SimModel Python API Library Compilation Guideline

Jun Cao[1], Reinhard Wimmer[1], Tobias Maile[1], James O'Donnell[2], Christoph van Treeck[1]
[1]Institute of Energy Efficient Building E3D, RWTH Aachen University, Germany
[2]School of Mechanical and Materials Engineering and Electricity Research Centre, University College Dublin, Ireland

1) Dependency: please install following external libraries or tools before compiling SimModel Python API library:

- The XML binding parser CodeSynthesis XSD 4.0, the user can download it from here: http://www.codesynthesis.com/products/xsd/download.xhtml
  For setting the CodeSynthesis compiler in Visual Studio, the user can check this link: http://wiki.codesynthesis.com/Using_XSD_with_Microsoft_Visual_Studio#Visual_Studio_2010_.2810.0.29
- The Python library: https://www.python.org/downloads/
  The user can install any Python version as they would use and reconfigure the Python environment settings as shown in later section. We use Python 3.4.4 32bit as the compilation tools in our demo.
- The C++ to Python binding library SWIG: http://sourceforge.net/projects/swig/files/swigwin/swigwin-3.0.7/
  You can install the latest SWIG version 3.0.8, which is published on Dec. 31, 2015. However, we have not tested this version. We selected SWIG 3.0.7 as the binding library for our Python binding. Furthermore, it is not necessary to install SWIG if you would not modify the SimModel schema structure. We have already generated the Python wrapper files for the SimModel schema model.

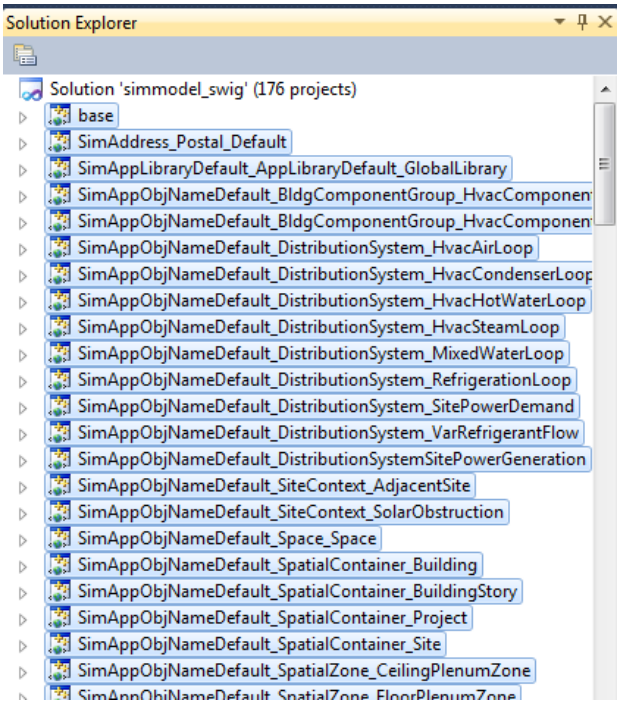
2) Source code compiling environment:

- The SimModel Python library is compiled on Windows by Visual Studio C++ 2010 (or higher version).
- You can also select the other OS compatible developing tools, e.g., Qt + mingw (the g++ Windows version), to compile the source code of the library. However, some bugs of these tools sometime will cause the program compilation failed.
- We will add the support for the other OS platform later.

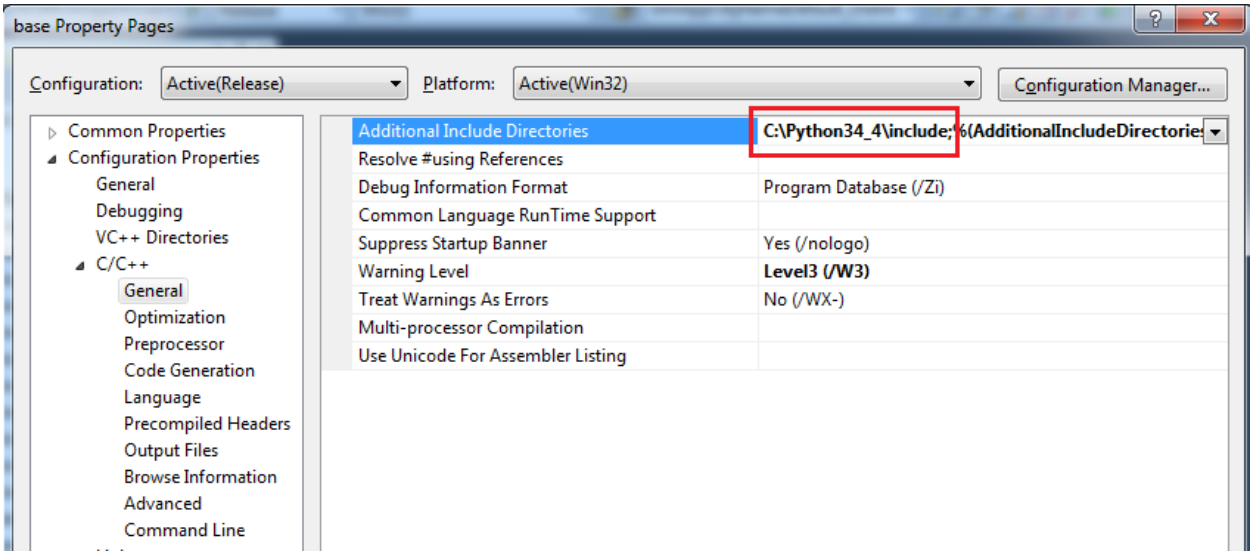
3) Compilation environment setting:

- Some features of the CodeSynthesis XSD 4.0 library header files are not supported by the SWIG binding parser. After installing CodeSynthesis, please copy the source code of the folder 'xsd4.0_source_code_update/tree' into your CodeSynthesis XSD installment directory, e.g., 'C:\xsd\xsd4.0\include\xsd\cxx\tree' to replace the header files saved in this subfolder.
- The subfolder 'simmodel_swig' saves the source code of the SimModel Python API library. Please open the library solution file 'simmodel_swig.sln' by your Visual Studio tool. The SimModel Python library only uses pure C++ coding, normal Visual Studio compiling environment higher than 2010 should load this .sln file properly.

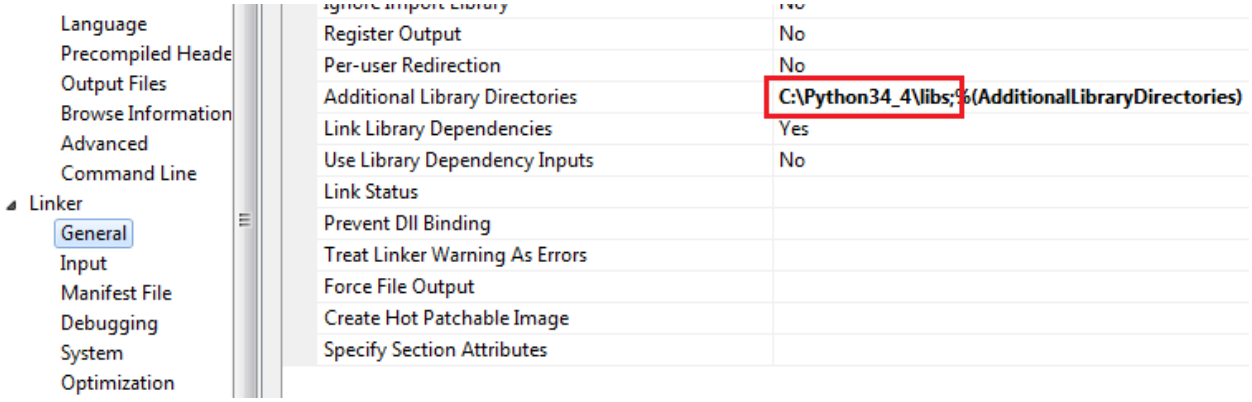
- After loading the ‘simmodel_swig.sln’ in Visual Studio, select all the projects listed in the solution (see the figure below), then right click the mouse, and select the ‘properties’ to set the compiling configuration.



- In the Property page, open ‘C/C++’ and select ‘General’ shortcut. Set your Python include directory at the ‘Additional Include Directories’, see the red box of the figure.



- Then select the ‘linker’, and go to the ‘General’ page. Set your Python library directory at the ‘Additional Library Directories’ (see figure below).



4) Build the source code: after setting the Python directory, press ‘F7’ to build the entire solution. A set of SimModel Python dll library (.pyd) files wrapping each SimModel classes will be output into the subfolder ‘simmodel_swig\Release’.

5) After building the code, open the subfolder ‘simmodel_swig\Release’, load the example code ‘runme.py’ to test the SimXML use case. Please read the comment of this file to learn how to import each SimModel class you would access.

6) The subfolder ‘simmodel_swig\Release’ also saves the use case ‘Boiler_Gas_VDI6020_V10.xml’ and the new version of the SimModel schema ‘simmodel_schema_15.10.30’.