madcap software

USER GUIDE

MADCAP LINGO 12

Source Control: Team Foundation Server

Copyright © 2026 MadCap Software. All rights reserved.

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of MadCap Software.

MadCap Software
1660 17th Street, Suite 201
Denver, Colorado 80202
858-320-0387
www.madcapsoftware.com

THIS PDF WAS CREATED USING MADCAP FLARE.

CONTENTS

CHAPTER 1

Introduction 5

CHAPTER 2

General Information for Team Foundation Server 6
- Common Source Control Terms 7
- Source Control Icons 8
- Bind Detection, Disabling Providers, and Unbinding Providers—TFS 10

CHAPTER 3

Process for Team Foundation Server 13
- Binding a Project 14
- Importing From Source Control 20
- Getting the Latest Version 22
- Checking Out Source Control Files 24
- Checking In Source Control Files 26
- Merging Source Control Files 29

CHAPTER 4

Other Activities for Team Foundation Server 31
- Binding a Project to Microsoft Visual Studio Team Services 32

Deleting Source Control Files 36
Disabling the Get Latest Prompt for Source Control 37
Disabling a Team Foundation Server Provider 38
Disconnecting From Source Control 40
Enabling Source Control Status Checks 43
Importing From Microsoft Visual Studio Team Services 44
Rolling Back to an Earlier Version of a File 47
Setting Color Options for Project File Differences 53
Unbinding a Team Foundation Server Provider 55
Undoing a Checkout of Source Control Files 57
Viewing Differences in Source Control Files 58
Viewing Files That are Checked Out 63
Viewing the History of Source Control Files 64

APPENDIX

PDFs 66
Cheat Sheets 66
User Guides 66

CHAPTER 1

Introduction

Microsoft Team Foundation Server (TFS) (or Azure DevOps) is a source control tool that integrates with Lingo. You can use the Lingo interface to perform various source control tasks for a project that is bound to TFS.

General Information

- "Common Source Control Terms" on page 7
- "Source Control Icons" on page 8
- "Bind Detection, Disabling Providers, and Unbinding Providers—TFS" on page 10

Process

1. Setting Up TFS (Done Outside of Lingo)
2. "Binding a Project" on page 14
3. (Other Team Members) "Importing From Source Control" on page 20
4. "Getting the Latest Version" on page 22
5. "Checking Out Source Control Files" on page 24
6. "Checking In Source Control Files" on page 26
7. "Merging Source Control Files" on page 29

CHAPTER 2

General Information for Team Foundation Server

There are various pieces of general information you should know if you plan to use this feature.

This chapter discusses the following:

Common Source Control Terms 7
Source Control Icons 8
Bind Detection, Disabling Providers, and Unbinding Providers—TFS . 10

Common Source Control Terms

Following are definitions for some of the common phrases used in Lingo's integrated source control with Microsoft Team Foundation Server.

- Bind This means to connect your project to Microsoft Team Foundation Server. After doing this, you can then take advantage of all the automated source control tasks (such as check in, check out, and so on).
- Check In This means to copy your local Lingo files to Microsoft Team Foundation Server, overwriting the copies stored in Microsoft Team Foundation Server. This enables other authors on your team to access and edit the latest files.
- Check Out This means to copy the latest source control files to your local Lingo project and remove the "Read Only" designation from them so that you can edit the files.
- Get Latest Version This means to copy the latest source control files to your local project without necessarily checking out the files. This means that the "Read Only" designation will remain associated with the files until you check them out.

NOTE Lingo integrates with multiple source control providers to provide built-in source control support. Each of the source control providers built-in to Lingo uses different terms. As such, Lingo's source control interface is different depending on which source control provider you use. Please refer to the sections for each source control provider if you need to see information about the terms used by other built-in systems.

Source Control Icons

Following are descriptions for the primary icons that you may see next to files when using source control.

Icon	Description
	Checked Out This indicates that the file has been checked out from source control. When your changes are done, you can check in the file. You may see this icon in conjunction with the icon, indicating that you have checked out and locked the file.
	New File (Pending Add) This indicates that you have a file in your project but have not yet added it to the integrated source control application. To resolve this, simply right-click on the file and select Source Control > Add.
	Not Checked Out This indicates that the file is added to source control but is not currently checked out, which means that it contains a "Read Only" designation in its properties. In order to edit the file, you must check it out from source control.
	Locked by Another User This indicates that another user has locked the file. You cannot check in the file while it is locked by another user.
	Locked by You This indicates that you have locked the file. You can lock a file when you check it out. Other users cannot check in the file while it is locked.

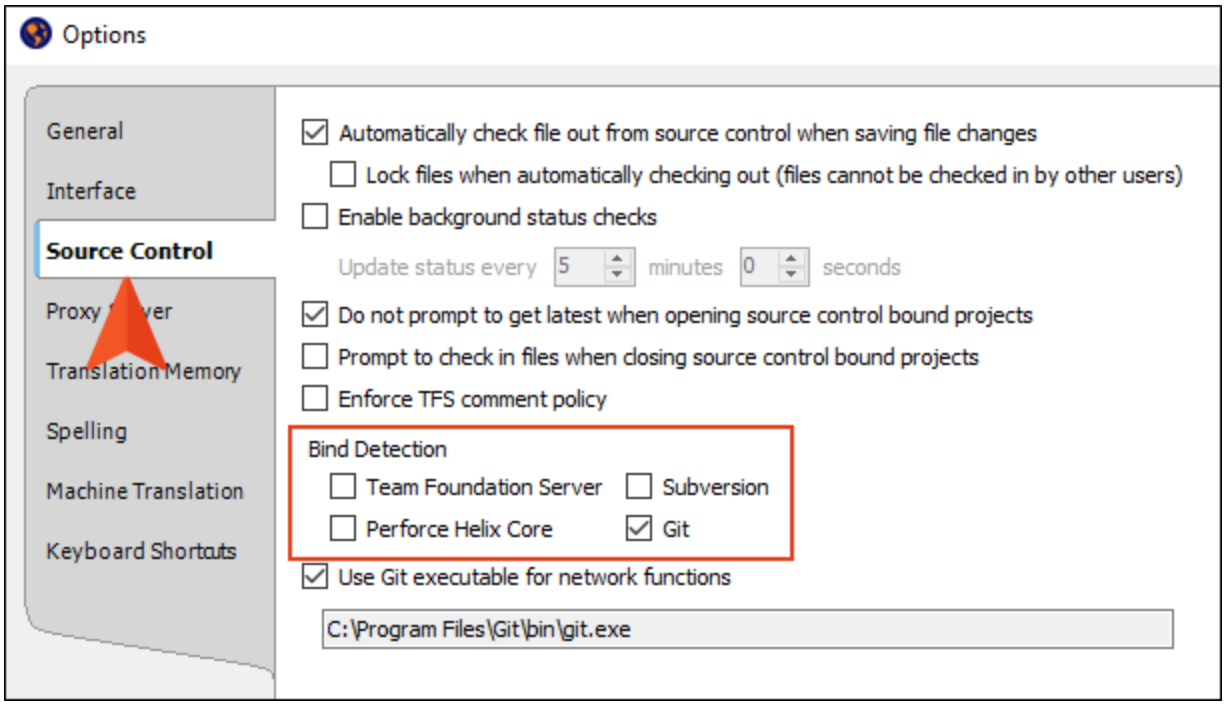
Icon	Description
	Checked Out By Other User This indicates that the file is currently checked out to another user. You may see this icon in conjunction with the icon, indicating that another user has checked out and locked the file.
	Out of Date This indicates that the file is not current (i.e., the local copy of the file is older than the source control copy). This might happen if another user checks out the file, makes changes to it, and checks it back in to source control. If this occurs, you can check the file out or get the latest version of the file from source control.
	Deleted This indicates that the file is deleted from source control, but you still have a copy of the file on your local machine. If necessary, you can add the file to source control.

Bind Detection, Disabling Providers, and Unbinding Providers—TFS

Lingo has options for bind detection, disabling providers, and unbinding providers. Although these are separate features, they are all somewhat related. This information is especially important if you are using an external tool to bind and manage your source control tasks.

Bind Detection

Lingo's bind detection settings are found on the Source Control tab of the Options dialog.



Bind detection scans your project when you load it to see if the project has been previously bound to source control. If a binding is detected, you then have the option of applying the binding and committing the project to source control. Depending on the provider you are using, Lingo may search the file system and its artifacts, as well as contact and query servers, to find potential source control bindings.

When you open a Lingo project that hasn't been bound to source control before, the bind detection option is disabled for Perforce Helix Core, Subversion, and Team Foundation Server. It is enabled by default for Git. If you bind a project to source control using the Lingo interface, the option is then automatically enabled.

NOTE You can use bind detection as an alternative to importing a Lingo project. If you have received a Lingo project file (e.g., by copying it from a server, by opening it from a network location), you can simply open the file and Lingo will search for and apply existing source control bindings.

TIP Detecting source control bindings may take a considerable amount of time. It is recommended that you select only the source control providers that you use to speed up the detection process.

Disabling Providers

By default, when a project is bound to source control, the provider (Git, Perforce Helix Core, Subversion, or Team Foundation Server) is enabled. This means that the source control interface elements in Lingo are visible, and you can use them to perform various tasks (e.g., commits, synchronize changes).

Disabling a provider means that the source control interface elements are no longer shown. This does not mean you cannot use source control. As long as the provider is still *bound* to the project, you can perform source control tasks in a third-party tool outside of Lingo.

For more details and steps, see "Disabling a Team Foundation Server Provider" on page 38.

Unbinding Providers

When you unbind a provider, it means you are removing the connection altogether between the Lingo project and the local repository.

You can unbind a provider via the Project Properties dialog or the Settings view in the Source Control Explorer. Click the Unbind Provider button.

For more details and steps, See "Unbinding a Team Foundation Server Provider" on page 55.

CHAPTER 3

Process for Team Foundation Server

Certain tasks must be completed in order when using this feature.

1. Setting Up TFS (Done Outside of Lingo)
2. "Binding a Project" on the next page
3. (Other Team Members) "Importing From Source Control" on page 20
4. "Getting the Latest Version" on page 22
5. "Checking Out Source Control Files" on page 24
6. "Checking In Source Control Files" on page 26
7. "Merging Source Control Files" on page 29

Binding a Project

Use the following steps if you have already created a Lingo project and want to bind ("connect") it to Microsoft Team Foundation Server (or Azure DevOps). You can also automatically detect existing source control bindings if your project has been previously connected to TFS.

> NOTE The following steps show how to bind a project using the Lingo interface. It is also possible to bind a project outside of Lingo (e.g., using Git Bash). If you decide to do this, you should be aware of some additional aspects of source control, such as bind detection and disabling providers.

How to Bind a Project Using the Project Properties Dialog

1. Open the project.
2. Select **File > Project Properties**. The Project Properties dialog opens.
3. Select the **Source Control** tab.
4. Click **Bind Project**. The Bind Project dialog opens.
5. From the drop-down, select **Microsoft Team Foundation Server**.
6. In the **Server** field, enter the name of the computer or the IP address of the server. You can also click ... to select a "Team Project Collection." If you click this button, the Select Team Foundation Server Project Collection dialog opens, and you can do the following.
   a. To add a server, click ➕.
   b. Enter the name or URL of the server.
   c. Enter the path and port number.
   d. Select the protocol (HTTP or HTTPS).

> NOTE You may need to obtain this information from your system administrator. Click **OK** in the dialogs until you return to the main wizard page.

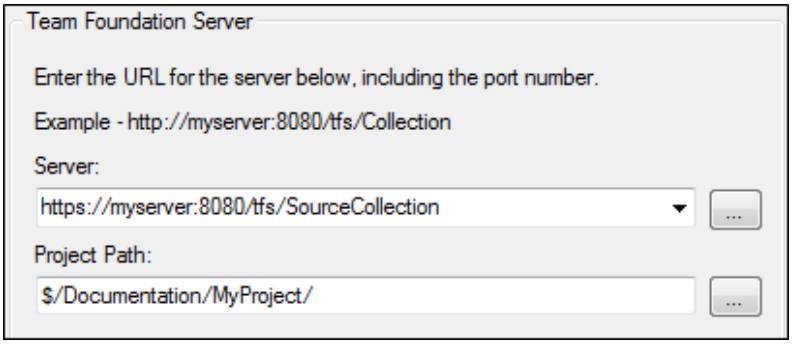
NOTE Be sure to specify the path deeper than the server name. For example:

```
.../tfs/DefaultCollection
```

or

```
.../tfs/SourceCollection
```

If you do not know this information, contact your system administrator.



NOTE If you run into issues in this dialog, chances are your user permissions do not have access to all of the collections on the TFS server. Please try to enter in the server/collection path manually.

7. Next to the **Project Path** field, click ▪▪▪.
8. Click on the Team Foundation Server folder to which you want to bind the Lingo project.
9. Click **OK**.
10. (Optional) In the **Comment** field, you can enter any internal comments.
11. (Optional) If you want the files in the project to be checked out when you are finished, click **Keep files checked out**.
12. In the Bind Project dialog, click **OK**.
13. If the Log In dialog opens, complete the **User name** and **Password** fields and click **OK**. Copies of the Lingo files are created and added to the folder you specified.
14. In the Project Properties dialog, click **OK**. The project is connected to Team Foundation Server, and you can now check files in and out as necessary.

How to Bind a Project Using the Explorer

1. Open the project.
2. Select **View > Source Control Explorer**. The Source Control Explorer opens.
3. From the drop-down or the Home pane, select **Settings**. The Settings pane opens.
4. Click **Bind**. The Bind Project dialog opens.
5. From the drop-down, select **Microsoft Team Foundation Server**.
6. In the **Server** field, enter the name of the computer or the IP address of the server. You can also click ... to select a "Team Project Collection." If you click this button, the Select Team Foundation Server Project Collection dialog opens, and you can do the following.
   a. To add a server, click ➕.
   b. Enter the name or URL of the server.
   c. Enter the path and port number.
   d. Select the protocol (HTTP or HTTPS).

> NOTE You may need to obtain this information from your system administrator. Click **OK** in the dialogs until you return to the main wizard page.

> NOTE Be sure to specify the path deeper than the server name. For example:
>
> `.../tfs/DefaultCollection`
>
> If you do not know this information, contact your system administrator.

> NOTE If you run into issues in this dialog, chances are your user permissions do not have access to all of the collections on the TFS server. Please try to enter in the server/collection path manually.

7. Next to the **Project Path** field, click ...
8. Click on the Team Foundation Server folder to which you want to bind the Lingo project.
9. Click **OK**.

10. (Optional) In the **Comment** field, you can enter any internal comments.
11. (Optional) If you want the files in the project to be checked out when you are finished, click **Keep files checked out**.
12. In the Bind Project dialog, click **OK**.
13. If the Log In dialog opens, complete the **User name** and **Password** fields and click **OK**. Copies of the Lingo files are created and added to the folder you specified. The project is connected to Team Foundation Server, and you can now check files in and out as necessary.

What’s Noteworthy?

TIP If you are having difficulty binding your project, try binding to a brand new directory in your source control provider. You should also ensure that the directory on your local machine (and its parent directories) is not already mapped to source control, as this may cause issues with binding.

NOTE You can also bind a new Lingo project to source control while creating it.

Importing From Source Control

This focuses on importing a Lingo project from source control. You might use this method, for example, if you are working on a multi-author project and another member of the team has placed the Lingo project in Microsoft Team Foundation Server.

How to Import a Project From Team Foundation Server

1. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select File > New Project > Import Project.
   - Source Control Explorer From the View ribbon, open the Source Control Explorer. From the drop-down, select the Home pane. Click Import Project.

   The Import Project from Source Control Wizard dialog opens.
2. From the drop-down, select Microsoft Team Foundation Server.
3. In the Server field, enter the name of the computer or the IP address of the server. You can also click ... to select a "Team Project Collection." If you click this button, the Select Team Foundation Server Project Collection dialog opens, and you can do the following.
   a. To add a server, click +. The Add Team Foundation Server dialog opens.
   b. Enter the name or URL of the server.
   c. Enter the path and port number.
   d. Select the protocol (HTTP or HTTPS).

> NOTE You may need to obtain this information from your system administrator. Click OK in the dialogs until you return to the main wizard page.

> NOTE If you run into issues in this dialog, chances are your user permissions do not have access to all of the collections on the TFS server. Please try to enter in the server/collection path manually.

4. Next to the **Team Project** field, click ···. If the Log In dialog opens, complete the **User name** and **Password** fields.
5. Click on the Team Foundation project.
6. Click **Next**.
7. Next to the **Project file** field, click **Browse**. The Browse Source Control Files dialog opens.
8. Find and click on the Lingo project file that you want to import. (You may need to log in with your user name and password.)
9. Click **OK**.
10. Click **Next**.
11. In the **Project name** field, the name of the project being imported is displayed. It is recommended that you leave the name as it is, especially if you are working with other authors on the project. However, you can enter a different project name if you want.
12. In the **Project folder** field, either accept the default location for the new project or click ··· to browse for and select a folder.
13. Click **Finish**. The project is imported and loaded into Lingo

Getting the Latest Version

After you bind a Lingo project to a source control application, you can get the latest version of any of the source control files. When you do this, you are copying the most current files stored in the source control application to your local Lingo project without necessarily checking out the files. This means that the "Read Only" designation will remain associated with the files until you check them out.

How to Get the Latest Version of Source Control Files

1. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select Source Control > Get Latest Version (for selected files) or Source Control > Get Latest Version All (for all files in the project).
   - Right-Click In the File List window pane, right-click the file you want to get and select Source Control > Get Latest Version.
   - Source Control Explorer With the Pending Changes pane open, right-click the file you want to update and select Get Latest Version.
2. If the local and server files are the same, a message tells you so. Click OK.

   If the local file is different from the file on the server, the Resolve Conflicts dialog opens. If you want to accept all of the differences between the local and server files, thus merging them, click Auto Merge All. If you want to review the differences in the files side by side and resolve each conflict, click Resolve.

What's Noteworthy?

NOTE By default, when you open a project that is bound to source control, a message automatically asks if you want to get the latest version of files. However, you can disable this prompt in the Options dialog (File > Options).

NOTE If you update your read-only files with read-only files and you do not perform a check out, the files will appear to be submitted without actually updating the server-side version.

Checking Out Source Control Files

When you need to work on any of the Lingo project files, you can check them out. Checking out files means to copy the latest source control files to your local Lingo project and remove the "Read Only" designation from them so that you can edit the files. A red check mark is displayed next to each file that is checked out.

You can check files out automatically or manually.

How to Check Out Files From Source Control Automatically

1. Select **File > Options**. The Options dialog opens.
2. Select the **Source Control** tab.

   > NOTE This tab will not be visible if your project is not yet bound to source control. See "Binding a Project" on page 14.

3. Click the check box **Automatically check file out from source control when saving file changes**. A check mark in the box indicates that the feature is enabled.
4. (Optional) If you want files that are automatically checked out to be locked as well, select **Lock files when automatically checking out**.

   Locking files means that other users cannot check in those same files when you have them checked out. You might use this feature to prevent conflicting changes from occurring and having to merge files as a result.

5. Click **OK**.

   Whenever you open a file that is not checked out, make changes, and save, not only will your changes be saved but the file will automatically be checked out to you.

How to Check Out Files From Source Control Manually

1. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select Source Control > Check Out (for selected files) or Source Control > Check Out All (for all files in the project).
   - Right-Click In the File List window pane, right-click the file you want to check out and select Source Control > Check Out (for selected files) or Source Control > Project > Check Out All (for all files in the project).

   The Check Out dialog opens. The selected files are listed with check boxes next to them.
2. Make sure to click the check box next to each file you want to check out so that it contains a check mark.
3. (Optional) If you want the files to be locked as well, select Lock files.

   Locking files means that other users cannot check in those same files when you have them checked out. You might use this feature to prevent conflicting changes from occurring and having to merge files as a result.
4. Click Check Out.

Checking In Source Control Files

When you are finished editing files, you can check them in to source control. Checking in a file overwrites the old copy of the file in the database with the new one from your local machine. Even if others are not working on a file, it is a good idea to periodically check in files for a backup in source control.

How to Check In Files to Source Control

1. Do one of the following, depending on the part of the user interface you are using:
   - Pending Changes Window Pane From the Source Control ribbon, open the Pending Changes window pane. Select the files in the window pane that you want to check in, and in the local toolbar click .
   - RibbonSelect Source Control > Check In (for selected files) or Source Control > Check In All (for all files in the project).
   - Right-Click In the File List window pane, right-click the file you want to check in and select Source Control > Check In (for selected files) or Source Control > Project > Check In All (for all files in the project).

   The Check In dialog opens. The selected files are listed with check boxes next to them.
2. (Optional) Enter an optional comment tied to the check-in. This enables you to keep an audit trail for a file. The comment can then be viewed from the History dialog, which can be accessed from the Source Control Explorer, the Source Control ribbon, or the Source Control button .
3. (Optional) If you want to see all files with pending changes (rather than only those you selected), click .
4. Make sure to click the check box next to each file you want to check in so that it contains a check mark.
5. If you want to keep the files checked out of source control, select Keep Checked Out. Doing this will overwrite the source control copies of the files so that they have the latest changes, but it lets you continue working on the files. This is a useful option if you are the only author working on the files in question.
6. Click Check In.

If no other users have also made changes to the file and checked it in while you were working on it, your version of the file is checked in.

However, if that is not the case, the Resolve Version Conflict dialog opens to let you know that another user has already checked in the file with changes. You can merge the files automatically if there are no conflicting changes (i.e., changes do not occur in the same location in the file). If there are conflicting changes, you can use the Merge Changes dialog to determine how changes are merged. See "Merging Source Control Files" on page 29.

How to Check In Files to Source Control Using the Explorer

1. Select **View > Source Control Explorer**. The Source Control Explorer opens.
2. From the drop-down or the Home pane, select **Pending Changes**.

   The Pending Changes pane opens. Files that will be checked in are listed under **Included Changes**, and files that will not be checked in are listed under **Excluded Changes**. You can identify changed files because they say **[edit]** next to the file name.
3. (Optional) In the **Comment** field, enter an optional comment tied to the check-in. This enables you to keep an audit trail for a file. The comment can then be viewed from the History dialog, which can be accessed from the Source Control Explorer, the Source Control ribbon, or the Source Control button .
4. (Optional) If you want to select the files or folders that you include in the check-in, right-click a file or folder and select one of the following options from the context menu.
   - **Exclude** Excludes the selected file from the check-in
   - **Exclude Unselected** Excludes all unselected files from the check-in
   - **Include** Includes the selected file in the check-in
   - **Include Unselected** Includes all unselected files in the check-in
5. Click **Check In Included** to check in all of the files in the Included Changes list.

   If no other users have also made changes to the file and checked it in while you were working on it, your version of the file is checked in. The Messages pane opens and displays a list of files that were checked in.

   However, if that is not the case, the Resolve Version Conflict dialog opens to let you know that another user has already checked in the file with changes. You can merge the files

automatically if there are no conflicting changes (i.e., changes do not occur in the same location in the file). If there are conflicting changes, you can use the Merge Changes dialog to determine how changes are merged. See "Merging Source Control Files" on the next page.

Merging Source Control Files

There may be times when you need to merge changes from different authors when checking in a file. The merge occurs automatically if there are no conflicting changes (i.e., changes do not occur in the same location in the file). If there are conflicting changes, a dialog opens, allowing you to determine how changes are merged.

How to Merge Source Control Files

1. Go through the process of getting the latest version of files from source control or checking in files. If your local copy of the file is different from the server copy (e.g., another author has already checked in the same file), the Resolve Conflicts dialog opens.
2. Click Auto Merge All. If changes from the other author do not conflict with your changes, this will merge all changes. A message lets you know that a backup of your local copy has been made. This lets you restore that file if you do not want to keep the merged version. You do not need to complete the rest of the steps below.

   However, if your changes conflict with those from another author, a message displays to tell you. In this case, continue with the next step.
3. Click OK on the conflict message.
4. In the Resolve Conflicts dialog click Resolve. The Resolve Version Conflict dialog opens. From this dialog, you can choose from the following options.
   - Merge changes for me Automatically merges changes within the same file that are not part of the same element. If changes have been made to the same element (e.g., the same <p> tag or <h1> tag), Lingo will display a prompt to merge the changes using the merge tool.
   - Merge changes in merge tool Opens a merging interface, which lets you see exactly what changes were made and choose which to keep.
   - Undo my local changes Automatically removes your changes and keeps changes from others.
   - Discard external changes Automatically removes changes from others and keeps your changes.
5. If you selected the option to use the merge tool, the Merge Changes dialog opens. Use this dialog to view and select changes. You can take actions in the following ways.

- **Click a change** You can click a change on either the remote or local side. This lets you select a particular change. Use the key at the top of the merge changes dialog, as well as the color coding on the local and server sides, to determine if a change has been added (new), deleted, changed, or moved.

  When you select a change, the change you selected will display with a solid colored background, and the conflicting change will display with a striped background. If you select the other change, the background shading will switch.

- **Type content** If you want to use your changes as well as those from another author, and even tweak the paragraph a bit more, you can click in the area at the bottom of the dialog and simply type content.

- **Previous/next conflict** When you are finished resolving the first conflict, you can use the "Previous Conflict" and "Next Conflict" buttons at the bottom of the dialog to work on other conflicts in the file.

> **NOTE** If you selected "Merge as Text" in the local toolbar and are working in the code, you can click on text with a hatched background to keep the change in it. After you click on text with a hatched background, the hatched lines are removed, leaving a solid color.

6. After all conflicts have been resolved, a message lets you know that a backup of your local copy has been made. This lets you restore that file if you do not want to keep the merged version. Click **OK**.

CHAPTER 4

Other Activities for Team Foundation Server

In addition to the many features already covered in this guide, there are many more tasks related to source control that you can perform in Lingo.

This chapter discusses the following:

Binding a Project to Microsoft Visual Studio Team Services 32
Deleting Source Control Files 36
Disabling the Get Latest Prompt for Source Control 37
Disabling a Team Foundation Server Provider 38
Disconnecting From Source Control 40
Enabling Source Control Status Checks 43
Importing From Microsoft Visual Studio Team Services 44
Rolling Back to an Earlier Version of a File 47
Setting Color Options for Project File Differences 53
Unbinding a Team Foundation Server Provider 55
Undoing a Checkout of Source Control Files 57
Viewing Differences in Source Control Files 58
Viewing Files That are Checked Out 63
Viewing the History of Source Control Files 64

Binding a Project to Microsoft Visual Studio Team Services

Lingo includes Microsoft Team Foundation Server support for Microsoft Visual Studio Team Services (VSTS). This is also known as Visual Studio Online, as well as Microsoft DevOps on Azure. If you manage your source control projects with Microsoft's online version of Visual Studio, you can bind your projects to Lingo and perform check-ins and check-outs of files.

Use the following steps if you have already created a Lingo project and want to bind ("connect") it to the Microsoft VSTS repository using Microsoft Team Foundation Server.

How to Bind a Project to Source Control

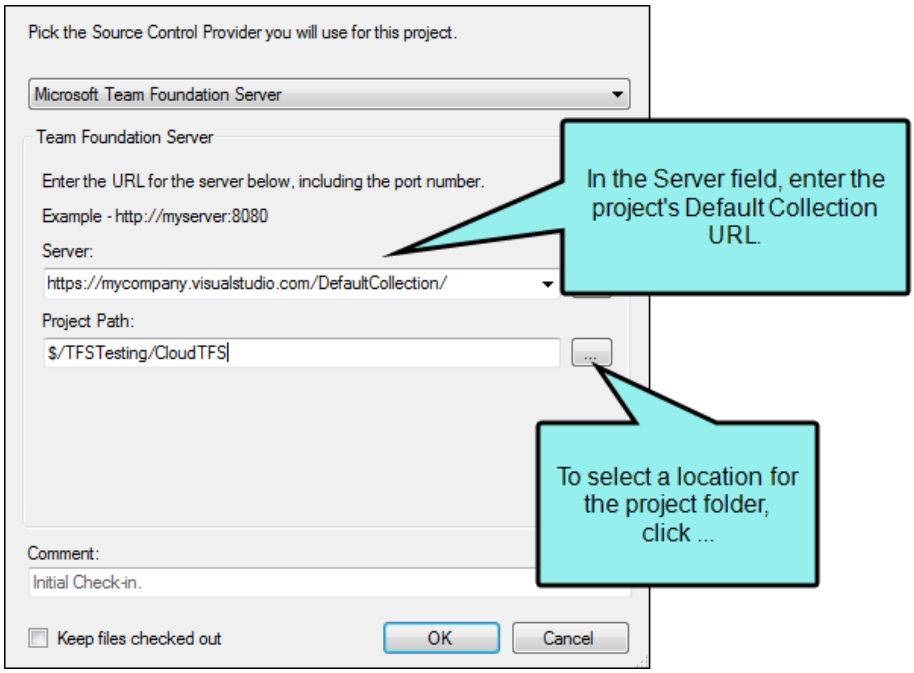
1. Open the project.
2. Select **File > Project Properties**. The Project Properties dialog opens.
3. Select the **Source Control** tab.
4. Click **Bind Project**. The Bind Project dialog opens.
5. From the drop-down, select **Microsoft Team Foundation Server**.
6. In the **Server** field, enter the URL of the project's "Default Collection."

> NOTE When binding to VSTS project using TFS, you must manually enter the URL in the **Server** field. You can only browse for a server location when binding to a local project using TFS.

> NOTE You may need to obtain this information from your system administrator. Click **OK** in the dialogs until you return to the main wizard page.

> **NOTE** If you run into issues in this dialog, chances are your user permissions do not have access to all of the collections on VSTS. Please try to enter in the server/collection path manually.

7. Next to the **Project Path** field, click ⊡. You may need to log in to Microsoft VSTS. If the Log In dialog opens, complete the **User name** and **Password** fields.



The Browse Source Control Folders dialog opens.

8. Click on the Team Foundation Server folder to which you want to bind the Lingo project.
9. Click **OK**.
10. (Optional) In the **Comment** field, you can enter any internal comments.
11. (Optional) If you want the files in the project to be checked out when you are finished, click **Keep files checked out**.
12. In the Bind Project dialog, click **OK**.
13. If the Log In dialog opens, complete the **User name** and **Password** fields and click **OK**. Copies of the Lingo files are created and added to the folder you specified.
14. In the Project Properties dialog, click **OK**. The project is connected to Team Foundation Server, and you can now check files in and out as necessary.

How to Bind a Project to Source Control Using the Explorer

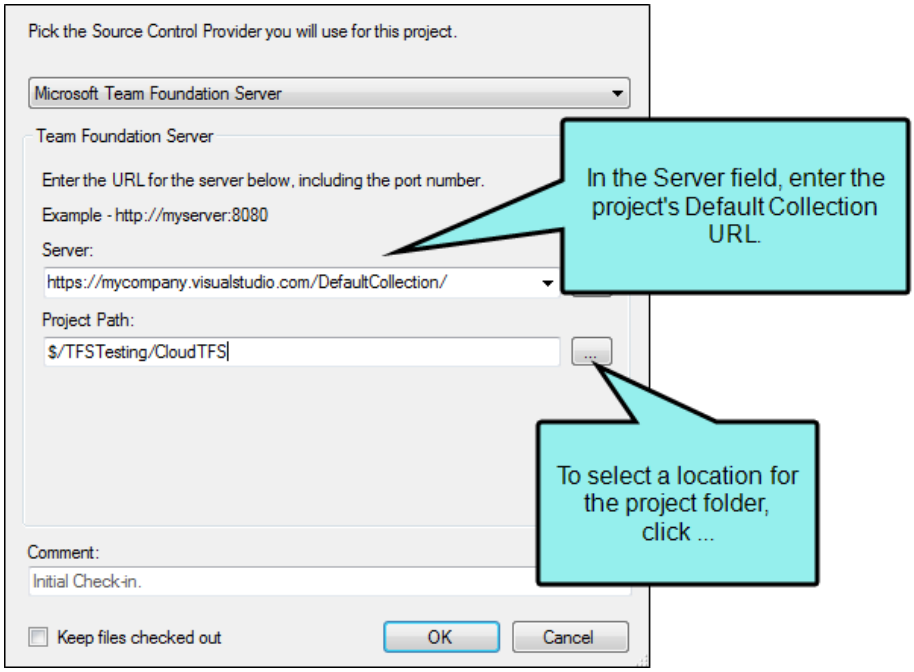
1. Open the project.
2. Select **View > Source Control Explorer**. The Source Control Explorer opens.
3. From the drop-down or the Home pane, select **Settings**. The Settings pane opens.
4. Click **Bind**. The Bind Project dialog opens.
5. From the drop-down, select **Microsoft Team Foundation Server**.
6. In the **Server** field, the URL of the project's "Default Collection."

   > NOTE When binding to VSTS project using TFS, you must manually enter the URL in the **Server** field. You can only browse for a server location when binding to a local project using TFS.

   > NOTE You may need to obtain this information from your system administrator. Click **OK** in the dialogs until you return to the main wizard page.

   > NOTE If you run into issues in this dialog, chances are your user permissions do not have access to all of the collections on VSTS. Please try to enter in the server/collection path manually.

7. Next to the **Project Path** field, click ⋯. You may need to log in to Microsoft VSTS. If the Log In dialog opens, complete the **User name** and **Password** fields.



The Browse Source Control Folders dialog opens.

8. Click on the Team Foundation Server folder to which you want to bind the Lingo project.
9. Click **OK**.
10. (Optional) In the **Comment** field, you can enter any internal comments.
11. (Optional) If you want the files in the project to be checked out when you are finished, click **Keep files checked out**.
12. In the Bind Project dialog, click **OK**.
13. If the Log In dialog opens, complete the **User name** and **Password** fields and click **OK**. Copies of the Lingo files are created and added to the folder you specified.The project is connected to Team Foundation Server, and you can now check files in and out as necessary.

Deleting Source Control Files

You can delete a file that is bound to source control. This also removes the file from Team Foundation Server.

How to Delete a File

1. In the File List window pane, select the relevant file(s).
2. On your keyboard press DELETE.
3. The Delete dialog opens. Select the bound files you want to delete.
4. Click Delete.
5. Check in your changes in order for the deletions to be reflected in remote projects.

Disabling the Get Latest Prompt for Source Control

By default, when you open a project that is bound to source control, a message automatically asks if you want to get the latest version of files. However, you can disable this prompt in the Options dialog. Therefore, in the future when you open the project you will no longer see the message, and the project will open without replacing any local files with the latest ones from source control.

How to Disable the Get Latest Prompt for Source Control

1. Select **File > Options**. The Options dialog opens.
2. Select the **Source Control** tab.

   > NOTE This tab will not be visible if your project is not yet bound to source control. See "Binding a Project" on page 14.

3. Click the check box **Do not prompt to get latest when opening source control bound projects**.
4. Click **OK**.

Disabling a Team Foundation Server Provider

By default, when a project is bound to source control, the provider (Git, Perforce Helix Core, Subversion, or Team Foundation Server) is enabled. This means that the source control interface elements in Lingo are visible, and you can use them to perform various tasks (e.g., commits, synchronize changes).

Disabling a provider means that the source control interface elements are no longer shown. This does not mean you cannot use source control. As long as the provider is still *bound* to the project, you can perform source control tasks in a third-party tool outside of Lingo.

How to Disable a Provider

Use this method if you want to disable a provider in just one project, rather than many projects.

1. Do one of the following, depending on the part of the user interface you are using:
   - Project Properties Select File > Project Properties.
   - Source Control Explorer Select View > Source Control Explorer. Then, in the window pane, click Settings.
2. Click Enabled to remove the check mark.
3. If you used the Project Properties dialog, click OK.

NOTE If you disable a Git provider, the local repository will continue to track your changes in case you later decide to enable the provider once again.

If you disable one of the other providers (Perforce Helix Core, Subversion, Team Foundation Server), your changes after that point will not be tracked. Therefore, if you later enable the provider again, it will not have recorded any changes made since the time that you disabled it.

NOTE When you disable a provider, that information is written to the registry on your computer.

NOTE If you disable a provider, but then perform one of the following actions in the Lingo interface, the provider will automatically become enabled once again.

- Bind an existing project
- Bind a new project
- Import a project from source control

NOTE Having a provider enabled in Lingo does not interfere with your workflow if you are performing source control actions exclusively outside of Lingo. Even if a provider is enabled in the project and the source control user interface elements are visible, this does not mean Lingo is automatically performing any source control actions with your files. It simply means Lingo is recognizing the binding, so it reflects your activities (e.g., the Pending Changes window is populated when you make edits in topics). However, if you prefer not to see any of this in Lingo, you can disable the provider.

Disconnecting From Source Control

There may be times that you need to disconnect from source control to work offline. You can disconnect from Microsoft Team Foundation Server and reconnect at any time.

Disconnecting from source control is beneficial because it lets you modify files when you would otherwise not have access to the source control system (e.g., you are out of the office with your laptop or you do not want to access source control over VPN). It also provides a fallback offline status in the event that the network is disconnected while you are working, so you are able to continue working on the files you have checked out until the network connection is restored.

How to Disconnect From TFS

When you want to work offline, you can disconnect from source control.

1. Check out the files you will need to work on while disconnected. See "Checking Out Source Control Files" on page 24.
2. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select **Source Control > Disconnect**.
   - **Right-Click** If you have the File List open, right-click on any file and select **Source Control > Project > Disconnect**.
3. A confirmation dialog appears. Click **Yes**. You will be disconnected from source control.

> **NOTE** When you disconnect from source control, you are not able to see the source control status of files or access source control functions.

NOTE If you attempt to modify a file that is not checked out, you will receive an error when you try to save the file. To avoid this, be sure you have all necessary files checked out before disconnecting from source control.

NOTE If you make a change to a file's properties (e.g., delete) while disconnected from source control, your changes may not be preserved when you reconnect to the network. To prevent errors, it is recommended that you do not make these kinds of changes until you reconnect to source control.

How to Reconnect to TFS

When you are finished working offline, you can reconnect to source control.

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select **Source Control > Reconnect**.
   - **Right-Click** If you have the File List open, right-click on any file and select **Source Control > Project > Reconnect**.
2. Check in the files you previously checked out. See "Checking In Source Control Files" on page 26.

What's Noteworthy?

NOTE Your current network connection status is indicated in the lower right corner of the Lingo interface. If you are connected you will see Connected; if you are disconnected you will see Disconnected.

Enabling Source Control Status Checks

If you are using source control integration in Lingo, you can check for frequent status changes automatically. You can specify the number of minutes and seconds when you want Lingo to ping the source control repository and get status changes for files that have been checked out, checked in, moved, deleted, etc. The upside of this feature is that you can ensure that the source control status information is always up to date. The downside is that you may experience slower performance due to this constant communication over the network.

How to Enable Source Control Status Checks

1. Select **File > Options**. The Options dialog opens.
2. Select the **Source Control** tab.

   NOTE This tab will not be visible if your project is not yet bound to source control. See "Binding a Project" on page 14.

3. Click the check box **Enable background status checks**. A check mark in the box indicates that the feature is enabled.
4. Enter the number of minutes and or seconds between each status update.
5. Click **OK**.

NOTE If you elect to disable this feature (disabled is the default setting), you can manually check for status updates by refreshing the Pending Changes window pane or Source Control Explorer. See "Viewing Files That are Checked Out" on page 63.

Importing From Microsoft Visual Studio Team Services

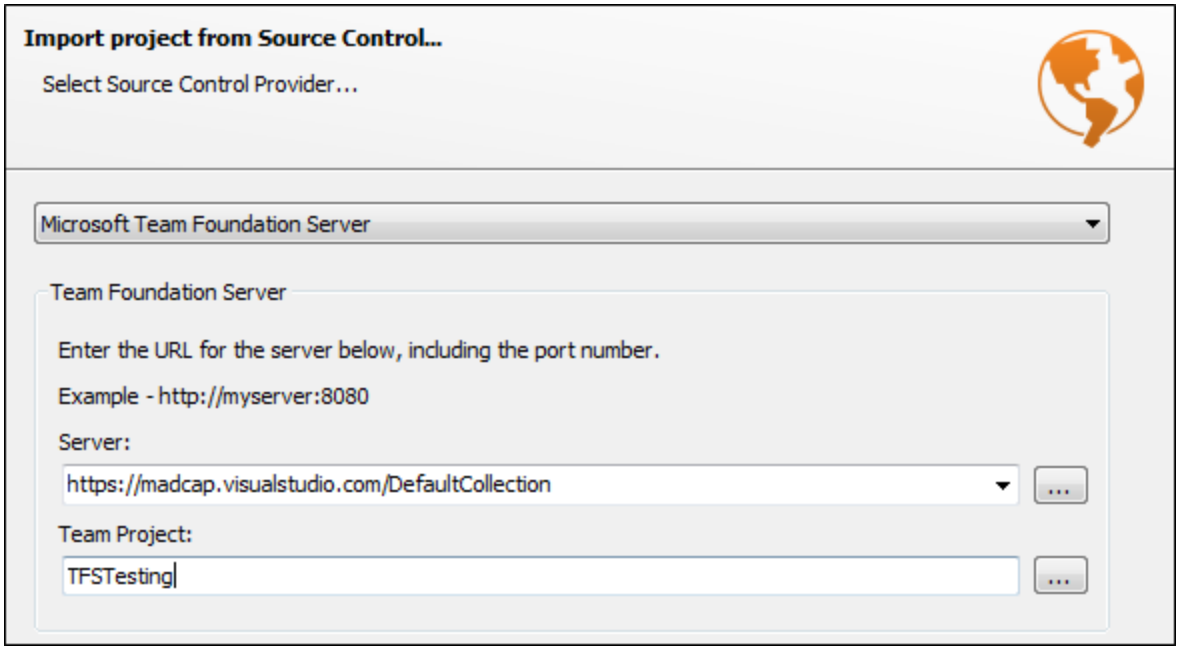
Lingo includes Microsoft Team Foundation Server support for Microsoft Visual Studio Team Services (VSTS). If you manage your source control projects with Microsoft's online version of Visual Studio, you can import these projects to Lingo.

Use the following steps if you have already created a Lingo project and want to import it to Lingo from Microsoft VSTS.

How to Import a Project From Microsoft Visual Studio Team Services

1. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select File > New Project > Import Project.
   - Source Control Explorer From the View ribbon, open the Source Control Explorer. From the drop-down, select the Home pane. Click Import Project.

   The Import Project from Source Control Wizard dialog opens.
2. From the drop-down, select Microsoft Team Foundation Server.
3. In the Server field, enter the URL of the project's "Default Collection."



> **NOTE** When binding to a VSTS project using TFS, you must manually enter the URL in the **Server** field. You can only browse for a server location when binding to a local project using TFS.

> **NOTE** You may need to obtain this information from your system administrator. Click **OK** in the dialogs until you return to the main wizard page.

> **NOTE** If you run into issues in this dialog, chances are your user permissions do not have access to all of the collections on the TFS server. Please try to enter in the server/collection path manually.

4. Next to the **Team Project** field, click ⋯. You may need to log in to Microsoft VSTS. If the Log In dialog opens, complete the **User name** and **Password** fields.

   The Browse Team Foundation Server dialog appears.

5. Click on the Team Foundation project.
6. Click **OK**.

7. Click **Next**.
8. Next to the **Project file** field, click **Browse**. The Browse Source Control Files dialog opens.
9. Find and click on the Lingo project file (LIPRJ) that you want to import. (You may need to log in with your user name and password.)
10. Click **OK**.
11. Click **Next**.
12. In the **Project name** field, the name of the project being imported is displayed. It is recommended that you leave the name as it is, especially if you are working with other authors on the project. However, you can enter a different project name if you want.
13. In the **Project folder** field, either accept the default location for the new project or click ... to browse for and select a folder.
14. Click **Finish**. The project is imported and loaded into Lingo.

Rolling Back to an Earlier Version of a File

One of the benefits of Lingo's integrated source control is that you can view the history and differences for a particular file. You can view code and content differences between two source control versions of the same file. This is useful if you need to roll back to an earlier version of a file.

> ☆ EXAMPLE You have been working on translating a particular file for a few days. Each day you check out the latest version of that file from source control, make your changes, and check the file back in to source control at the end of the day. At a certain point, you determine that you need to "roll back" to an earlier version of the file, using it to replace the latest version. Therefore, you use this feature to view the highlighted differences between the current version and an older version of the file. Once you have identified the older version that you want to use, you can perform a "get" of that version.

How to Roll Back to an Earlier Version of a File

1. In the Source Control Explorer or File List, select the relevant file(s).

   OR

   Open a file.

2. Do one of the following, depending on the part of the user interface you are using:

   - Ribbon Select Source Control > View History.
   - Right-Click If you have the File List open, right-click the file you want to roll back and select Source Control > View History.
   - Source Control Explorer With the Pending Changes pane open, right-click the file you want to roll back and select View History.

   The History dialog opens.

3. (Optional) View the differences between two versions of the file. This may help you decide which version of the file you want to retrieve. (Another way is to look at the dates for each version in the History dialog.)

   To view the differences, follow these steps.

   a. Select the first file version from the list.

   b. Hold the CTRL key and select the second file version from the list.

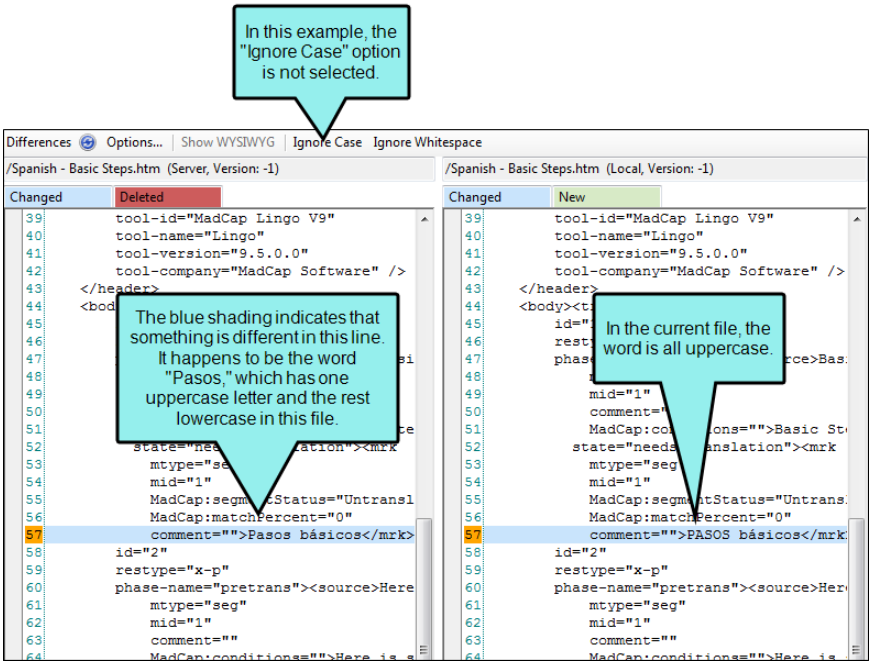
   c. Click Show Differences. The Differences Editor opens to the right, showing content from the backup file on the left and the current version of the file on the right.

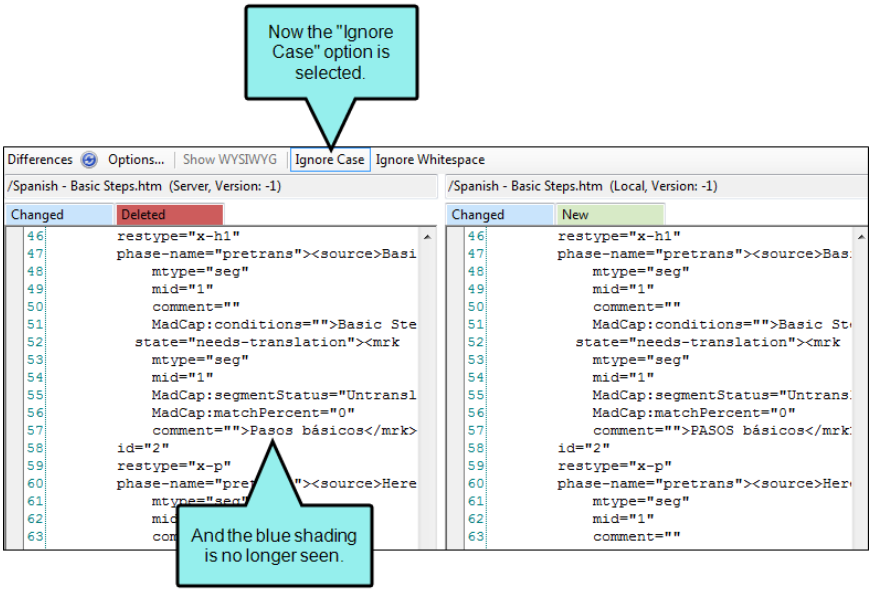
   d. In the local toolbar of the Differences Editor, you can click any of the following to make adjustments.

      - Options Click this to open the File Differences dialog, which lets you change the colors used to display content differences between the files.

- **Ignore Case** Click this to ignore changes in case when viewing differences.

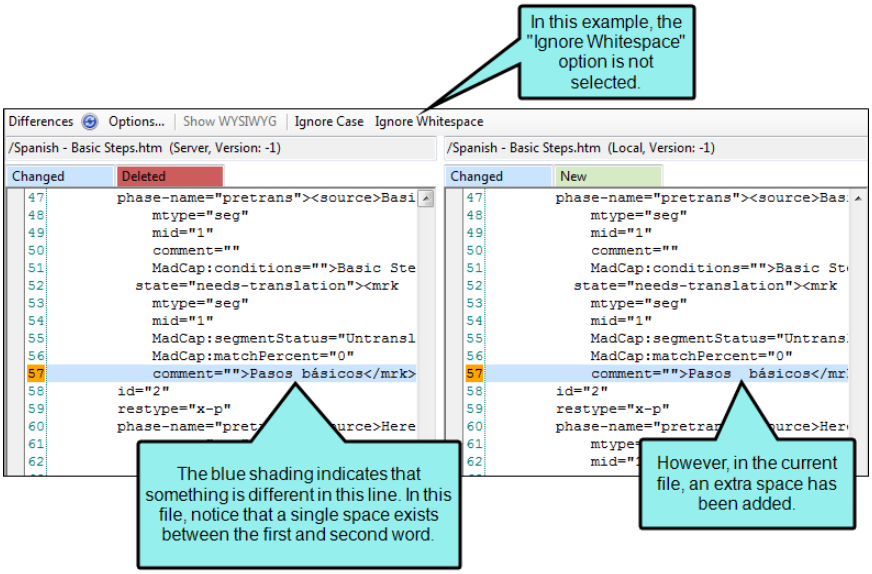
> ☆ EXAMPLE If a word is not capitalized in the original file but it is capitalized in the current file, this option does not highlight those differences.




Now the "Ignore Case" option is selected.
Differences
Options...
Show WYSIWYG
Ignore Case
Ignore Whitespace
/Spanish - Basic Steps.htm (Server, Version: -1)
/Spanish - Basic Steps.htm (Local, Version: -1)
Changed
Deleted
Changed
New
And the blue shading is no longer seen.

- **Ignore Whitespace** Click this to ignore whitespace when viewing differences.

☆ EXAMPLE A segment is identical in both files, except for an extra space that was added within the segment in one of those files. If you click this option, that difference is not highlighted.



e. When you are finished viewing the differences, close the window pane.

4. In the History dialog, select the version of the file to which you want to roll back.
5. Click **Get Selected Version**. That file is downloaded from source control and replaces the local copy of the file in your project.
6. In the History dialog, click **Close**.

Setting Color Options for Project File Differences

If you are using Lingo's integrated source control features, you can view differences between files in various ways. One way is to view file differences between a local version of a Lingo project and the source control version.

When viewing file differences between a local version of a Lingo project and the source control version, you can select color options to display the files. Color coding makes it easier to discern where differences between files occur.

> ☆ EXAMPLE By default the files that are included only in your local copy are displayed as green in the Differences Editor, and the files that are included only in source control are displayed in red. You can use this dialog to change the local-only files to blue and the source control–only files to yellow.

How to Set Color Options for Project File Differences

1. In the Source Control Explorer or File List, select the relevant file(s).
2. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select Source Control > Show Differences.
   - Right-Click If you have the Source Control Explorer or File List open, right-click the file you want to view and select Source Control > Show Differences.

   The Differences Editor opens.
3. In the local toolbar of the Differences Editor, click Options. The File Differences dialog opens.
4. Change the text or background color for any of the difference types. To do this, simply click in the appropriate Text or Background cell and select Pick Color. In the Color Picker dialog, choose the new color.
5. Click OK.

Unbinding a Team Foundation Server Provider

When you unbind a provider, it means you are removing the connection altogether between the Lingo project and the local repository.

How to Unbind Using the Project Properties Dialog

1. Open the project.
2. Select File > Project Properties.
3. Select the Source Control tab.
4. Click Unbind Provider. (If your project is dual-bound, you will also see a section for the other binding.)
5. Click OK.

How to Unbind Using the Source Control Explorer

1. Open the project.
2. Select View > Source Control Explorer.
3. From the drop-down or the Home pane, select Settings. The Settings pane opens.
4. Click Unbind Provider. (If your project is dual-bound, you will also see a section for the other binding.)
5. Click Yes.

What’s Noteworthy?

NOTE You can also disable a provider, which retains the binding but hides source control elements from the user interface.

Undoing a Checkout of Source Control Files

If you have files checked out from source control but do not want them checked out anymore, you can use the "Undo Check Out" option instead of checking in the files. While checking in the file would save your changes to source control, undoing a checkout returns the files to their previous state and does not check in any of your changes to source control.

How to Undo a Checkout of Source Control Files

1. In the Source Control Explorer or File List, select the relevant file(s).
2. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select Source Control > Undo Check Out (for selected files) or Source control > Undo Check Out All (for all files in the project).
   - Right-Click In the File List, right-click the files you want to undo and select Source Control > Undo Check Out (for selected files) or Source Control > Project > Undo Check Out All (for all files in the project).

   The Undo Check Out dialog opens. The selected files are listed with check boxes next to them.
3. (Optional) If you want to see all files with pending changes (rather than only those you selected), click .
4. Make sure to click the check box next to each appropriate file so that it contains a check mark.
5. Click Undo Check Out.

Viewing Differences in Source Control Files

One of the benefits of Lingo's integrated source control is that you can view the history and differences for a particular file.

Ways to View Differences Between Files

You can view differences between files in the following ways:

- Two Versions of Same Source Control File (History/Roll Back) You can view code and content differences between two source control versions of the same file. This is useful if you need to roll back to an earlier version of a file.

  > ☆ EXAMPLE You have been working on translating a particular file for a few days. Each day you check out the latest version of that file from source control, make your changes, and check the file back in to source control at the end of the day. At a certain point, you determine that you need to "roll back" to an earlier version of the file, using it to replace the latest version. Therefore, you use this feature to view the highlighted differences between the current version and an older version of the file. Once you have identified the older version that you want to use, you can perform a "get" of that version.

- Local Versus Source Control Version of a File You can view code and content differences between the local version of a file and the source control version of that file.

  > ☆ EXAMPLE You check out a topic from source control and then make changes to some of the translations in your local copy of that file. You save your changes. Later that day, you want to revisit the new translations, but you cannot remember exactly which translations you added and which were there before. Therefore, you use this feature to highlight the text differences between your local checked-out version of the file and the version stored in the source control application. The new text is highlighted on the side displaying the local version of the file.

- **Local Versus Source Control Version of All Files in a Folder** You can view file differences between the local version of the files in a folder and the source control version. Most likely, you will use this to view all of the differences between the local files and source control files in a specific content folder.

> ☆ EXAMPLE You are working on a large Lingo project. During the course of the day, you edit several files in the project. At the end of the day, you check in most of your files, but forget to check in a few of them. Later, you realize that you missed a few files, and now those files are out-of-date. Therefore, you use this feature to see the file-level differences between your local copy of the folder in the File List and the source control copy. The differences are color coded, so you can easily identify the files in question. (By default, the files that are included only in your local copy are green, and the files that are included only in source control are red.)

How to View Differences Between Two Versions of the Same Source Control File

1. In the Source Control Explorer or File List, select the relevant file(s).
2. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select **Source Control > View History**.
   - **Right-Click** In the File List, select the file(s) you want to view. In the local toolbar of the File List, click , then select **View History** .

   The History dialog opens.
3. From the list, select the first file version that you want to compare.
4. Hold the **CTRL** key and select the second file version from the list.
5. Select **Show Differences**.The Differences Editor opens.
6. (Optional) In the Differences Editor, use the buttons in the local toolbar to customize the information shown in the editor.
7. When you are finished viewing the differences, close the window.

How to View Differences Between the Local Version of a File and the Source Control Version

1. In the Source Control Explorer or File List, select the relevant file(s).
2. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select Source Control > Show Differences.
   - Right-Click In the File List window pane, right-click the file(s) you want to view and select Source Control > Show Differences.

   The Differences Editor opens.
3. (Optional) In the Differences Editor, use the buttons in the local toolbar to customize the information shown in the editor.
4. When you are finished viewing the differences, close the window.

How to View Differences Between the Local Version of All Files in a Folder and the Source Control Version

1. In the Content Explorer or Project Organizer, select the relevant folder.
2. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select Source Control > Show Differences.
   - Right-Click Right-click the folder you want to view and select Source Control > Show Differences.

   The Differences Options dialog opens.
3. (Optional) Use this dialog to specify the type of information that you want to see in the Differences Editor. (You can also choose these options from the local toolbar of the Differences Editor.)
   - Show files that are only in the Left folder Displays the files on the left side of the Differences Editor. The left side is used to show the local copies of your project files.
   - Show files that are only in the Right folder Displays the files on the right side of the Differences Editor. The right side is used to show the source control copies of your project files.
   - Show files that are different in both folders Displays the files where differences occur between the local copy and source control copy of the project. For example, the left side might display files that you have created in your local copy but have not yet been added to source control.
   - Show files that are the same in both folders Displays the files that are the same in the local copy as they are in the source control copy.
   - Recursive Displays files recursively. In other words, if you have files contained within folders, selecting this button will ensure that you see all of the files, not just the folders.
4. Click OK. The Differences Editor opens.
5. (Optional) In the Differences Editor, use the buttons in the local toolbar to customize the information shown in the editor.

What’s Noteworthy?

NOTE When you check out a file from source control, you are actually checking out the file's corresponding XLF file. You will see the XLF file if you open the Source Control Explorer. This is because you need the XLF file available in order to make changes, view the file's history, or view differences.

Viewing Files That are Checked Out

You can use the Source Control Explorer to view all of the files that you have checked out and need to check in.

How to View Files That You Have Checked Out

1. Select **View > Source Control Explorer**. The Source Control Explorer opens.
2. From the drop-down or the Home pane, select **Pending Changes**.

   The Pending Changes pane opens. Files that you have changed appear in the **Included Changes** or **Excluded Changes** section (depending on whether you are going to include or exclude them in your next check-in).
3. Take note of the file's status. The status is written in brackets next to the file name (e.g., edit, add).

> NOTE You can click the refresh navigation button in the Source Control Explorer to make sure you have the most recent status for each file. Another option is that you can use a feature to automatically ping the source control repository periodically, thus refreshing this information frequently. However, you may experience slower performance with this automatic status update option set.

> NOTE When you check out a file from source control, you are actually checking out the file's corresponding XLF file. You will see the XLF file if you open the Source Control Explorer. This is because you need the XLF file available in order to make changes, view the file's history, or view differences.

> NOTE When you modify a file in source control, you may sometimes see a SKL file alongside the XLF and original files. This skeleton file is a placeholder file. Be sure to check in all three associated files together. Checking in just the SKL file may result in errors.

Viewing the History of Source Control Files

One of the benefits of Lingo's integrated source control is that you can view the history for a particular file, including who checked in the file and when it was checked in. You can also view differences between different versions of the file and roll back to an older version if necessary.

How to View the History of a Source Control File

1. In the Source Control Explorer or File List, select the relevant file(s).

   OR

   Open a file.

2. Do one of the following, depending on the part of the user interface you are using:
   - Ribbon Select Source Control > View History.
   - Right-Click If you have the Source Control Explorer or File List open, right-click the file you want to view and select Source Control > View History.
3. The History dialog opens. The following are explanations of the different parts of this dialog.
   - Version Displays a number for each version of the file. The lower the number, the older the version. The higher the number, the more recent the version.
   - Users Displays the name of the user who has been working on the file.
   - Date Displays the date and time when the action has occurred.
   - Action Displays the action that has taken place for the file (e.g., commit).
   - Comment Displays the comment (if any) associated with the file. A comment can be added to a file when you commit that file to source control. This enables you to maintain an audit trail for the file's history.
   - Get Selected Version Retrieves a particular version of a file, thus rolling back to that version of the file. The local version of the file is replaced with the source control version that you selected.
   - Show Differences Opens a dialog that lets you view the differences between two versions of a file. If you select one row in the History dialog and view the differences, you will see the content differences between the version that you selected and the

version of the file in your local copy of the Lingo project. If you select two files in the History dialog (by holding down the CTRL key) and view the differences, you will see the content differences between those two versions of the file.

4. In the History dialog, click **Close**.

PDFs

The following PDFs are available for download from the Help system.

Cheat Sheets

Shortcuts Cheat Sheet

User Guides

Alignment Guide

Getting Started Guide

Key Features Guide

Source Control Guide: Git

Source Control Guide: Perforce Helix Core

Source Control Guide: Subversion

Source Control Guide: Team Foundation Server

Termbases Guide

Touring the Workspace Guide

Translation Guide

What's New Guide