Using a Debugger During Test Execution

This topic explains how to step through C++test test cases with a debugger to better examine the code's internal state during a given test. For example, you might want to debug test cases to learn more about how C++test obtained an unexpected outcome, or to determine why a test case failed. You do not need to manually add breakpoints to the code. C++test will automatically set the breakpoints at the beginning of each test case that you select for execution.

Sections include:

- Configuring Debugger Settings
- Executing Tests with a Debugger

Configuring Debugger Settings

To enable debugging inside Visual Studio IDE

1. Open test configuration you use for running test cases
2. Choose the **Execution> Runtime** tabs
3. Enable the **Run tests in debugger(*)** option.

Executing Tests with a Debugger

This section covers debugging tests in native development environments. For information about debugging tests in embedded environments see Debugging Test Cases.

To execute a test with the Visual Studiodebugger:

1. Prepare a "Debug Unit Tests" Test Configuration.
    - Either use the built-in "Debug Unit Tests" Test Configuration or develop a custom Test Configuration by duplicating the built-in configuration and customizing it as described in the Configuring Test Configurations and Rules for Policies.
2. Run your preferred "Debug Unit Tests" Test Configuration as follows:
    a. Select test case(s) that you want to debug in one of the following ways:
        - Select the test case in the Test Case Explorer.
        - Select the test case function name in the code editor.
        - Select the test case function in the Class view. Note that these functions will be located under the Testsuite classes, not under the classes corresponding to the original source.
    b. Launch your preferred "Debug Unit Tests" Test Configuration. For example, right-click the selection, then use the **Parasoft** shortcut menu to run the preferred Test Configuration. C++test will then launch appropriate debugger and automatically set the break-points at the beginning of each selected test case function.

    c. If you are using Microsoft Visual Studio 2003 with Service Pack 1, perform the following steps:

        - Select the test process in the **Processes** dialog and click **Attach**.
        - Confirm the application type (Native) by clicking **OK** in the Attach to Process dialog.
        - Close the Processes dialog.

3. Use standard debugger features to step through the test case.

> - The debugger will be activated only if the debugging Test Configuration is run on a selection of one or more test cases. Otherwise, the following warning will appear in the console output and the execution will continue without debugging:
> ```
> Warning: debugger will not be activated - no valid breakpoints found. Select (test case)
> function definitions to set breakpoints.
> ```