Use Case 2 Analysis of a Windows Application with C or Cpp

Analysis of a Windows application with C/C++

> Please note that these instructions may be outdated!

In this use case, we will focus now on how to analyze a Microsoft Windows application:

Sample application

From 'Help – Samples' option, in Visual Studio 13 Community edition, you can download the source code of our sample application.

Database Query From MFC

Technologies SQL Server, Visual Studio 2012

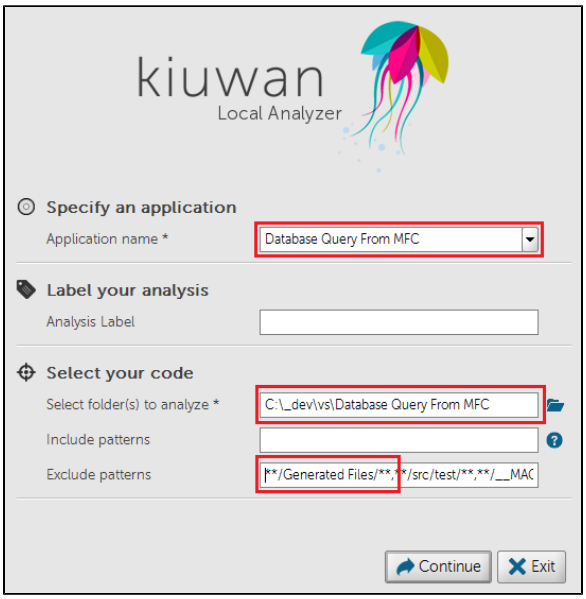
Topics User Interface, Data Access, Database Synchronization, Database, Database Connectivity

Platforms Desktop, Data

Analizing the code

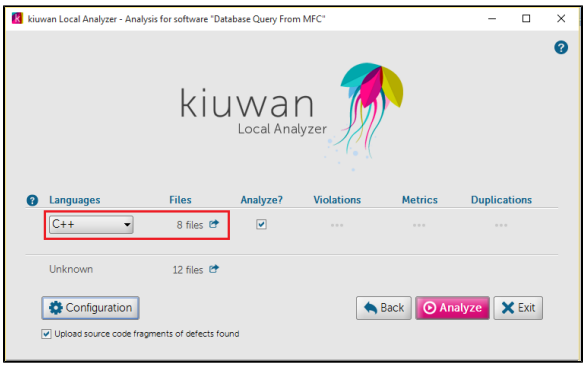
First of all, start 'Kiuwan Local Analyzer' with the {KiuwanLocalAnalyzer_install_dir}\kiuwan.cmd script.

Then, create a new application and set the source code directory.

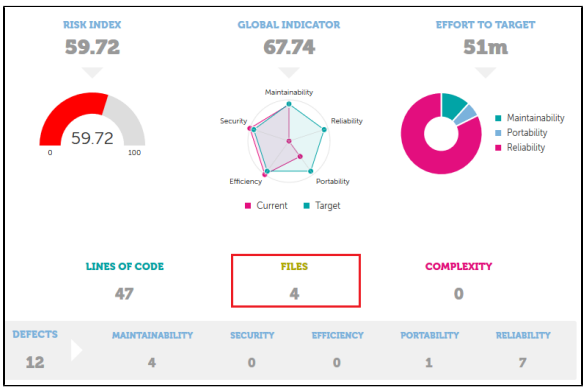


In the 'exclude patterns' option, add the '**/Generated Files/**' path. Visual Studio generates temporal files in this directory and we need to exclude them from the analysis because we don't want to analyze files we haven't coded.

Click 'Continue', select 'C++', and analyze…



Once the analysis has finished, you can see the results on Kiuwan:



Everything seems just ok, but if you look closely, only four files were analyzed out of the eight files detected by Kiuwan Local Analyzer.

If we go to the log screen to find out what happened, we can see which files were left out:

Solving parser problems

Looking in the log directory for this analysis ({KiuwanLocalAnalyzer_install_dir}\temp\Database Query From MFC.1444389776635-0\results), we can find the file cpp.unresolved.headers.log.

The content of this file are the header files that Kiuwan couldn't find during the analysis. Most probably because they are not in our project folders:

<SDKDDKVer.h>

<afxwin.h>

<afxext.h>

<afxdisp.h>

<afxdtctl.h>

<afxcmn.h>

<afxcontrolbars.h>

"afxwin.h"

<afxdb.h>

"afxdialogex.h"

In fact, these are Windows and Visual Studio header files our application uses, but are located in external directories. As we can see in the previous post, the resolution of these files is necessary to get a complete and reliable analysis.

With the last version of Kiuwan Local Analyzer, you can autoconfigure the header files paths and macro definitions using the log file that Visual Studio generates on the compilation phase.

To get the default Visual Studio header files paths, run the following commands from a DOS console:

> cd C:_dev\vs\Database Query From MFC

> call “C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\vcvarsall.bat”

> echo %INCLUDE% > include.txt

(update the source code and Visual Studio directories with those of your installation).

To get Visual Studio log file we need to compile the application from the command line:

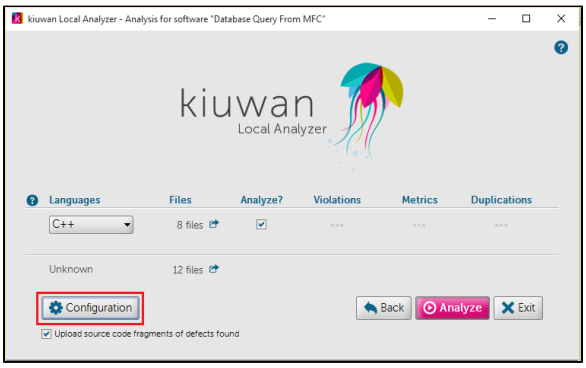
> cd C:_dev\vs\Database Query From MFC

> call “C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\vcvarsall.bat”

> msbuild.exe “Database Query From MFC.sln” /nologo /target:Rebuild /p:Configuration=Release /p:Platform=Win32 > msbuild.log

Now, we can use the generated ‘include.txt’ and ‘msbuild.log’ files to configure Kiuwan Local analyzer.

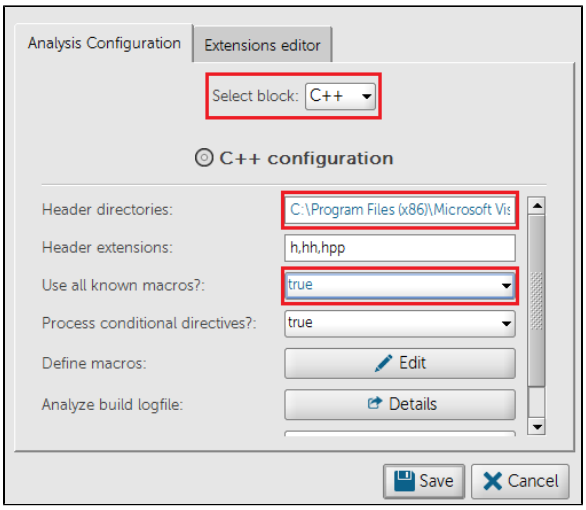
Before clicking on ‘Analyze’, open the configuration section:



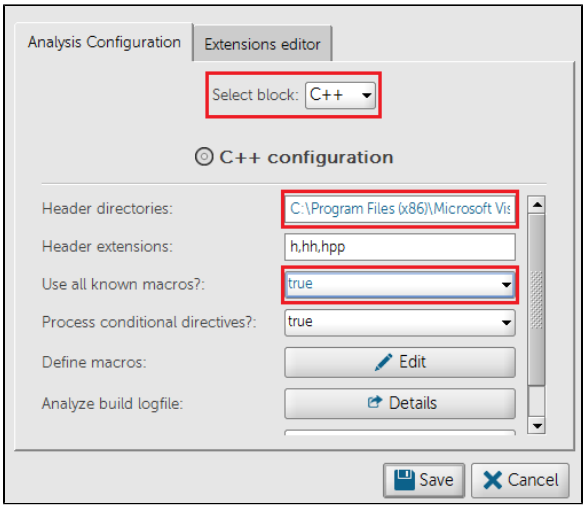
Copy the content of ‘include.txt’ file in ‘Header directories’ option. You need to replace the semicolon separator by a comma separator before the copy.

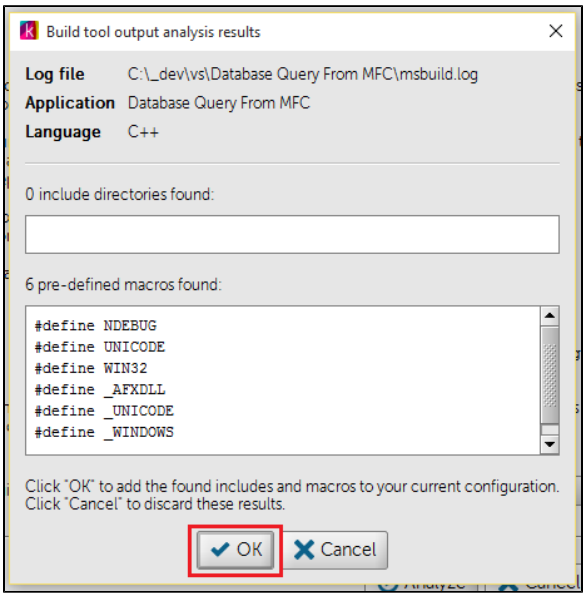
C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\INCLUDE,C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\ATLMFC\INCLUDE,C:\Program Files (x86)\Windows Kits\8.1\include\shared,C:\Program Files (x86)\Windows Kits\8.1\include\um,C:\Program Files (x86)\Windows Kits\8.1\include\winrt

You should also set the ‘Use all know macros?’ to true.

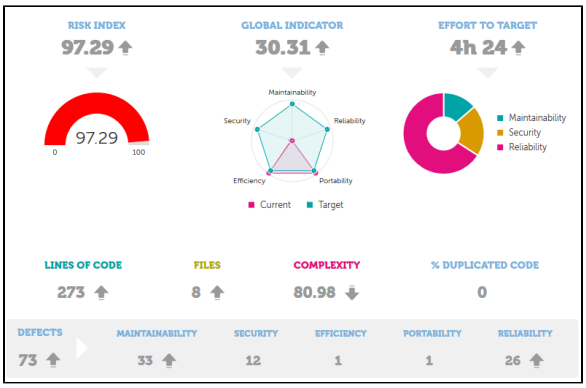


Now, in this same screen, click 'Analyze build logfile – Details' button, to load the msbuild output logfile:





After these steps, run the analysis again and check the new results on Kiuwan:



Now, all the eight C++ files of this sample application were correctly analyzed.