Working With Projects

SilkTest®
2008 R2

Borland®

Table of Contents

Overview of SilkTest Projects

Projects in SilkTest organize all the resources associated with a test set and present them visually in the Project Explorer, making it easy to see, manage, and work within your test environment.

SilkTest projects store relevant information about your project, including references to all the resources associated with a test set, such as plans, scripts, data, options sets, .ini files, results, and frame/include files, as well as configuration information, Editor settings, and data files for attributes and queries. All of this information is stored at the project level, meaning that once you add the appropriate files to your project and configure it once, you may never need to do it again. Switching among projects is easy - since you need to configure the project only once, you can simply open the project and run your tests.

When you create a new project, SilkTest automatically uses the SilkTest Agent that supports the type of application that you are testing. For instance, if you create an Adobe Flex project, SilkTest uses the Open Agent.

Each project is a unique testing environment

By default, new projects do not contain any settings, such as enabled extensions, class mappings, or Agent options. If you want to retain the settings from your current test set, save them as an options set by opening SilkTest and choosing **Options/Save New Options Set**. You can include the options set when you create your project. You can create a project manually or have SilkTest automatically generate a project for you, based on existing files that you specify.

Storing project information

SilkTest stores project-related information in two project files:

- *projectname*.vtp: the project file has a .vtp (Verify Test Project) extension and is organized as an .ini file; it stores the names and locations of files used by the project.

- *projectname*.ini: the project initialization file, similar to the partner.ini file, stores information about options sets, queries, etc. included in your project.

These files are created in the *projectname* folder. When you create your project, SilkTest prompts you to store your project in *<SilkTest installation directory>*\Project, the default location. (You can specify another location if you do not want to use the default Project folder.) SilkTest creates a *projectname* folder within this directory, saves the *projectname*.vtp and *projectname*.ini files to this location, and copies the extension .ini files (appexpex.ini, axext.ini, domex.ini, and javaex.ini) to the projectname\extend subdirectory.

> **Note**   If you are using Windows Vista, project data is saved in the C:\Users\<username>\AppData\Local\VirtualStore\Program Files\Borland\SilkTest <version> folder, and not in the Program Files folder where SilkTest is installed. When you open a project from SilkTest, project data is displayed as expected. However, if you use the File Explorer to access the project, you need to click the Compatibility Files icon to access the project. To store project data in the installation folder, as with previous Windows versions, you can disable the Vista User Access Control.

Note that the extension .ini files (appexpex.ini, axext.ini, domex.ini, and javaex.ini) located in your *<SilkTest installation directory>*\extend folder are copied to your *projectname*\extend directory, regardless of what extension you have enabled. Do not rename your *projectname*\extend directory; this directory must exist in order for SilkTest to open your project.

You can have SilkTest automatically enable the appropriate extension using the Basic Workflow bar (**Workflows/Basic**), or you can manually enable extensions; see Enabling extensions in the *online Help*. The current project uses the extension options in the extension .ini file copied to the *projectname\extend* directory. Any modifications you make to the options for this enabled extension will be saved to the copy stored within the current project, in your *projectname\extend* directory.

The *projectname\extend* directory is used only for local testing on the host machine. If you want to test on remote Agent machines, you must copy the .ini files from the *projectname\extend* directory to the *<SilkTest directory>\extend* directory on the target machines.

File references
Whether you are emailing, packaging, or adding files to a project, it is important to understand how SilkTest stores the file's path. SilkTest's .vtp files use relative paths for files on the same root drive and absolute paths for files with different root drives. The use of relative and absolute file paths is not configurable and cannot be overridden. If you modify the .vtp file to change file references from relative paths to absolute paths, the next time you open and close the project it will have relative paths and your changes will be lost.

Accessing files within your project

Working with SilkTest projects makes it easy to access your files - once you have added a file to your project, you can open it by double-clicking it in the Project Explorer. The Project Explorer contains two tabs:

- **Files tab**: lists all of the files included in the project. From the Files tab, you can view, edit, add, and remove files from the project, as well as right-click to access menu options for each of the file types.
- **Global tab**: displays all the resources that are defined at a global level within the project's files; for example, testcases, functions, classes, window declarations, etc. When you double-click an object on the Global tab, the file in which the object is defined opens and your cursor appears at the beginning of the line in which the object is defined. You can also run and debug testcases and application states from the Global tab. You can also sort the elements that appear within the folders on the Global tab.

For more information about the Files and Global tabs, see the Project Explorer dialog in the *online Help*.

Note: Existing test sets do not appear in the Project Explorer by default; you must convert them into projects; see "Converting existing tests into projects for details" later in this tutorial.

Sharing a project among a group
Follow these guidelines when sharing a project among a group:

- Create the project in the location from which it will be shared; for example on a network drive
- Ensure that testers create the same directory structure on their machines

Creating a new project

You can create a new project and add the appropriate files to the project, or you can have SilkTest automatically create a new project from an existing file. This exercise walks you through creating a new project using the Basic Workflow bar.

Before you begin

Since each project is a unique testing environment, by default new projects do not contain any settings, such as extensions, class mappings, Agent options, etc. If you want to retain the settings from your current test set, save them as an options set by opening SilkTest and choosing **Options/Save New Options Set**. You can add the options set to your project.

To create a new project

1 Click **Open Project/New Project** on the Basic Workflow bar.

2 On the New Project dialog, click the type of project that you want to test.

   The type of project that you select determines the default Agent. For instance, if you specify that you want to create an Adobe Flex project, the Open Agent is automatically set as the default agent. SilkTest uses the default Agent when recording a testcase or enabling extensions.

3 Click **OK**.

4 On the Create Project dialog, type the Project Name and Description.

5 Click **OK** to save your project in the default location, *<SilkTest installation directory>*\Project. SilkTest will create a *projectname* folder within this directory, save the *projectname*.vtp and *projectname*.ini to this location and copy the extension .ini files (appexpex.ini, axext.ini, domex.ini, and javaex.ini) to a *projectname*\extend subdirectory. If you do not want to save your project in the default location, click **Browse** and specify the folder in which you want to save your project.

6 SilkTest creates your project and displays nodes on the Files and Global tabs for the files and resources associated with this project.

You are now ready to add existing files or create new files and add them to your project; see the next exercise, Adding files to a project, for more details.

Adding files to a project

You can add existing files to a project or create new files to add to the project.

While we recommend adding all referenced files to your project so that you can easily see and access the files, and the objects contained within them, referenced files do not have to be included in the project. Plans and scripts will continue to run, provided the paths that are referenced are accurate.

To add an existing file to a project

1   If your project is not already open, click **Open Project/New Project** on the Basic Workflow bar (**Workflows/Basic**), select the project to which you want to add a file, and then click **Open**.

2   On the Project Explorer, select the **Files** tab, right-click the node associated with the type of file you want to add, and then click **Add File**. For example, to add a script file to the project, right-click **Script**, and then click **Add File**.

3   On the Add File to Project dialog, specify the file you want to add to the open project, and then click **Open**. The file name, followed by the path, appears under the appropriate category on the Files tab and is associated with the project via the *projectname*.vtp file. If files exist on a network drive, they are referenced using Universal Naming Conventions (UNC).

You can also add existing files to the project by choosing **Project/Add File**. SilkTest automatically places the file in the appropriate node, based on the file type; for example if you add a file with a .pln extension, it will appear under the Plan node on the Files tab.

We do not recommend adding application .ini files or SilkTest .ini files (qaplans.ini, propset.ini, and the extension.ini files) to your project.

If you add object files (.to, .ino) to your project, the files will appear under the Data node on the Files tab. Objects defined in object files will not appear in the Global tab. You cannot modify object files within the SilkTest editor because object files are binary. To modify an object file, open the source file (.t or .inc), edit it, and then recompile.

To add a new file to a project

1   If your project is not already open, click **Open Project/Open Project** on the Basic Workflow bar (**Workflows/Basic**), select the project to which you want to add a file, and then click **Open**.

2   On the Project Explorer, select the **Files** tab, right-click the node associated with the type of file you want to add, and then click **New File**. For example, to create a script file associated with the project, right-click **Script** and then click **New File**. The new file opens in the 4Test Editor.

3   Save the file and click **Yes** when SilkTest asks if you want to add the file to the open project. The file name, followed by the path, appears under the appropriate category on the Files tab and is associated with the project via the *projectname*.vtp file. If files exist on a network drive, they are referenced using Universal Naming Conventions (UNC).

If you chose to create a new test frame file, the .inc file is automatically added to the project; SilkTest does not prompt you to save it to the project.

You can also create a new file via the File menu; for example, **File/New/4Test Script**.

Converting existing test sets to a project

Before you begin

Since each project is a unique testing environment, by default new projects do not contain any settings, such as extensions, class mappings, or Agent options. If you want to retain the settings from your current test set, save them as an options set by choosing **Options/Save New Options Set**. You can include the options set when you create your project.

SilkTest can automatically generate a new project from an existing file or files, such as plan, script, or suite files. SilkTest scans the specified files, gathers all references to other files contained within them; for example, references to subplans, include files, options sets, scripts, window children, functions, and so on, and adds them to the project that it generates. SilkTest searches for the following statements within files: use, include:, script:, framefile:, optionset:, and usefiles=. See *Overview of Autogenerate Project* and *Automatically generating a new project* in the *online Help* for complete details.

To convert existing test sets to a project

1   Click **Open Project/New Project** on the Basic Workflow bar (**Workflows/Basic**).

2   On the New Project dialog, under Generic, click **Autogenerate**.

3   On the Auto-Generate Project dialog, type the Project Name and Description.

4   Specify the location in which you want to save your project. We recommend the default location, <*SilkTest installation directory*>\Project. SilkTest will create a *projectname* folder within this directory, save the *projectname*.vtp and *projectname*.ini to this location and copy the extension .ini files (appexpex.ini, axext.ini, domex.ini, and javaex.ini) to a *projectname*\extend subdirectory. If you do not want to save your project in the default location, click **Browse** and specify the folder in which you want to save your project.

5   Click **Add** to select the file from which you want SilkTest to generate a project and then click **Open**. The file is added to the list in the Generate from box. Click **Add** again to select additional files, if applicable. To add an options set, click **Add**, select **All Files** in the Files of type box, select the options set you want to add, and then click **Open**. To remove a file from the list, select the file, and then click **Remove**.

6   Click **OK** when you have finished adding files. SilkTest automatically generates a new project from the file or files you selected. This may take a few moments, depending on the size of the files you are converting into projects. SilkTest creates *projectname*.vtp and *projectname*.ini files in the location you specified in the Save in text box and displays the files and resources included in the automatically-generated project in the Project Explorer.

7   If you added an options set and want to use it, you must load it into memory: on the Files tab, expand the Profile node, right-click the options set you want to load, and then click **Open Options Set**. Note that double-clicking an options set opens it for editing, but does not load it into memory.

Note   Results files, data files, bitmaps, and files included or referenced via File Functions are not automatically added to your project. You must add these files manually, see "Adding files to a project" earlier in this tutorial. You can also convert existing tests to a project by creating a new project and manually adding the files to the project.

Working with Projects

Autogenerate and .opt files: If an .opt file is referenced in a .pln file or selected as a file to Generate from, Autogenerate will parse the .opt file, find references to UseFiles=, and include any files referenced here in the Project Explorer, with the exception of .inc located in the SilkTest extend directory. It will not automatically add …\extend\filename.inc files to the project. However, you can manually add these files to your project.