This type of error can occur in three circumstances: if you run <code>ss Dir</code> and no items are found; if you run <code>ss Status</code> and at least one item is checked out; if you run <code>ss Diff</code> , and at least one file that you compare is different. All of these circumstances indicate that what you want to do next may fail, even though this Visual SourceSafe command did not.
0	Indicates that Visual SourceSafe executed the command successfully.

The following is a sample batch file that uses the Status exit code. This batch file runs the Status command, using the `-O-` parameter to disable all output; and then gives its own form of Status display, based on the exit code from SourceSafe. Although not particularly useful, this does give a very simple example of how to use the exit code for more constructive purposes. (Note: the following syntax is MS-DOS-specific. This file could not be used, as such, under the Macintosh or UNIX, although the concept could be applied to write similar scripts.)

```
@echo off
ss stat -O- %1 %2 %3 %4 %5 %6 %7 %8 %9
if errorlevel 100 goto BAD_FAILURE
if errorlevel 1 goto CHECKED_OUT

rem Exit code 0, nothing is checked out
echo No specified files are checked out
goto END

rem Exit code 1, something is checked out
:CHECKED_OUT
echo One or more files are Checked out
goto END

rem Exit code 100, something went badly wrong
:BAD_FAILURE
echo SourceSafe could not run successfully

:END
```

See Also

[Diff \(Command Line\)](#)

[Folder \(Command Line\)](#)

[Status \(Command Line\)](#)

Using Alternate Commands on the Command Line

See Also

Visual SourceSafe provides two ways to customize Visual SourceSafe to accept alternate command syntax in place of built-in Visual SourceSafe command syntax.

You can define alternate commands and place them in the initialization files under the [Commands] heading. The syntax takes the form `Alternate Command = Visual SourceSafe Command`. For instance, a Microsoft Windows NT user might place the following in SS.INI.

```
[Commands]
Grep = Find
Ls = Dir
```

You can type the alternate command in place of Visual SourceSafe command, with the same parameters and results; Visual SourceSafe commands continue to work too.

Visual SourceSafe also provides hard-coded alternate commands. The following table lists alternate commands and their Visual SourceSafe equivalents.

Alternate	Visual SourceSafe
CD	CP
ChPro	CP
Copy	Share and Branch
Remove	Delete
Grep	Find
LS	Directory
MD	Create
MkPro	Create
MP	Create
Type	View
Undelete	Recover
WhereIs	Locate
Scan	FindInFiles
Separate	Branch
Uncheckout	UndoCheckout
Update	CheckIn

See Also

[CP \(Command Line\)](#)

[Create \(Command Line\)](#)

[Delete \(Command Line\)](#)

[Folder \(Command Line\)](#)

[Find \(Command Line\)](#)

[Locate \(Command Line\)](#)

[Recover \(Command Line\)](#)

[Share \(Command Line\)](#)

[View \(Command Line\)](#)

About (Command Line)

Displays information about your copy of Microsoft Visual SourceSafe, including the version number and the copyright, legal and licensing notices.

Access Rights

Any Visual SourceSafe user can use this command.

Starting Visual SourceSafe from the Command Line

You can use command-line options to start the Visual SourceSafe client against a specified database. With these options, you can easily assign different Visual SourceSafe icons to different SourceSafe local or remote databases.

The following command-line options are supported:

Command-Line Option	Description
-y user,pwd	Identifies the user's name and password, avoiding the Login dialog box.
-s \\server\share\path	Identifies a specific SRCSAFE.INI file, overriding Visual SourceSafe's normal search method for SRCSAFE.INI.
-d var	Identifies a data path to use in the SRCSAFE.INI file. If specified, this value specifies a DATA_PATH variable used in the SRCSAFE.INI file.
-p project	Identifies the project to open Visual SourceSafe in. For example, SSEX -p\$/Word/Code would launch Visual SourceSafe in that project.

All of these options are optional, and none are dependant on others.

Remarks

To use these options effectively, assign the appropriate command line to different SourceSafe icons in your Visual SourceSafe program group. For example, to use one icon to access a SourceSafe database at \\DEV1\SourceSafe, assign the following line to one icon:

```
SSEX -s\\DEV1\SourceSafe
```

The -d option is a powerful feature that can be used to configure a SRCSAFE.INI file with multiple data path variables. (See the Visual SourceSafe Administrator's Help file for more information on the Data_Path variable.) For example:

```
Data_Path (Word) = \\Office\WORD\SSDB  
Data_Path (Excel) = \\Office\EXCEL\SSDB  
Data_Path (PowerPoint) = \\Office\PPT\SSDB
```

Using an icon to launch the SourceSafe Explorer on the Powerpoint database, for example, you would assign this command line to a SourceSafe icon:

```
SSEX -s\\Office\Source -dPowerpoint
```

This command causes SourceSafe to search the root folder of the \\Office\Source share for SRCSAFE.INI. IN SRCSAFE.INI, SourceSafe would look for the Powerpoint Data_path variable assignment and would be directed to the SourceSafe database at \\Office\PPT\SSDB.

The -p option can be used in email software (that supports icons) to send a pointer to your Visual SourceSafe project, or to create a Windows icon that always opens Visual SourceSafe in a particular project.

Cloak (Command Line)

See Also

Hides a project from recursive Get, Check Out, Check In, Undo Check Out, and Project Differences commands.

Syntax

ss Cloak *Visual SourceSafe project path*

Examples

Cloaks the \$/Word/Code project:

```
ss Cloak $/Word/Code
```

Access Rights

Any Visual SourceSafe user can use this command.

Cloak Command See Also

General Project Properties Tab

Cloaking Visual SourceSafe Projects

Decloak Command Line Command

Decloak (Command Line)

See Also

Removes the cloaked attribute from a project.

Syntax

ss Decloak *Visual SourceSafe project path*

Examples

Decloaks the \$/Word/Code project:

```
ss Decloak $/Word/Code
```

Access Rights

Any Visual SourceSafe user can use this command.

Cloak Command See Also

General Project Properties Tab

Cloaking Visual SourceSafe Projects

Cloak Command Line Command

Deploy (Command Line)

See Also

Deploys a project to a server or ftp location specified by the Deploy_Path initialization variable.

Syntax

ss Deploy *Visual SourceSafe project path* [-O] [-R] [-V] [-G] [-W]

Examples

Deploys a web project (\$/SSAFE) to the deployment location specified by the Deploy_Path initialization variable:

```
ss Deploy $/SSAFE
```

The following table describes the command-line options available with this command.

Option	Description
-O	Paginate or redirect the command output.
-R	Recursively deploy an entire project list.
-V	Deploy an old project version.
-G	Set options for the retrieved copy.
-W	Make the retrieved local copy writable.
-, -H	Request online Help on a command.

Access Rights

You must have the Read **access right** to use this command.

Remarks

The Deploy command places files into a location specified by the Deploy_Path variable. This variable can be set directly in the SRCSAFE.INI file, or by using the Administrator program. Deploy is very similar to the Get Latest Version command, except that you are not limited to putting the files into your working folder and you cannot avoid overwriting existing files. Used recursively on a project (the -R option), Visual SourceSafe deploys all the files in the project and subprojects, and places them into subfolders of the same name.

By default, Deploy copies the most recent version of the projects you specify. However, you can deploy earlier versions as well by using the -V option. (Note that "ss Deploy -V*" does not work.)

Note Deploying an earlier version of a project gets the files that were in the project at a particular time, as they were at that time. For instance, if you use the Deploy command to retrieve version 7 of the project \$/WORD, you may get TEST.HTML version 3, HELP.HTM version 10, and TEST.MAP version 5.

Deploy Command See Also

Deploy Command

Get Latest Version Command Line Command

Deploy_Path Initialization Variable

WorkFold

Sets the working folder. In the Visual SourceSafe Explorer, use the [Set Working Folder](#) command.

Syntax

SS Workfold [Project] [Folder]

Examples

Sets the \$/Website project working folder to c:\External Web\Website:

```
ss Workfold $/WEBSITE c:\External Web\Website
```

The following table describes the command-line options available with this command.

Option	Description
Project	Specifies the Visual SourceSafe project you want to set a working folder for.
Folder	Specifies the working folder for the specified project.
-, -H	Request online Help on a command.

Access Rights

You must have the Read [access right](#) to use this command.

Remarks

This command makes the specified folder the working folder for the specified project.

Note that both the project and the directory are optional. If either one is not specified, the *current* project or folder is used. That is, if you do not specify a folder, the current folder is used. If you do not specify a project, the current project is used. If you specify more than one project or more than one folder, that is an error.

The output of the command is simply the string:

```
%s successfully set as the working folder for %s.
```

The Force_Dir and Force_Prj initialization variables are completely ignored by this command!

YNC

Compares the contents of a project with your **working folder** and reconciles the differences.

Syntax

ss YNC *Visual SourceSafe projects* [-?]

Examples

Compares the Visual SourceSafe project WORD to your working folder and reconciles differences:

```
ss YNC $/WORD
```

Access Rights

You must have Read **access right** to use this command.

Remarks

Reconciles project differences using the user-specified reconciliation behavior stored in the Reconcile_Project initialization variable.

Accessibility for People with Disabilities

Microsoft is committed to making its products and services easier for everyone to use. This help file provides information about the following features, products and services, which make Microsoft Windows, Microsoft Windows NT and Microsoft Visual SourceSafe more accessible for people with disabilities:

[Microsoft services for people who are deaf or hard-of-hearing.](#)

[Access Packs for either Microsoft Windows or Microsoft Windows NT, a software utility that makes using Windows and Windows NT easier for people with motion or hearing disabilities.](#)

[Keyboard layouts for single-handed users.](#)

[Microsoft software documentation on audio cassette, floppy disk and compact disc \(CD\).](#)

[Third-party utilities to enhance accessibility.](#)

[Hints for customizing Microsoft Windows or Microsoft Windows NT.](#)

[More information about products and services for people with disabilities.](#)

Note The information in this section applies only to users who purchased Windows or Windows NT in the United States. If you purchased Windows or Windows NT outside the United States, your Windows package contains a subsidiary information card listing Microsoft support services telephone numbers and addresses. You can contact your subsidiary to find out whether the type of products and services described in this help file are available in your area.

Microsoft Services for People Who Are Deaf or Hard-of-Hearing

Through a text telephone (TT/TDD) service, Microsoft provides people who are deaf or hard-of-hearing with complete access to Microsoft product and customer services.

You can contact Microsoft Sales Information Center on a text telephone by dialing (800) 892-5234 between 6:30 A.M. and 5:30 P.M. Pacific time. For technical assistance you can contact Microsoft Support Network on a text telephone at (206) 635-4948 between 6:00 A.M. and 6:00 P.M. Pacific time, Monday through Friday, excluding holidays. In Canada, dial (905) 568-9641 between 8:00 A.M. and 8:00 P.M. Eastern time, Monday through Friday, excluding holidays. Microsoft support services are subject to Microsoft prices, terms, and conditions in place at the time the service is used.

Access Packs for Microsoft Windows and Microsoft Windows NT

Microsoft distributes Access Packs for Microsoft Windows and Microsoft Windows NT, which provide people with motion or hearing disabilities better access to computers running Windows or Windows NT. (If you are running Microsoft Windows 95, these same Access Pack features are already built in. See online Help for more information.) Microsoft Windows and Microsoft Windows NT contain features that:

- Allow single-finger typing of SHIFT, CTRL, and ALT key combinations.
- Ignore accidental keystrokes.
- Adjust the rate at which a character is repeated when you hold down a key, or turn off character repeating entirely.
- Prevent extra characters if you unintentionally press a key more than once.
- Enable you to control the mouse cursor by using the keyboard.
- Enable you to control the computer keyboard and mouse by using an alternate input device.
- Provide a visual cue when the computer beeps or makes sounds.

Access Pack for Microsoft Windows is included on the Microsoft Windows Driver Library in the file ACCP.EXE. Access Pack for Microsoft Windows NT is included in the Microsoft Application Note WNO789. If you have a modem, you can download ACCP.EXE or WNO789.EXE, which are self-extracting archive files from the following network services:

- CompuServe®
- GENie™
- Microsoft OnLine
- Microsoft Download Service (MSDL), which you can reach by calling (206) 936-6735 any time except between 1:00 A.M. and 2:30 A.M. Pacific time. Use the following communications settings:

For this setting	Specify
Baud rate	1200, 2400, 9600 or 14400
Parity	None
Data bits	8
Stop bits	1

- Various user-group bulletin boards (such as the bulletin-board services on the Association of PC User Groups network)
- In /SOFTLIB/MSLFILES on the Internet servers FTP.MICROSOFT.COM and WWW.MICROSOFT.COM

People within the United States who do not have a modem can order the Access packs by calling Microsoft Sales Information Center at (800) 426-9400 (voice) or (800) 892-5234 (text telephone). In Canada, you can call (905) 568-3503 (voice) or (905) 568-9641 (text telephone).

#K\$+ Keyboard Layouts for Single-Handed Users

Microsoft distributes Dvorak keyboard layouts that make the most frequently typed characters on a keyboard more accessible to people who have difficulty using the standard "QWERTY" layout. There are three Dvorak layouts: one for two-handed users, one for people who type with their left hand only, and one for people who type with their right hand only. The left-handed or right-handed keyboard layouts can also be used by people who type with a single finger or a wand. You do not need to purchase any special equipment in order to use these features.

Microsoft Windows and Microsoft Windows NT already support the two-handed Dvorak layout, which can be useful for coping with or avoiding types of repetitive-motion injuries associated with typing. To get this layout use the Windows Control Panel; consult your on-line documentation for detailed instructions. The two layouts for people who type with one hand are distributed as Microsoft Application Note GA0650. It is also contained in file GA0650.EXE on most network services and on the Microsoft Download Service. For instructions on oining this application note see [Access Packs for Microsoft Windows and Microsoft Windows NT](#).

K\$+ Microsoft Documentation in Alternative Formats

People who have difficulty reading or handling printed documentation may obtain many Microsoft publications from Recording for the Blind, Inc. Recording for the Blind distributes these documents to registered eligible members of their distribution service either on audio cassettes or on floppy disks. The Recording for the Blind collection contains more than 80,000 titles, including Microsoft product documentation and books from Microsoft Press. You can contact Recording for the Blind at the following address or phone numbers for information on eligibility and availability of Microsoft product documentation and books from Microsoft Press:

Recording for the Blind, Inc.
20 Roszel Road
Princeton, NJ 08540

Phone: (609) 452-0606
Fax: (609) 987-8116

Manf the *Product Name* version x documents are also available on a CD that comes with the package.

K\$+ Third-Party Utilities to Enhance Accessibility

A wide variety of third-party hardware and software products are available to make personal computers easier to use for people with disabilities. Among the different types of products available for the MS-DOS, Microsoft Windows, and Microsoft Windows NT operating systems are:

- Programs that enlarge or alter the color of information on the screen for people with visual impairments.
- Programs that describe information on the screen in braille or synthesized speech for people who are blind or have difficulty reading.
- Hardware and software utilities that modify the behavior of the mouse and keyboard.
- Programs that enable users to “type” using a mouse or their voice.
- Word or phrase prediction software that allows one to type more quickly and with fewer keystrokes.
- Alternate input devices, such as single switch or puff-and-sip devices, for those who cannot use a mouse or a keyboard.

For more information on obtaining third-party utilities, see [Getting More Information](#). For more information about customizWindows and Windows NT for people with low vision, see [Customizing Windows or Windows NT](#).

K\$+ Customizing Windows or Windows NT

There are many ways you can adjust the appearance and behavior of Windows or Windows NT to suit varying eyesight and motor skills without requiring any additional software or hardware. These include ways to adjust the appearance as well as the behavior of the mouse and keyboard. The specific methods available depend on which operating system you are using. Application notes are available describing the specific methods available for each operating system.

See the appropriate application note for information related to customizing your operating system for people with disabilities. For information on obtaining application notes, see [Access Pack for Microsoft Windows and Microsoft Windows NT](#).

Operating system	Application note number
Microsoft Windows 3.0	WW0786.TXT
Microsoft Windows1	WW0787.TXT
Microsoft Windows for Workgroups 3.1	WG0788.TXT
Microsoft Windows NT 3.1 and 3.5	WN0789.EXE
Microsoft Windows95	WN1062

K\$+ Getting More Information

For more information on Microsoft products and services for people with disabilities, contact:

Microsoft Sales Information Center
One Microsoft Way
Redmond, WA 98052-6393

Voice telephone: (800) 426-9400
Text telephone: (800) 892-5234
Fax: (206) 635-6100

The Trace R&D Center at the University of Wisconsin-Madison produces a book and a compact disc that describe products that help people with disabilities use computers. The book, titled *Trace ResourceBook*, provides descriptions and photographs of about 2,000 products. The compact disc, titled *Co-Net CD*, provides a database of more than 18,000 products and other information for people with disabilities. It is issued twice a year. To obtain these directories, contact:

Trace R&D Center
S-151 Waisman Center
1500 Highland Avenue
Madison, WI 53705-2280

Voice telephone: (608) 263-2309
Text telephone: (608) 263-5408
Fax: (608) 262-8848

For general information and recommendations on how computers can help specific people, you should consult a trained evaluator who can best match your needs with the available solutions. An assistive technology program in your area will provide referrals to programs and services that are available to you. To locate the assistive technology program nearest you, you can contact:

National Information System
Center for Developmental Disabilities
Benson Building
University of South Carolina
Columbia, SC 29208

Voice/text telephone: (803) 777-4435
Fax: (803) 777-6058

delta

Delete command

Destroy command

development environment

drag and drop

file header

file list

file pane

G–K

inheritance

history

keyword

keyword enabled file

keyword expansion

L

label

local copy

locking

log on

M

master copy

merging

multiple check out

N–Q

parent project

password

project

project list

project pane

Purge command

R

read-only file

recursive operation

reverse delta

[rights propagation](#)

[root project](#)

[Results pane](#)

S

[security](#)

[shadow folder](#)

[shadow variable](#)

[shared file](#)

[share link](#)

[source code control](#)

[status bar](#)

[subproject](#)

T–V

[user list](#)

[username](#)

[version control](#)

[version number](#)

[version tracking](#)

[Visual merge](#)

[Visual SourceSafe administrator](#)

[Visual SourceSafe database](#)

[Visual SourceSafe Explorer](#)

W–Z

[web site project](#)

[wildcard characters](#)

[working folder](#)

user list

The list of users who can use the Visual SourceSafe database. This list is maintained by the Visual SourceSafe administrator and displayed in Visual SourceSafe Administrator's main window.

access rights

The levels of permission to use the Visual SourceSafe database that users are granted by the Visual SourceSafe administrator when project security is enabled. The levels of access rights are Read, Check Out, Add, and Destroy. For more information, see the Administration chapter of the *Microsoft Visual SourceSafe User's Guide*.

Advanced dialog box

A dialog box that contains optional or additional settings. Many SourceSafe dialog boxes have an Advanced button that displays these additional settings.

automatic merge

When multiple users have the same file checked out, their changes to the file are merged by Visual SourceSafe during check in.

branched file

A file whose share link has been broken using the Branch command.

branching

The process of sharing a file with another project and then separating it into two (or more) branches. Branching creates two paths of the file's history, both with a common version as the origin, but different subsequent versions.

checked-in file

A file stored in the Visual SourceSafe database and unavailable for modification.

checked-out file

A file that has been reserved for work by a user. Users check out files so changes can be made to them. In the default configuration, Visual SourceSafe allows only one user at a time to check out a file. Checking out a file copies its latest version into the user's working folder.

check out folder

The folder to which a file is checked out in Visual SourceSafe. It is important to distinguish this from the working folder. If you check out a file, the file is checked out to your working folder. From the perspective of another user, the file is in the check out folder. The check out folder is displayed in the Check Out Folder column of the file pane of Visual SourceSafe Explorer; the working folder is displayed under the toolbar.

Clipboard

A temporary storage location used by Microsoft Windows to transfer text, graphics, and code.

cloaking

Hiding a project from being affected by certain commands, namely Get, Check Out, Check In, Undo Check Out, and Project Show Difference.

column format

A formatting option used with keyword expansion within a file. This option (using two colons instead of one in the keyword instructions) tells Visual SourceSafe to align columns of keyword information for a more attractive and readable display.

conflict

Two or more differing changes to the same line of code in a multiple check out situation. Visual SourceSafe recognizes conflicts during a merge operation, either explicitly when a user clicks the Merge Branches command, or implicitly when checking in a file that multiple users have checked out.

conflict marker

A symbol used to designate conflicting changes to a file. These symbols are:

- <<<<<< SourceSafe version
- ===== Conflict separator
- >>>>>> Local version

Visual SourceSafe places these markers in the file after a conflicting check-in or merge operation, so that you can find and resolve the conflict more easily.

cross-platform development

Visual SourceSafe supports transparent file-compatibility across multiple processors and operating systems.

current project

The project selected in the project pane of the Visual SourceSafe Explorer window.

current version

The version of a file most recently stored in the Visual SourceSafe database. The current version has the highest version number of a file in Visual SourceSafe.

delta

Changes. In Visual SourceSafe, a delta is the difference between version X of a file and version X-1 of the same file. Visual SourceSafe uses reverse delta technology to store changes.

Delete command

Removes files and projects from a Visual SourceSafe project, and marks them as deleted; the items still exist, however, and can be recovered using the Recover command.

Destroy command

Permanently removes deleted files and projects from the Visual SourceSafe database. Once destroyed, the items cannot be recovered.

development environment

A set of software development tools, presented as a unified environment in which the software developer can efficiently work. Microsoft Visual Basic and Microsoft Visual C++ are examples of such environments, as they combine a coding language with tools and controls. Visual SourceSafe can be integrated into these environments, providing access to its source code control features.

drag and drop

A combination of features that allows the user to drag an item and drop it onto another item using the mouse. An item can be a source (the item the user drags) or a target (the item on which the user drops a source).

To drag an object, click the object, then while holding down the mouse button, move it to the new location. Release the mouse button to drop the object.

file header

Text at the beginning of a file that contains information, such as the revision number of the file, the author, the date and time of last modification, and so forth. So it doesn't interfere with compiling, this information usually appears as comments in programming language files.

file pane

The right side of the Visual SourceSafe Explorer window. This pane contains the file list, a list of all the files in the current project.

file list

The list of files in the current project, found in the file pane of the Visual SourceSafe Explorer window.

history

A record of changes to a file since it was initially added to Visual SourceSafe, which Visual SourceSafe keeps by using reverse delta technology. With the file history, you can return to any point in the file's history and recover the file as it existed at that point. The History of Project dialog box shows the record of significant events in the current project, such as labeling, and deletion or addition of files and subprojects.

inheritance

The inherited effect of variables in Visual SourceSafe initialization files that subprojects receive from their parent projects. Variables in these files can be grouped under headings to specify behavior. If a variable is set before any group heading in the initialization file, the variable affects all projects; the effect is inherited by the subproject variables.

keyword

A word with a special meaning to Visual SourceSafe. You can use keywords in your text files to automatically create a file header with version-specific information. Use Visual SourceSafe keywords to place information from Visual SourceSafe directly into a text file.

keyword enabled file

A file that contains Visual SourceSafe keyword information.

keyword expansion

The process Visual SourceSafe uses to replace a Visual SourceSafe keyword with meaningful header information when you use the Check In and Add Files commands. For example, the string "\$Revision: \$" tells Visual SourceSafe to expand the **Revision** keyword automatically with the current file version number.

label

A user-defined name you can attach to a specific version number of a file or project.

local copy

A copy of a file stored in your working folder on your local computer. The local copy may differ from the Visual SourceSafe master copy if the local copy has been changed since the last check out, or if the master copy was changed by another user while you were working on the local copy.

locking

A system of ensuring that two processes do not try to affect the same record in a database at the same time. To coordinate record access, Visual SourceSafe applies native locking, which uses native operating system functions. Visual SourceSafe also can be set to use lockfiles, which create temporary files in the LOCKS folder.

log on

The process of entering and verifying a user's name and password to access the Visual SourceSafe database.

master copy

The most recently checked-in version of a file stored in the Visual SourceSafe database, as opposed to the local copy of a file in your working folder.

merging

The process of combining differences in two or more changed copies of a file into a new version of the file. A merge involves at least two different files (which may be different versions of the same file or changes made to the same version of the file) and creates a new file made up of the results of the merge.

multiple check out

Simultaneous check outs by two or more users. Multiple check out is not possible unless it is enabled by the Visual SourceSafe administrator.

parent project

A project that contains one or more subprojects. A project can be both a parent project and a subproject at once, if it is in the middle of the project hierarchy.

password

A text string used as security to verify the identity of a user. A user password is often required to use the Visual SourceSafe database.

project

A group of related files, typically all the files required to develop a software component. Files can be grouped within a project to create subprojects. Projects can be defined in any way meaningful to the user(s)—as one project per version, or one project per language, for example. In general use, projects tend to be organized in the same way file directories are.

project pane

The left side of the Visual SourceSafe Explorer window. The project pane contains the project list, a list of all available projects in the Visual SourceSafe database.

project list

A list, in the left pane of the Visual SourceSafe Explorer window, of all the projects available in the Visual SourceSafe database.

Purge command

Permanently removes previously deleted files and projects from the Visual SourceSafe database. Once purged, the items cannot be recovered.

read-only file

A file marked as read-only in its file attributes. Such a file can be viewed in an appropriate text editor, but cannot be modified. Visual SourceSafe marks the file as read-only when you use the Check In and Get Latest Version commands.

recursive operation

An operation that is applied both to a parent project and to all the subprojects of that parent. For example, you can use the Check Out command recursively to check out all the files in the project list simultaneously, and avoid selecting each file individually.

Results pane

A portion of the Visual SourceSafe Explorer window where results from Visual SourceSafe operations are shown. For example, when you check in a file, this pane shows the file name being checked in.

reverse delta

The change-storage technology Visual SourceSafe uses, in which incremental changes to a baseline file are stored, rather than each successive version of the file in its entirety. In Visual SourceSafe, the current version of a file is used as the baseline, and the changes from the previous versions are saved. This technology results in reduced disk storage requirements and faster access times, because only the current version is always stored in the database in its entirety.

rights propagation

The default assignment of user-access rights in subprojects based on rights assigned in the parent project. This default assignment can be changed.

root project

The highest-level project, with the name \$/ in the project list. All projects in a Visual SourceSafe database are subprojects of the root project.

security

Visual SourceSafe has two levels of security: default security and project security. Default security provides two types of access rights: Read-Write and Read-Only. When project security is enabled, four access rights are available per user, per project: Read, Check Out, Add, and Destroy. Each succeeding right includes all rights preceding it. The Destroy access right provides unlimited access and is equivalent to Read-Write rights under default security.

shadow folder

A central, optional folder that contains current versions of all the files in a project. The shadow folder does not contain the master copy of a file or the local copy of a file. Instead, it provides a central location from which to view the overall structure of the project and serves as a convenient place to build or compile the project.

shadow variable

An initialization file variable used to specify the location of a shadow folder.

shared file

A file simultaneously used by, and part of, more than one project.

share link

The link between a file and the one or more projects it may be shared with. This link serves to update the shared file with any checked-in changes, regardless of which project the file was checked out from.

source code control

The management of a file's change history and the file's relation to a larger grouping of related files known as a project. Source code control is a vital part of the efficient development of software applications. Visual SourceSafe is a project-oriented type of source code control.

subproject

A project within a parent project.

Visual SourceSafe administrator

The person responsible for the Visual SourceSafe database. The administrator uses Visual SourceSafe Administrator to control the location of the database, the user list, and access rights of each user, and performs setup and backup duties on the database. The administrator's user name is always Admin.

Visual SourceSafe database

The central database where all master copies, history, project structures, and user information is stored. A project is always contained within one database; multiple projects can be stored in one database, and multiple Visual SourceSafe databases can exist to store multiple projects.

Visual SourceSafe Explorer

The main user interface of Visual SourceSafe, by default comprising two panes – the left project pane and the right file pane, as well as the toolbar, status bar, and menus, etc. Visual SourceSafe Explorer is displayed when you click the Visual SourceSafe icon.

status bar

A bar at the bottom of the Visual SourceSafe Explorer window used for displaying project and file status information and current command information.

username

A unique identifying string for a given user. Used for logging on.

version control

The control of users' changes to a file, as well as the maintenance of version history on the file.

version number

A number that indicates the number of revisions a file has undergone since it was added to Visual SourceSafe. This number is displayed in the History dialog box to help identify specific versions of a file or project. Version numbers are always whole numbers.

version tracking

The record keeping process of tracking a file's history from the initial version to the current version. Changes to a file are tracked as part of this process.

wildcard characters

The asterisk (*) and question mark (?) are wildcard characters. You can use these characters to match patterns. You can also use wildcard characters and matching characters to further refine a search.

Symbol	Example	Usage
*	wh* finds what, white, and why; *at finds cat, bat, and what	Like the MS-DOS asterisk (*) wildcard character, this asterisk matches any number of characters.
?	b?ll finds ball, bell, and bill	Like the MS-DOS (?) wildcard character, this symbol matches any single character.

A backslash preceding an asterisk or question mark indicates a literal asterisk or question mark: \`*` or \`?` (This is necessary if you want to search for actual asterisks, question marks, or backslashes.) A double backslash (\\) indicates a backslash. Any other use of a backslash is ignored.

working folder

A specified folder on a user's local computer used to store files when they are checked out of the Visual SourceSafe database. A user makes changes to files in the working folder, and then checks the modified files back into the Visual SourceSafe database for version tracking.

web site project

A project marked as a web site project in the Visual SourceSafe Administrator. Such a designation allows special web site commands, such as Deploy, to be used in this project.

visual merge

A merge operation where conflicts are resolved visually, in an easy-to-use graphical interface.

Append_EOL Initialization Variable

See Also

Indicates whether all text files must end with an end-of-line character. The default is No.

Syntax

Append_EOL = {Yes|No}

Example

Tells Visual SourceSafe not to append an end-of-line character to text files:

```
Append_EOL = No
```

Remarks

Some compilers require that all the files they compile end in an end-of-line character. If you are working in such an environment, you may want to set this variable to Yes; it causes Visual SourceSafe to append an end-of-line character whenever it retrieves a text file that does not already end with one.

You can also set this variable in your SS.INI file by using the Append End-of-Line to All Text Files check box on the Local Files tab in the SourceSafe Options dialog box.

See Also

Local Files Options Tab

Backup_Ext Initialization Variable

See Also

Specifies the extension used to create a backup file when **merging**. The default is .ORG, and this variable can be set only by editing SS.INI directly.

Syntax

Backup_Ext = *file extension used for merge backups*

Example

Sets .ORG as the extension to be used for backup files:

```
Backup_Ext = .ORG
```

Remarks

When Visual SourceSafe merges a branch of a file and finds a conflict, it replaces the original file with a merged copy that flags all conflicts, so that you can resolve them. If you specify a backup extension, the original (pre-merged) file is saved with the extension specified in this variable.

For example, when checking in MYFILE.CPP, Visual SourceSafe creates MYFILE.ORG. It contains the contents of MYFILE.CPP before Visual SourceSafe begins the check in, merge, or get operation.

See Also

Merging Files and Projects

Binary_Masks Initialization Variable

See Also

Sets binary file storage for certain file types.

Syntax

Binary_Masks = *mask* [, *additional masks*]

Example

Tells Visual SourceSafe that any files with the .OBJ and .XLS extensions are binary files:

```
Binary_Masks = *.OBJ, *.XLS
```

Remarks

This variable overrides Visual SourceSafe's automatic detection of text and binary files, so that you can indicate that certain file types are always binary. You can also set this variable in your SS.INI file by using the Binary File types option on the File Types tab in the SourceSafe Options dialog box.

Normally, Visual SourceSafe checks the contents of a file during an add or check in to see if it contains text or binary data; but this check can be time consuming for large files. Setting the Binary_Masks variable speeds up the Add Files and Check In commands for files that you know to be of one type or the other.

See Also

[File Types Options Tab](#)

[Setting the File Type: Text and Binary Files](#)

Checkout_Comment Initialization Variable

See Also

Sets the -C command-line option for the Check Out command. Applies only to the Visual SourceSafe command-line program (SS.EXE).

Syntax

Checkout_Comment = *valid -C setting*

Example

Causes check out to not prompt for comments by default:

```
Checkout_Comment = -
```

Remarks

Visual SourceSafe gives you the option of providing a comment for each file you check out. This is the default behavior. The check-out comment tells other users why you checked out the file, and is then erased when you check in the file with a permanent comment describing the change you made.

The Checkout_Comment variable controls Visual SourceSafe's prompt for a check-out comment. The following settings are all valid.

Does not prompt for a comment at check out:

```
Checkout_Comment = -
```

Prompts for one comment for all checked out files:

```
Checkout_Comment =
```

Reads the specified text file as a comment:

```
Checkout_Comment = @filename
```

Uses the specified comment:

```
Checkout_Comment = comment
```

You can also set this variable in your SS.INI file under Check Out Comments on the Command Line Options tab in the SourceSafe Options dialog box.

See Also

[Check Out Command](#)

[Command Line Options Tab](#)

Comment_Editor Initialization Variable

See Also

Determines which editor is used to enter comments. Applies only to the Visual SourceSafe command line program (SS.EXE)

Syntax

Comment_Editor = *editor executable file*

Example

Sets B.EXE as the editor for entering comments:

```
Comment_Editor = C:\EDITORS\B.EXE
```

Remarks

Visual SourceSafe's normal method for receiving comments is a command-line prompt. This variable overrides that, telling Visual SourceSafe to bring up the specified editor whenever it has to prompt for a comment. This enables you to enter comments in a familiar editor.

If the Editor initialization variable is set, you can set the Comment_Editor to override it on comments; setting it to blank restores prompt-driven comment entry while allowing an editor for viewing. You can also set this variable in your SS.INI file under the Comment Prompt options on the Command Line Options tab in the SourceSafe Options dialog box.

See Also

Command Line Options Tab

Editor Initialization Variable

Comment_Editor_Prompt Initialization Variable

See Also

Controls the prompt used when entering a comment with an editor. Applies only to the Visual SourceSafe command line program (SS.EXE)

Syntax

Comment_Editor_Prompt = {Yes|No}

Example

Turns the editor prompt on:

```
Comment_Editor_Prompt = Yes
```

Remarks

When you are entering a file comment in the Visual SourceSafe command line, the Editor and Comment_Editor initialization variables invoke an editor or word processor in which you can enter your comment. When the editor is displayed, it provides a prompt such as the following:

```
Enter a comment for TEST.C; then save and exit.
```

This prompt is not included in your comment; you do not have to delete it (although you can, with no harm, delete it). You can set this variable to No (the default is Yes), so the prompt does not appear in your editor.

You can also set this variable in your SS.INI file under the Comment Prompt options on the Command Line Options tab in the SourceSafe Options dialog box.

See Also

Command Line Options Tab

Comment_Template Initialization Variable

Sets a template file to be used for comments in **Visual SourceSafe Explorer** or the comment editor.

Syntax

Comment_Template = *template file*

Example

Sets TEMPLATE.TXT as the template to be used when entering comments:

```
Comment_Template = R:\SS\TEMPLATE.TXT
```

Remarks

This variable points to a template—a text file that you create containing a comment template. When the user is prompted for a comment, either in Visual SourceSafe Explorer or in a Comment_Editor used by the command line, the template is placed into the comment field.

Compare Initialization Variable

See Also

Indicates how Visual SourceSafe does its file comparisons, for the purpose of determining whether to get a file.

Syntax

Compare = {Contents|Checksum|Time}

Example

Compares the full contents of the file to determine if a file has changed:

```
Compare = Checksum
```

Remarks

When you get or **check out** a file, Visual SourceSafe does not re-copy the file into your **working folder** if you already have an up-to-date copy there. This variable sets how Visual SourceSafe determines if your **local copy** is up-to-date. The following options are in order of increasing speed:

- **Contents**—Compares the full contents of your local copy to the **master copy**. This is the slowest method, but the only one that works in all cases.
- **Checksum**—Compares by a checksum stored by Visual SourceSafe. If two files have the same checksum values, but are different, this method fails (this happens rarely).
- **Time**—Compares the date/time flag. This is the fastest setting, but its accuracy is influenced by the time and date settings of other computers using the file. (This setting is usually used only when your network supports synchronizing all computers on the network.)

These options, though increasingly fast, increase the risk that Visual SourceSafe may incorrectly retrieve a file. Accordingly, it is recommended that you use the Contents option.

When using Compare=Time, the exact method of date comparison depends on the setting of the SetTime initialization variable. If SetTime is set to Check In, Visual SourceSafe replaces the local copy if the time of the last check in (or add) is more recent than the modification date of your local copy. If SetTime is set to Current (the default) or Modification, Visual SourceSafe replaces the local copy if the Visual SourceSafe file's modification time is different from the local copy's modification time.

You can also set this variable in your SS.INI file by using the Compare Files By option on the Local Files tab in the SourceSafe Options dialog box.

See Also

[Check Out Command](#)

[Get Latest Version Command](#)

[Local Files OptionsTab](#)

[SetTime Initialization Variable](#)

Delete_Local Initialization Variable

See Also

Indicates during Add, Check In, and Undo Check Out commands whether or not you are maintaining a **local copy** of a complete **project**.

Syntax

Delete_Local = {Yes|No}

Example

Tells Visual SourceSafe not to delete local copies when you are done with them:

```
Delete_Local = No
```

Remarks

By default, when you are working on a file, Visual SourceSafe keeps a copy of the entire project in your **working folder**. When you set Delete_Local to Yes, Visual SourceSafe only keeps copies of the files that you are currently working on. Therefore, when you add, check in, or undo the check out of a file, instead of flagging it read-only, Visual SourceSafe deletes your local copy of the file.

When this variable is set, you cannot compile out of your working folder, because crucial files may not be in that folder. For this reason, when you set Delete_Local to Yes, you also frequently set a **shadow folder** for that project.

Even if Delete_Local is set, you can still use the Get Latest Version command to retrieve a file into your working folder. In that case, the file remains in your working folder until you delete it, or check out and then check in.

You can also set this variable in your SS.INI file by using the Delete Local Files After Add or Check In option on the Local Files tab in the SourceSafe Options dialog box.

See Also

Add Files Command

Check In Command

Check Out Command

Local Files Options Tab

Using Shadow Directories

Dir Initialization Variable

See Also

Associates a **working folder** with a specific **project**. This variable is automatically set by the Set Working Folder command. It is recommended that you do not set this variable manually.

Syntax

Dir = *folder path*

Example

Associates the folder C:\WORD with the project \$/WORD:

```
[$/WORD]  
Dir (PC) = C:\WORD
```

Remarks

When you are working on a project, you generally have a local folder, on your hard disk or on the network, where you keep your source code and work on the project.

The Dir command displays where your working folder for a particular project is stored. When you change to a new project, Visual SourceSafe automatically sets your working folder to this folder.

The use of the Dir variable can be further strengthened by setting either or both the Force_Dir and Force_Prj variables to Yes.

See Also

Force_Dir Initialization Variable

Force_Prj Initialization Variable

Set Working Folder Command

Editor Initialization Variable

See Also

Determines which editor is used to enter comments and view files.

Syntax

Editor = *editor executable file*

Example

Sets the comment and view editor to ME.EXE:

```
Editor = C:\ME\ME.EXE
```

Remarks

When you use the View command, Visual SourceSafe looks for the application it should run to display the file. The last place it looks is in the initialization files, for the Editor variable. Because this variable is not extension-specific, it is not generally the best place to specify an editor. However, it can be useful to set a default editor for all files whose extensions are not found.

The Visual SourceSafe command line also uses the variable specified here for comments, unless you override it with the Comment_Editor initialization variable.

You can also set this variable in your SS.INI file by using the File Viewer option on the General tab in the SourceSafe Options dialog box.

See Also

Comment_Editor Initialization Variable

General Options Tab

EOL Initialization Variable

Sets the end-of-line character used when Visual SourceSafe gets text files.

Syntax

EOL = {n|rn}

Example

Uses a carriage-return/line-feed pair at the end of each text line:

```
EOL = rn
```

Remarks

Different operating systems use different end-of-line characters in text files. UNIX uses a line feed; MS-DOS uses a carriage-return/line-feed pair.

No matter what end-of-line character Visual SourceSafe finds in a file, it reads it correctly. This initialization variable affects what character Visual SourceSafe outputs when it gets a text file. By default, Visual SourceSafe uses the standard for its operating system. If you set this variable, Visual SourceSafe uses whatever character you specify.

File_Types Initialization Variable

See Also

Fills the List Files of Type combo box in the Add Files dialog box in Windows.

Syntax

File_Types = *file types*

Example

Places two items in the combo box, C Files and Batch files:

```
File_Types = C Files (*.C; *.H), Batch files (*.BAT)
```

Remarks

In the Visual SourceSafe Add Files dialog box, the List Files of Type box is used to filter the **file list** to show only files of certain types. Showing all files is always an option. If you set the Relevant_Masks initialization variable, that setting is an option as well. This initialization variable is used to set other options that appear in the combo box. In the above example, for instance, C Files and Batch Files would both appear as options; if you choose C Files, the file list shows all files with the extension C or H.

You can also set this variable in your SS.INI file by using the File Groups option on the File Types tab in the SourceSafe Options dialog box.

See Also

Add Files Command

File Types Options Tab

Relevant_Masks Initialization Variable

Force_Dir Initialization Variable

See Also

Strengthens the use of the Dir initialization variable.

Syntax

Force_Dir = {Yes|No}

Example

Indicates normal use of the Dir variable:

```
Force_Dir = No
```

Remarks

The Dir initialization variable is used to specify your **working folder** for a **project**. The Dir variable controls where **Visual SourceSafe Explorer** retrieves files. If Force_Dir is set to No, that is the only effect of the Dir variable. However, if you set Force_Dir to Yes (which it is by default), the Dir variable is used in two other circumstances.

1. On the Visual SourceSafe command-line, When you get or check out a file, the file goes into the current folder. But if Force_Dir is set to Yes, the file goes into your working folder for the project, even if that is not the current folder.
2. When you perform a recursive get on a project, subdirectories are created for each subproject. For instance, for the subproject HELLO, a folder called HELLO is created, so the entire project list is copied into a folder tree. But if Force_Dir is set to Yes, subprojects go into their own respective working directories instead of into subdirectories under the main folder.

You can also set this variable in your SS.INI file by using the Assume a Folder, Based on the Current Project option on the Command Line Options tab in the SourceSafe Options dialog box.

See Also

Command Line Options Tab

Get Latest Version Command

Force_Prj Initialization Variable

Keep_Checkedout Initialization Variable

See Also

Indicates whether you want to keep files **checked out** when you check in or add them.

Syntax

Keep_Checkedout = {Yes|No}

Example

Instructs Visual SourceSafe not to check all files back out to you immediately after updating or adding them:

```
Keep_Checkedout = No
```

Remarks

After you add or check in a file, you cannot immediately modify it, because it is no longer checked out. In some situations—particularly on one-person projects

—you always want to have all the files checked out. In that case, set Keep_Checkedout to Yes, which instructs Visual SourceSafe to immediately check the file back out to you whenever you check in or add a file.

In **Visual SourceSafe Explorer**, Keep_Checkedout sets the default state of the Keep Checked Out check box in the Check In dialog box, which you can toggle for specific cases. (On the command line, you can override this variable with the -K (or -K-) command-line option.)

You can also set this variable in your SS.INI file by using the Always Keep Files Checked Out option on the General tab in the SourceSafe Options dialog box.

See Also

General Options Tab

Getting, Checking In, Checking Out, and Viewing Files

Lowercase_Files Initialization Variable

See Also

Controls how filenames are displayed in Visual SourceSafe Explorer.

Syntax

Lowercase_Files = {Yes|No}

Example

Instructs the Visual SourceSafe Explorer to show all files in uppercase:

```
Lowercase_Files = No
```

Remarks

In its normal mode, Visual SourceSafe displays files in the case in which they were added. If this variable is set to Yes, however, Visual SourceSafe displays files and directories in all lowercase.

Note that this variable has no effect on the display of project names, which retain the case with which you created them.

You can also set this variable in your SS.INI file by using the Show Filenames in Lowercase Letters option on the View tab in the SourceSafe Options dialog box.

See Also

[View Options Tab](#)

Menu_Help Initialization Variable

This variable is no longer supported. Indicates whether the Visual SourceSafe status bar displays automatic help on selected menu items and icons.

Syntax

Menu_Help = {Yes|No}

Example

Instructs Visual SourceSafe to always display a help line on the status bar when a menu item or icon is selected:

```
Menu_Help = Yes
```

Remarks

When this variable is set to Yes, as you scroll through a menu or click an icon, the status bar at the bottom of **Visual SourceSafe Explorer** shows a brief description of the currently selected command.

Output Initialization Variable

See Also

Sets the default method for displaying output.

Syntax

Output = *valid -O setting*

Example

Paginates all output by default:

```
Output =
```

Sends all output to the file OUTLOG.TXT by default:

```
Output = OUTLOG.TXT
```

Remarks

On the Visual SourceSafe command line, the Output initialization variable sets a default for the -O command-line option to any function. The first example above is equivalent to -O, and tells Visual SourceSafe to paginate its output. The second is equivalent to -oOUTLOG.TXT, an instruction to redirect all output into a file.

See Also

[-O Command-Line Option](#)

Refresh_Interval Initialization Variable

See Also

Sets the time between **file list** auto-refreshes.

Syntax

Refresh_Interval = *number of seconds*

Example

Sets the refresh interval to 300 seconds (5 minutes):

```
Refresh_Interval = 300
```

Remarks

Periodically, **Visual SourceSafe Explorer** refills its file list, so that any changes other users have made are reflected in the display. Refreshing keeps the Visual SourceSafe screen up-to-date, but it also causes a momentary delay.

You can force refreshes manually with the Refresh File List option in the View menu, but if automatic refresh is on (as it is by default), this variable can be used to change the time delay between refreshes.

You can also set this variable in your SS.INI file by using the Automatically Refresh File List option on the View tab in the SourceSafe Options dialog box.

See Also

[View Options Tab](#)

Relevant_Masks Initialization Variable

See Also

Sets the Visual SourceSafe Add command mask.

Syntax

Relevant_Masks = *masks*

Example

Restricts the Add command to files with the extensions .C, .H, or none:

```
Relevant_Masks = *.C, *.H, *
```

Restricts the Add command to all files except those with the extension .OBJ:

```
Relevant_Masks = !*.OBJ
```

Remarks

A typical Visual SourceSafe **working folder** contains a number of important code files that are in the Visual SourceSafe **project**, and a number of unimportant intermediate files that are not in the project. For instance, in a C folder, the .C and .H files contain the code; the .OBJ files, which are byproducts of the process, are not placed in the project.

Relevant_Masks indicates which files you would typically place into a project. This does not prevent you from placing other files into the project; however, it does have several potentially useful effects.

First, if you add an entire folder into Visual SourceSafe in one command (by typing `ss add *` on the Visual SourceSafe command line, or dragging a folder from File Manager into Visual SourceSafe), only files matching the relevant masks are added. Other files can then be added individually.

Second, when you run the Visual SourceSafe Show Differences command on a project level to compare the **current project** to your working folder, files that do not match the relevant masks are not displayed as differences.

Finally, the relevant masks are used as the default option in the Add Files of Type combo box in the Add Files dialog box; therefore, by default, only these files are displayed.

You can also set this variable in your SS.INI file by using the File Types tab in the SourceSafe Options dialog box.

See Also

Add Files Command

File Types Options Tab

Reuse_Comment Initialization Variable

See Also

Instructs Visual SourceSafe Explorer to fill in comments with the last comment entered.

Syntax

Reuse_Comment = {Yes|No}

Example

Turns off last-comment use:

```
Reuse_Comment = No
```

Remarks

When Visual SourceSafe Explorer prompts you for a comment, if you do not have a Comment_Template set, the comment is initially blank. However, if you set this variable to Yes, the comment field is automatically filled in with the most recent comment you entered. This makes it easy to use the same comment repeatedly; or, you can start typing to replace the previous comment with a new one.

You can also set this variable in your SS.INI file by using the Reuse Last Comment option on the General tab in the SourceSafe Options dialog box.

See Also

General Options Tab

Smart_Mode Initialization Variable

See Also

Enables and disables Visual SourceSafe's SmartMode feature.

Syntax

Smart_Mode = {Yes|No}

Example

Disables Visual SourceSafe's SmartMode feature:

```
Smart_Mode = No
```

Remarks

Smart_Mode tells Visual SourceSafe whether or not to use its SmartMode features. The SmartMode features are:

- When you create or recover a **project**, Visual SourceSafe sets the new project as the **current project**.
- When you delete or destroy a file, Visual SourceSafe deletes the corresponding local copy.
- When you rename a file, Visual SourceSafe renames the corresponding **local copy**.

You can also set this variable in your SS.INI file by using the Remove Local Files After Delete option on the Local Files tab in the SourceSafe Options dialog box.

See Also

Local Files Options Tab

-S Command-Line Option

Update_No_Change Initialization Variable

See Also

Controls what happens when you **check in** a file that you have not changed. The default is Uncheckout.

Syntax

Update_No_Change = {Update|Uncheckout|Ask}

Example

Undoes check out of files that have not changed:

```
Update_No_Change = Uncheckout
```

Remarks

When you check in a file that has not changed since you checked it out, Visual SourceSafe has three options: It checks in the file, undoes the check out of the file or asks you how to handle each file.

Because no changes have been made, the only difference between the first two cases is that check in creates an entry in the file **history** with a comment but no change record; undo check out does not.

- Update—Checks in files even if they have not changed.
- Uncheckout—Undoes the check out of unchanged files.
- Ask—Asks whether Visual SourceSafe should check in an unchanged file.

You can also set this variable in your SS.INI file by using the Check In Unchanged Files option on the General tab in the SourceSafe Options dialog box.

See Also

Check Out Command

General Options Tab

Use_ReadOnly Initialization Variable

See Also

Controls Visual SourceSafe's use of the **read-only** flag for files that are not checked out.

Syntax

Use_ReadOnly = {**Yes|No**}

Example

Flags **local copies** of non-checked-out files as read-only:

```
Use_ReadOnly = Yes
```

Remarks

By default, Visual SourceSafe assumes the files in your **working folder** that you do not have checked out—that is, files you should not be modifying

—are read-only. Visual SourceSafe sets the read-only flag when you get, add, check in, or undo check out of a file, and uses it as a signal that you have not been modifying the file (so Visual SourceSafe can overwrite it). If you set this variable to No, Visual SourceSafe does not set the read-only flag on files in your working folder.

You can also set this variable in your SS.INI file by using the Use Read-Only Flag for Files That Are Not Checked Out option on the Local Files tab in the SourceSafe Options dialog box.

See Also

Local Files Options Tab

Warnings Initialization Variables

See Also

Controls warnings given by **Visual SourceSafe Explorer**. Does not apply to the Visual SourceSafe command line (that is, no warnings are generated on the command line regardless of this variable's setting.)

Syntax

Warn_Destroy = {Yes|No}
Warn_Exit = {Yes|No}
Warn_Multiple_Checkout = {Yes|No}
Warn_Purge = {Yes|No}
Warn_Remove = {Yes|No}
Warn_Rollback = {Yes|No}
Warn_UnCheckout = {Yes|No}

Example

Displays a warning before **deleting** files or **projects**:

```
Warn_Remove = Yes
```

Remarks

Whenever you take an action that could result in the loss or accidental deletion of data, Visual SourceSafe displays a warning. While most users find these warnings useful for preventing accidental changes, some users prefer to eliminate the extra step of answering the warning.

These variables can be set (independently of each other) to turn various warnings off. By default, all warnings are considered on, except Warn_Exit. Note that the Multiple_Checkouts initialization variable warns you when you are about to **check out** a file that another user has checked out; this is only relevant when Multiple_Checkouts = Yes.

You can also set these variables in your SS.INI file by using the Warnings tab in the SourceSafe Options dialog box.

See Also

Warnings Options Tab

CP_OnSelection Initialization Variable

See Also

Indicates whether selecting a **project** in **Visual SourceSafe Explorer** automatically sets the **current project**.

Syntax

CP_OnSelection = {Yes|No}

Example

Indicates that selecting a project automatically switches to it:

```
CP_OnSelection = Yes
```

Remarks

If this variable is set to Yes, every time you select a project in Visual SourceSafe Explorer (either with the keyboard arrows or with a mouse click) it becomes your current project. If it is set to No, you must press Enter (or double-click with the mouse) to make it your current project.

You can also set this variable in your SS.INI file by using the Make Selected Project the Current Project option on the View tab in the SourceSafe Options dialog box.

See Also

[View Options Tab](#)

Diff_Format Initialization Variable

See Also

Controls the format used with the Show Differences (File) command on the Tools menu.

Syntax

Diff_Format = {SS|UNIX|Visual}

Example

Tells Visual SourceSafe to use a visual (side-by-side) difference display:

```
Diff_Format = Visual
```

Remarks

When you use the Show Differences (File) command to show the differences between two files, Visual SourceSafe can display the file differences in three ways.

- SS –Lists lines added, deleted, or changed.
- UNIX –Displays the standard output form of the UNIX Diff utility.
- Visual –Shows two files side by side, with different lines color-coded.

If you do not specify a preferred format, **Visual SourceSafe Explorer** defaults to Visual, and the command line defaults to the SS display. However, if you set this variable, both Visual SourceSafe Explorer and the command line switch their default to your setting. In any case, you can always override the default with command-line options and in the SourceSafe Options dialog box on the Tools menu.

See Also

Show Differences Command (File)

Diff_Ignore Initialization Variable

See Also

Controls the factors that are considered or ignored when using the Show Differences (File) command on the Tools menu.

Syntax

Diff_Ignore = *list of options*

Example

Ignores end-of-line differences, and displays all others:

```
Diff_Ignore = c-es-w-
```

Remarks

The Visual SourceSafe Show Differences (File) command is used to display all the relevant changes to a file. This variable defines what is a relevant change. You can control four issues in this variable.

- C
- case sensitivity.
- E
- end-of-line characters.
- S
- small differences.
- W
- white-space changes.

The format of the variable is a list of these four letters, with a dash after any variable that should not be ignored. In the preceding example (also the default), E is the only letter that is not followed by a dash; consequently, end-of-line characters are the only differences ignored.

See Also

Show Differences (File) Command

Expand_Keywords_Locally Initialization Variable

See Also

Controls whether Visual SourceSafe automatically replaces **keyword-expanded** files in your **working folder**.

Syntax

Expand_Keywords_Locally = {Yes|No}

Example

Performs the Get Latest Version command automatically after any Check In or Add command that causes keyword expansion:

```
Expand_Keywords_Locally = Yes
```

Remarks

When you check in a file and have keyword expansion enabled, Visual SourceSafe may replace certain keywords in your file with **version** information. The modified file is copied back into Visual SourceSafe. If this variable is set to Yes, the modified file is then copied directly back into your folder. If you set this variable to No, the file is not copied back; this improves performance, but the copy in your working folder may not be completely up-to-date.

You can also set this variable in your SS.INI file by using the Copy Keyword-Expanded Files Into Working Folder option on the Local Files tab in the SourceSafe Options dialog box.

See Also

Add Files Command

Check In Command

Keyword Expansion

Local Files Options Tab

FileType Initialization Variable

See Also

Indicates which type of file is used.

Syntax

FileType = {Binary|Text|Auto-Detect}

Example

Automatically detects the file type:

```
Filetype = Auto-Detect
```

Remarks

This variable overrides Visual SourceSafe's automatic detection of text and binary files, and indicates that all files are text or binary. Generally, this variable is defined globally for all projects. If you use it, you should apply it only to specific subprojects when you know all files in that subproject are of a specific type. For example, if you have a project named `$/Executables` with two subprojects `$/Executables/DLLs` and `$/Executables/EXEs`, you may want to put the `FileType=Binary` variable setting under the heading `[/Executables]` to speed up Add Files and Check In operations.

See Also

[Setting the File Type: Text and Binary Files](#)

Force_Prj Initialization Variable

See Also

Automatically sets the **current project**, based on the current folder. Applies only to the Visual SourceSafe command line program (SS.EXE).

Syntax

Force_Prj = {Yes|No|Ask}

Example

Instructs Visual SourceSafe not to automatically change the current project:

```
Force_Prj = No
```

Remarks

- Yes—First looks at your current folder whenever you execute any Visual SourceSafe command. If this is the **working folder** for a Visual SourceSafe project, Visual SourceSafe automatically changes to that project before executing the command: You can change directories to move around your projects, without executing the SS CP command.
- No—Does not change to your current project.
- Ask—Prompts before changing your project.

If you set Force_Prj = Yes, it is highly recommended that you do not give two projects the same working folder, because Visual SourceSafe won't know which project to switch into.

You can also set this variable in your SS.INI file by using the Assume Project Based on Working Folder option on the Command Line Options tab in the SourceSafe Options dialog box.

See Also

Command Line Options Tab

Long_Filenames Initialization Variable

See Also

Controls Visual SourceSafe's use of filenames longer than MS-DOS standard.

Syntax

Long_Filenames = {Yes|No}

Example

Instructs Visual SourceSafe to work with pathnames up to 255 characters long:

```
Long_Filenames = Yes
```

Remarks

This variable toggles Visual SourceSafe between long filename mode (in which complete paths, including filenames, must not exceed 255 characters) and short filename mode (in which MS-DOS format names are used for all files). The default is Yes. It is particularly relevant under Windows NT, which supports both long-filename file systems and the MS-DOS FAT system.

You can also set this variable in your SS.INI file by using the Always Display MS-DOS Format Filenames option on the View tab in the SourceSafe Options dialog box.

See Also

[View Options Tab](#)

Mark_Merges Initialization Variable

See Also

This variable is no longer supported. Indicates whether Visual SourceSafe should make a note of all **merges**, including those with no conflicts. The default is No.

Syntax

Mark_Merges = {Yes|No}

Example

Instructs Visual SourceSafe to note conflict areas only:

```
Mark_Merges = No
```

Remarks

When you merge two files and a conflict occurs, Visual SourceSafe creates a temporary file with marks for all the conflicts. If you set this variable to Yes, Visual SourceSafe marks all the changes it has detected and merged, including those that did not result in a conflict of any kind. It also makes each mark more explicit, by showing the original code as well as the two changed sets.

If you set this variable to No, Visual SourceSafe marks only the changes that conflict. It uses **conflict markers** to indicate the changes so you can resolve the conflicts.

See Also

General Options Tab

Merging Files and Projects

Merging Options

Resolving Merge Conflicts

Merge_Ext Initialization Variable

See Also

Sets the extension used by Visual SourceSafe to create a temporary **merge** file. The default is .MRG.

Syntax

Merge_Ext = *extension of temporary merge file*

Example

Gives all temporary merge files the extension .MRG:

```
Merge_Ext = .MRG
```

Remarks

When Visual SourceSafe performs an automatic merge, it creates a temporary file with this extension in which to perform the merge. When the merge is complete, Visual SourceSafe either renames this file or deletes it; so, in general, you should never see this file. However, in case of failure during a merge, you can look for the .MRG copy of this file.

See Also

Merging Files and Projects

Merging Options

Resolving Merge Conflicts

Project Initialization Variable

Stores a user's **current project**. Set automatically when you exit Visual SourceSafe.

Caution Do not set this variable manually.

Syntax

Project = *current Visual SourceSafe project*

Example

Indicates that the **root project** is your **current project**:

```
Project = $/
```

Remarks

Your current project is stored in this variable in your SS.INI file.

Recursive Initialization Variable

See Also

Indicates whether Visual SourceSafe operations act recursively by default.

Syntax

Recursive = {Yes|No}

Example

Indicates that commands operate only on the current or specified **project**, without going into all **subprojects**:

```
Recursive = No
```

Remarks

Many Visual SourceSafe commands (such as Get Latest Version or Show History) can be performed on a specific project (and all the files directly in it), or recursively down an entire project tree. You can always make a particular command recursive by choosing the appropriate check box in **Visual SourceSafe Explorer**, or specifying the -R command-line option. If you set this variable to Yes, however, all commands are recursive by default.

You can also set this variable in your SS.INI file by using the Act on Projects Recursively option on the General tab in the SourceSafe Options dialog box.

See Also

[-R Command-Line Option](#)

[General Options Tab](#)

Refresh_Display Initialization Variable

See Also

Controls whether Visual SourceSafe automatically refreshes its **file list** in the **file pane** of **Visual SourceSafe Explorer**.

Syntax

Refresh_Display = {Yes|No}

Example

Instructs Visual SourceSafe Explorer to periodically rebuild its file list:

```
Refresh_Display = Yes
```

Remarks

Periodically, Visual SourceSafe Explorer refills its file list, so that any changes users have made are reflected in the display. Refreshing keeps the Visual SourceSafe screen up-to-date, but it also causes a momentary delay. This variable controls whether Visual SourceSafe automatically refreshes its file list; if it is set to Yes, Refresh_Interval controls how often. If this variable is set to No, you may want to manually refresh the file list periodically, using the Refresh File List command on the View menu.

You can also set this variable in your SS.INI file by using the Automatically Refresh File List option on the View tab in the SourceSafe Options dialog box.

See Also

Refresh Interval Initialization Variable

View Options Tab

Replace_Checkout Initialization Variable

See Also

This variable is no longer supported. Determines whether Visual SourceSafe overwrites a **checked out** writable file. The default is No.

Syntax

Replace_Checkout = {Error|Merge|Replace|Ask|Skip}

Example

Instructs Visual SourceSafe not to overwrite a checked out writable file:

```
Replace_Checkout = No
```

Remarks

When you get or check out a file, Visual SourceSafe copies the file into your **working folder**. If there is already a checked out writable (non-read-only) copy of the file in your working folder, overwriting the file may destroy your work.

Visual SourceSafe provides four ways to handle this situation.

- Error—Does not replace the file and reports an error.
- Replace—Replaces the file.
- Merge—Merges the changes between the checked out-file and the one you are now getting or checking out.
- Ask—Gives you an option each time Visual SourceSafe encounters this situation.
- Skip—Does not replace the file, but does not generate an error.

You can also set this variable in your SS.INI file by using the Replace Checked Out Files option on the Local Files tab in the SourceSafe Options dialog box.

Use the Replace_Writable initialization variable to control how Visual SourceSafe treats writable files in your working folder when you get or check out files that you don't have checked out.

See Also

[Check Out Command](#)

[Get Latest Version Command](#)

[Local Files Options Tab](#)

[Replace_Writable Initialization Variable](#)

Replace_Writable Initialization Variable

See Also

Controls what happens when you attempt to get or check out a file on top of a writable file. The default is Ask.

Syntax

Replace_Writable = {Merge|Replace|Ask|Skip}

Example

Instructs Visual SourceSafe not to overwrite writable files:

```
Replace_Writable = No
```

Remarks

When you get or check out a file, Visual SourceSafe copies the file into your **working folder**. If there is already a writable (non-read-only) copy of the file in your working folder—for instance, if you have the file checked out

—overwriting the file may destroy your work.

Visual SourceSafe provides four ways to handle this situation.

- Merge—Merges any differences between the **master copy** of the file and the file in your working folder together in your working folder copy.. This is useful when you want to ensure that you are getting other users' changes and you have the file checked out. If you have this option set and don't have the file checked out, SourceSafe automatically treats that file as if you had the Skip option set.
- Replace—Replaces the file.
- Ask—Gives you an option each time Visual SourceSafe encounters this situation.
- Skip—Does not replace the file, and does not generate an error.

You can also set this variable in your SS.INI file by using the Replace Writable Files option on the Local Files tab in the SourceSafe Options dialog box.

See Also

[Check Out Command](#)

[Get Latest Version Command](#)

[Local Files Options Tab](#)

OutputWindow Initialization Variable

See Also

Controls the use of the bottom pane which displays the results of Visual SourceSafe operations. Note that the old name of this variable, Results_Window, is obsolete.

Syntax

OutputWindow = {Yes|No}

Example

Displays the Results pane:

```
OutputWindow = Yes
```

Remarks

When you perform a command such as Get Latest Version, Visual SourceSafe displays the results of the command in the Results pane.

If you set this variable to No, the Results pane is not displayed. Errors are reported in the standard error message window.

You can also set this variable in your SS.INI file by using the Show Results pane option on the View tab in the SourceSafe Options dialog box.

See Also

[View Options Tab](#)

SetTime Initialization Variable

See Also

Controls the date and time given to your **local copies** by a Get Latest Version command.

Syntax

SetTime = {**Current|Mod|Update**}

Example

Gives files a new date/time stamp when Visual SourceSafe gets them:

```
SetTime = Current
```

Remarks

- **Current**—Sets the date and time of the file to the current date and time by default, whenever Visual SourceSafe gets a file (with Get Latest Version, Check Out, Undo Check Out, or many other commands). This signals the compiler that the file has just changed, and has to be recompiled.
- **Modification**—Sets the **local copy** to the date and time the file was last modified.
- **Check In**—Sets the local copy to the date and time the file was last checked in.

You can also set this variable in your SS.INI file by using the Set Date/Time on Local Files option on the Local Files tab in the SourceSafe Options dialog box.

See Also

[Get Latest Version Command](#)

[Local Files Options Tab](#)

Skip Initialization Variables

See Also

Controls the use of options dialog boxes on commands in Visual SourceSafe Explorer.

Syntax

Skip_*variable name* = {Yes|No}

Examples

```
Skip_FCheckout = No  
Skip_PCheckout = Yes  
Skip_FGet = No  
Skip_PGet = Yes  
Skip_PDeploy = Yes  
Skip_FUncheckout = No  
Skip_PUncheckout = Yes  
Skip_FDdiff = No  
Skip_PDdiff = Yes  
Skip_FHist = No  
Skip_PHist = No
```

Remarks

When you execute a command in Visual SourceSafe Explorer, you can choose whether to display a dialog box that gives you a variety of options to set. The advantage of the dialog box is that it enables you to treat each execution of a command differently. The disadvantage is that it adds at least one extra step to every common operation.

By default, Visual SourceSafe does not display each dialog box when you perform a command on a file; it displays the dialog box when you perform a command on a **project**. These variables can be used to alter those defaults; for instance, if you set Skip_PGet = Yes, no dialog box is displayed when you perform a project-level Get Latest Version command. If you set Skip_FCheckout = No, Visual SourceSafe displays the Check Out command dialog box when you use the file-level Check Out command.

You can also set these variables in your SS.INI file by using the options on the Command Dialogs tab in the SourceSafe Options dialog box.

See Also

Command Dialogs Options Tab

Temp_Path Initialization Variable

See Also

Sets the folder where Visual SourceSafe stores temporary files.

Syntax

Temp_Path = *path to Visual SourceSafe temporary folder*

Example

Indicates that the temporary path is a folder called Temp immediately under the folder in which SRCSAFE.INI is located:

```
Temp_Path = Temp
```

Remarks

The Temp_Path variable is most often defined by the Visual SourceSafe administrator in the SRCSAFE.INI file. However, you can use it to store temporary files locally on your computer by defining Temp_Path in your SS.INI file. Storing temporary files locally can reduce storage requirements placed on your Visual SourceSafe database server and may improve performance.

You can also set this variable in your SS.INI file by using the Folder for Temporary Files option on the General tab in the SourceSafe Options dialog box.

See Also

General Options Tab

Toolbar Initialization Variable

See Also

Shows or hides the toolbar in Visual SourceSafe.

Syntax

Toolbar = {Yes|No}

Example

Displays the toolbar:

```
Toolbar = Yes
```

Remarks

At the top of **Visual SourceSafe Explorer** is the toolbar, or icon bar, which gives quick mouse access to some of Visual SourceSafe's most common features.

You can also set this variable in your SS.INI file by using the Show Toolbar option on the View tab in the SourceSafe Options dialog box.

See Also

[View Options Tab](#)

Toolbar_Size Initialization Variable

See Also

Sets the size of the toolbar command buttons.

Syntax

Toolbar_Size = {16|24}

Example

Selects the small button size as the default:

```
Toolbar_Size = 16
```

Remarks

The numbers 16 and 24 refer to the number of pixels used for the height and width of the small and large buttons, respectively. You can also change the setting by using the Large Buttons check box in the Customize Toolbar dialog box on the Tools menu.

See Also

Customize Toolbar Command

ComputerName Initialization Variable

This variable is no longer supported as of version 5.0. You can delete it from your .INI file or ignore it since NT and Win95 use your actual computer name as specified on the Network dialog box in the Control Panel.

Sets a user-specified computer name for optional display of **checked-out files** in the **file pane** of **Visual SourceSafe Explorer**.

Syntax

ComputerName = *computer name string*

Example

Sets your computer name to AccountingMachine1:

```
ComputerName = AccountingMachine1
```

Remarks

Spaces are allowed in the computer name; quotation marks are not necessary to delimit the string.

Visual SourceSafe displays the computer name in the Check Out Folder column of Visual SourceSafe Explorer. However, Visual SourceSafe cannot obtain the computer name information when running on MS-DOS, Windows 3.x, or the Macintosh. On these platforms, SourceSafe retrieves the computer name from the ComputerName variable and uses it in the Visual SourceSafe Explorer display. If no computer name is specified with this variable, Visual SourceSafe cannot display the computer name.

You can assign the computer name variable value to an environment variable if desired. For example, (on MS-DOS)

```
ComputerName = %COMPUTERNAME%
```

The % character instructs Visual SourceSafe to read the environment variable to determine the name of the computer.

Note The ComputerName variable is ignored by Windows NT and Windows 95. These platforms always retrieve the computer name from the operating system itself.

Extension_Association Initialization Variable

Associates a file extension with the editor used when viewing a file.

Syntax

extension association = name of executable editor file

Example

Sets the editor to WINHELP.EXE when viewing files with the .hlp extension:

```
.hlp = winhelp.exe
```

Remarks

This variable is used to override (for Visual SourceSafe only) the extension associations you have in Windows File Manager or Windows 95 Explorer. You only have to use this variable for Visual SourceSafe running on MS-DOS.

Status_Bar Initialization Variable

See Also

Shows and hides the status bar in Visual SourceSafe Explorer.

Syntax

Status_Bar = {Yes|No}

Example

Displays the status bar:

```
Status_Bar = Yes
```

Remarks

The status bar shows information on the files displayed in the **file pane** of Visual SourceSafe Explorer, such as sort order, check out name, number of files in the project, and so forth. It also displays status and Help messages.

You can also set this variable in your SS.INI file by using the Show Status Bar option on the View tab in the SourceSafe Options dialog box.

See Also

Status Bar

View Options Tab

Store_Deltas Initialization Variable

Sets whether Visual SourceSafe stores change information between versions of a file. The default is Yes.

Syntax

Store_Deltas = {Yes|No}

Example

Stores all changes between file versions:

```
Store_Deltas = Yes
```

Remarks

You should use this variable rarely. Setting this variable to No saves disk space on the Visual SourceSafe server, but doesn't allow you to re-create past versions of files.

When you add files to Visual SourceSafe, the Add Files command lets you decide if Visual SourceSafe retains its old version of a file or only the latest version with the Store Only Latest Version check box in the Advanced options in the Add Files dialog box. Setting the Store_Deltas variable controls the default setting of this option. The default is Yes, but you can manually override this setting for individual files in the Add Files dialog box.

ToolTips Initialization Variable

See Also

Sets whether ToolTips are displayed as you drag the mouse pointer over a toolbar button.

Syntax

ToolTips = {Yes|No}

Example

Displays ToolTips:

```
ToolTips = Yes
```

Remarks

You can also set this variable using the Show ToolTips check box with the Customize Toolbar command on the Tools menu.

See Also

Customize Toolbar Command

Visual_Diff_InsertClr Initialization Variable

See Also

This variable is no longer supported. Sets the color used to indicate inserted lines in the Differences for File dialog box, available with the Show Differences (File) command on the Tools menu. These colors are also used in the Visual Merge dialog box.

Syntax

Visual_Diff_InsertClr = {R,G,B|Color}

Example

Sets the inserted lines color to green:

```
Visual_Diff_InsertClr = <0,255,0>
```

Sets the inserted lines color to green:

```
Visual_Diff_InsertClr = Green
```

Remarks

If you set this variable, make sure you set Visual_Diff_DeleteClr and Visual_Diff_ChangeClr to something else. You can specify any color with the RGB syntax (3 numbers between 1–255 separated by commas), or use the following color names: Black, Blue, Red, Green, Cyan, Magenta, Gray, White, Yellow, Light Blue, Light Yellow, Light Green, Light Red, Light Magenta, Light Cyan.

See Also

[The Visual Merge](#)

[Show Differences \(File\) Command](#)

[Visual_Diff_ChangeClr Initialization Variable](#)

[Visual_Diff_DeleteClr Initialization Variable](#)

Visual_Diff_DeleteClr Initialization Variable

See Also

This variable is no longer supported. Sets the color used to indicate deleted lines in the Differences for File dialog box, available with the Show Differences (File) command on the Tools menu. These colors are also used in the Visual Merge dialog box.

Syntax

Visual_Diff_DeleteClr = {R,G,B|Color}

Example

Sets the inserted lines color to red:

```
Visual_Diff_DeleteClr = <255,0,0>
```

Sets the inserted lines color to red:

```
Visual_Diff_DeleteClr = Red
```

Remarks

If you set this variable, make sure you set Visual_Diff_InsertClr and Visual_Diff_ChangeClr to something else. You can specify any color with the RGB syntax (3 numbers between 1–255 separated by commas), or use the following color names: Black, Blue, Red, Green, Cyan, Magenta, Gray, White, Yellow, Light Blue, Light Yellow, Light Green, Light Red, Light Magenta, Light Cyan.

See Also

[The Visual Merge](#)

[Show Differences \(File\) Command](#)

[Visual_Diff_ChangeClr Initialization Variable](#)

[Visual_Diff_InsertClr Initialization Variable](#)

Visual_Diff_ChangeClr Initialization Variable

See Also

This variable is no longer supported. Sets the color used to indicate changed lines in the Differences for File dialog box, available with the Show Differences (File) command on the Tools menu. These colors are also used in the Visual Merge dialog box.

Syntax

Visual_Diff_ChangeClr = {R,G,B|Color}

Example

Sets the changed lines color to blue:

```
Visual_Diff_ChangeClr = <0,0,255>
```

Sets the changed lines color to blue:

```
Visual_Diff_ChangeClr = Blue
```

Remarks

If you set this variable, make sure you set Visual_Diff_InsertClr and Visual_Diff_DeleteClr to something else. You can specify any color with the RGB syntax (3 numbers between 1–255 separated by commas), or use the following color names: Black, Blue, Red, Green, Cyan, Magenta, Gray, White, Yellow, Light Blue, Light Yellow, Light Green, Light Red, Light Magenta, Light Cyan.

See Also

[The Visual Merge](#)

[Show Differences \(File\) Command](#)

[Visual_Diff_DeleteClr](#)

[Visual_Diff_InsertClr](#)

History_Include_Files Initialization Variable

See Also

Sets whether files are included automatically in project history reports.

Syntax

History_Include_Files = {Yes|No}

Example

Sets SourceSafe to automatically include files in the project-level Show History command:

```
History_Include_Files = Yes
```

Remarks

You can also set this variable using the Include Files check box in the Project History Options dialog box.

See Also

Show History Command

Save_Settings Initialization Variable

See Also

Sets whether a file-level Show Difference command's optional settings are saved between Visual SourceSafe sessions.

Syntax

Save_Settings = {Yes|No}

Example

Instructs SourceSafe to save the file-level Show Difference settings between sessions:

```
Save_Settings = Yes
```

Remarks

Affects only the file-level settings for the Show Differences command (Tools menu).

See Also

Show Differences (File) Command

Visual_Merge Initialization Variable

See Also

Sets whether the Visual Merge feature is used when merging files.

Syntax

Visual_Merge = {Yes|No|Only if there are conflicts}

Example

Instructs SourceSafe to use the Visual Merge feature only when conflicts between changes in the two files being merged exist, otherwise use the standard merge:

```
Visual_Merge = Only when there are conflicts
```

Remarks

You can also set this variable using the Use Visual Merge check box on the General tab of the Tools Options dialog box.

See Also

[The Visual Merge](#)

[Merging Files and Projects](#)

[Merging Options](#)

[Resolving Merge Conflicts](#)

[Mark_Merges Initialization Variable](#)

[Merge_Ext Initialization Variable](#)

Visual_Merge_Sound Initialization Variable

See Also

This variable is no longer supported. Sets whether the Visual Merge sound is played when accepting changes during a visual merge operation.

Syntax

Visual_Merge_Sound = {.WAV filename}

Example

Instructs SourceSafe to use the LASER.WAV sound file when you accept changes from one of the top two panes in a visual merge operation:

```
Visual_Merge_Sound = LASER.WAV
```

Remarks

We recommend a fairly small .WAV file to maintain performance.

See Also

[The Visual Merge](#)

[Merging Files and Projects](#)

[Merging Options](#)

[Resolving Merge Conflicts](#)

Reconcile_Project Initialization Variable

See Also

Determines default behavior when reconciling differences between projects with the Show Differences command.

Syntax

Reconcile_Project = {Get|Ignore, Delete from folder|Recover deleted file|Add to project|Ignore, Add to project|Ignore, Check In|Undo Check Out|Get|Ignore, Get|Ignore}

Example

Instructs SourceSafe to use the normal default behaviors for reconciling project differences in the Reconcile All Differences dialog box:

```
Reconcile_Project = Get, Delete, Add, Check In, Get
```

Remarks

Changing this variable allows you to customize the default choices in the Reconcile All Differences dialog box. The setting for this variable are used when you specify the -F option for the Project Difference command line command.

The first three settings are to specify behavior for reconciling files that appear in only one list:

For files that are not in your working folder. This specifies what you want to happen when Visual SourceSafe finds files that are not in your working folder, but are in the project. Specify Get (the default) to get files missing from your working folder. Specify Ignore to take no action with these files.

Files that are deleted in the project. This specifies what you want to happen when Visual SourceSafe finds files that are in your working folder, but have been deleted in the project. Specify Delete from folder (the default) to delete these files from your working folder. Specify Recover deleted file to recover the deleted file and place a copy in your working folder. Specify Add to project to add these files back to your project. Specify Ignore to take no action with these files.

Files that are not in the project at all. This specifies what you want to happen when Visual SourceSafe finds files that are not in your project, but are in your working folder. Specify Add to Project (the default) to add these files to your Visual SourceSafe project. Specify Ignore to take no action with these files.

The last two settings specify behavior for reconciling files that appear in both lists:

Files that you have checked out. This specifies what you want to happen when Visual SourceSafe finds files that are checked out to you, and are different in your working folder. Specify Check In (the default) to check these files into your Visual SourceSafe project. Specify Undo Check Out to undo the check out on these files. Specify Get Latest Version to overwrite your checked out file with a read-only copy (losing any changes you may have made). Specify Ignore to take no action with these files.

Files that you do not have checked out. This specifies what you want to happen when Visual SourceSafe finds files that are not out to you, but are different in your working folder. Specify Get Latest Version (the default) to get the latest version of these files. Specify Ignore to take no action with these files.

See Also

[Project Difference Dialog Box](#)

[Show Differences \(Project\) Command](#)

[Difference \(Project\) Command Line](#)

Double_Click Initialization Variable

See Also

Determines default behavior when you double-click on a filename in the Visual SourceSafe Explorer.

Syntax

Double_Click= {View|Edit|Ask}

Example

Instructs SourceSafe to present a dialog box asking whether you want to edit or view a file double-clicked in the Visual SourceSafe Explorer:

```
Double_Click = Ask
```

Remarks

Ask is the default setting.

See Also

Tools Options General Tab

Edit File Command

Cloak Initialization Variable

See Also

Hides cloaked projects from recursive Get, Check Out, Check In, Undo Check Out, and Project Difference commands.

Syntax

Cloak= {Yes|No}

Example

Instructs SourceSafe to cloak a project so that a recursive Get command, for example, would not affect that project:

```
Cloak = Yes
```

Remarks

No is the default setting. This variable can also be set from the File General Properties tab (when a project is selected)

See Also

General Project Properties Tab

Cloak Command Line Command

Merge_Ignore Initialization Variable

Tells Visual SourceSafe to ignore white space characters when merging files.

Syntax

```
Merge_Ignore = w
```

Example

Instructs SourceSafe to ignore white space characters like tab and space while performing a merge operation:

```
Merge_Ignore = w
```

Remarks

You can also set this variable using the Tools Options Difference tab in the Visual SourceSafe Explorer. Check the "Ignore white space differences when merging files" checkbox to do this.

IMG_File Initialization Variable

Specifies the graphics file you want to use to represent a file when generating a site map from a web site project.

Syntax

```
IMG_File = HTMLFILE.GIF
```

Example

Instructs SourceSafe to use the REDDOT.JPG file instead of the default HTMLFILE.GIF in the site map it generates:

```
IMG_File = REDDOT.JPG
```

IMG_Folder Initialization Variable

Specifies the graphics file you want to use to represent a folder when generating a site map from a web site project.

Syntax

```
IMG_Folder = FOLDER.GIF
```

Example

Instructs SourceSafe to use the BLUEDOT.JPG file instead of the default FOLDER.GIF in the site map it generates:

```
IMG_Folder = BLUEDOT.JPG
```

Visual SourceSafe: Project-Oriented Version Control for Microsoft Access

See Also

Microsoft Visual SourceSafe is a version control system for team development of applications using Microsoft Access. Version control systems track and store changes to database objects so developers can review an object's history, return to earlier versions of an object, and develop objects and databases concurrently. Microsoft Visual SourceSafe does this using **reverse delta** technology, and stores only the changes to an object, not each complete version of the object itself. And, unlike other version control systems, Visual SourceSafe is project-oriented. Project-orientation means you can operate on your entire database as a unit when it is convenient – you are not restricted to operations on a single object only.

Access Databases as Visual SourceSafe Projects

When working with a Microsoft Access database, you can store the database file (.MDB) and all its contained objects in Visual SourceSafe. Each of your databases is stored in a SourceSafe project. Query, form, report, macro, and module objects are stored as text files in the project associated with the database they are used in. Other objects (tables, relationships, toolbars, database properties, import/export specifications, etc.) are stored in a special file (.ACB) in the SourceSafe project. This .ACB file should be treated gingerly – you should not rename or share this file in Visual SourceSafe, or you risk database corruption. You also cannot compare two .ACB files using the Show Differences command.

What's the difference? SourceSafe lets you see changes between versions of text files, but not binary files. On the other hand, SourceSafe can show you each difference, no matter how small, between versions of your module objects.

One object can simultaneously exist in many projects – it is **shared**. When you modify an object in one project, the change is automatically propagated to all the projects sharing the object. You can share a critical form between databases or have multiple teams developing a module simultaneously.

Visual SourceSafe Version Control

Visual SourceSafe can maintain multiple versions of an object, including a record of the changes to the object from version to version. Version control addresses:

- Team coordination – making sure, by default, that only one person at a time is modifying an object, to prevent accidental replacement of another user's changes. An optional feature allows you to enable multiple simultaneous checkouts by different users for text files (such as code modules).
- Version tracking – archiving and tracking old versions of databases and other objects, which can be retrieved for bug tracking and other purposes.
- Reusable or object-oriented code – tracking which databases use which modules so that code can be reused.

See Also

[Getting Started With Visual SourceSafe](#)

[Understanding Visual SourceSafe Projects](#)

Getting Started With Visual SourceSafe

See Also

Microsoft Visual SourceSafe is a project-oriented version control system for all types of files, including text files, graphics files, binary files, sound files, and video files. Using Visual SourceSafe, you can track changes made to a file from the moment it was created. And you can merge changes from two or more different versions of a file into one file that contains them all.

Setting Up Visual SourceSafe

Getting started with Visual SourceSafe begins with setting up and connecting to a Visual SourceSafe project. When you use Visual SourceSafe, your files are stored in a project providing both safe storage and important historical information and version tracking.

Generally, your **Visual SourceSafe administrator** installs Visual SourceSafe on a network server for you and then adds your name to the database **user list**. Then, you start the **Visual SourceSafe Explorer** from your computer and access the server where your files are stored.

More and more, developers are accessing SourceSafe's functions from within their development environment. Visual SourceSafe can be easily integrated with Microsoft Visual Basic, Microsoft Visual C++, Microsoft Access, Microsoft Visual FoxPro, and other development tools. If Visual SourceSafe is integrated into your development environment, you do not need to run Visual SourceSafe separately to realize the advantages of source code control.

Note If a Visual SourceSafe database is not yet installed, see your administrator. If you are the administrator, read the Administration chapter of the *Microsoft Visual SourceSafe User's Guide* before proceeding.

Setting Up Projects

The next step is setting up a project. A project is a group of related database objects, typically all the objects required to develop a database.

Use the Add Project to SourceSafe command on the SourceSafe menu to create a project or subproject. Use the Add Objects command to add objects to it.

Working With Objects

When you want to modify an object, you check it out of the project. Visual SourceSafe copies the object from the project into your working folder. You can then edit the object. If anyone else attempts to check out the same object for editing, Visual SourceSafe generates a message stating the object is already checked out (unless multiple checkouts are enabled for text files in that project.) This simple checkout protocol ensures that conflicts do not arise among multiple users working on the same object.

After you are done editing the object, you check it into Visual SourceSafe using the Check In command. This copies the modified object from your folder into the Visual SourceSafe's project, making your changes accessible to other users. However, Visual SourceSafe stores all the changes that have been made to the object—the most recent copy is always available, but earlier versions can be retrieved as well. Visual SourceSafe's **reverse delta** technology ensures that all versions of a object are available, but uses a minimum of disk space.

If you want to retrieve a copy of an object, but don't need to edit it, you can use the Get command to get the most recent version of the object into your working folder. You can use SourceSafe's Show History command to conveniently view the object or project's history, and the Show Differences command to determine differences between an object in your local folder and the latest version of that object stored in the SourceSafe project.

See Also

[Visual SourceSafe: Project-Oriented Version Control for Microsoft Access](#)

[Add Files Command](#)

[Check In Command](#)

[Check Out Command](#)

[Create Project Command](#)

[Installing Visual SourceSafe](#)

[Understanding the Working Folder](#)

[Understanding Visual SourceSafe Projects](#)

[Visual SourceSafe Explorer](#)

[Visual SourceSafe Naming Syntax](#)

Add Objects to SourceSafe Command

See Also

Copies a database object, or group of objects, from your Microsoft Access database into a Visual SourceSafe **project**.

– **To use the Add Objects to SourceSafe command**

- After you've added a Microsoft Access database to source code control, on the Tools menu, click the name of your source code control provider, and then click Add Objects to SourceSafe.

Remarks

Use the Add Objects To SourceSafe dialog box to place Microsoft Access objects under source control. You can use this command only with objects that are already part of your Microsoft Access database and that have not yet been added to source code control.

Access queries, forms, reports, macros, and modules are saved in the source control project as text files. All other Microsoft Access objects are saved in a special file (.ACB) in the source control project. Do not rename the .ACB file!

Dialog Box Options

Object Type

Filters the Object Name list box by object type.

Objects to be added:

Lists all objects in the database of the type specified in the Object Type list box that haven't been added to SourceSafe yet. This command is available only when the database has been added to source code control but some of the objects in it have not.

OK

Confirms your selections.

Cancel

Cancels your selections.

Select/Deselect

Selects or Deselects the highlighted item in the Object Name list box.

Select All

Selects all objects listed in the Object Name list box.

Deselect All

Deselects all objects listed in the Object Name list box.

Advanced

Displays the Advanced options dialog box, if your source control provider supports them. Visual SourceSafe does not.

Comment

Allows you to enter a comment relevant to your operation.

See Also

[Add Database To SourceSafe Command](#)

[Getting Started with Visual SourceSafe \(Visual SourceSafe\)](#)

Show Differences Command

See Also

Compares a selected Access database object to the current version stored in the source code control provider project and shows you the differences, if any.

– **To use the Show Differences command**

- After you've added a database to your source code control provider, on the Tools menu, click the name of your source code control provider, and then click Show Differences.

Remarks

Most commonly, you use the Show Differences command immediately before you check in a object to compare your modified copy of the object to the master copy. This gives you a list of the changes you have made, which can be helpful for generating your comment.

When you use the Show Differences command on a Microsoft Access object, your source code control provider displays a color-coded list of line differences between the source-controlled version of the object and your local copy. A text file representation of the Microsoft Access object is created. The Show Differences command shows you a text-file representation of the differences between the current object on your local drive and the latest version checked in to SourceSafe.

For binary files, your source code control provider tells you if the local copy of the file is different from the master copy, but cannot display the differences. For the Data and Misc. objects, this command is disabled because those objects cannot be compared reliably.

Show History Command

See Also

Shows the history of an object or database under source code control.

– **To use the Show History command**

- After you've added a Microsoft Access database to your source code control provider, on the Tools menu, click the name of your source code control provider, and then click Show History.

Remarks

Every time you check in an object to your source code control provider, the object is given a new version number. Use the Show History command to display a list of an object's or project's past versions.

Note By default, the Show History command displays the File History Options dialog box. Hold down the Shift key when you choose the Show History command to get the SourceSafe Project History dialog box. The Project History dialog box shows history detail for all objects in the project, rather than for just one item.

The versions listed start with the most recent and include all versions back to the creation of the file or project. Visual SourceSafe provides information about the version number, which can be used with other commands (such as Get) to indicate a specific version.

By default, the history of a project includes the histories of all its objects, but is not recursive and does not include **subprojects**.

File History Options Dialog Box

See Also

Labels Only

Includes only **labels** in the project or object history.

From

Type a time, date, version, or label. Use the prefix D to indicate a date or time, and the prefix L to indicate a label. For example "D02/29/95;12:15" or "LBeta1."

To

Type a time, date, version or label. Use the prefix D to indicate a date or time, and the prefix L to indicate a label. For example "D02/29/95;12:15" or "LBeta1."

User

Type a **username** to show a history of objects worked on by the specified user.

OK

Displays the History of File dialog box.

Cancel

Cancels the Show History command.

Project

Displays the Project History Options dialog box. The Project History Options dialog box contains the same options as those in the File History Options dialog box, along with two additional options: Include Files includes object information in the History of Project display; Recursive includes subprojects and all objects within the subprojects. Click OK to display the History of Project dialog box.

See Also

[History of File Dialog Box \(Visual SourceSafe\)](#)

[History of Project Dialog Box \(Visual SourceSafe\)](#)

Share Objects Command

See Also

Lets you add a new object from a different SourceSafe project to the Microsoft Access database you are working on.

– **To use the Share Objects command**

- After you've added the current Microsoft Access database to your source code control provider, on the Tools menu, click your source code control provider, and then click Share Objects.

Remarks

Note To share an object, you create a share **link** among **projects**. The object then exists simultaneously in two (or more) projects. When you check in the object to any one of the projects, your changes are automatically checked in to all of them.

If you create a branch, changes made in the **branched object** no longer propagate to the originating project, and vice versa.

When you share a project, you create a new project under the current project that is an exact duplicate of the specified project. All the objects in the new project are shared with the corresponding objects in the originating project, and changes to one are reflected in the other during check in.

Warning Do not share the .ACB file.

Options Command

See Also

Customizes your source code control provider environment.

– **To use the Options command**

- On the Tools menu, click the name of your source code control provider, and then click Options.

Remarks

The Options command displays the Source Code Control Options dialog box.

Dialog Box Options

For each option, you can specify Yes, No, or Ask.

- Get latest **checked in** versions of objects when opening a database? (No is the default.)
- Check in objects when closing the database? (No is the default.)
- Add objects to source control when adding them to Microsoft Access? (Yes is the default.)
- Remove objects from source control when deleting them from Microsoft Access? (Ask is the default.)

Advanced

Displays advanced options specific to your source control provider. If your provider doesn't support advanced options, this button is hidden.

OK

Confirms selections.

Cancel

Cancels selections.

See Also

[Advanced General Options \(Visual SourceSafe\)](#)

[Advanced Local Files Options \(Visual SourceSafe\)](#)

Create Database from SourceSafe Project Dialog Box

Create a new Microsoft Access database in this folder

Enter the name of the folder in which the database will be placed so you can work on it.

SourceSafe project to download

Visual SourceSafe project list. Click the project you want to open.

OK

Visual SourceSafe gets the files, places them in your working folder, imports them into a new Microsoft Access database, turning them from text files into their corresponding Microsoft Access objects and opens the Microsoft Access database so that you can edit it.

Cancel

Cancels the Create Database from SourceSafe Project command.

Run SourceSafe Command

Starts Visual SourceSafe. If SourceSafe is already running, it is brought to the front of your desktop, not re-launched.

– **To use the Run SourceSafe Command**

- On the Tools menu, click SourceSafe, and then click Run. In some cases, you may need to type your username and (if applicable) your password, and then click OK.

Remarks

Note When you use the Run Visual SourceSafe command, you have access, through Visual SourceSafe Explorer, to all of the Visual SourceSafe features and commands. The Visual SourceSafe commands available from the Tools menu are a subset of the full Visual SourceSafe command set. Of course, if you are using a different source control provider, your options will be different.

Add Database To SourceSafe Project Command

Adds the open Microsoft Access database to a SourceSafe project.

– **To use the Add Database To SourceSafe Project command**

- On the Tools menu, click SourceSafe, and then click Add Database To SourceSafe.

Remarks

The Add Database To SourceSafe command takes the selected objects in the Microsoft Access database and creates a new source code control project containing those objects. This command is available only when you have a database not already under source code control open. Each selected object is automatically added at this time. All objects are selected by default.

Add SourceSafe Project

Project

Type the name of the Visual SourceSafe project to which to add the database, or click a project name in the **project list**. You can enter a full project path specification (as in \$/CODE/RESOURCES) or just type a project name to have a project created under the current project.

OK

Adds the current Microsoft Access database to Visual SourceSafe.

Cancel

Cancels the Add SourceSafe Project command.

Create

Creates a new Visual SourceSafe project. Type a name under Project and click Create.

Get Latest Version Command

See Also

Gets the selected objects from your source code control provider and brings them into the open Microsoft Access database. The objects are not checked out to you. For objects which you already have checked out, Get has no effect.

– **To use the Get Latest Version Command**

- On the Microsoft Access Tools Source Code Control menu, click Get.

Dialog Box Options

Object Type

Filters the Object Name list box by object type.

Objects to Get

Lists all objects in the database of the type specified in the Object Type list box. If other users have added new objects to the SourceSafe project, they will appear in the list of objects, but they will not be added to your database until you get them.

OK

Confirms your selections.

Cancel

Cancels your selections.

Select/Deselect

Selects or Deselects the highlighted item in the Object Name list box.

Select All

Selects all objects listed in the Object Name list box.

Deselect All

Deselects all objects listed in the Object Name list box.

Advanced

Displays the Advanced options dialog box if your source control provider supports them.

See Also

[Get Latest Version Command Advanced Options \(Visual SourceSafe\)](#)

[Getting, Checking In, Checking Out, and Viewing Files](#)

[Local Files Options Tab](#)

Check Out Command

See Also

Checks out the selected object(s) from your source code control provider and places a writable copy in the open Microsoft Access database.

– **To use the Check Out command**

- On the Tools menu, click the name of your source code control provider and then click Check Out.

Dialog Box Options

Object Type

Filters the Object Name list box by object type.

Object Name

Lists all objects in the database of the type specified in the Object Type list box. If other users have added new objects to the SourceSafe project, they will appear in the list of objects, but they will not be added to your database until you get them. Objects already checked out to you are omitted.

OK

Confirms your selections.

Cancel

Cancels your selections and the Check Out command.

Select/Deselect

Selects or Deselects the highlighted item in the Object Name list box.

Select All

Selects all objects listed in the Object Name list box.

Deselect All

Deselects all objects listed in the Object Name list box.

Advanced

Displays the Advanced options dialog box.

Comment

Allows you to enter a comment relevant to your operation.

See Also

[Check Out Advanced Options \(Visual SourceSafe\)](#)

[Getting, Checking In, Checking Out, and Viewing Files](#)

Undo Check Out Command

See Also

Overwrites your changes to a object, and copies the latest checked in version to your Microsoft Access database.

– **To use the Undo Check Out command**

- On the Microsoft Access Tools Source Control menu, click Undo Check Out.

Dialog Box Options

Object Type

Filters the Object Name list box by object type.

Object Name

Lists all checked out objects in the database of the type specified in the Object Type list box.

OK

Confirms your selections.

Cancel

Cancels your selections.

Select/Deselect

Selects or deselects the highlighted item in the Object Name list box.

Select All

Selects all objects listed in the Object Name list box.

Deselect All

Deselects all objects listed in the Object Name list box.

Advanced

Displays the Advanced options dialog box.

See Also

[Getting, Checking In, Checking Out, and Viewing Files](#)

[Local Files Options Tab](#)

[Undo Check Out Advanced Options \(Visual SourceSafe\)](#)

Undo Check Out Advanced Options

Local Copy

- **Replace**—Replaces the object in your working folder
—it performs the Get Latest Version Command on the file. This keeps your working folder up-to-date with the contents of the project; all your changes are ignored.
- **Delete**—Deletes the object from your working folder.
- **Leave**—Leaves the object alone.
- **Default**—Replaces the object in your working folder with the last version from the project, unless you have the Delete Local Files After Add or Check In set on the Local Files Options tab.

Check In Command

Checks in changes made in your Microsoft Access database working folder to your source control project.

Dialog Box Options

Object Type

Filters the Object Name list box by object type.

Object Name

Lists all checked out objects in the database of the type specified in the Object Type list box.

OK

Confirms your selections.

Cancel

Cancels your selections and the command.

Select/Deselect

Selects the highlighted item in the Object Name list box.

Select All

Selects all objects listed in the Object Name list box.

Deselect All

Deselects all objects listed in the Object Name list box.

Advanced

Displays the Advanced options dialog box.

Comment

Allows you to enter a comment relevant to your operation.

Source Code Control Results Window

Displays the results of the last source code control command.

Dialog Box Options

Close

Closes the Source Code Control Results window.

Integration Options Tab

Displays information about the integration between your Microsoft Access database and the corresponding SourceSafe project.

– To display the Integration Options tab

1. On the Tools Source Code Control menu, click Options, and then click the Advanced button. Then, choose the Integration tab.

Access Rights

You must have the Read **access right** to use this tab.

Remarks

The Integration Options tab is most useful for seeing how your SourceSafe projects and Microsoft Access databases are connected. However, you can also set other options on this tab: whether to display intermediate dialog boxes for the History and Show Differences command, and whether to prompt for the SourceSafe project to use when opening a new database.

Tab Options

SourceSafe project

Shows the full name of the SourceSafe project associated with this Microsoft Access database.

Database name

Displays the name (if any) of the Visual SourceSafe project the database is stored in.

Database path

Displays the path to the SourceSafe project the database is stored in.

Data specifier

Displays the data folder for a specific project. (Your SourceSafe administrator can set up different data directories for different projects using the Data_Path variable.)

Display dialog box for

Check the box corresponding to the command if you want an options dialog box displayed when you select that command.

Choose SourceSafe database

Set to Use Default Database to always look in the default SourceSafe database for your project information. Set to Prompt to display a prompt dialog box, allowing you to specify the database name, path, and data specifier each time you open a project.

OK

Sets the options as specified on the tab.

Cancel

Closes the Integration Options tab.

Connect Dialog Box

SourceSafe database path

Type the full path and name of the SourceSafe database you want to connect to. Use the Browse button to browse for the full name.

Data path (optional)

Type the identifier for the data project you want. See your SourceSafe administrator for more assistance with this identifier. (Your SourceSafe administrator must set up this identifier using the Data_Path variable.)

OK

Connects to the specified database, if available.

Cancel

Cancels the database connection.

Refresh Object Status

Refreshes the object status display in the Microsoft Access Database window.

– **To use the Refresh Object Status command**

- On the Microsoft Access Tools Source Code Control menu, click SourceSafe Refresh Object Status.

Remarks

Use this command when you want to check an object's status in a multi-developer environment where other source code control users may have changed that object's status.

SourceSafe Properties

Displays SourceSafe object properties such as comments, checkout status, links to other projects, etc.

– **To use the SourceSafe Properties command**

- On the Microsoft Access Tools Source Code Control menu, click SourceSafe Properties.

Remarks

Use this command to see if you have checked an object in or out, or want to check the project status in a multi-developer environment where other source code control users may have changed an object's status. This command also lets you see if this object is shared with another Microsoft Access database, or see the comment entered when this object was checked in.

A project by this name already exists in the source code control project <project>.

This message has the following cause and solution:

- You tried to add a database to source control in an existing project.

Choose Yes to reconnect to the existing project, No to cancel this operation.

Failed to update project location information in the host application.

This message has the following cause and solution:

- Another application is using the .MDB file.

Make sure no other application is using the database (be sure to check for use via DAO programmatically also).

Unable to load a source code control provider.

This message has the following cause and solution:

- The attempt to load a source control provider failed.

Re-run Setup for your source control provider.

Unable to load the source code control provider - you need a newer version.

This message has the following cause and solution:

- You are using a version of the source control provider older than the supported version.

Re-install the latest version of your source control provider.

The source code control provider did not initialize properly.

This message has the following cause and solution:

- The source control provider was loaded, but was unable to initialize.

Re-run Setup for your source control provider.

<source control provider> did not disconnect properly.

This message has the following cause and solution:

- The source control provider could not disconnect from your project.

Exit Access, and then restart Access to clear this error.

The project could not be properly closed in <source control provider>.

This message has the following cause and solution:

- The source control provider could not close the source control project.

Exiting Access and re-starting it will usually clear this error.

Would you like to get the latest version of all files in project
<project>?

This message has the following cause and solution:

- This query is asking if you want to get all the latest file versions in your project's working folder.

Choose Yes to get the latest versions of all files in the project. Choose No to cancel. You can use the Options command to turn this query message off.

Since you have not logged into <source control provider>, source code control features are not enabled for this session.

This message has the following cause and solution:

- You did not log on to the source control provider.

Re-open your project, and log on to the source control provider to use source control features.

The project <project> no longer exists under source code control. You must either recover or re-create the project before you can use this project with source code control.

This message has the following cause and solution:

- Your project has been deleted (or moved) by another user.

Click OK if you want to try to find the project. If the project has been deleted, choose Cancel. Once you click Cancel, the database will be detached from source code control, and you can re-add the project to source control (but you will lose any historical information on this project).

If you can find out who deleted or moved your project, you may be able to recover the project and its history (for example, if your source control administrator has archived your project to a backup tape).

The object <object> has not been checked out. Do you want to edit it in design mode knowing that all changes will be overwritten with the next Get?

This message has the following cause and solution:

- <this error message isn't being used at the moment>

If you see this error, please report it to Microsoft Product Support.

Could not create a working folder for this Microsoft Access database.

This message has the following cause and solution:

- Your source control provider could not create the working folder on the file system that it needed.

Check the read-write permissions on your files and directories, check your disk space, and ensure the database is on a writable drive.

Unable to access file <file>.

This message has the following cause and solution:

- The source control provider could not make a temporary file writable.

Check the read-write permissions on your files and directories, check your disk space, and ensure the database is on a writable drive.

Failed to export <object> to a file.

This message has the following cause and solution:

- The source control provider could not successfully export a module.

Check that the module is not open in an editor, and is not being used by another process or application.

Failed to import file <file> into Microsoft Access.

This message has the following cause and solution:

- The source control provider could not successfully import an Access object.

Check that the object is not corrupted, and that the object is consistent with the database version.

Would you like to add file <object> to the current source code control project?

This message has the following cause and solution:

- The specified object is not part of your current project.

Choose Yes to add the object to source code control, or No to leave it uncontrolled.

Save all files before adding this project to source code control.

This message has the following cause and solution:

- You are adding a Microsoft Access project to source control without saving your changes first. Save your changes or discard them before adding the file(s).

Do you want to check out <object> before opening it in Design view?

This message has the following cause and solution:

- An object you have selected is not checked out to you.

Choose Yes to check out the object, or No to leave the object checked in.

Cannot check out <object>.

This message has the following cause and solution:

- An object you have selected could not be checked out.

You may not have the proper permissions to check out this object.

Would you like to add file <file> to the current source code control project?

This message has the following cause and solution:

- The specified file is not part of your current project.

Choose Yes to add the file to source code control, or No to leave it uncontrolled.

Would you like to check out the file <file>?

This message has the following cause and solution:

- An object you have selected is not checked out to you.

Choose Yes to check out the object, or No to leave the object checked in.

Failed to add file '<file>' to source code control.

This message has the following cause and solution:

- There is a file with the selected name already in the source control project.

You can save your new file as a different name, or accept the file already in source code control (overwriting any changes).

Would you like to get the latest version of all objects in this database?

This message has the following cause and solution:

- This query is asking if you want to have all the latest object versions in your project's working folder on your computer.

Choose Yes to get the latest versions of all objects in the project. Choose No to cancel. You can turn off this query by using the Options command.

All available objects have already been added to source code control.

This message has the following cause and solution:

- You tried to add an object to source control, but all objects have already been added to source control.

Choose OK to verify.

No objects are available to get from source code control.

This message has the following cause and solution:

- You tried to get an object from an empty database, or all objects are already checked out. Choose OK to try another database, or Cancel to cancel the Get operation.

You do not have any objects to check in to source code control.

This message has the following cause and solution:

- You tried to check in an object that wasn't checked out to you.

Choose OK to try another object, or Cancel to cancel the Check In operation.

No objects in this database are available to be checked out from source code control.

This message has the following cause and solution:

- You tried to check out an object from an empty database, or all other objects are checked out to other users..

Choose OK to try another object, or Cancel to cancel the Check Out operation..

No objects are available to be checked out from source code control.

This message has the following cause and solution:

- None of the objects you selected are checked out, so you can't check them back in yet.

Failed to add <object> to source code control.

This message has the following cause and solution:

- The name of an object you added to your project already exists in the project.

You may save your new object with a different name, or overwrite your new object with the object already in source code control.

Since this database is read-only, source code control features are not enabled.

This message has the following cause and solution:

- You opened the database as read-only when it was under source code control.

Close the database, and change its attribute to writable. Then, re-open the database.

The object '<object>' is checked out in source code control, but not in Microsoft Access.

This message has the following cause and solution:

- This error is usually caused by checking out an object from Visual SourceSafe's Explorer instead of from within Access. It can also be caused if you saved your database, checked out a file, then replaced your database with the saved version.

Choose Yes to undo the checkout, or No to change the status of the local object to checked out.

The object '<object>' is checked out in Microsoft Access, but not in source code control.

This message has the following cause and solution:

- This error is usually caused by checking out an object from Access and then checking it in using the Visual SourceSafe Explorer. It can also be caused if you saved your database, checked in a file, then replaced your database with the saved version.

Choose Yes to undo the checkout, or No to make it checked out (which may overwrite changes other users have already checked in.)

Would you like to remove the object '<object>' from source code control?

This message has the following cause and solution:

- You deleted an object from your project, and your source control provider wants to know if it should be deleted from the source control provider as well.

Choose Yes to remove the object from source control, or No to leave the object in your source control project.

Do you want to check out the Data and Misc. objects?

This message has the following cause and solution:

- You tried to open an object without checking it out first.

Choose Yes to check the objects out, No to edit it (without saving your changes), or Cancel to cancel design mode.

Unable to check out file '<file>' because it is already checked out.

This message has the following cause and solution:

- You tried to check out a file that is already checked out by another user.

Files of this type may be checked out to only one user at a time. You can wait for the other user to check it in, or get your SourceSafe administrator to undo their checkout (which will destroy their changes.)

The following item(s) cannot be added to source code control because their names contain invalid characters:.

This message has the following cause and solution:

- An object you have selected has invalid characters in its name. Visual SourceSafe restricts the use of some characters in names.

Choose OK to rename the object (or leave it out of source control), or Cancel to cancel the operation.

Failed to create a new project based on a <provider> project.

This message has the following cause and solution:

- You encountered an internal error while trying to browse.

Close and restart Access, and Visual SourceSafe if it is running.

Would you like to check in all files which are still checked out to you in project <project>?

This message has the following cause and solution:

You are closing a project leaving some files checked out.

Choose Yes to check in all files, or No to close the project with the files checked out to you.

Not all options could be set correctly.

This message has the following cause and solution:

- You encountered an error writing to your system registry.

Ensure that you have enough space on your drive for changes to the system registry.

An error occurred reading file '<file>' for CRC computation.

This message has the following cause and solution:

- You have encountered an internal source control error.

With Visual SourceSafe, close SourceSafe and re-open it. If that doesn't solve the problem, re-install Visual SourceSafe and then re-boot your computer. If that doesn't solve your problem, check for adequate memory and disk space, check disk space on the server, and check the network connection to your computer.

An error occurred opening file '<file>' for CRC computation.

This message has the following cause and solution:

- You have encountered an internal source control error.

With Visual SourceSafe, close SourceSafe and re-open it. If that doesn't solve the problem, re-install Visual SourceSafe and then re-boot your computer. If that doesn't solve your problem, check for adequate memory and disk space, check disk space on the server, and check the network connection to your computer.

An error occurred during a SourceSafe operation.

This message has the following cause and solution:

- You have encountered an internal source control error.

With Visual SourceSafe, close SourceSafe and re-open it. If that doesn't solve the problem, re-install Visual SourceSafe and then re-boot your computer. If that doesn't solve your problem, check for adequate memory and disk space, check disk space on the server, and check the network connection to your computer.

Properties could not be shown.

This message has the following cause and solution:

- You encountered an internal error while trying to browse.

Close and restart Access, and Visual SourceSafe if it is running.

<provider> does not support this operation.

This message has the following cause and solution:

- Your source code control (SCC) provider does not support the menu item you chose. Contact your source code control vendor, or purchase Microsoft Visual SourceSafe.

Differences could not be shown.

This message has the following cause and solution:

- You encountered an internal error while trying to check differences between objects. Close and restart Access, and Visual SourceSafe if it is running.

Would you like to save your files before proceeding with this
<provider> operation?

This message has the following cause and solution:

- You did not save files before performing a source code control provider operation.
Click Yes to save your files.
Click No to allow Visual SourceSafe to overwrite changes made since the last save operation.

There is already an existing database with the same name in the folder you chose. Do you want to overwrite it?

This message has the following cause and solution:

- You have chosen a name that already exists.

Choose Yes to overwrite the existing database, or No to choose another name.

Unable to check out <object> because it is already checked out.

This message has the following cause and solution:

- You tried to check out an object that is already checked out by another user.

Only modules can be checked out to more than one user simultaneously. Files of this type may be checked out to only one user at a time. You can wait for the other user to check it in, or get your SourceSafe administrator to undo their checkout (which will destroy their changes.)

'<object>' is checked out somewhere else - you cannot rename it now.

This message has the following cause and solution:

- You are trying to rename a file that is currently checked out.

You can wait for the other user to check the file in, or get your SourceSafe administrator to undo their checkout (which will overwrite their changes.)

'<object>' is checked out somewhere else - you cannot rename it now.

This message has the following cause and solution:

- You are trying to rename an object that is currently checked out.

You can wait for the other user to check the object in, or get your SourceSafe administrator to undo their checkout (which will overwrite their changes.)

Could not delete "<file>" from the working folder.

This message has the following cause and solution:

- On opening an Access database, the source code provider attempts to create a temporary working folder. If the folder already exists, then the source code provider attempts to delete any files that may already exist in the folder. If any error occurs, this message will be displayed. Note that the source code provider will silently delete read-only files. One potential cause of this error is if the named file is a running executable file.

The working folder is for the exclusive use of the source code provider. Don't attempt to store any files in this folder, or otherwise access this folder.

Could not delete working folder "<folder>".

This message has the following cause and solution:

- On closing an Access database, the source code provider attempts to delete the temporary working folder and all files in it. You may have a command prompt with its current folder set to the working folder, or a running executable in the folder. Check that your network is functioning properly.

The working folder is for the exclusive use of the source code provider. Don't attempt to store any files in this folder, or otherwise access this folder.

This database must be closed before you can add it to <provider>. Do you want Microsoft Access to close this database?

This message has the following cause and solution:

- To provide source code control Access needs to know that the database is only being used by one developer at a time. Therefore, when you first place a database under source code control, if it was not opened exclusively, Access closes the database and reopens it exclusively, after first asking for confirmation with this message.

Answer Yes to have Access close the database, reopen it exclusively, and proceed with adding the database to source code control. Answer No to cancel the operation.

Failed to create a new database based on a <provider> project.

This message has the following cause and solution:

- Visual SourceSafe was unable to create anew database.

This error can be caused by several things, such as file or folder permissions, network access, or file locking. Check that you have the proper permissions to perform this operation, that your network is working, and that one or more of the database files are not locked by another user.

The command completed successfully.

This message has the following cause and solution:

- You are creating a database from the source control provider and the the source control provider project includes an .ACB file, or you are checking in the .ACB file, or you are adding the .ACB file to the source control provider.

No corrective action is necessary. Just remember to check out the Data and Misc. objects before making changes to tables, relationships, command bars, database properties, or import/export specs that you wish to be saved.

The Data and Misc. objects may have changed. Undo check out and lose changes?

This message has the following cause and solution:

- This prompt is displayed whenever you use the Undo Checkout command on the Data and Misc. objects.

It is not possible for the source control provider to determine whether the Data and Misc. objects have actually changed or not, so this warning is provided to remind you to save any changes you want to keep.

Answer Yes if you wish to replace your local Data and Misc. object with the latest version from the source control provider, or No to leave the local version in place and checked out.

Unable to use the selected source code control project because it already contains a Microsoft Access database.

This message has the following cause and solution:

- When you add an Access database to source code control, you must select a source control project into which the files will be placed. The add-in checks the selected source control project for any files that already exist that belong to an Access database. If any such files are found, this error results.

Don't attempt to store more than one Access database in a single source control project. Choose an empty project (or one that contains only non-Access files), or create a new project.

Unable to use the selected source code control project.

This message has the following cause and solution:

- When you add a host project to source control, you must select an source control project into which the files will be placed. The add-in checks the selected source control project to make sure it is suitable for the operation. If a problem is found, this error results.

Follow the rules imposed by the specific add-in regarding suitable projects for the Add command.

The saved checkout status for some of your objects does not match <object>.

This message has the following cause and solution:

- You moved your database from one machine to another, did a checkout from the Visual SourceSafe Explorer, or are logging into this database as a different user.

Would you like to resolve the conflicts now? Choose Yes to resolve the conflicts or No to exit and restore all objects to their state before you started this operation.

The copy of the database file in the version control database has changed, you can no longer save your current changes. You must check-out the project file and remake your changes.

This message has the following cause and solution:

- This is a warning that another user changed the project file while you were working. Check out the new version of the project file and redo your changes.

Failed to add objects to <project>.

This message has the following cause and solution:

- Visual SourceSafe couldn't save the files that you want to add, and they contain unsaved changes. The Visual SourceSafe operation may have failed, or you could have run out of memory. Save your files manually and try again. If that fails, check your Visual SourceSafe connection and try again.

It is not valid to share a Data and Misc. objects file.

This message has the following cause and solution:

- You tried to share a Data or Misc. objects file into a project which already had one. You can have only one of these in any Visual SourceSafe project.

Run your source control provider and remove the Data or Misc. object you just shared to avoid an invalid database.-

You do not have any objects checked out from source code control.

This message has the following cause and solution:

- You chose the Add, Checkout, Get, Check In, Undo Check Out command, but after careful checking, no files were available for the operation.

This error sometimes occurs if someone deletes the file that you wanted to operate on (or deletes the whole project, for that matter)- make sure your project is still there. Next, make sure you have something checked out.

The project '<project>' no longer exists under source code control. You must either recover or re-create the project before you can use this database with source code control.

This message has the following cause and solution:

- You are trying to use source control on a database which does not exist.

Your database has perhaps accidentally been deleted, so you can try to recover the project, or recreate the project.

Would you like to add the new object(s) to the current source code control project?

This message has the following cause and solution:

- You are adding, either directly or indirectly through a wizard or other tool, several files to source control.

Choose Yes to add the file(s), or No to cancel the operation.

Add this project to <project>?

This message has the following cause and solution:

- You created a new project, or opened a project which hasn't been under source control before.
Choose Yes to add the project, or No to cancel the operation.

Another user (<user>) has placed this project under source code control and is the only person who should work with it.

This message has the following cause and solution:

- Another user has this database under source control and you opened it. You will overwrite all of the other user's changes unless you exit immediately. If the other user had anything checked out, his or her changes will be lost the next time they use the Get Command.

Exit immediately, or sacrifice the other user's changes.

No files are available to get from source code control.

This message has the following cause and solution:

- You chose the Get command, but after careful checking, no files were available for the operation.

This error sometimes occurs if someone deletes the file that you wanted to operate on (or deletes the whole project, for that matter)- make sure your project is still there. Next, make sure you have something checked out.

<file> does not have a valid filename. It must have a legal filename to be added to source code control.

This message has the following cause and solution:

- Although the name of this object is valid for the host application, it isn't a valid file system name (it contains illegal characters).

Rename the object, and then add it to source control.

The share operation succeeded, but the objects could not be added to the current database.

This message has the following cause and solution:

- The share operation succeeded, but trying to add the file to the project failed.
Ensure that you shared a valid Access file, not one with an extension Access doesn't recognize.

The operation failed because some file(s) could not be saved.
Save your project and try again.

This message has the following cause and solution:

- Some files could not be saved to disk.

Ensure that have permission and space to save files on the destination disk. Try to save the files manually, then try the operation again.

This project was named '<name>' but has been moved, copied, or renamed.

This message has the following cause and solution:

- Once you place an Access database under source code control, you are not allowed to rename it or move it to a different directory, or to work with copies of it. When you open a database under source code control which has been moved, renamed, or copied, this message will be displayed and all source code control functions will be disabled for this database.

Do not work with a copy of an Access database which is under source code control. Use the Create Database from SCC command to make a new database in the desired location.

If the database has been moved, return it to its original location and reopen it. Source code control functions will return to normal. Likewise, if it has been renamed, restore its original name.

It is not valid to share a database name file.

This message has the following cause and solution:

- After files were shared from another project, it was found that one of the new files was a database name file (e.g., db.acn). In general, every project should already contain a database name file, and no project should ever have more than one, so sharing one into a project is not recommended.

Run Visual SourceSafe and use the source code control tool to remove the newly shared database name file from the current database.

This source code control project contains a Data and Misc. objects file and a database name file whose names do not match.

This message has the following cause and solution:

- While the source code control status of the Access objects in the current project was being refreshed, it was found that the Data and Misc. objects file (e.g., db.acb) and the database name file (e.g., db.acn) don't match. They should have the same base name (e.g., db.acb and db.acn).

Use Visual SourceSafe to fix the project by making the Data and Misc. objects file match the database name file.

The source code control project contains more than one file representing the Data and Misc. objects of a database.

This message has the following cause and solution:

- While the source code control status of the Access objects in the current project was being refreshed, it was found that the project contains more than one Data and Misc. objects file (e.g., db.acb).

Use Visual SourceSafe to remove all but one of the Data and Misc. objects files.

The source code control project contains more than one database name file.

This message has the following cause and solution:

- While the source code control status of the Access objects in the current project was being refreshed, it was found that the project contains more than one database name file (e.g., db.acn). Use Visual SourceSafe to remove all but one of the database name files.

Failed to add the database to <provider>.

This message has the following cause and solution:

- An error occurred when attempting to reopen the Access database for exclusive read/write access. The database may be in use by another user, or the file may be read-only.

Ask all other users to close the database. Change the file attribute to read/write.

This source code control project contains neither a Data and Misc. objects file nor a database name file.

This message has the following cause and solution:

- You need to add all the database files to the project for source control to function properly.
You will not be able to create a new Access database from this project.

The share operation succeeded, but the objects could not be added to the current database.

This message has the following cause and solution:

- An error occurred while the objects just shared were being imported into Microsoft Access. Check that the shared files came from a valid Access project.

Would you like to check in all objects which are still checked out to you in project <project>?

This message has the following cause and solution:

- You have set the "Check in objects when closing the database" option to "Ask." This informational message asks if you would like to check in all objects still checked out to you.

Choose Yes to check in all objects, or No to leave checked out objects checked out.

The new name cannot be used because an object with that name already exists in <provider>.

This message has the following cause and solution:

- You are attempting to rename a file/object, and the new name is the name of an object/file that already exists in source code control.

Choose a different name for the file/object, or rename/delete the existing object that has the desired name.

Microsoft Access has detected that you don't have a source code control program (such as Microsoft Visual SourceSafe) installed on your machine.

This message has the following cause and solution:

- You have not added source control programs to your computer.

The source code control commands are therefore not available. Choose Yes to display this warning the next time you try to use source control commands, or No to disable this warning message.

For Visual SourceSafe users, run NETSETUP.EXE from your SourceSafe folder to get source control capabilities. For other source control providers, re-run Setup for your provider.

The file <file> is no longer under source code control, the operation failed.

This message has the following cause and solution:

- Some user deleted a file from your Visual SourceSafe database during your session, and you tried to use a context menu to get it.

Re-add the file to source control, check to find out who deleted it and why, etc.

Visual SourceSafe Explorer or one of its components could not be found.

This error has the following cause and solution:

- The Visual SourceSafe folder is not where expected (in the same folder as SSVB.DLL), or may not be on your path.

Verify that the Visual SourceSafe folder is on your path. If the problem persists, rerun Visual SourceSafe SETUP.EXE and choose the Client option.

File <filename> (file <filename> in Visual SourceSafe) was not found.

This error has the following cause and solution:

- You attempted to perform a command on a file that is not in Visual SourceSafe, but may be on your local drive.

Verify that the file exists in your Visual SourceSafe project. If the file does not exist, you can add it to Visual SourceSafe and retry your operation.

File <filename> maps to a name over the maximum Visual SourceSafe filename size of <size>.

Visual SourceSafe must map your local filename to a Visual SourceSafe project path. This error has the following cause and solution:

- The local filename is valid, but when mapped, it exceeds the maximum size for a Visual SourceSafe project path.
You can either shorten the name of the file, or rename the Visual SourceSafe parent project to a shorter name.

Project <project> was not found. Would you like to browse for the project?

Visual SourceSafe tried to open a project that could not be found in your Visual SourceSafe database. This error has the following cause and solution:

- You moved or renamed the Visual SourceSafe project in Visual SourceSafe Explorer, then opened your Visual Basic or Visual C++ project that refers to the old Visual SourceSafe project name.

Click Yes to browse for the project if you know the name of the new project.

Click No if you don't want the project to be under source code control. If you click No, you must manually re-add the project to Visual SourceSafe in order to use source code control features in the future.

File <filename> could not be mapped to the Visual SourceSafe project <project>. Copy the file to <folder>?

Visual SourceSafe must map the local copy into a Visual SourceSafe project path for this operation to work. This error has the following cause and solution:

- You added a file from a folder other than your project folder (or one of its subdirectories) and SourceSafe can't determine which SourceSafe project corresponds to your new file's home folder. Click Yes if you want to copy the file to the folder named.

Copy the file to a subfolder at or below your project. If the file is already in Visual SourceSafe in a different project, share the file to your current project and try the operation again.

Click No if you do not want to continue this operation. You can later copy or share the file and try the request again.

File <filename> could not be mapped to the Visual SourceSafe project <project>.

This error has the following cause and solution:

- You added a file from a folder other than your project folder (or one of its subdirectories) and SourceSafe can't determine which project corresponds to your new file's home folder.

Visual SourceSafe must map the local copy into a Visual SourceSafe project path for this operation to work. Copy the file to a subfolder at or below your IDE project. If the file is already in Visual SourceSafe in a different project, share the file to your current project and try the operation again.

You will continue to get this message for the file as long as the name cannot be mapped. If this happens, you should remove the file from your project file, copy or share it, and then re-add the file.

Unable to load the source code control provider.

This error has the following causes and solutions:

- The registry strings used to find the source code control provider dynamic-link library—for example, SSSCC.DLL—aren't correct, or you have the wrong version of Visual Basic, or the file SSSCC.DLL is missing.
- You moved the source code control folder after setup without updating the registry.
- The setup program failed.
Check your source code provider's setup instructions.
Rerun the Visual SourceSafe setup program.

The source code control provider did not initialize properly.

This error has the following solution:

- Your system is low on memory or system resources.
Close some applications and try again.
If that doesn't help, reinstall the source code control provider.

The source code control add-in did not disconnect from Visual Basic.

This error has the following solution:

- Close Visual Basic and re-open it; this is most likely a temporary internal error.

The source code control provider was unable to connect to Visual Basic.

This error has the following causes and solutions:

- An internal error occurred with OLE and Visual Basic.
- You are using an incorrect version of Visual Basic.
Reinstall Visual Basic and the source code control provider.

The source code control provider was not able to successfully install menus.

This error has the following cause and solution:

- There was an internal OLE error.
Reinstall Visual Basic first, the source code control provider next (if necessary), and then OLE (if necessary).

The source code control provider was not able to connect.

A Visual Basic OLE Automation method failed. This error has the following cause and solution:

- There was an internal OLE, Visual Basic, or source code control provider error.

Reinstall the source code control provider.

Out of memory; could not complete operation.

You do not have sufficient computer memory to perform the desired operation. This error has the following causes and solutions:

- You have too many applications, documents, or source files open.
Close any unnecessary applications, documents, or source files that are open.
- You are running Microsoft Windows in standard mode.
Restart Microsoft Windows in enhanced mode.
- You are running Microsoft Windows in enhanced mode, but have run out of virtual memory.
Increase virtual memory by freeing some disk space, or ensure that some space is available.
- You have terminate-and-stay-resident programs running.
Eliminate terminate-and-stay-resident programs.
- You have many device drivers loaded.
Eliminate unnecessary device drivers.

The file could not be automatically renamed inside the source code control provider.

This error has the following solution:

- Open the source code control provider application and manually rename the file.

Unable to automatically remove the file from the source code control provider.

This error has the following solution:

- Open the source code control provider application and manually remove the file.

Could not connect the source code control project to the current Visual Basic project.

This error has the following cause and solution:

- The source code control provider could not associate the current Visual Basic project with a source code control provider project.

Run Visual SourceSafe and get the project (if it is already in Visual SourceSafe).

Try to perform the same operation on another project. If this works, rename the original project and perform the operation again. If this doesn't work, call [Microsoft Technical Support Services](#).

Unable to add the new project to source code control.

This error has the following cause and solution:

- There are insufficient source code control provider access rights.
Ask your source code control administrator to increase your access rights, and try again, or try to add the project to a different place in the project tree.
Log on again to Visual SourceSafe with the proper password.
(If you have had other errors just proceeding this one, review their causes for more possible causes.)

The previously loaded project could not be properly closed.

This error has the following cause and solution:

- While attempting to open a new project, the source code control provider could not properly close the last one.
Exit and restart Visual Basic.

Unable to set up the source code control menus correctly for this project.

This error has the following cause and solution:

- There was an internal error during set up of the source code control menu system.
Exit and restart Visual Basic, or exit and restart Windows.

This operation is not supported by the source code control provider.

Some source code providers don't support all possible source code control commands. This error has the following cause and solution:

- You attempted to launch an unsupported operation.
Exit and restart Windows, and report this error to [Microsoft Technical Support](#).

Add Files command failed: The files were not added to source code control.

This error has the following solution:

- Open the source code control provider application and manually add the files to the project.

Get command failed: The files were not downloaded from source code control.

This error has the following solution:

- Open the source code control provider application and manually get the files.
Check the Results window for other error conditions. If found, correct them and retry this operation.

Check In command failed: The files were not checked in.

This error has the following solution:

- Open the source code control provider application and manually check in the files. Check the Results window for other error conditions. If found, correct them and retry this operation.

Check Out command failed: The files were not checked out.

This error has the following solution:

- Open the source code control provider application and manually check out the files.
Check the Results window for other error conditions. If found, correct them and retry this operation.

Undo Check Out command failed. Check files to verify their version status.

This error has the following solution:

- Open the source code control provider application and manually undo the check out of the files. Check the Results window for other error conditions. If found, correct them and retry this operation.

Source code control could not refresh some files.

This error has the following cause and solution:

- An error occurred during a Get Latest Version command.
Check for free disk space, or restart Windows.
Check the Results window for other error conditions. If found, correct them and retry this operation.

Some files could not be checked in.

This error has the following cause and solution:

- An error occurred during a Check In command.
Check for free disk space, or restart Windows.
Check the Results window for other error conditions. If found, correct them and retry this operation.

Some files could not be checked out.

This error has the following cause and solution:

- An error occurred during a Check Out command.
Check for free disk space, or restart Windows.
Check the Results window for other error conditions. If found, correct them and retry this operation.

Some files could not have their check out status undone.

This error has the following cause and solution:

- An error occurred during an Undo Check Out command.
Check for free disk space, or restart Windows.
Check the Results window for other error conditions. If found, correct them and retry this operation.

The visual difference operation failed.

This error has the following solution:

- Open the source code control provider application and manually use the differences command.

The Open Project command from the source code control provider failed.

This error has the following solution:

- Open the source code control provider application, and then manually get the project. Then, open the project in Visual Basic.

Unable to start the source code control provider.

This error has the following cause and solution:

- Visual Basic is not able to start a component of the source code control provider.
Run the source code control provider from the icon, Windows 95 Explorer, or File Manager.

Unable to obtain history for this item.

This error has the following cause and solution:

- Visual Basic was not able to get the history for this file or project from the source code control provider.
Run the source code control provider and get the history from there.

Unable to determine the properties of this item.

This error has the following cause and solution:

- Visual Basic was not able to find the properties of this item.
Run the source code control provider and get the properties from there.

Unable to successfully share a file from the source code control provider.

This error has the following solution:

- Run the source code control provider and share files from there.

Not all requested files could be added into the source code control project.

This error has the following cause:

- While adding files to the source code control provider, not all files could be properly replaced. Check the Results window for other error conditions. If found, correct them and retry this operation.

You must check out the project file before you can perform this command.

This error has the following cause and solution:

- You attempted to perform a command that requires check out of the Visual Basic project file before you can check out the project file.
Check out the Visual Basic project file before proceeding, and then try the command again.

This operation requires the Visual Basic project file to be checked out first.

Many operations, such as adding a new file, removing a file, renaming a file, and so forth, require that the Visual Basic project file be checked out first.

- You performed an operation without first checking out the Visual Basic project file.
Check out the Visual Basic project file before proceeding, and then try the command again.

Your project has not been saved.

This error has the following cause and solution:

- You have to save your project before using it with source code control.
Save the project, then try your command again.

Not all options could be set correctly.

This error has the following solution:

- In most cases, options cannot be set correctly due to low memory or disk space. Free some memory or disk space and try again.

Would you like to get the latest checked-in copy of the files in this project?

Click Yes to get the latest version of the file(s).

Click No to keep the file versions you already have.

Add this project to source code control?

In most cases, click Yes to keep the Visual Basic and source code control projects synchronized. You may want to click No if you are experimenting with a file that you may add at a later date, or a file that you do not want to add.

Click Yes to add the file to source code control.

Click No to add the file to your Visual Basic project without adding it to the source code control project.

The file has been removed from the Visual Basic project. Would you also like to remove the file from source code control?

Click Yes to remove the file from the source code control project. (Deleted files can be recovered later if necessary.)

Click No to remove the file from your Visual Basic project, but keep it in the source code control project.

Would you like to check in all changed files now?

Click Yes to check in all files.

Click No to defer check in until a later time.

All available files have been added to the source code control project.

All files in the Visual Basic project have already been added to the source code control provider.

No files are available to get in this project.

This error has the following cause and solution:

- There are no files in the current Visual SourceSafe project.
Use the Add Files command to add files.

You do not have any files to check in.

This error has the following cause and solution:

- You attempted to check in files, but there are none checked out.

If you have made changes to a file that you want to check in, run Visual SourceSafe Explorer, select the file, and on the SourceSafe menu, click Check Out. Be sure to click the Don't Get Local Copy option in the Check Out dialog box so you don't lose your changes.

All available files have already been checked out.

You attempted to check out a file, but all files are already checked out.

The Results window is full; early entries will be removed from the Results window.

When performing an operation on multiple files, the Results window may quickly fill with status messages. This warning message has the following cause:

- The Results window is full. You won't be able to view some of the earliest entries.

Some old results have been removed to make space for newer results.

This message notifies you that old results have been removed; no action is required.

Because you have not logged on to <provider>, source code control features are not enabled for this project.

This error has the following cause and solution:

- You canceled your logon to the source code control provider; the source code provider features are not available.

Open a different project, then return to this project and log on.

Would you like to save your files before proceeding with this <provider> operation?

This error has the following cause and solution:

- You did not save files before performing a source code control provider operation.
Click Yes to save your files.
Click No to allow Visual SourceSafe to overwrite changes made since the last save operation.

The project <project> no longer exists under <provider>.

This error has the following cause and solution:

- The specified project has been deleted (or moved). You must recover (or select) the specified project before any source code control commands can be applied to it.
If the project was moved, you receive a dialog box to help you search for the new project. If the project was deleted, you must run the source code control provider to reinstate the project.

The file <filename> could not be added to the Visual Basic project.

This error has the following cause and solution:

- The file could not be added to Visual Basic.
Try another filename.

File <file> is already checked out.

This error has the following cause and solution:

- .FRM files can be checked out to only one user at a time.
If necessary, run your source code control provider to find out who the file is checked out to.

The file <filename> could not be mapped to the source code control provider project <project>.

This error has the following cause and solution:

- The source code control provider does not know which source code control provider project to place the file into, or get the file from.

The source code control provider determines this based on which folder the file is in. Though the source code control provider can map projects to directories within a project/folder subfolder, files in subdirectories or on other drives create problems, unless there already is an association between the folder that the file is coming from and a source code control provider project.

Add the specified file to a folder at the same level as, or under, the specified project. If the file is already in the source code control provider, share the file to the current project.

Project <project> was not found.

This error has the following cause and solution:

- A project was renamed, moved, or deleted.
Click Browse if you want to look through the source code control provider database for the project.

Can't add .FRX file to source code control.

This error has the following cause and solution:

- An .FRX file was automatically generated to hold resources for the .FRM file, but the .FRX file could not be added to source code control.
Add the file manually.

Can't reload file <file> in the Visual Basic project.

This error has the following cause and solution:

- The file cannot be re-loaded in the Visual Basic project.

Be sure the Visual Basic project file is checked out. Check the filename, and if it is correct, you may have to reload the file in the Visual Basic project. This may require getting a new copy of the file from your source code control provider and placing it into your working folder, or may require restarting Visual Basic.

Only one copy of the source code control provider can run at a time.

This error has the following cause:

- Although multiple, simultaneous sessions of Visual Basic can be run, only one source code control provider can run at a time; this session of Visual Basic has source code control disabled.

File <file> could not be mapped to the Visual SourceSafe project <project>.

This error has the following cause and solution:

- You must have the file in Visual SourceSafe before continuing the operation.
Click Yes to copy the file to the specified project and continue.
Click No to cancel the operation.

File <file> could not be mapped to the Visual SourceSafe project <project>.

This error has the following cause:

- The code development environment prohibits copying files directly into Visual SourceSafe.

Your project file has not been saved. Save your project file before using source code control operations.

This error has the following cause and solution:

- Source code control requires that you save your project file so that it has a true file to work with.

Your project has not been saved. Save your project before adding it to source code control.

This error has the following cause and solution:

- Source code control requires that you save your project so that no project changes are lost.

No corresponding Visual Basic project file for this SourceSafe project.

This error has the following cause and solution:

- The source code control provider could not locate the Visual Basic project file (.VBP or .MAK file). A project file is required because it stores the list of files included in the project, and the source code control provider uses this list to keep track of file status.

This project is not under source code control.

This error has the following cause and solution:

- Visual SourceSafe has determined that the current project is not stored in the Visual SourceSafe database.

If you want to add this project to source code control now, save your project and on the Add-Ins source code control provider menu, click Add Project.

Would you like to add file <file> to the current source code control project?

In most cases, click Yes to keep the Visual Basic and source code control projects synchronized. You may want to click No if you are experimenting with a file that you may add at a later date, or a file that you do not want to add.

Click Yes to add the file to source code control.

Click No to add the file to your Visual Basic project without adding it to the source code control project.

The source code control project already has a makefile with the same name as the one you are adding. Is this the same file?

This message has the following cause and solution:

- You are attempting to add a makefile with the same name as a file already in the source code control provider project.

Check the filename, and rename if necessary.

Click Yes to add the currently selected file, overwriting the old one with the same name.

Click No to cancel the operation.

That source code control project didn't contain the loaded Visual Basic project file. Try again?

This error has the following cause and solution:

- You selected a source code control project that didn't contain the proper Visual Basic project file.
Click Yes to resume browsing through the source code control database for the correct project.
Click No to cancel the operation.

The change you just made has changed the Visual Basic project file, but you have not checked the Visual Basic project out yet.

This error has the following cause and solution:

- You made a change to your Visual Basic project file, but you did not check out the file from the source code control provider first.

Click Yes to check out the project file so you can save your changes.

Click No to keep the project file checked in. You will not be able to save your changes in the source code control provider.

The file <file> is not a normal file type for adding to a Visual Basic project.

This error has the following cause and solution:

- The file you are adding to the source code control provider is of a type not generally recognized by Visual Basic.

Click Yes to continue adding it to the source code control provider.

Click No to cancel the add operation.

The file <file> already exists in the source code control project, but your local copy is different.

You (or another user) has added this project to the source code control provider. However, when you tried to add files to the project, a different file with the same name was already in the project. You should use the source code control provider's difference command to see what the differences are, and then check in the proper version.

Rather than using the Add Project command, you could also get the entire project to your local drive, then check out the files you need. Using this method, you would get back in synchronization with the project, but would lose any local changes you had made to the project files.

The file rename operation has been stopped, not all files were saved.

This error has the following cause and solution:

- You choose not to allow your source code control provider to check out the project file you are renaming.

If you choose not to allow the source code control provider to check out the project file while renaming a file, the operation cannot proceed so that the file does not get renamed in the source code control provider.

You have created a new project.

This error has the following cause and solution:

- You choose to save your project to a different folder, but your source code control provider can't map the new folder name directly.

The old project is still under source code control, but the new one is not. Use the Add Project command to manually add the new project name to source code control.

The project cannot be added to source code control because the path is too long.

This error has the following cause and solution:

- The path name for this project is too long.

Use a shorter path name and try again.

SRCSAFE.INI file not found in folder <folder>.

This error has the following cause and solution:

- You have chosen a folder path in the Connect to Visual SourceSafe dialog that does not contain a SRCSAFE.INI file.

You need to choose the folder where their database information exists. This is typically on the server where SourceSafe is installed.

<string> contains an invalid character.

This error has the following cause and solution:

- You have entered a data path value in the Connect to Visual SourceSafe dialog that contains an invalid character.

Data path names should not contain the () [] or blank characters.

Project <project> has been removed and must be recovered before opening.

This error has the following cause and solution:

- You have deleted a project and then tried to open it from Visual Basic or Visual C++.

You cannot open a deleted project until it is recovered.

File <file> has not been changed.

This error has the following cause and solution:

- You are attempting to check in a checked out file that you have not changed.
Click Yes, and Visual SourceSafe checks it in anyway. (Clicking Yes All checks in all selected files.) A check in creates a new version of the file, with an optional comment.

Click No, and Visual SourceSafe undoes the check out of the file. (Clicking No All undoes the checkout for all selected files.) No new version of the file is created in this case.

Internal error: please exit and restart Visual Basic.

This error has the following cause and solution:

- An internal error has occurred in Visual Basic or Visual SourceSafe.

Save your work, exit from Visual Basic and Visual SourceSafe, and restart those programs if you still need to use them.

Some files have not been named yet.

This error has the following cause and solution:

- You have not named all your files.

Name your files using the Visual Basic Save File command before proceeding with your operation.

The file '<file>' is not under source code control.

This error has the following cause and solution:

- You have selected a file that is not under source code control.

Use the Add Files command to add the file to your source code control provider before trying to check it out or use source code control provider functions on it.

The file '<file>' is no longer connected to this source code control project.

This error has the following cause and solution:

- You have selected a file that is no longer under source code control in this project.

The file has possibly been deleted from the source code control provider, or moved to a different project. Use the Add Files command to add the file to your source code control provider in the proper project, or search the source code control database for the file if it may be part of another.

Please save and check-in your Visual Basic project file.

This error has the following cause and solution:

- You have made changes that modify your Visual Basic project file.

You need to save and check-in your makefile. Otherwise, your project will contain references to the copied-in files in their old locations instead of the current locations.

Errors during load of file <file>.

This error has the following cause and solution:

- Errors were encountered while loading the file.

The file may or may not have been loaded successfully by Visual Basic. Try opening the file again, or check the file for corruption.

You have not added the makefile to source code control. This project has not been connected to source code control.

This error has the following cause and solution:

- You added a new project to source control, but did not include the makefile.

Re-add the project and include the makefile.

A VB project by this name already exists in the source code control project <project>. Would you like to reconnect this VB project to the existing source code control project?

This error has the following cause and solution:

- You tried to add a project, but a project with that name was already under source control.

Choose Yes to reconnect to the existing project, or choose No if the project in source control is a different project by the same name. Then, choose a new name for the new project.

Would you like to check out file '<file>'?

This error has the following cause and solution:

- You tried to edit the Visual Basic file <file>, without having it checked out.

Choose Yes to check out the file, or No to cancel the operation.

You can share another makefile into this source code control project, but the makefile has not been added to the VB project.

This error has the following cause and solution:

- You tried to share a makefile from another project into the current project, but have not added the makefile into SourceSafe.

Add the makefile into the proper project first.

The Get operation failed.

This message has the following cause and solution:

- You have unsaved objects that would be lost if the operation proceeded.

Save all objects in your project to a file, and then try again.

The Check In operation failed.

This message has the following cause and solution:

- You have unsaved objects that would be lost if the operation proceeded.

Save all objects in your project to a file, and then try again.

The Undo Check Out operation failed.

This message has the following cause and solution:

- You have unsaved objects that would be lost if the operation proceeded.

Save all objects in your project to a file, and then try again.

The Share operation failed.

This message has the following cause and solution:

- You have tried to share a file that has the same name property as one in the target project.

There are several possible causes for this message. One is if you have not assigned the name property in one of your files, you could be conflicting with a file in the target project (where the name property also was not assigned or is identical). In this case, assign the name property of the file in your current project, and then try again.

Other possibilities are that the project file is checked out to someone else, or you were not able to check out the project file. Check the file checkout status to determine if this is the case. Or, there could have been a Visual SourceSafe internal error.

You have tried to change the Visual Basic project file without checking it out first.

This message has the following cause and solution:

- You need to check out the Visual Basic project file before you can make changes to it.

You must check out the project file and then make your changes.

This operation requires the Visual Basic project file to be checked out first. Would you like to check out the Visual Basic project file?

This message has the following cause and solution:

- You need to check out the Visual Basic project file before you can make changes to it.

You must check out the project file and then make your changes.

Unable to initialize the host application for source code control.

This message has the following cause and solution:

- SourceSafe could not connect properly to your development host.

Rerun the setup program for your development environment, and then rerun the Visual SourceSafe setup program.

Unable to connect Visual Basic to source control. Please check the installation of the source code control add-in.

This message has the following cause and solution:

- SourceSafe could not connect properly to your development host.

Rerun the setup program for Visual SourceSafe.

There is no makefile associated with this project, unable to create a Visual Basic project from this <project> project.

This message has the following cause and solution:

- You tried to create a Visual Basic project from source control, but specified an invalid project.

Check the project name, and try again.

Another user (<user>) has placed this database under source code control and is the only person who should work with it.

This message has the following cause and solution:

- You are trying to open a database owned by another user.

You were not the last user to use this file. If you continue, you will overwrite their changes and checkout status. Log out from Visual SourceSafe, and then get your own copy of the file using the Create <object> From SourceSafe command.

Could not create a valid file name for <file>.

This message has the following cause and solution:

- You can only add modules to Visual SourceSafe if the module name is a valid filename. The file name you chose is not valid.

Rename your object to something with a valid filename.

If you mistyped the file name, please try again with a valid file name. If you believe the file name is valid, and are working in a mixed-operating system environment, the problem may be that the file name you chose does not translate properly in one of the other operating system environments connected to your Visual SourceSafe database. In this case, try another filename.

You have not added the .ACB file to source code control to have a complete database.

This message has the following cause and solution:

- You added an Access project, but chose not to add a .ACB file.

Re-add the project to source control, and include the .ACB file.

A database by this name already exists in the source code control project <project>. Would you like to reconnect to it?

This message has the following cause and solution:

- You tried to add a project, but a project with that name was already under source control.

Choose Yes to reconnect to the existing project, or choose No if the project in source control is a different project by the same name. Then, choose a new name for the new project.

The History operation failed.

This message has the following cause and solution:

- You have unsaved objects that would be lost if the operation proceeded.

Save all objects in your project to a file, and then try again.

Required supporting files could not be added to source code control, the Check In operation failed.

This message has the following cause and solution:

- You tried an Add operation, but it failed when SourceSafe tried to add supporting project files (for example, the .frx form file).

Add each file separately to source control.

The item was not removed from the <provider> project.

This message has the following cause and solution:

- This is usually caused by an internal SourceSafe error.

Remove each file separately from source control.

The item was not renamed in the <provider> project.

This message has the following cause and solution:

- This is usually caused by an internal SourceSafe error.

Rename each file separately in source control.

The file could not be reloaded into the host IDE.

This message has the following cause and solution:

- This is usually caused by an internal application error.

Close the project and re-open it.

You have created a new project with a different name. The new project is no longer associated with source code control.

This message has the following cause and solution:

- You have renamed a project and the renamed project is no longer in source control.

Add the new project to source control, if desired.

The file that was renamed had a tagalong file which could not be automatically renamed in source control.

This message has the following cause and solution:

- This is usually caused by an internal SourceSafe error.

Manually rename the tagalong file.

You have chosen not to check out the Visual Basic project file, all the changes you have made to the project during this session will be lost.

This message has the following cause and solution:

- This is a warning that if you do not check out the project file, any changes you make in this editing session will not be saved in source control.

Check out the project file if you want to save your changes.

The copy of the Visual Basic project file in the version control database has changed, you can no longer save your current changes. You must check-out the project file and remake your changes.

This message has the following cause and solution:

- This is a warning that another user changed the project file while you were working.

Check out the new version of the project file and redo your changes.

Failed to initialize the current project. Please exit and reload this project.

This message has the following cause and solution:

- This is usually caused by an internal SourceSafe error.

Exit and reload the project. If that doesn't solve the problem, exit and reload your development environment application. If the problem persists, reboot your computer.

Out of Memory - could not complete operation.

This message has the following cause and solution:

- Your computer has run out of memory.

Close unused applications. If the problem persists, restart your computer.

Only a single project can be under source control at a time with this source code control provider.

This message has the following cause and solution:

- Some source control providers can handle only one project at a time.

Contact your vendor for an updated version of the source code control provider. Or, choose a more advanced source control provider, or don't use multiple projects..

Unable to initialize the host application for source code control.

This message has the following cause and solution:

- The source provider could not connect properly to the host development environment..

Rerun the setup programs for your host development environment and source control provider..

Failed to initialize the source code control provider.

This message has the following cause and solution:

- The source provider could not connect properly to the host development environment..

Rerun the setup programs for your source control provider..

Failed to reload the project.

This message has the following cause and solution:

- An internal application error occurred.

Close this project from the file menu and re-open it to get the correct version of all files loaded into memory.

Failed to add files to <project>.

This message has the following cause and solution:

- Visual SourceSafe couldn't save the files that you want to add, and they contain unsaved changes.

The Visual SourceSafe operation may have failed, you may not have add rights in this project, or you could have run out of memory. Save your files manually and try again. If that fails, check your Visual SourceSafe connection and try again.

You do not have any files checked out from source code control.

This message has the following cause and solution:

- You chose the Add, Checkout, Get, Check In, Undo Check Out command, but after careful checking, no files were available for the operation.

This error sometimes occurs if someone deletes the file that you wanted to operate on (or deletes the whole project, for that matter)- make sure your project is still there. Next, make sure you have something checked out.

Would you like to add the new file(s) to the current source code control project?

This message has the following cause and solution:

- You are adding, either directly or indirectly through a wizard or other tool, several files to source control.

Choose Yes to add the file(s), or No to cancel the operation.

The share operation succeeded, but the files could not be added to the current project.

This message has the following cause and solution:

- The share operation succeeded, but trying to add the file to the project failed.

Check that the name property of the file does not conflict with the name property of an existing form, module, class module, or other Visual Basic file.

The filename '<file>' contains illegal characters. The file will be ignored.

This message has the following cause and solution:

- You put one or more illegal characters (like a semicolon, slash, backslash, colon, etc) in the specified filename.

Rename your file using legal characters.

This project was named '<name>' but has been moved, copied, or renamed.

This message has the following cause and solution:

- Once you place an Access database under source code control, you are not allowed to rename it or move it to a different directory, or to work with copies of it. When you open a database under source code control which has been moved, renamed, or copied, this message will be displayed and all source code control functions will be disabled for this database.

Do not work with a copy of an Access database which is under source code control. Use the Create Database from SCC command to make a new database in the desired location.

If the database has been moved, return it to its original location and reopen it. Source code control functions will return to normal. Likewise, if it has been renamed, restore its original name.

Could not create a working folder for this project.

This message has the following cause and solution:

- The working folder you specified could not be created.

You may not have permission to create a folder on the computer you specified, or the drive may be full, or the name you choose may already be in use. Try specifying another name.

The file <file> already exists in the source code control project, but your local copy is different.

This message has the following cause and solution:

- You tried to add a file to your project when a file by that name already existed in the project. The file was not added, and your local copy has not been changed.

Choose a different name for the file you want to add.

You have not added enough files to source code control to have a complete project.

This message has the following cause and solution:

- You tried to add some files to source control, but they are not enough to completely define a project. This project has not been connected to source code control.

Visual SourceSafe needs to maintain all the files in the project. Add the complete set to the Visual SourceSafe project.

The <file> may have changed. Undo checkout and lose changes?

This message has the following cause and solution:

- This confirmation message asks if you want to continue with your Undo Check out command, even though it may mean losing changes you have made to your local file.

Choose Yes to undo your checkout, and No to cancel the operation.

Some file(s) could not be saved. The project was not added to <provider>.

This message has the following cause and solution:

- You are adding a project to source control without saving changes.

SourceSafe was not able to save some of your changes. You may not have write permission on some files, or you may have a network error.

An error occurred while trying to connect to the <project> project.

This message has the following cause and solution:

- The source control provider could not initialize the source control project.

Exit your development environment and re-start it. If this does not clear the error, re-run your source control provider's setup program. Check your system resources, especially memory and disk space.

Add this project to <project>?

This message has the following cause and solution:

- You have not added this project to source control.

Choose Yes to add this project to source control. Choose No to leave this project out of source control (for instance, to debug code first).

You don't have a source code control program (such as Microsoft Visual SourceSafe) installed on your machine.

This message has the following cause and solution:

- You have not added source control programs to your computer.

The source code control commands are therefore not available. Choose Yes to display this warning the next time you try to use source control commands, or No to disable this warning message.

For Visual SourceSafe users, run NETSETUP.EXE from your SourceSafe folder to get source control capabilities. For other source control providers, re-run Setup for your provider.

The new name cannot be used because an object with that name already exists in <provider>.

This message has the following cause and solution:

- You are attempting to rename a file/object, and the new name is the name of an object/file that already exists in source code control.

Choose a different name for the file/object, or rename/delete the existing object that has the desired name.

The new name cannot be used because it is not a valid filename.

This message has the following cause and solution:

- You are attempting to rename a file/object, and the new name is not a valid filename (typically because it contains invalid characters).

Choose a different name for the file/object.

No files are available to be checked out from source control.

This error has the following cause and solution:

- There are no unchecked-out files in the current Visual SourceSafe project.
To check out a file not yet added to source control, use the Add Files command to add files. If the file is already in your project, make sure it is not already checked out.
This error sometimes occurs if someone deletes the file that you wanted to operate on (or deletes the whole project, for that matter)- make sure your project is still there.

All available files have already been added to source control.

This error has the following cause and solution:

- All the files in your current development project have already been added to source control. If you want to check in a file that is not yet in your development project, first add the file to your development project. Visual SourceSafe will prompt you to add it to source control.

Would you like to remove the file '<file>' from source code control?

This message has the following cause and solution:

- You have removed an object from the development project.

Choose Yes to remove the object from source control as well, or No to leave the object in the source control project.

The file <file> is no longer under source code control, the operation failed.

This message has the following cause and solution:

- Some user deleted a file from your Visual SourceSafe database during your session, and you tried to use a context menu to get it.

Re-add the file to source control, check to find out who deleted it and why, etc.

The <project> project that you selected does not contain this IDE project.

This message has the following cause and solution:

- This warning informs you that while trying to browse for a deleted project you picked the wrong project.

Re-try the browse operation.

The <project> project that you selected does not contain this VB project.

This message has the following cause and solution:

- This warning informs you that while trying to browse for a deleted project you picked the wrong project.

Re-try the browse operation.

Would you like to get the latest version of all files in project
<project>?

This message has the following cause and solution:

- This query is asking if you want to get all the latest file versions in your project's working folder.

Choose Yes to get the latest versions of all files in the project. Choose No to cancel. You can use the Options command to turn this query message off.

Since you have not logged into <source control provider>, source code control features are not enabled for this session.

This message has the following cause and solution:

- You did not log on to the source control provider.

Re-open your project, and log on to the source control provider to use source control features.

The project <project> no longer exists under source code control. You must either recover or re-create the project before you can use this project with source code control.

This message has the following cause and solution:

- Your project has been deleted (or moved) by another user.

Click OK if you want to try to find the project. If the project has been deleted, choose Cancel. Once you click Cancel, the database will be detached from source code control, and you can re-add the project to source control (but you will lose any historical information on this project).

If you can find out who deleted or moved your project, you may be able to recover the project and its history (for example, if your source control administrator has archived your project to a backup tape).

Add this project to <project>?

This message has the following cause and solution:

- You created a new project, or opened a project which hasn't been under source control before.
Choose Yes to add the project, or No to cancel the operation.

Another user (<user>) has placed this project under source code control and is the only person who should work with it.

This message has the following cause and solution:

- Another user has this database under source control and you opened it. You will overwrite all of the other user's changes unless you exit immediately. If the other user had anything checked out, his or her changes will be lost the next time they use the Get Command.

Exit immediately, or sacrifice the other user's changes.

No files are available to get from source code control.

This message has the following cause and solution:

- You chose the Get command, but after careful checking, no files were available for the operation.

This error sometimes occurs if someone deletes the file that you wanted to operate on (or deletes the whole project, for that matter)- make sure your project is still there. Next, make sure you have something checked out.

This is an integration only project; checking out files is not allowed from the SourceSafe Explorer.

This message has the following cause and solution:

- You chose the Check Out command on files restricted from checking out except while in the host development environment.

Open the project from within Visual Basic, Access, or other development environment and check the files out from there.

Add Files to Source Code Control Command (Development Environment)

See Also

Copies a file, or group of files, from your **development environment** project into a Visual SourceSafe **project**.

- **To use the Add Files to Source Code Control command**
 - After you've added a development environment project to source code control, on the Tools SourceSafe menu, click Add Files to Source Code Control.

Remarks

Use the Add To Source Code Control dialog box to place a development environment file into the source code control database. You can use this command only with files that are already part of your development environment project. If you have a file you want to add to source code control that isn't part of any project, add it to your development environment project first.

Note Development environment files already added to the current source code control project are not displayed under Files To Be Added.

Dialog Box Options

Files To Be Added

Files available to be added to the source code control database. Click the check box next to the file(s) you want to add.

Comment

Type an optional comment for all selected files.

OK

Adds file(s) to source code control.

Cancel

Cancels the Add Files To Source Code Control command.

Select All

Selects all files listed.

See Also

[Add Project To Source Code Control Command](#)

[Adding Files to the Visual SourceSafe Database \(Visual SourceSafe\)](#)

[Getting Started with Visual SourceSafe \(Visual SourceSafe\)](#)

Show Differences Command (Development Environment)

See Also

Compares a selected file in your **development environment** project to the current version stored in the source code control provider database and shows you the differences, if any.

– **To use the Show Differences command**

- After you've added a project to your source code control provider, on the Tools SourceSafe menu, click Show Differences.

Remarks

Most commonly, you use the Show Differences command immediately before you check in a file to compare your modified copy of the file to the master copy. This gives you a list of the changes you have made, which can be helpful for generating your comment.

When you use the Show Differences command on an ASCII text file, your source code control provider displays a color-coded list of line differences between the database-stored version of the file and your local copy.

For binary files, your source code control provider tells you if the local copy of the file is different from the master copy, but cannot display the differences.

See Also

[File Difference Dialog Box \(Visual SourceSafe\)](#)

[Show Differences \(File\) Command \(Visual SourceSafe\)](#)

[Show Differences \(Project\) Command \(Visual SourceSafe\)](#)

File Difference Dialog Box (Visual SourceSafe)

See Also

By default, Visual SourceSafe uses the Visual display option, with line width calculated to fit on your screen, and no context lines. If you specify a UNIX display, Visual SourceSafe uses the Ignore Case and Ignore White Space options. The Visual display shows all lines (infinite context).

Visual

Shows the entire file, with annotations of the changes.

SourceSafe

Shows a list of lines that were added, deleted, and changed, optimized for readability.

Unix

Shows differences information in the format used by the UNIX Diff utility.

Ignore Case

Shows differences without regard for upper- and lowercase differences.

Ignore White Space

Shows differences without regard for spacing characters, such as tabs.

Report

Sends a report to the **Clipboard**, a text file, or a printer.

Project

Displays the Project Difference dialog box.

Advanced Options

Ignore OS Differences

Tells Visual SourceSafe to ignore operating system line-break character differences. This is useful if you are comparing files across a network where one computer is a Macintosh, for example, and the other is operating with Microsoft Windows NT.

Show Context Lines

Includes the specified number of context lines in the differences display. Context lines are the lines in your file surrounding the line identified as different by Visual SourceSafe. For example, setting Show Context Lines to three means you'll see three context lines above and three context lines below the difference line.

Save Settings

Saves your selected settings for future use.

See Also

Show Differences (Project) Command (Visual SourceSafe)

Show History Command (Development Environment)

See Also

Shows the history of a file or project under source code control.

– **To use the Show History command**

- After you've added a **development environment** project to your source code control provider, on the Tools SourceSafe menu, click Show History.

Remarks

Every time you check in a file or project to your source code control provider, the file or project is given a new version number. Use the Show History command to display a list of a file's or project's past versions.

Note By default, the Show History command displays the File History Options dialog box; click Projects to display the Project History options dialog box.

The versions listed start with the most recent and return to the creation of the file or project. Visual SourceSafe provides information about the version number, which can be used with other commands (such as Get Latest Version) to indicate a specific version.

By default, the history of a project includes the histories of all its files, but is not recursive and does not include **subprojects**.

See Also

[File History Options Dialog Box \(Visual SourceSafe\)](#)

[History of File Dialog Box \(Visual SourceSafe\)](#)

[History of Project Dialog Box \(Visual SourceSafe\)](#)

File History Options Dialog Box (Visual SourceSafe)

See Also

Labels Only

Includes only **labels** in the project or file history.

From

Type a time, date, version, or label. Use the prefix D to indicate a date or time, and the prefix L to indicate a label. For example "D02/29/95;12:15" or "LBeta1."

To

Type a time, date, version or label. Use the prefix D to indicate a date or time, and the prefix L to indicate a label. For example "D02/29/95;12:15" or "LBeta1."

User

Type a **username** to show a history of the files worked on by the specified user.

OK

Displays the History of File dialog box.

Cancel

Cancels the Show History command.

Project

Displays the Project History Options dialog box. The Project History Options dialog box contains the same options as those in the File History Options dialog box, along with two additional options: Include Files includes file information in the History of Project display; Recursive includes subprojects and all files within the subprojects. Click OK to display the History of Project dialog box.

See Also

[History of File Dialog Box \(Visual SourceSafe\)](#)

[History of Project Dialog Box \(Visual SourceSafe\)](#)

Share Files Command (Development Environment)

See Also

Makes the selected file a member of the **current project**, or creates a new branch project.

– **To use the Share Files command**

► After you've added a **development environment** project to your source code control provider, on the Tools SourceSafe menu, click Share Files.

Remarks

Note To share a file, you create a share linkGloss_sharelink among projectsGloss_project. The file then exists simultaneously in two (or more) projects. When you check in the file to any one of the projects, your changes are automatically checked in to all of them.

If you create a branch, changes made in the **branched file** no longer propagate to the originating project, and vice versa.

When you share a project, you create a new project under the current project that is an exact duplicate of the specified project. All the files in the new project are shared with the corresponding files in the originating project, and changes to one are reflected in the other during check in.

Note When sharing a project with Visual Basic form files (.FRM), be sure to share the associated .FRX file created by Visual Basic. If you don't, the shared project will not have all the files it needs and will not work correctly.

See Also

[Share Files Dialog Box Options \(Visual SourceSafe\)](#)

[Branching Files and Projects \(Visual SourceSafe\)](#)

[Sharing Files \(Visual SourceSafe\)](#)

[File Types Options Tab \(Visual SourceSafe\)](#)

Options Command (Development Environment)

See Also

Customizes your source code control provider environment.

– **To use the Options command**

- After you've added a **development environment** project to your source code control provider, on the Tools SourceSafe menu click Options.

Remarks

The Options command displays the Source Code Control Options dialog box.

Dialog Box Options

For each option, you can specify Yes, No, or Ask.

- Get **checked in** version of project files when opening a project? (No is the default.)
- Check in files when closing the project? (Ask is the default.)
- Add files to source code control when adding them to Visual Basic? (Yes is the default.)
- Remove files from source code control when deleting them from Visual Basic? (Ask is the default.)

Advanced

Displays advanced options.

OK

Confirms selections.

Cancel

Cancels selections.

See Also

[Advanced General Options \(Visual SourceSafe\)](#)

[Advanced Local Files Options \(Visual SourceSafe\)](#)

Create Project from SourceSafe Command (Development Environment)

See Also

Copies a project to your hard disk when the project is in source code control but not on your hard disk.

– **To use the Open New Project command**

- On the Tools SourceSafe menu, click Download SourceSafe Project.

Note: After you locate a project you want, this command opens it in your **development environment** and automatically gets all of its files from the Visual SourceSafe database.

See Also

[Open SourceSafe Project Dialog Box \(Visual SourceSafe\)](#)

Create local project from SourceSafe Dialog Box (Visual SourceSafe)

Creates a new development project from an existing SourceSafe project.

– **To use the Download SourceSafe Project command**

- On the Tools menu, click the name of your source code control provider, and then click Download SourceSafe Project.

Note Use the Download SourceSafe Project dialog box to specify an existing SourceSafe project containing a project you need, and then choose a folder on your local computer in which to place the new project files. Microsoft Access will open the newly created project upon completion.

Dialog Box Options

Create a new project in this folder

The **working folder** for the selected project. Click Browse to create or change the working folder.

SourceSafe Project to download

Visual SourceSafe project list. Click the project you want to open.

OK

Visual SourceSafe gets the files, places them in your working folder, and opens the project in your development environment so that you can edit it.

Cancel

Cancels the Download SourceSafe Project command.

Run Command (Development Environment)

Starts your source code control provider.

– **To use the Run command**

- On the Tools SourceSafe menu, click Run. In some cases, you may need to type your username and (if applicable) your password, and then click OK.

Remarks

Note When you use the Run Visual SourceSafe command, you have access, through Visual SourceSafe Explorer, to all of the Visual SourceSafe features and commands. The Visual SourceSafe commands available from the Tools SourceSafe menu are a subset of the full Visual SourceSafe command set.

Add Project To Source Code Control Command (Development Environment)

See Also

Adds the current **development environment** project to the source code control database.

– **To use the Add Project To Source Code Control command**

- On the Tools SourceSafe menu, click Add Project to Source Code Control.

Note If you specify the name of your source code control provider, that name is displayed in place of "Source Code Control."

Remarks

The Add Project to Source Code Control command takes the selected files in the development environment project window and creates a new source code control project containing those files. This command is available only when you select a project not already under source code control. Each selected file is automatically added at this time.

See Also

[Add SourceSafe Project Dialog Box \(Visual SourceSafe\)](#)

Add SourceSafe Project Dialog Box (Visual SourceSafe)

Project

Type the name of the Visual SourceSafe project you want, or click in the **project list** to choose a project. You can enter a full project path specification (as in `$/CODE/RESOURCES`) or just type a project name to have a project created under the current project.

OK

Adds the current **development environment** project to Visual SourceSafe.

Cancel

Cancels the Add Project to SourceSafe command.

Create

Creates a new Visual SourceSafe project. Type a name under Project and click Create.

Get Latest Version Command (Development Environment)

See Also

Gets the selected files from your source code control provider and places a read-only file in the **development environment** project.

- **To use the Get Latest Version command**
 - On the Visual Basic Tools SourceSafe menu, click Get Latest Version.

Dialog Box Options

Files to Get/Check In

Selects the file or files you want to get or check in.

OK

Confirms your selections.

Cancel

Cancel your selections.

Select All

Selects all files listed.

See Also

[Get Latest Version Command Advanced Options \(Visual SourceSafe\)](#)

[Getting, Checking In, Checking Out, and Viewing Files](#)

[Local Files Options Tab](#)

Get Latest Version Command Advanced Options (Visual SourceSafe)

Sets default Get Latest Version command options.

Replace Writable

Determines Visual SourceSafe's behavior when you attempt to get a file that is already marked as writable in your working folder. Select one of the following in the list box:

- Ask—Asks you what to do in each case.
- Replace—Visual SourceSafe replaces the file with a read-only version.
- Skip—Skips the Get Latest Version operation.
- Merge—Merges the changes between the checked out-file and the one you are now getting or checking out.
- Default—Uses the default set on the Local Files Options tab in the SourceSafe Options dialog box (Tools menu).

Set File Time

Changes the time assigned to the Get Latest Version operation. Select one of the following in the list box: current time, the last file modification time, the last file check-in time, or the default time (set on the Local Files Options tab).

End of Line

Specifies the end-of-line character that Visual SourceSafe gives to each line in the file it retrieves. Select one of the following in the list box: line feed (LF) for the UNIX platform, carriage return (CR) for the Macintosh, carriage-return/line-feed (CR/LF) for MS-DOS/Windows, or the default (set on the Local Files Options tab).

Make Writable

Retrieves the file and makes it writable. This is not equivalent to a Check Out operation, because the Get Latest Version command is non-exclusive— one or more people can perform the Get Latest Version command on the same file. You cannot check in changes after a Get Latest Version operation with this option set.

Check Out Command (Development Environment)

See Also

Checks out a copy of the selected file(s) from your source code control provider and places a writable copy in the corresponding **development environment** working folder.

– **To use the Check Out command**

- On the Visual Basic Tools SourceSafe menu, click Check Out.

Dialog Box Options

Files to be Checked Out

Lists the files available for check out. Click the files you want to check out, or click Select All to select all files listed.

Comment

Type a brief description of the reason for the check out.

OK

Confirms Check Out.

Cancel

Cancels the Check Out command.

Select All

Selects all files listed.

See Also

[Check Out Advanced Options \(Visual SourceSafe\)](#)

[Getting, Checking In, Checking Out, and Viewing Files](#)

[Check Out Status Properties Tab](#)

[Local Files Options Tab](#)

Check Out Advanced Options (Visual SourceSafe)

Sets default Check Out command options.

Replace Writable

Determines how Visual SourceSafe behaves when you try to check out a file that is already marked as writable in your working folder. Select one of the following in the list box:

- Ask—Asks you what to do with each file.
- Replace—Replaces the file with a read-only version.
- Skip—Skips the check out operation.
- Merge—Merges the changes between the checked out-file and the one you are now getting or checking out.
- Default—Uses the defaults set on the Local Files Options tab in the Options dialog box (Tools menu).

Set File Time

Changes the time assigned to the check out operation. Select one of the following in the list box: current time, the last file modification time, the last file check-in time, or the default time (set on the Local Files Options tab).

End of Line

Specifies the end-of-line character that Visual SourceSafe gives to each line in the file it retrieves. Select one of the following in the list box: line feed (LF) for the UNIX platform, carriage return (CR) for the Macintosh, carriage-return/line-feed (CR/LF) for MS-DOS/Windows, or the default (set on the Local Files Options tab). Visual Basic users should always set this to CR/LF.

Don't Get Local Copy

Tells Visual SourceSafe not to copy the file to your working folder. Visual SourceSafe records that the file is checked out, but does not give you a new copy.

Undo Check Out Command (Development Environment)

See Also

Overwrites your changes to a file, and copies the latest checked in version to your **development environment** working folder.

– **To use the Undo Check Out command**

- On the Visual Basic Tools SourceSafe menu, click Undo Check Out.

Dialog Box Options

Files to Un-Check Out

Lists the currently checked out files. Click the files whose check out you want to undo, or click Select All to select all the listed files.

OK

Confirms the selection.

Cancel

Cancels the selection.

Select All

Selects all files listed.

See Also

[Getting, Checking In, Checking Out, and Viewing Files](#)

[Local Files Options Tab](#)

[Undo Check Out Advanced Options \(Visual SourceSafe\)](#)

Undo Check Out Advanced Options (Visual SourceSafe)

Local Copy

- **Replace**—Replaces the file in your working folder
—it performs the Get Latest Version command on the file. This keeps your working folder up-to-date with the contents of the project; all your changes are ignored.
- **Delete**—Deletes the file from your working folder.
- **Leave**—Leaves the file alone.
- **Default**—Replaces the file in your working folder, unless you have the Delete Local Files After Add or Check In set on the Local Files Options tab.

Check In Command (Development Environment)

Checks in changes made in your **development environment** working folder to your source code control database.

– **To use the Check In command**

- On the Visual Basic Tools SourceSafe menu, click Check In.

Dialog Box Options

Files To Be Checked In

Lists the files available for check in. Click the files you want to check in, or click Select All to select all the files listed.

Comment

Type an optional comment.

Keep Checked Out

Updates the source code control database with your changes but keeps the file checked out to you.

OK

Checks in the selected files.

Cancel

Cancels the Check In command.

Find MakeFile Dialog Box (Development Environment)

Identifies your **development environment** project file so the source code control provider can properly perform its operations. Appears only when you perform operations on projects that contain more than one makefile is in the same folder.

Dialog Box Options

Select the Visual Basic project file

Select the development environment project file for the project you want to work with.

OK

Opens the selected project file.

Cancel

Cancels the command.

Source Code Status Dialog Box (Development Environment)

Displays the status of the current source code control command.

Dialog Box Options

Cancel

Cancels the current source code control operation.

Source Code Control Results Window (Development Environment)

Displays the results of the last source code control command.

Dialog Box Options

Close

Closes the Source Code Control Results window.

Integration Options Tab

Displays information about the integration between your project and the corresponding Visual SourceSafe project.

– **To display the Integration Options tab**

- On the Tools SourceSafe menu, click SourceSafe Options, and then click the Advanced button. Then, choose the Integration tab.

Access Rights

You must have the Read **access right** to use this tab.

Remarks

The Integration Options tab is most useful for seeing how your SourceSafe and development projects are connected. However, you can also set a couple options on this tab: whether to display intermediate dialog boxes for the History and Show Differences command, and whether to prompt for the SourceSafe database to use when opening a new project.

Tab Options

SourceSafe project

Shows the full name of the SourceSafe project associated with this development project.

Database name

Displays the name (if any) of the **Visual SourceSafe database** the project is stored in.

Database path

Displays the path to the SourceSafe database the project is stored in.

Data specifier

Displays the data folder for a specific project. (Your SourceSafe administrator can set up different data folders for different projects using the Data_Path variable.)

Display dialog box for

Check the box corresponding to the command if you want an options dialog box displayed when you select that command.

Choose SourceSafe database

Set to Use Default Database to always look in the default SourceSafe database for your project information. Set to Prompt to display a prompt dialog box, allowing you to specify the database name, path, and data specifier each time you open a project.

OK

Sets the options as specified on the tab.

Cancel

Closes the Integration Options tab.

Open Database Dialog Box

Available databases

This list shows the name and path of the databases currently available to open. You can use the Browse button to search for a database that doesn't appear in the list.

Open

Opens the specified database, if available.

Cancel

Cancel the database connection.

Remove

Removes the database connection from the available list. (The database is not physically deleted.)

Refresh File Status (Development Environment)

Refreshes the file status display in the Visual Basic project window.

– **To use the Refresh File Status command**

- On the Visual Basic Tools SourceSafe menu, click SourceSafe Refresh File Status.

Remarks

Use this command when you have checked a file in or out, or want to check the project status in a multi-developer environment where other source code control users may have changed a project file's status.

Find Makefile Dialog Box

Allows you to locate a project file so you can open it under source control.

Once the project file (make file) is located, click OK to open it under source control, or click Cancel to cancel the operation.

