eggIntegration for Visual Studio

eggIntegration for Visual Studio is an extension to Microsoft Visual Studio that enables you to run eggPlant Functional test scripts as Generic Tests from a Visual Studio Test Project.  
  
Teams using the Microsoft ALM systems of Team Foundation Server (TFS) or Visual Studio Online can then execute eggPlant Functional Tests from Microsoft Test Manager or from a Visual Studio build.

The Visual Studio Generic tests calling the eggPlant Functional scripts can integrated into the standard Microsoft ALM workflows and Team working in the same way as any other Microsoft ALM automation test. See the Microsoft documentation:

* [Running Tests in Microsoft Test Manager](https://msdn.microsoft.com/en-us/library/dd286680(v=vs.110).aspx)
* [How to: Run Tests from Microsoft Visual Studio](https://msdn.microsoft.com/en-us/library/ms182470.aspx)
* [Setting Up Test Machines to Run Tests](https://msdn.microsoft.com/en-us/library/dd293551(v=vs.120).aspx)

# Requirement

* Visual Studio 2013 Premium or Ultimate to create and execute the Generic tests running eggPlant Functional scripts.

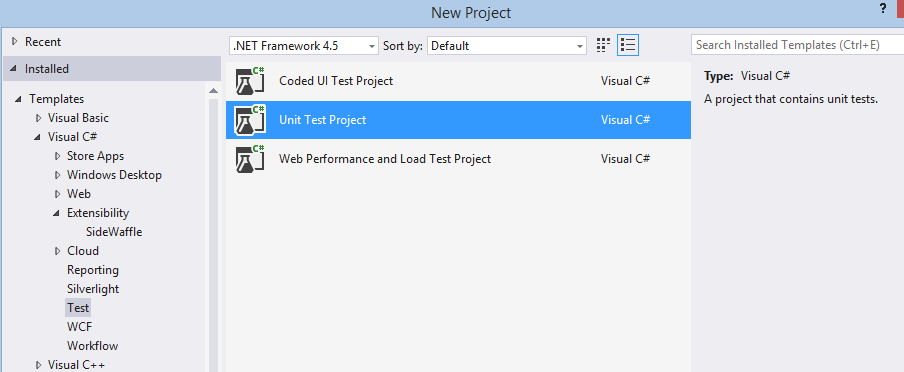
Not supported: Visual Studio Professional, Express Editions or Microsoft Test Professional **do not** support creating Generic Tests.

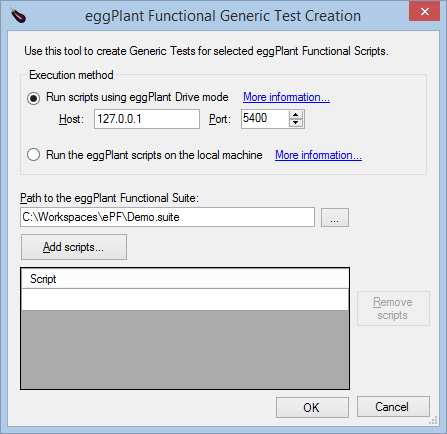
# Installation

As a user with local administrator permissions, execute eggIntegrationVS\_setupX.exe (where X is a version number). Allow the installer to execute the Visual Studio Package Installers on completion.

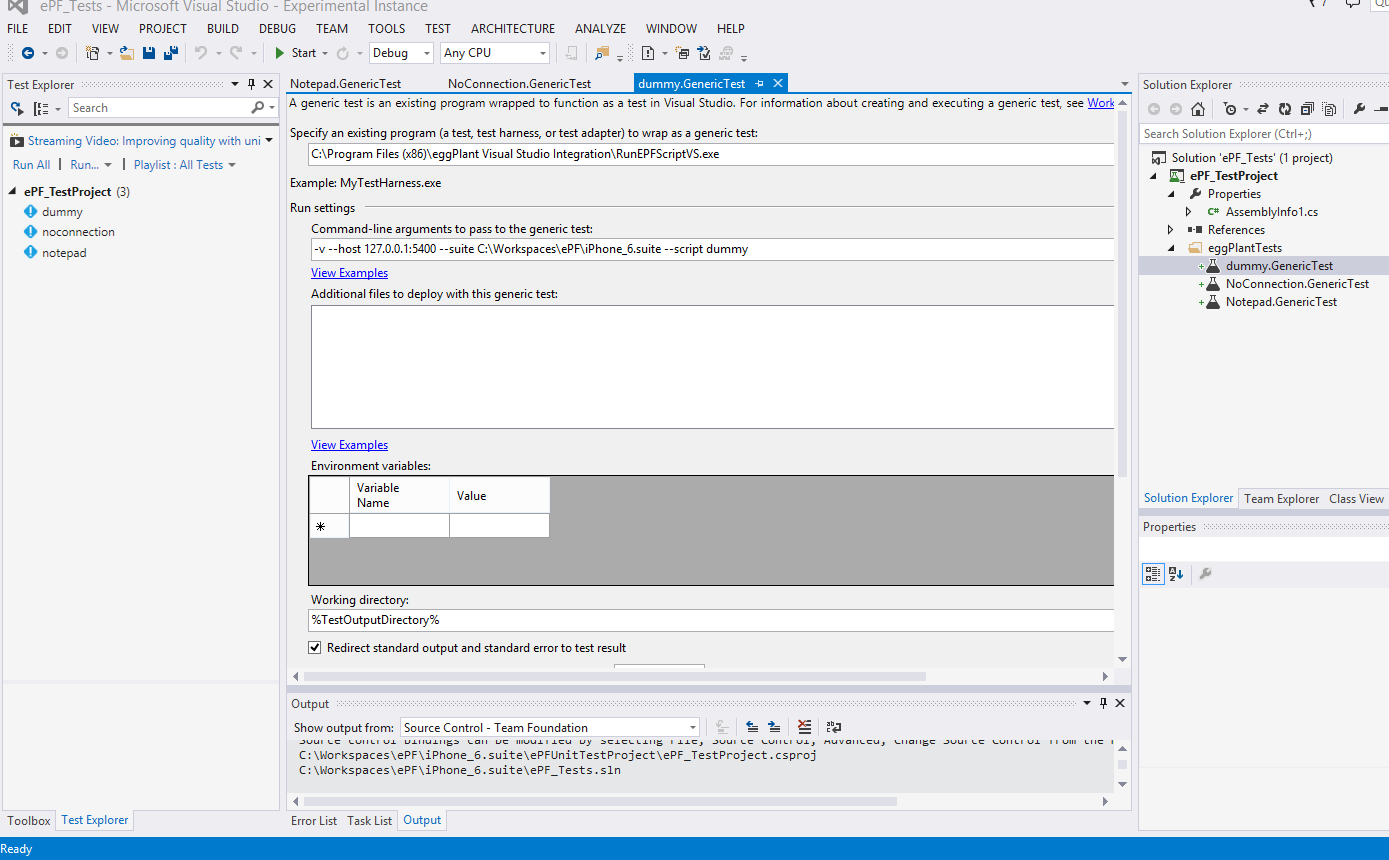
# Adding Tests to Visual Studio

From Visual Studio you will need to create a Test Project. From the menu File, New, Project, select Visual C# Test and the Unit Test Project. You can also select Visual Basic but not C++. You can delete the generated UnitTest1.cs file.  
  
Note: The project must be a **Test Project** otherwise you cannot execute the tests even though you can add them to a project.



After opening the project, from the TOOLS Menu select “eggPlant Functional Tests….  
The following tool dialog is displayed:  


If you select to run scripts via eggDrive and the Host is a remote machine then the suite path and Add buttons are disabled and you must manually enter the scripts and suite path on the remote machine. Selecting the option to run eggPlant scripts on the local machine will use command line execution of eggPlant.

After selecting the suite and scripts and clicking OK a Generic Test is created in the project for each script:  


To open the Test Explorer window, select the Menu, TEST, Windows then Test Explorer.

Open or double click a GenericTest in the Solution Explorer to view or edit the command line arguments to RunEPFScriptVS.exe, this is the process that executes the script and collects the results.

If you have a host:port specified then the execution will use [eggDrive](http://docs.testplant.com/?q=content/eggdrive) otherwise it will run eggPlant Functional through the [command line](http://docs.testplant.com/?q=content/running-command-line) on the local machine.

# Running tests

Right click a test in the Test Explorer window to run the test or use one of the Run menu items.  
The eggPlant script log output is translated into the Visual Studio format and copied to the Visual Studio Test location. Click the Output link at the bottom of the Test Explorer window to see the results, errors and any screenshots.

To run tests from Microsoft Test Manager you will need to setup a test controller and test agents as described in the Microsoft documentation: [Setting Up Test Machines to Run Tests](https://msdn.microsoft.com/en-us/library/dd293551(v=vs.120).aspx).

## Notes

* The Visual Studio package installer is copied to the installation folder, e.g. C:\Program Files (x86)\eggPlant Visual Studio Integration
* You can manage packages from the Visual Studio Menu, TOOLS, Extensions and updates… where you can disable or uninstall packages.
* The Test Creation dialog disables the drive search button and Add button if the Host is not an IP address or hostname for the local system.
* If the suite is on a remote system you must enter the suite path manually and it must be a path recognised by eggPlant Functional running in Drive mode on that remote system.
* The dialog saves the selecting settings for the user between sessions.
* Double click the rows in the Scripts list to edit entries or click in the bottom blank row to type a new script name. You only need the name of the script in the suite.
* The Console Application RunEPFScriptVS.exe is used to execute the scripts. This takes various command line argument as shown by running it with no arguments. For diagnostic purposes you can run it manually from a command window.
  + If the *–host host:port* argument is given, RunEPFScriptVS.exe will use XMLRPC to connect to eggPlant Functional running in Drive mode. It selects the suite and calls *RunWithNewResults scriptName*
  + If a host argument is not given, the script will be executed by calling RunScript.bat with the command line arguments:  
    runscript.bat “fullScriptPath” -GlobalResultsFolder Results [optional other parameters]
  + The files created in the suite GlobalResultsFolder “Results” are not deleted and will be visible using the eggPlant Functional UI.
  + The location of eggPlant Functional is found by searching for eggPlant\runscript.bat under %ProgramFiles%, %ProgramFiles (x86)%, and %EGGPLANT%. You can use the latter environment variable if eggPlant Functional is not installed in either of the first two locations.