Microsoft® SCC Integration User's Guide

StarTeam 15.0

Borland®

THE OPEN ALM COMPANY

Contents

Preface

The StarTeam Microsoft SCC Integration integrates the StarTeam file component with integrated development environments (and other third-party applications) that use the Microsoft Source Code Control (SCC) Application Programming Interface (API) to provide version control for files within your application. These applications include, but are not limited to:

- SilkPerformer 2007 Release 2
- SilkPerformer 2008
- SilkTest 2007 Relese 2
- SilkTest 2008
- Sybase PowerBuilder 10.5 and 11
- Sybase PowerBuilder .NET 11.2

Important   Be aware that applications do not always implement the SCC API in the same way. As a result, the Integration may work differently with different applications.

This guide does not cover these applications in depth. Please check the Borland web site for periodic updates to both the Integration and its documentation.

The installation program for this Integration copies an online version of this guide to the installation folder. This online manual is in Adobe's Portable Document Format (.PDF) and requires version 4.0 or higher of the Adobe Acrobat Reader. The free reader is available from the Adobe web site at: www.adobe.com.

Contacting Borland Support

Borland Software Corporation is committed to providing world-class services in the area of consulting and technical support. We have over 15 years of experience in supporting developers and enterprise customers. Our qualified technical support engineers are prepared to handle your support needs on a case-by-case basis or in an ongoing partnership. Borland provides support worldwide, delivering timely, reliable service to ensure every customer's business success.

For more information about Borland's support services, please see our web site at http://support.borland.com.

From the Web site, you can also access many newsgroups where users exchange information, tips, and techniques. See http://info.borland.com/newsgroups/ for the latest list of free product newsgroups.

When contacting support, be prepared to provide complete information about your environment, the version of the product you are using, and a detailed description of the problem.

For support on third-party tools or documentation, contact the vendor of the tool.

Documentation Conventions

The application documentation uses the following conventions.

Choose File > Exit	Indicates a menu selection followed by a submenu selection. The greater-than symbol (**>**) separates the commands to be selected from subsequent menus. For example, the Choose the File > Exit command means to choose File from the menu bar and then choose Exit from the drop-down menu.
`Fixed-Space Font`	Text appearing in Courier font represents information that you need to enter and messages from the system.
italics	Syntax appearing in italics represents information that you replace with the names of your files, child folders, etc.
	Italics are also used for the names of dialogs and books and for emphasis.
Bold	Syntax appearing in bold represents information that you must use exactly as shown.
[ ]	Square brackets surround optional syntax.
|	A vertical bar separates mutually exclusive choices in syntax.

Note	Identifies supplemental information.
Tip	Identifies information on alternative procedures or other helpful but nonessential information.
Important	Identifies information that is essential to the completion of a task.
Caution	Identifies actions that may result in loss of data, or procedures that must be followed to ensure that data is not lost.

2

Introducing StarTeam

Software Configuration Management (SCM) is critical to managing the software life cycle process. It includes version control, build management, change management, and so on.

Source control, also called version control, is a way to keep multiple versions of the files that comprise a project. Team members need easy, well-organized, and reliable access to the current and historical revisions of a file. Source control is simply the management and maintenance of these revisions in an archive. A user checks out a revision (usually the most recent—also called the tip—revision), modifies it, and checks it in as a new revision without losing either the former revision or anyone else's work.

StarTeam's version control has been integrated into a number of the most popular integrated development environments. Thus, you can use the application's version control without leaving the development environment with which you are most familiar. The application simply adds a few menu commands and toolbar buttons. It enables each team member to:

- Develop applications in the preferred development environment while simultaneously checking files in and out of the application.

- Link each file to a StarTeam change request, requirement, or task as part of the process of adding or checking in that file.

The team leader can:

- Keep track of who checked in which files and when.

- Keep track of which file revisions contain the changes needed to complete a specific change request, requirement, or task.

The entire team can:

- Store everything related to a product in one location.

Be aware that if the StarTeam projects have been wrapped around PVCS archives or VSS projects, your PVCS or VSS password must be either a blank password or identical to your StarTeam password.

The Integration connects to the application Server using the TCP/IP (Sockets) protocol. Misconfiguration of this protocol may prevent successful connection to the server.

Understanding StarTeam and Its Terminology

Each application Server instance runs a server configuration that specifies where and how your files are stored. The following sections provide an overview of the project structure controlled by an instance of the Server.

Repositories

The server, running a server configuration, can manage any number of StarTeam projects that use the same repository. A repository consists of a database to store project information and a vault that contains the hives in which archive files are stored.
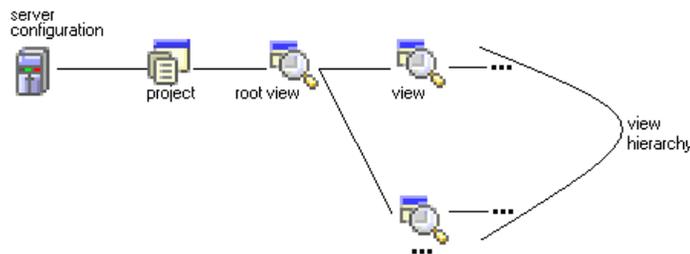
Servers and Projects

A project is a container that allows files to be grouped according to the purpose they serve. For example, a project may include all files related to a software application created by your company or all files related to a component of that software application.
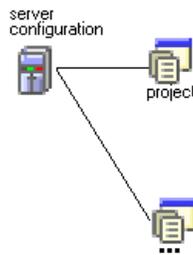
Views

Each project has one root (or main) view and any number of child views.

An administrator will create child views to:

- Provide access to a subset of the files in a project. For example, a developer may not need to be aware of files stored in that project for marketing purposes and a market researcher may not need to be aware of the source code files. They can both work on the same project, but in different views, each of which is a subset of the root view. Such views are called reference views.

- Provide branches for the project. For example, suppose the root view of a project stores source code for Release 1.0 of a software application. After the application is released, assume that work on Release 2.0 continues in the root view. In this situation, it may be desirable to create an additional view for a maintenance release, such as Release 1.0 Service Pack 1. Branching views enable development teams to work on files with the same names but different code or content. The application enables you to merge such files when required.

StarTeam Folders 📁

The root view and the child views each have a single folder as a root folder.
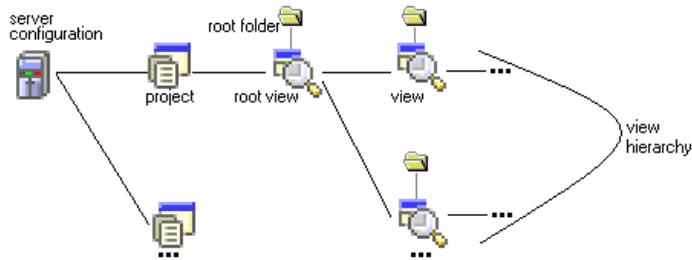
A root folder can have a hierarchy of child folders, which is called the StarTeam folder hierarchy.
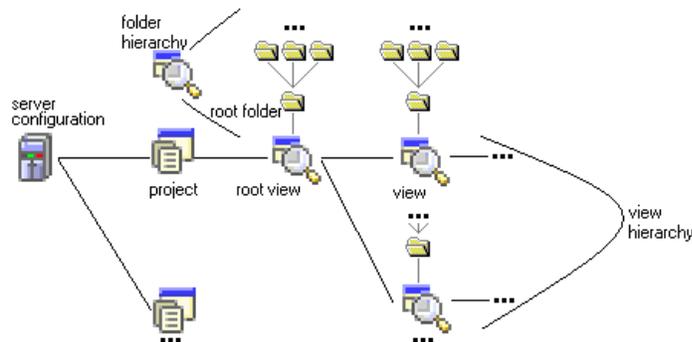
When you create a StarTeam project, its root view and root folder are created automatically and given the same name as the project. For example, if the project name is Great App, the root view name is initially Great App, and the root folder name is initially Great App. If you have the correct access rights, you can change any of these names using the application.
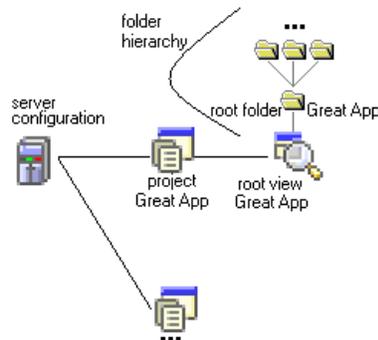
Working Folders 📁

Each application folder has a corresponding working folder. The working folder is the location to which files are checked out onto your workstation. The path to the working folder may have little or no correlation to the path to the application folder.

For example, assume that the root folder for a view is Great App, and it has a child folder named Source Code. Great App has the working folder C:\Great App and Source Code has the working folder C:\Great App\Source Code or even E:\VBstuff\sc. The StarTeam folder Source Code may be a child folder of the StarTeam folder named

Great App, but its working folder is not necessarily a child folder of the Great App working folder.

In the following figure, the working folders form a hierarchy that is similar to that of the StarTeam folder hierarchy.
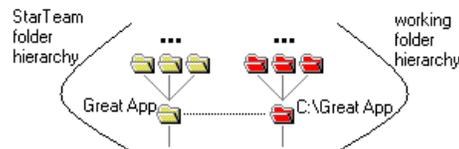
However, as the next figure shows, paths to working folders are not necessarily identical to those of the StarTeam folders.
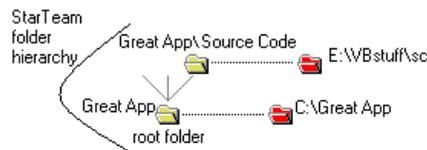
Server Descriptions

Before you can use an instance of StarTeam Server, you must create a server description that identifies that instance. The server description can have a name of your choosing, but it must identify the computer on which the instance is running and the TCP/IP endpoint used for access to that instance.

If your workstation has both the StarTeam Cross-Platform Client and the Integration installed, both applications must use the same server description for the same instance of the server. After the server description has been defined by either application, both can use it.

Note   For more information on projects and views, see the *StarTeam Administrator's Guide*.

Storing Information in StarTeam Projects

Before using StarTeam, with your development environment application, you should plan ahead. Some issues consider are:

- Whether to store your application's workspaces or projects in the same StarTeam Server.

  Workspaces or projects that share a server configuration can share common code very easily. However, the space required for the vault and database may be a consideration.

- Whether to store your application's workspaces or projects in the same StarTeam project.

  StarTeam view labels become attached to every file in a project view. Although desirable for related sets of files that are built at the same time, this feature may be confusing for unrelated sets of files or even for related files that are built at different times.

- Whether your application's workspaces or projects should store all of their files in the root folder of the project view.

  It may be preferable to place different types of information in child folders of the root folder. For example, you may want to put the source code in a child folder named Source Code, the marketing materials in a folder named Marketing, and so on.

Accessing StarTeam

To access the Server from the development environment using the StarTeam Microsoft SCC Integration, you need:

- A user name and password with which to log onto a server configuration.
- The machine name or IP address for the computer running that server configuration.
- The port number for the TCP/IP protocol for the server configuration.

To add workspaces or projects from your development environment to the server configuration, you need the access rights required to:

- Create projects and views or see existing ones.
- Create folders for the project views being accessed.

To use workspaces or projects from your development environment that are stored by the server configuration, you need the rights to:

- See the project view and the appropriate folders.
- See and modify files in those folders.

If process rules are enforced for the StarTeam project, you need the rights to:

- See and modify change requests, requirements, and tasks in the project view.
- Create and modify links on files, change requests, requirements, and tasks.

Understanding the Effects of File Status

In most cases, the status of your files in StarTeam determines the operations that you can perform on them. The following table lists all file statuses and presents a description of each status.

Table 2.1   StarTeam File Statuses

Status	Description
Missing	The file is part of the project view, but it is not in your working folder. If you want to place the file into a working folder on your hard drive, you must check this file out.
Merge	The working file has been altered but is not based on the tip revision (most recent revision) of this file. This status usually appears when you and another team member work on a file at the same time, but the other person checks the file in before you do. When you check this file in, the application prompts you to merge your file with the tip revision.
Modified	The working file has been altered and is based on the tip revision of this file. You may want to check this file in so that the changes are placed in the application.
Out Of Date	The working file is a copy of a prior revision of the file. If you need the tip revision, you should check it out.
Not In View	The file is in your working folder but not in the project view. You may want to add this file to the view. If you are no longer using the file, you may want to delete it from the working folder.
Current	The tip revision of this file is in the working folder.
Unknown	The file in the working folder has the same name as a file in the view, but it was not checked out from the repository. You may have copied it from another location. Use Update Status to determine the correct status.

During the file check-out process, the Integration copies a file revision from the repository to a working folder. Checking in a file places a new revision in the repository. In many cases, the status of a file affects the check in or check out process.

The next table summarizes the relationship between file status and the check-in and check out operations for the application. A successful check in or check out operation changes the status of your working file to Current, unless you have checked out a historical revision or deleted the working file as part of the operation. (continued)

Table 2.2   File Status and Check-in and Check-out Operations

Status	Check In	Check Out
Current	No considerations.	No considerations.
Merge	Starts the merge utility unless you force the check in. File Compare/ Merge, the default merge utility, enables you to merge text files. Merge status means that someone else has checked in this file after your last check out. You do not have their changes in your working file. As a result, their changes will be lost unless the files are merged.	Not allowed unless you force the check out. Your changes will be lost if you check out this file.
Missing	Not applicable. If a file has Missing status, it is not in your working folder so nothing can be checked in.	No considerations. If a file has Missing status, the application asks whether you want to check it out when you open it. You can also check it out manually.
Modified	No considerations. Unless someone else has the file locked, you can check in the file.	Not allowed unless you force the check out. Your changes will be lost if you check out this file because it overwrites the file in your working folder.
Not In View	Not applicable. A file with Not in View status cannot be checked in. You can add it to the project with the Add Files command.	Not applicable. A file with Not in View status is not in the repository, so nothing can be checked out.
Out Of Date	Not allowed unless you force the check in. If you check in an Out Of Date file, that the tip revision will no longer have the changes made to the file after your working copy became Out Of Date.	No considerations. Checking out an Out Of Date file makes your working file Current.
Unknown	Not allowed unless you force the check in. Checking in your working file makes it the tip revision in the repository. If the file status is Unknown, the consequences of this action are also unknown. Use Update Status with an MD5 checksum to see if the file can be identified. If the file cannot be identified, you may want to compare your working file to the tip revision.	Not allowed unless you force the check out. Performing a check out overwrites your working file with the tip revision in the repository. If a file's status is Unknown, the consequences of this action are also unknown. Use Update Status with an MD5 checksum to see if the file can be identified. If the file cannot be identified, you may want to compare your working file to the tip revision.

Moving StarTeam Folders

StarTeam often stores working folder paths from development environment applications as relative paths. For example, ..\sc may be the working folder for a project's Source Code folder.

If you move a folder to another location in the StarTeam hierarchy, its working folder may end up in an unexpected location. This result occurs because the application applies the relative path to the working folder path for thenew parent folder. Therefore, if you move a folder, you may want to specify a working folder path that is not relative, to avoid accidentally changing the working folder path on users' workstations.

3

Introducing the Integration

The StarTeam Microsoft SCC Integration (the Integration) interoperates with various third-party products that use the Microsoft Source Code Control (SCC) Application Programming Interface (API). This API, originally designed by Microsoft to allow applications to work with Microsoft Visual SourceSafe, enables you to perform version control operations, such as checking files in and out, using StarTeam as the SCC provider.

In addition to installing your development environment application or third-party SCC compliant product and the Integration, you can also optionally install the StarTeam Cross-Platform Client on your workstation. Having a copy of the StarTeam Cross-Platform Client installed enables you to change StarTeam personal options or StarTeam project properties.

Note   For installation instructions, refer to the Release Notes (`readme_SCC.html`).

The Integration is called from the development application. How it operates depends upon:

- The application's implementation of the API.

- The options selected in the development application.

- The options selected in StarTeam.

Understanding the Additions to the Development Environment

If your development application uses the SCC API, it usually includes one or more of the following:

- A Source Control menu or submenu.

- A source control toolbar.

- Options for source code control.

- A way to select a specific source control or version control provider.

Understanding the Options that Affect the Integration

Several options selected in the client can affect the Integration. These settings include both project properties and personal options.

To set project properties:

1   Open the StarTeam project view that contains your files.

2   Choose Project > Properties from the menu bar.

3   In the *Project Properties* dialog, select the Options tab or the Process Rules tab.

4   Set the appropriate options. See "Using Project Options" on page 16 and "Using Project Process Rules" on page 17 for more information on these options.

To set personal options:

1   Open the StarTeam project view that contains your files.

2   Choose Tools > Personal Options from the menu bar.

3   In the *Personal Options* dialog, select the File tab.

4   Set the appropriate options. See "Using Personal Options" on page 20 for a description of these options.

Using Project Options

Settings in the following areas of the Options tab of *Project Properties* dialog the affect the Integration:

- Keyword expansion.

  The keywords in your files are expanded as the files are added or checked in, if both of the following actions have been performed:

  - Select the Keyword Expansion check box.

  - Include the extensions for your source code in the Expand Keywords for These File Extensions text box.

- Revision comments.

  The Integration passes along any comment associated with the files to the Server. If the Require Revision Comment When Files Are Checked In check box is selected, files cannot be checked in without a comment. The Comment often provides a reason for the check in.

- Exclusive locks.

  The Require Exclusive Lock When Files Are Checked In check box can be selected to reduce the chances that users will overwrite other users' files. If this is done, you *cannot* check files in from your development environment unless you have exclusively locked them (preferably when you checked them out).

This project option overrides individual users' settings for the Use Non-exclusive Locks in Integrations personal option. If this project option has been selected, users should also select the Exclusively Lock Files On Check-out personal option.

- Read-only attribute.

  To reduce the likelihood that a user will edit a file that is *not* locked, The Mark Unlocked Working Files Read-only check box can be selected to reduce the chances that users will edit unlocked files.

  Working copies of the files that change from locked to unlocked (for example, when they are checked in) become read-only.

  When this project option is selected, it overrides the personal option with the same name — Mark Unlocked Working Files Read-only.

Using Project Process Rules

Process rules provide more control over files that are added or checked into StarTeam. Administrators can ensure that every file is linked to a process item (a change request, requirement, or task). The idea is that the process item provides information about why this file needs a place in your project or needs to be changed.

In StarTeam, you can use process rules even when they are not enforced at the project level. However, if you want the Integration to request the selection of a process item when you place files under version control, process rules must be enforced for the project.

Using process items enables users to clearly distinguish the file revisions that meet or complete a specific change request, requirement, or task. Having this information makes checking the files out for a build much easier.

No process rules are set by default before you create a project from your development application. If you want to use process rules, you must decide whether to add the files from the development application's workspace or project before or after setting process rules. If you add files:

- Before you set process rules, none of the files will be linked to an explanatory process item.

- After setting process rules, a process item must be selected for the files. Therefore, you must both set the process rules and create one or more process items before returning to your development application.

If you are adding a workspace or project to an existing project, process rules may already be enforced. In this case, you need to create the appropriate process items before you add files or turn off the process rules until the files are added.

If others are working on files already in the project, turning off the rules, even momentarily, may be impossible. In this situation, make sure you have added at least one appropriate change request, requirement, or task to the Star Team project. For example, you might create a change request with "Add workspace to StarTeam" as its synopsis.

Understanding Process Item Restrictions

The restrictions set on process items control what you can select. For example, you may be limited to selecting change requests only. In addition, you may be able to select only change requests with specific statuses. If you do not have the access rights required to review project properties, ask your administrator to inform you about the restrictions placed on process items.

To review restrictions on process items:

1 From StarTeam, choose Project > Properties from the menu bar.

2 In the *Project Properties* dialog, select the Process Rules tab.

3 Note whether you can select change requests, requirements, or tasks.

4 Note if a type of process item has been limited to particular statuses. For example, a change request may be limited to:

- Open status.

- In Progress status.

- Both the Open and In Progress status.

If no statuses are selected, then the process item can have any status.

When status restrictions have been placed on process items, you cannot use a fixed change request, completed requirement, or finished task as a process item, unless you change that item's status in StarTeam. For example, if an unexpected change needs to be made to a file and the corresponding change request has already been marked as fixed, you may need to reopen the change request before checking in your file.

Pre-selecting Active Process Items

If you switch back and forth between your development application and StarTeam during the work day, remember that you can select an item as the active process item while in StarTeam.

Selecting an active process item convenient, as it can save you time when adding files or checking them in. For example, after reviewing a task, you usually know whether you are going to work on it soon. If you already have the task selected in StarTeam, making it the active process item is a simple operation.

Only one item at a time can be specified as an active process item.

To pre-select an active process item:

1 From the client, select an appropriate item from the upper pane. When process rules are enforced, you must select a permitted type of item with a permitted status.

2 Do one of the following:

- Choose Change Request > Set Active Process Item.

- Choose Requirement > Set Active Process Item.

- Choose Task > Set Active Process Item.

- Click the Set Active Process Item icon on the toolbar.

The status bar displays the Active Process Item icon (the grinding gears), followed by the term Change Request, Requirement, or Task, and the number.

Notes   Although a Clear Active Process Item command also appears on the Change Request, Requirement, and Task menus, you may never use it because selecting a second active process item clears the first.

Pre-selecting an active process item makes it the default process item in the *Add Files* and *Check In* dialogs. However, you can select a different item at any time.

Opening Projects in Branch Views

To make it easier to log on, the StarTeam Toolbar caches user credentials and server connection information. When switching to a different branched view, the credentials saved in the StarTeam Toolbar are used automatically. To change to a branched view, you need to clear the stored information in the StarTeam Toolbar.

To open projects in branched views:

1 Right-click on the StarTeam Toolbar icon ⬛ in the system tray.

2 Choose **Exit** from the context menu.

3 Log on to project, and select your desired view.

Using Process Items in the Development Environment

Process rules set in StarTeam may require you to select a change request, requirement, or task as part of adding files to source control or checking in files. The selected process item becomes linked to each of the files. The process item can later be marked as fixed or finished.

When using process rules, you will see one or two additional dialogs. The first dialog in the following procedure appears only if you have pre-selected an active process item.

To use process items:

1 If you pre-selected an active process item, the *Active Process Item* dialog appears.

- For a change request, this dialog displays its number and synopsis.

- For a requirement, this dialog displays its number and synopsis.

- For a task, this dialog displays its number and name.

2 To use the pre-selected item:

a (Optional) If the change request is now fixed, the requirement is completed, or the task is now finished, select the Mark Selected Change Request As Fixed, the Mark Selected Requirement As Complete, or the Mark Selected Task As Finished check box.

b The name of this check box changes depending on the type of item selected as the process item.

c Click Use Active Process Item.

3 To edit the selected process item:

a Click the Edit Active Process Item button. This action displays the dialog for the item.

b Make the necessary changes.

c Click OK.

4 To select a different process item or create a new one:

a Click Select a Different Process Item. The *Select Process Item* dialog appears if you have not preselected an active process item, if you have decided not to use the preselected process item, or if the preselected process item is no longer acceptable (for example, its status may have changed).

This dialog allows you to select an existing change request, task, or requirement for use as a process item or create a new item.

b To select an existing item:

1 Do one of the following:

- Select the Linked Items radio button to see a list of items currently linked to the selected file.

- Select the List All Permitted Items radio button to display all existing items that can be used as process items. The list contains all change requests, requirements, and tasks that have the required statuses. If no statuses are required, this list contains all change requests, requirements, or tasks.

- Select the List All Permitted Items Assigned To Me radio button to display all items for which you are responsible that can be used as process items. If a task is selected as a process item, this list displays tasks for which you are a resource, as well as the tasks for which you are responsible.

- Select the List All radio button to see a complete list of items, regardless of status.

2 Select a specific item. If you select an item that is not permitted or no longer permitted as a process item, the Integration notifies you of this fact. You can resolve this problem by changing the item status or selecting a different item.

3 Click OK. The selected item becomes the new active process item. Information about this item appears on the status bar after a refresh.

c To create a new change request, task, or requirement that can be used as a process item:

1 On the *Select a Process Item* dialog, click New.

2 Select the Change Request, Requirement, or Task tab.

3 Enter the required information.

4 Click OK.

When you select a process item or create a new item using the Integration, that item becomes the process item. However, the item does not appear in StarTeam until a refresh (Shift+F5) occurs.

Note   When you select Project > Source Control > StarTeam Properties, the *SCC Properties* dialog appears. This dialog is similar to the *Select Process Item* dialog. It allows you to create a new change request, task, or requirement, edit items, or select an item as the active process item. You can also use it to refresh the list of process items.

Using Personal Options

A number of personal options affect the StarTeam Microsoft SCC Integration. The tab of the *Personal Options* dialog appears when you choose the Tools> Personal Options command in the StarTeam Cross-Platform Client.

The following personal options are important:

- Exclusively Lock Files on Check-Out.
  Use Non-exclusive Locks in Integrations.

  When you work from a development application, files are automatically locked as you check them out, regardless of the setting for the Exclusively Lock Files On Check-out check box.

  If you clear the Use Non-exclusive Locks in Integrations check box, they are exclusively locked.

  If you set the Use Non-exclusive Locks in Integrations check box, they are non-exclusively locked

Note   If the project property Require Exclusive Lock When Files Are Checked In is selected, files are exclusively locked when they are checked out from a given project. The application ignores your setting for the Use Non-exclusive Locks in Integrations check box.

  If neither of these options is set, the default behavior is that files are exclusively locked during the check out operation and unlocked during the check in operation.

- Clear File Locks On Check in.

The Integration always unlocks files as they are checked in, regardless of this setting.

- Mark Unlocked Working Files as Read-only.

  To reduce the likelihood that a user will edit a file that is *not* locked, the Mark Unlocked Working Files Read-only check box can be selected. This option causes working copies of the files that are changed from locked to unlocked to become read-only.

  The project property with the same name, Mark Unlocked Working Files Read-only, overrides this personal option for a specific project if it is selected.

- Use Last Modification Time For Checked-Out Files.

  When you check out a file, its time stamp can be set to the time at which it was last saved or to the current time.

  As a rule, you should select the Use Last Modification Time For Checked-Out Files check box. If you use the current time, your development environment may treat the file as though it has changed after it was last saved.

Using an Alternate Working Path

When you open a StarTeam project to copy the IDE solution or project files to your computer, you can change the local destination.

To use an alternate working folder path destination:

1   In the **Select StarTeam** folder dialog box, click **Change Working Path**. The **Select an alternate directory** dialog box opens.

2   Choose an alternate directory. This alternate working folder path is local to the Integration, and it is not saved after the session ends.

Note   If the working path does not match the location of the project files, you do not need to use the StarTeam Cross-Platform Client to set the folder path or alternate folder path and then get the files from your IDE. The path set using the StarTeam Cross-Platform Client is persisted.

Using File Compare/Merge

File Compare/Merge is a file comparison and merge utility provided by StarTeam. This utility is started by the Integration when you need to compare or merge text files. File Compare/Merge is typically started though a **Compare Contents** command on the source control menu or other dialog. File Compare/Merge starts automatically when a user attempts to check in or check out a file that is no longer based on the latest version in the repository.

Many software applications include their own file comparison and merge utilities. If you want to use these alternate utilities in place of File Compare/Merge, configure StarTeam as explained in .

File Compare/Merge can also be used as stand-alone utilities. For details about using File Compare/Merge, see the *StarTeam User's Guide.*

Using Alternate Comparison and Merge Utilities

To use an alternate comparison or merge utility:

1   In StarTeam, choose Tools > Personal Options.

2 In the *Personal Options* dialog, select the File tab.

3 To use an alternative merge utility, select the Merge Utility check box.

   a Enter the path to or browse for the merge utility in the text box.

   b In the Options text box, add the appropriate command-line options for the utility. Use the following terms to identify the files the application will pass to the merge utility:

   $branchtip
   A place holder for the path to the tip revision of the file to be merged.

   $usertip
   A place holder for the path to the local working file to be merged.

   $basefile
   A place holder for the path to the common ancestor for the $branchtip and $usertip files.

   $resultfile
   A place holder for the path to the file that will store the output from the merged file.

For example, suppose the Merge Utility text box contains "D:\Programs\Merge Utility 5\Mergeutil.exe" and the Options text box contains "-s $branchtip $usertip $basefile $resultfile". The application assumes that the options are appropriate for the utility to be used. Each time that it needs to call the merge utility, it replaces the terms that start with $ with the actual paths to those files and asks the operating system to execute the completed command.

4 To use a comparison utility other than File Compare/Merge, select the Comparison Utility check box.

   a Enter the path to or browse for the comparison utility in the text box.

   b In the Options text box, add the appropriate command-line options for that utility. Use the following terms to identify the files that the application will pass to the comparison utility:

- **$file1**
  A place holder for the path to the first of the two files to be compared.

- **$file2**
  A place holder for the path to the second of the two files to be compared.

For example, suppose the Comparison Utility text box contains "D:\Programs\ Comparison Utility 8\Diffutil.exe" and the Options text box contains "$file1 $file2". In this case, the application assumes that the options are appropriate for the utility to be used. Each time that the application needs to call the comparison utility, it replaces the terms that start with $ with the actual paths to those files and asks the operating system to execute the completed command.

Accessing StarTeam Servers

To access data under source control, users must create a connection to a server configuration from within their development environment. The connection information is then stored in a file named starteam-servers.xml. If you accidentally lose the connection to the server while working in an application, the Integration automatically re-establishes the connection.

For existing server configurations, the Integration uses the MPX profile defined for the configuration. If you add a new server configuration using the Integration, this configuration will use the StarTeamMPX default profile. If you want to view or change this profile, you must use the client. To do this, log onto StarTeam, select Project > New or Open > Server Properties and click on the MPX button.

Depending upon the Integration you use, the procedure for creating a connection varies. The following section explains how to enter the connection information in the *Existing StarTeam Servers* dialog.

Tip   When more than one person accesses the development environment files within a project, all team members should use the same description for the server configuration that stores that project.

To create or modify a connection to a server configuration:

1   From the *Create New StarTeam Project* dialog or *Open Existing StarTeam Project* dialog, click Servers.

2   In the *Existing StarTeam Servers* dialog, click New or Modify.

The *StarTeam Server* dialog appears.

3   In the Server Description text box, enter a unique descriptive name for the server. It is case-insensitive and cannot contain colons (:). It can be any name, but a common choice is the name of the computer on which the server is running. This name should match the description used by others to access this server.

4   In the Server Address text box, enter the computer name or IP address.

5   In the TCP/IP Endpoint text box, enter the appropriate information.

6   Click OK to return to the *Existing StarTeam Servers* dialog.

7   Click OK to return to the *Create New StarTeam Project* dialog or *Open Existing StarTeam Project* dialog, which should now preselect the server description in the StarTeam Server drop-down list box.

Note   If you want to view or change the MPX profile for the new server configuration, you must use the client. To do this, log onto StarTeam, select Project > New or Open > Server Properties and click on the MPX button.

Using the starteam-client-options.xml File

In the starteam-client-options.xml file, the Integration finds the location of the user's settings for the StarTeam personal options, as well as the alternate working folders to be used for a given view or folder.

If your starteam-client-options.xml file is missing, the Integration uses its own defaults for the personal options and assumes that no alternate working folders exist. The next time you use the application, StarTeam will automatically recreate the starteam-client-options.xml file without consulting you. However, the recreated file will contain only the default settings for the personal options and no alternate working folder information.

Note   Borland recommends that you backup this file or place it under version control.

You can tell that the starteam-client-options.xml file is missing or corrupted when:

▪   Your personal options are no longer correct.

▪   Changes you made to personal options disappear when you restart StarTeam.

▪   Even though files have been checked out, they don't change, because they have been copied to the wrong working folders.

▪   StarTeam indicates that old files are missing and does not show new files (it is looking for them in the wrong location).

Preventing Inactivity Timeouts

Due to inactivity timeouts, some customers have experienced problems maintaining a connection to their StarTeam Server when it is protected by a firewall. The Integration

now enables a user to overcome this problem by creating a specific entry in the Windows Registry.

To prevent inactivity timeouts:

1 Open the Windows Registry Editor.

2 In the left pane, locate and select the following key:
```
My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\StarTeam Integrations\StarTeam
Microsoft SCC Integration
The value of Ping is set to 1.
```

3 Do either of the following:

   ▪ In the left pane, right-click on the key, and choose New > DWORD Value from the menu.

   ▪ In the right pane, right-click on an empty area, and choose New > DWORD Value from the menu.

   A new value is added to the right pane, and its name is selected for in-place editing.

4 Enter `Ping` for this value's name, and press Enter.

5 Do one of the following:

   ▪ Double-click the `Ping` value's name.

   ▪ Right-click on the `Ping` value's name, and choose Modify from the menu.

6 In the *Edit DWORD Value* dialog, enter `1` (or some other non-zero value) in the Value Data field, and click OK.

7 Exit the Registry Editor.

Preventing Java VM Conflicts with PowerBuilder

When you connect to a source control system, PowerBuilder instantiates a Java VM by default. For certain SCC programs, such as Borland StarTeam or Serena TrackerLink, the Java VM instantiated by PowerBuilder conflicts with the Java VM instantiated by the SCC program.

To prevent Java VM conflicts, add the following section and parameter setting to the PB.INI file:

[JavaVM]

CreateJavaVM=0

Identifying the StarTeam Runtime Version

If you have trouble with the Integration, customer support may have you check the StarTeam Runtime version you are using.

To determine the StarTeam runtime version:

1 Navigate to the C:\Program Files\Borland\StarTeam SDK xxx\Lib folder and locate the StarTeamSDKxxx.dll. For example, C:\Program Files\Borland\StarTeam SDK 10.4\Lib\StarTeamSDK104.dll.

2 Right click on the StarTeamSDKxxx.dll, and select **Properties** from the context menu. The Properties dialog box opens and displays the required version information.

Index