Advanced Instrumentation Configuration Options

The following table lists common Advanced Options related to instrumentation configuration. These options are set in the Project Options panel. To review and modify settings:

1. Right-click the project tree node for the project whose settings you want to review and modify, then choose

   Properties from the shortcut menu. The Properties dialog will open.

2. Select **Parasoft> C++test> Other Settings** in the left pane.
3. Use the available controls to add, import, reorder, or remove advanced options.

Option	Description

`xharness.`
`testCasePrimaryTestObjectVarName`
`string`
	Specifies the name of the variable holding the primary test object in automatically-generated test cases.
	
	The default value is `cpptest_TestObject`

`testrunner.addSourceLine string`	Inserts an additional source line into the testrunner source file for test cases. The line will be inserted at the beginning of the file (after the line including the "portinfo.h" header).
	
	Examples:
	
	The following option includes the `using` declaration / directive in the testrunner source:
	
	`testrunner.addSourceLine namespace Test`
	`{} using namespace Test;`
	
	Alternatively, you can use the following two options in concert (be sure to replace `TestSuiteClass.hxx` with the actual name of your test suite header file):
	
	`testrunner.addSourceLine #include "TestSuiteClass.hxx"`
	
	`testrunner.addSourceLine using namespace Test;`

`testrunner.debuggerCommandLine /usr` `/X11R6/bin/xterm -e xxgdb -x %s &`	Enables support for the xxgdb debugger.
`testrunner.debuggerCommandLine ddd -x` `%s &`	Enables support for the ddd debugger.
`testrunner.earlyRuntimeInitialization` `false`	Prevents C++test from adding a call to the CppTest_InitializeRuntime() function before a global object constructors call or at the beginning of the main() routine.
`edgtk.useEdgPreprocessor false`	Configures C++test to use the original compiler executable as the preprocessor. By default, C++test uses an internal preprocessor for Microsoft Visual C/C++ compilers.