Solution Orphan Checker

There is a nice little utility in the bin directory called SolutionOrphanChecker.exe that can run through all the projects and files in our source tree looking for potential problems that our build system doesn’t detect:

* Files with no compile actions. These are not necessarily a problem, but as a general rule, any cs file in a project without a compile action is bad. This check wass mainly added to detect FDO files that Randy kept around for reference purposes but which were no longer needed in the new FDO.
* Project References with SpecificVersion set to true. Anything we build as part of the FW build process should have SpecificVersion set to false.
* “Orphaned” source files that are not part of any project. These can be local files that you might create and forget to delete, or they can be checked-in files that are removed from a project but not deleted (or deleted files that come back as the result of an integration).

# Instructions for setting up the SolutionOrphanChecker

You can run SolutionOrphanChecker at the command line or set it up as an External Tool in Visual Studio. It takes a single command line argument, which is the full path of the root of the source tree you want to check, e.g., c:\Projects\fwr\Src. If you set it up to run as an External Tool, select the *Use Output window* checkbox.

# When to use it

It’s probably overkill to run this before every check-in, but you might run it periodically just to keep things clean. Three specific situations when it would be beneficial are:

* If you’ve done a lot of complex refactoring that involves moving/renaming files, or changing around a lot of project references.
* Any time you add a new project.
* Before submitting an integration that is adding files or includes changes to project files.