FAQs and Troubleshooting

In this section:

How can I get quick access to information about usage on the command line?

Use the `-help` command line switch:

```
dottestcli.exe -help
```

Why is dotTEST unable to find build artifacts in Visual Studio 2017 when Lightweight Solution Load is enabled?

By default, dotTEST attempts to build solutions and projects prior to analyzing them to obtain the required build artifacts, such as .exe or .dll files. If the Lightweight Solution Load option is enabled in Visual Studio 2017, it prevents dotTEST from building a solution. As a result, the required build artifacts are not available or are not up to date, and the analysis may not be performed or some rules may not be executed. You can prevent this in one of the following ways:

- Disable the Lightweight Solution Load option.
- Delegate the build to MSBuild by configuring the `dottest.build.builder_id` option and setting its value to `msbuild`; see Building Solutions and Projects for details.
- Build the solution prior to analysis and configure dotTEST to skip the building phase by setting the `dottest.build.nobuild` option to `true` or passing the `-nobuild` command line switch; see Building Solutions and Projects for details.

Why do I get notified that Parasoft DTP Plugin is slowing Visual Studio 2017?

Visual Studio 2017 introduced new performance management options to monitor and manage extensions that can affect startup time. As a result, a pop-up alert may appear when Parasoft DTP Plugin is installed to indicate that Visual Studio startup time has increased. You can click the Manage Visual Studio Performance link in the pop-up window to review the startup times and disable displaying the alert.

Why does Visual Studio 2015 not display code markers?

If the Parasoft DTP Plugin for Visual Studio is reinstalled, Visual Studio 2015 may fail to display code markers that indicate analysis findings or code coverage in the editor. This is caused by a Visual Studio known issue (see MEFCache bug breaks text adornments in any package for details).

To ensure that code markers are properly shown, clear the Visual Studio MEF cache each time you reinstall or updated the Parasoft DTP Plugin. Use one of the following options:

- Use the Clear MEF Component Cache extension that will automatically clear the MEF cache (download here).
- Manually delete the cache directory. Depending on your installation details, the cache directory may be available in the following locations:
  - If the Plugin has been installed with the default options: %localappdata%\Microsoft\VisualStudio\14.0\ComponentModelCache
  - If the Visual Studio registry suffix has been specified with the `<SUFFIX>` parameter during installation: %localappdata%\Microsoft\VisualStudio\14.0<suffix_name>\ComponentModelCache

Why does dotTEST not report results for some rules and metrics when integrated with Visual Studio 2013?

Some static analysis rules and metrics require Visual Studio Update 5 to be installed on your machine if you run analysis with dotTEST from Visual Studio 2013. You may need to install Visual Studio Update 5, or run analysis from the command line.

What if I get the api-ms-win-crt-runtime-l1-1-0.dll is missing error when trying to execute unit tests with VSTest?

Install update for Universal C Runtime in Windows (KB2999226). See https://support.microsoft.com/en-us/help/2999226/update-for-universal-c-runtime-in-windows for information about the update and installation.

What if dotTEST cannot collect coverage information?

- Ensure that appropriate PDB files are available when coverage is collected. Each analyzed assembly must have a corresponding PDB file generated during the same build (dotTEST does not support the portable format of PDB files).
- Some versions of .NET Core may have a bug that prevents dotTEST from collecting coverage data for .NET Core applications. To ensure that the application coverage is collected, copy the dottest.Hooks assembly shipped with dotTEST to the application folder where IIS is deployed. The assembly is shipped in [INSTALL DIR]/integration/iis/bin/dottest/dotnet.

What if dotTEST fails to run analysis and generates an empty report?

Some machine setups may fail to provide dotTEST with the path to Visual Studio and its version, which prevents dotTEST form opening and analyzing projects. In such a case, you may need to manually set the following environment variables:

- SET VSINSTALLDIR=[path to your Visual Studio installation directory]
  Example: `SET VSINSTALLDIR=C:\Program Files (x86)\Microsoft Visual Studio\2017\Professional`
- SET VISUALSTUDIOVERSION=[version number of your Visual Studio]
  Example: `VISUALSTUDIOVERSION=15.0`

Visual Studio Version Numbers

Product Name	Version Name
Visual Studio 2017	15.0
Visual Studio 2015	14.0
Visual Studio 2013	12.0
Visual Studio 2012	11.0

How can I prevent my machine ID from floating?

Changes in the network environment may affect the interface that is used to compute your machine ID and result in machine ID instability. You can use the PARASOFT_SUPPORT_NET_INTERFACES environment variable to specify a stable interface and prevent the machine ID from floating.

1. Set up the **PARASOFT_SUPPORT_NET_INTERFACES e**nvironment variable.
2. Set the variable value to a stable Ethernet network interface. Do not use virtual, temporary or loopback interfaces.
   - On Windows: Set the value to the MAC address of your network card. You can use the `ipconfig -all` command to obtain the address. For example:

```
SET PARASOFT_SUPPORT_NET_INTERFACES=00-10-D9-27-AC-85
```

If the problem persists, you can obtain diagnostic information by setting up the environment variable **PARASOFT_DEBUG_NET_INTERFACES** a nd setting its value to true. This will print to the standard output the checking procedure that can be shared with technical support, as well as the interface that is used to compute your machine ID. The interface will be marked with the [SELECTED] prefix.

How can I work with dotTEST via proxy?

Typically, if you connect through a proxy server, you need to configure the connection by passing protocol-specific system properties to the JVM – using the `-D` command line option.

To work with dotTEST, ensure that the system properties for the HTTPS protocol (`https.proxyHost` and `https.proxyPort`) are configured. Your command line may resemble the following:

```
java -Dhttps.proxyHost=myserver.example.com -Dhttps.proxyPort=8080
```

The proxy mode is not supported for Visual Studio.

Why are suppressions of some rules no longer available on DTP after dotTEST was upgraded to a newer version?

Suppressions associated with rules whose messages changed between releases may not be available on DTP and the rules must be re-suppressed.

You can restore legacy messages for BD category rules in version 10.4.1 and later by configuring the following settings in your .properties file: `flowanalysis.legacy.messages.for.<rule_ID>=true`.

For example:

```
flowanalysis.legacy.messages.for.BD.PB.ARRAY=true
flowanalysis.legacy.messages.for.BD.PB.ZERO=true
```