

IBM VisualAge TeamConnection



Source Code Control User's Guide

Version 2.0

IBM VisualAge TeamConnection



Source Code Control User's Guide

Version 2.0

First Edition (September 1997)

© Copyright International Business Machines Corporation 1997. All rights reserved.

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Notices

Note

Before using this document, read the general information under “Notices” on page vii .

Contents

Notices	iii
Notices	vii
Trademarks	ix
Chapter 1. Differences between other source code control providers and TeamConnection	1
Projects vs Families.	1
Installing the TeamConnection source code control DLL	2
Chapter 2. Connecting TeamConnection to Visual Basic 4.0	3
Removing the TeamConnection Source Code Control DLL	3
Chapter 3. Using TeamConnection as your source code control provider	5
Before you start	5
Opening a project	5
Integrated features	5
Check-in	6
Check-out	6
Uncheck-out	6
Get Version.	6
Properties	6
Full features of TeamConnection	6
MIGRATING DATA BASE.	7

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY, USA 10594.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact the Site Counsel, IBM Corporation, P.O. Box 12195, 3039 Cornwallis Road, Research Triangle Park, NC 27709-2195, USA. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

IBM may change this publication, the product described herein, or both. These changes will be incorporated in new editions of the publication.

This publication contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX	MVS/XA
BookManager	NetView
Common User Access	Operating System/2
CUA	OS/2
C/370	TeamConnection
IBM	VisualAge
MVS	XGA
MVS/ESA	

The following terms are trademarks of other companies:

ObjectStore

ObjectStore Design, Inc.

UNIX X/Open Company Limited

Microsoft, Windows, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

Chapter 1. Differences between other source code control providers and TeamConnection

The purpose of this document is to help Visual Basic, Visual C++, and Power Builder users, make TeamConnection their Visual environments source code control provider. This document assumes the reader is a new user of TeamConnection, but has some familiarity with source code control.

Projects vs Families

Most source code control providers group all code into projects. TeamConnection uses an object oriented approach that provides much more control over the software product while allowing greater flexibility. Projects have one dimension of control. Development environments like Visual Basic group all of their files into projects. Using projects to group source code has several limitations. First, the source code control system is limited to providing just version control. While version control is useful, once the enterprise-size organization is reached, it is often not sufficient to control just versions of the source code. TeamConnection provides not only versioning but defect and feature tracking, build and driver management, access control, and much more. TeamConnection uses families, releases, components, and work areas for management and control.

TeamConnection uses several layers of control. The highest level is the family. The family is the name of the data base, where TeamConnection stores all of the code, the versions, and all other information related to the code. A family represents a complete and self-contained collection of TeamConnection users and development data. Data within a family is completely isolated from data in all other families. One family cannot share data with another. It is important to know the name of the family where TeamConnection will store your code and associated information.

A part in TeamConnection is a collection of data that is stored by the family server. This can include files, text, objects, binary objects, or modeled objects. Parts can be stored by a user, a tool, or generated from other parts, such as when a linker generates an executable file.

Components are used to organize the data in a family. Components are arranged in a hierarchical tree structure, with a single top component called root. The component owns the parts that may be in it, and controls access to the parts. Once you are given access to a component, you have access to all the parts and subcomponents in that component. The component also controls the process that TeamConnection uses, for example, to report and fix a defect. Within each family, development data is organized into groups called components. The component hierarchy of each family includes a single top component, initially called root, and descendants of that root. Each child component has at least one parent component; a child can have multiple parents.

The release is somewhat analogous to a project. A release is a logical grouping of the components that make up a product. An application is likely to contain parts from more than one component. Because you probably want to use some of the same parts in more than one application, or in more than one version of an application, TeamConnection groups parts into releases. A release is a logical organization of all parts that are related to an application; that is, all parts that must be built, tested, and distributed together. Each time a release is changed, a new version of the release is created. Each version of the release points to the correct

version of each part in the release. Each part in TeamConnection is managed by at least one component and contained in at least one release. One release can contain parts from many components; a component can span several releases. Each time a new development cycle begins, you can define a separate release. Each subsequent release of an application can share many of the same parts as its predecessor. You need to know the name of the release.

A work area is basically a view of a release. For example, a work area can be opened for each defect that needs to be fixed. More than one programmer can work in the same work area at the same time. A programmer can have more than one work area active at a time. A release contains the latest integrated version of each of its parts. As users check parts out of the releases, update them, and then check them back in, TeamConnection keeps track of all these changes, even when more than one user updates the same part at the same time.

You need to know the name of the work area in which you will be working. A good practice is to create and name a work area after the defect being addressed in the work area. For example, name work area W1557 for defect 1557. You can create a work area if you have the authority in TeamConnection, but this must be done through the TeamConnection GUI.

For more information about families, releases, components, work areas, parts, and what you can do with them, see your TeamConnection Documentation.

Installing the TeamConnection source code control DLL

Before you can use the integrated support from your development environment you must install TeamConnection and the TeamConnection Source Code Control DLL. If you are using TeamConnection Version 2.08 or later, the source code control DLL is already installed.

Note: If you have not already done so, follow the directions and install the TeamConnection client for your workstation. The following directions assume that you have successfully installed the TeamConnection client.

Chapter 2. Connecting TeamConnection to Visual Basic 4.0

If you are using TeamConnection Version 2.08 or later, the source code control add-in for Visual Basic is already included.

Removing the TeamConnection Source Code Control DLL

To change the default source code control system for Visual Basic, change the value in the ProviderRegKey to the registry key of another provider.

To remove TeamConnection, leave the value in the ProviderRegKey blank.

Chapter 3. Using TeamConnection as your source code control provider

Once the installation procedure is complete, starting your development environment automatically links the TeamConnection Source Code Control DLL.

Before you start

There are several things you must know before you can start using TeamConnection as your source code control provider. If you are not sure of this information, contact your family administrator. Your family administrator can help you find the following information:

- Family
- Component
- Release
- WorkArea

You also need to know the project name. The project name is used by your development tool to relate to the TeamConnection attributes of family, release, work area, and component by the Source Code Control DLL.

Opening a project

One of the few differences you see when using TeamConnection as your source code control provider occurs when you open a project. When you open a new project, the TeamConnection Source Code Control Settings window opens. At the top of this window is a field with your development project name. In addition to the project name field, there are fields for family, work area, release, and component. If this is a new project, these fields are blank. If this is not a new project, the fields contain the previous values. You can change these values only when this window is open. If at anytime you decide to change any of these values, you must first close the project and reopen it.

Once all the fields are filled in, select **OK**. The project will open. If you select **Cancel**, the source code control system disconnects from the development environment until another project is opened.

Under some versions of Visual Basic, projects automatically close and open after certain operations. This causes this TeamConnection Source Code Control Settings window to open at times when it may appear unnecessary. When this occurs, select **OK**. If you select **Cancel**, you will be left in a state that requires shutting down and restarting Visual Basic to reconnect the source code control system.

Integrated features

Once you open a project you can use the integrated features of the development environment to access your files in TeamConnection. The development environment keeps track of the files that are known to TeamConnection, and the checkout status of each file. For example, the development environment keeps track of files checked out to other users.

The exact steps necessary to perform each of the following actions depend on the development environment being used. However, for a given environment, the steps are the same regardless of the source code control provider. For example, if you check out a file in the Visual C++ development environment when it is connected to Visual SourceSafe, the steps you use are exactly the steps you use when C++ is connected to TeamConnection.

Check-in

The steps to check-in a file vary by the development environment. In most cases pressing mouse button 2 when the mouse pointer is over a file icon of a file checked out to you, brings up a menu that includes the file check-in option. Selecting the file checkin option opens the Check-In window. Checking the **keep checked out** box on the Check-In window sets the keep locked flag, TeamConnection saves the file, but keeps it checked out to you. Selecting **OK** causes the TeamConnection part check-in function to execute and the file is checked in.

Check-out

Similar to check-in, the check-out action can be started by pressing mouse button 2 on the file icon of a file not already checked-out. Check-out calls the TeamConnection Part Check-out function.

Uncheck-out

A checked out file can be unchecked out. Again this action can often be started by right clicking the file icon of a file that is checked out. Uncheck-out calls the part unlock function in TeamConnection.

Get Version

Rather than check out a file, you can also get the latest version of the file. Get Version calls the TeamConnection Part Extract function.

Adding Files to source code control

Adding a file that is not already under source code control places the selected file into the source code control system. Add calls the TeamConnection part create function.

Properties

Selecting **Properties** from a pull-down menu opens the properties GUI. Information that TeamConnection needs to correctly check out and check in parts is provided here. For example, the work area field changes each time an existing work area is integrated and a new work area is created.

Full features of TeamConnection

Most development environments allow you to evoke TeamConnection from the pull-down menus. In Visual Basic, TeamConnection appears as an option in the Add-Ins pull-down menu. In Visual C++, TeamConnection appears in the Source Code Control option of the Tools pull-down menu. From the TeamConnection GUI

you can create new work areas (if you have the correct authority), retrieve previous versions of a part, open or process defects, and perform many other actions against parts.

MIGRATING DATA BASE

One key issue for programmers and project managers moving from another source code control system to TeamConnection is how to migrate the database of projects. The following describes one way to bring the current level of code for a small to medium sized project into TeamConnection.

Migrating an existing project

The following example illustrates the simplest way to migrate an existing Visual Basic source code control database into TeamConnection. Lets say we were using the ABC source code control system, and we are going to migrate our project, Austin, to TeamConnection. The idea is to extract all the files in Austin using the ABC source code control system, and then add them as parts in TeamConnection. Follow the steps below to perform this migration.

1. Make ABC the default source code provider. To do this, set the registry key ProviderRegKey to point to the registry entry for source code control provider ABC. See "Installing the TeamConnection source code control DLL" on page 2 for more information on how to perform this step. Once you complete this step, ABC will be the Source Code Control provider when we open Visual Basic.
2. Start the Visual Basic development environment.
3. Open project Austin.
4. Extract all the files to your system.
5. Exit the Visual Basic development environment.
6. Edit the registry key ProviderRegKey to be:
SOFTWARE\IBM\TeamConnection\

See "Installing the TeamConnection source code control DLL" on page 2 for more information on how to perform this step. TeamConnection is now the default source code control provider and is attached when the development environment starts.

7. Restart the Visual Basic development environment.
8. Open project Austin again.
9. When the TeamConnection Source Code Control Settings window opens it will have Austin listed as the project. Fill in the values for family, release, component, and work area, then select **OK**.
10. Add the files to TeamConnection following the steps in the Visual Basic development environment.
11. Repeat these steps until all of your projects are migrated to TeamConnection.

Starting a new project: Starting a new project in Microsoft Visual Basic or Visual C++ is essentially the same regardless of the source code provider. The only operational difference is that the TeamConnection Source Code Control Settings window opens at some point. When the TeamConnection Source Code Control Settings window opens, enter the names of your family, component, work area, and release.

Starting Visual Basic: To create a new project under Visual Basic, do the following:

1. Start Visual Basic
2. Create and save a new project
3. Select the **Add Project to TeamConnection** option from the TeamConnection option in the Add-Ins pull-down menu. The TeamConnection Source Code Control Settings window opens. Fill in the family, release, component, and work area then select **OK**.
4. The Add To TeamConnection window opens. Select the files you want to add. Type a comment in the comment field. Visual Basic requires that a comment be entered. Select **OK**.

Starting Visual C++: To create a new project under Visual C++, do the following:

1. Start the Visual Developers Studio as normal.
2. On the File pulldown, select New. A new window will open.
3. On the New window select Project Workspace, then **OK**.
4. The New Project Workspace window will open. On the New Project Workspace window, do the following:
 - a. Select the type of project
 - b. Type a name
 - c. Select create.
5. The TeamConnection Source Code Control Settings window will open. On the TeamConnection Source Code Control Settings window, enter the family name, release, component, and work area. Then, select **OK**.
6. Files can now be added to the project using the Insert menu pull-down.
7. To place the files under source code control, select the add to source code control option of the Tools menu pull-down.



Part Number: 33H2571



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

33H2571

