Tutorial for VC++ UDF Studio 2019R3   [Chinese Version]

Supported Features:

Windows version	Trial Version	Registered Version
WinXP~Win10 (x86/x64)	✔	✔

Fluent	Trial Version	Registered Version
6.3 (x86/x64)	✔	✔
12.0 (x86/x64)	✔	✔
12.1 (x86/x64)	✔	✔
13.0 (x86/x64)	✔	✔
14.0 (x86/x64)	✔	✔
14.5 (x86/x64)	✔	✔
15.0 (x86/x64)	✔	✔
16.0 (x64)	✔	✔
16.1 (x64)	✔	✔
16.2 (x64)	✔	✔
17.0 (x64)	✔	✔
17.1 (x64)	✔	✔
17.2 (x64)	✔	✔
18.0 (x64)	✔	✔
18.1 (x64)	✔	✔
18.2 (x64)	✔	✔
19.0 (x64)	✔	✔
19.1 (x64)	✔	✔
19.2 (x64)	✔	✔
2019R1 (x64)	✔	✔
2019R2 (x64)	✔	✔
2019R3 (x64)	✔	✔

*Fluent 17.0~2019R3 are recommended

Visual Studio (English & Chinese version)	Trial Version	Registered Version
2005 and older	✖	✖
2008 without SP1	✖	✖
2008 with SP1	✔	✔
2010	✔	✔
2012	✔	✔
2013	✔	✔
2015~2019	✖	✖

*Visual Studio 2010 is recommended

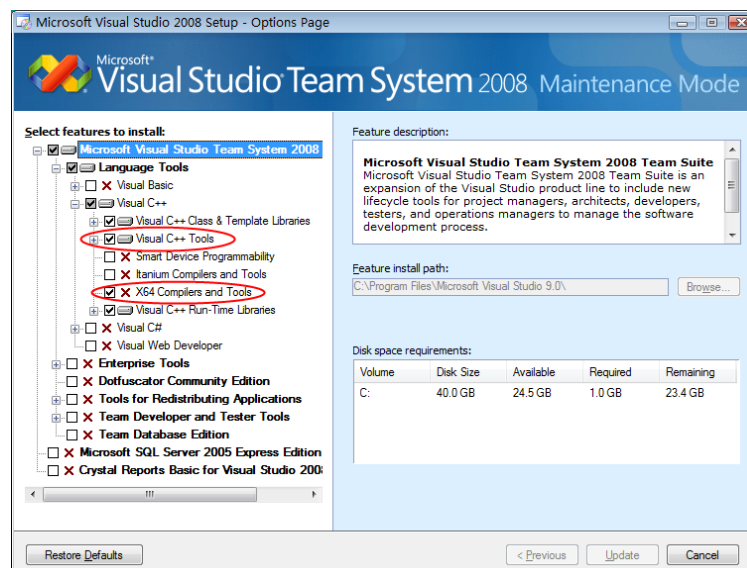| Fluent architecture	Trial Version	Registered Version
Serial (2d, 3d)	✅ max. 2 udf macros	✅ unlimited udf macros
Serial (2ddp, 3ddp)	❌	✅
Parallel (2d, 2ddp, 3d, 3ddp)	❌	✅

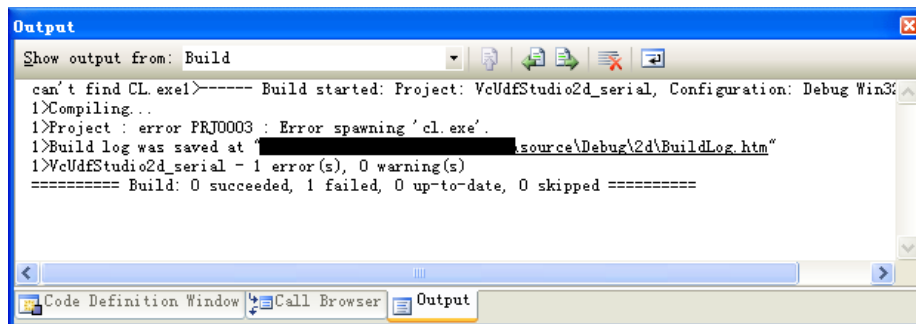## Important Notes on Visual Studio Installation:

1. Visual C# is recommended to be installed along with Visual C++, though it is not necessary for some Visual Studio version (e.g. VS2008 SP1 ultimate edition). But for VS2010, installation of Visual C# is mandatory, otherwise error will occur when starting Visual Studio.
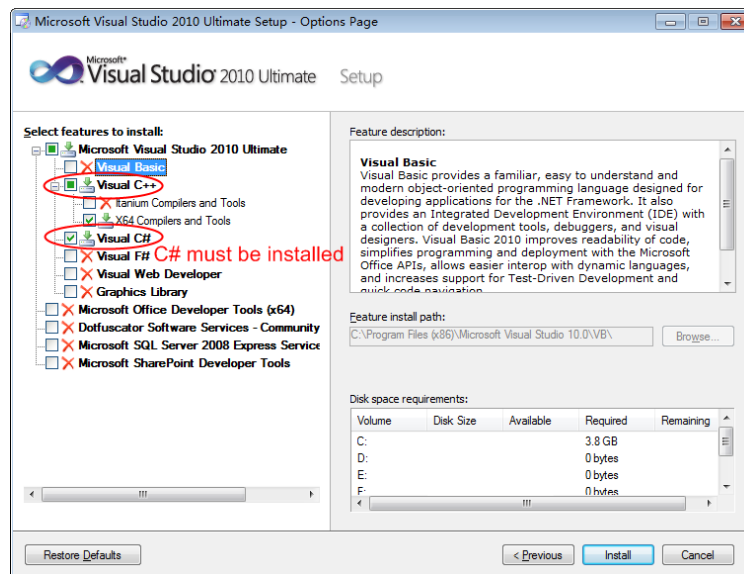
For VS2008, please assure "Visual C++ Tools" to be installed. In addition, for VS2008 standard edition, Visual C# is also necessary to avoid the "can't find cl.exe" error. In addition, VS 2008 Service Pack 1 must be installed based on the original version.

For 64 bit Windows, you have to install 64bit Fluent and please assure that "X64 compilers and Tools" will be installed.
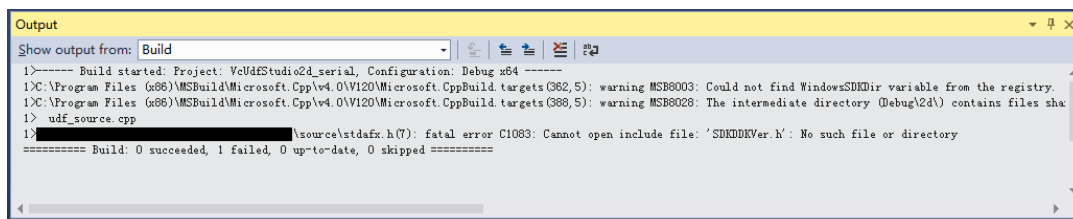
2. For Visual Studio 2013, please first assure the internet connection during installation. Otherwise, "WindowSDKDir" variable may be lost as following figure shows.
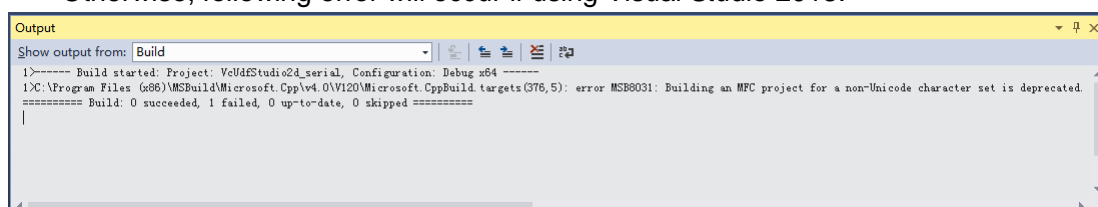

Second, please download and install "Visual C++ MFC Multi-Byte-Character-Set Library" first for Visual Studio 2013.

Website: https://www.microsoft.com/en-US/download/details.aspx?id=40770

Otherwise, following error will occur if using Visual Studio 2013.

Step-by-step Example:

1. Assure "Visual C#", "Visual C++" and "Visual C++ Tools" selected when selecting Visual Studio components to be installed (See previous section "Notes on Visual Studio Installation" for detailed requirements).

2. Install Service Pack 1 if you are using Visual Studio 2008. Other Visual Studio version can skip this step.

3. Start the "VC++ UDF Studio" launcher, and select the FLUENT version and VC version you'd like to use and click "OK". Click the "Browse" button to select a FLUENT installation directory whose corresponding version is not in the list but you want to run.
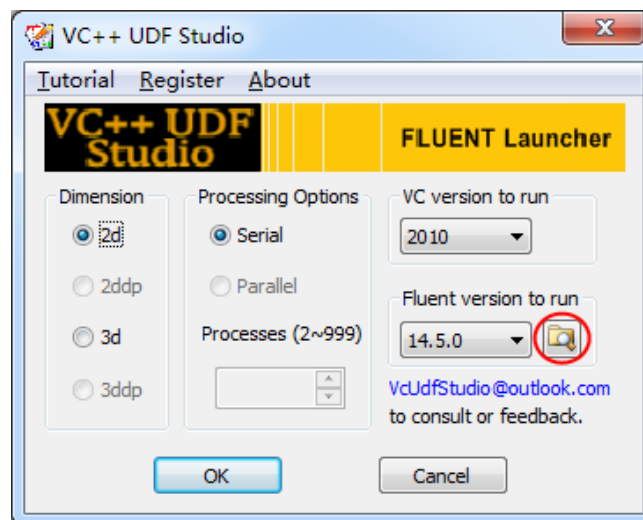

4. Read a Fluent case then click "Start Visual Studio" menu and begin to code UDF.
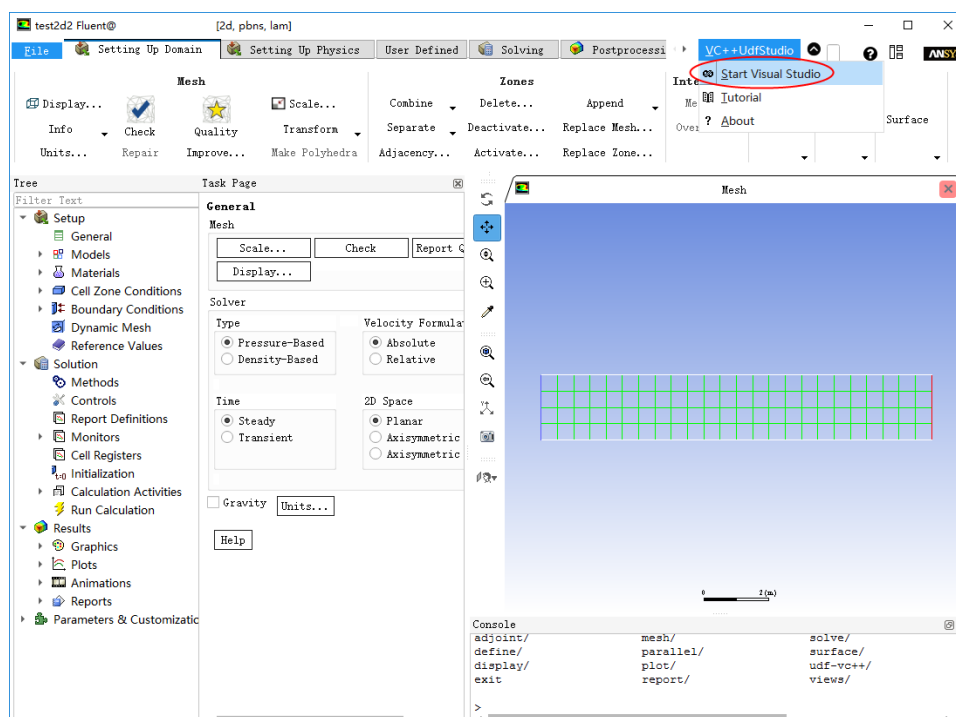
"VC++ UDF Studio" toolbar and menu will be shown in the Visual Studio. In the meantime, a folder called "source" will be created in the case directory containing all the source codes.
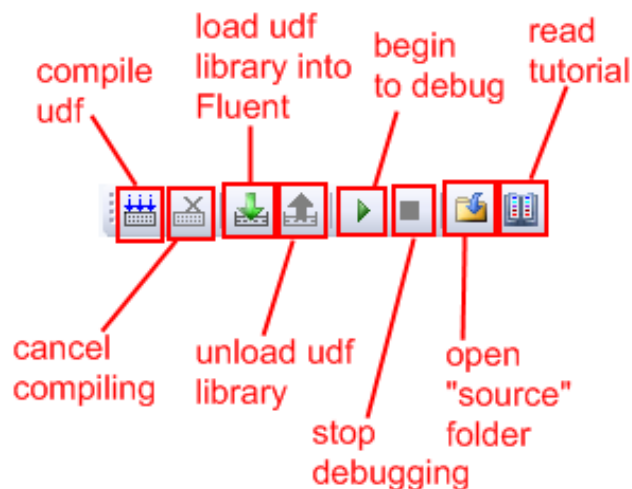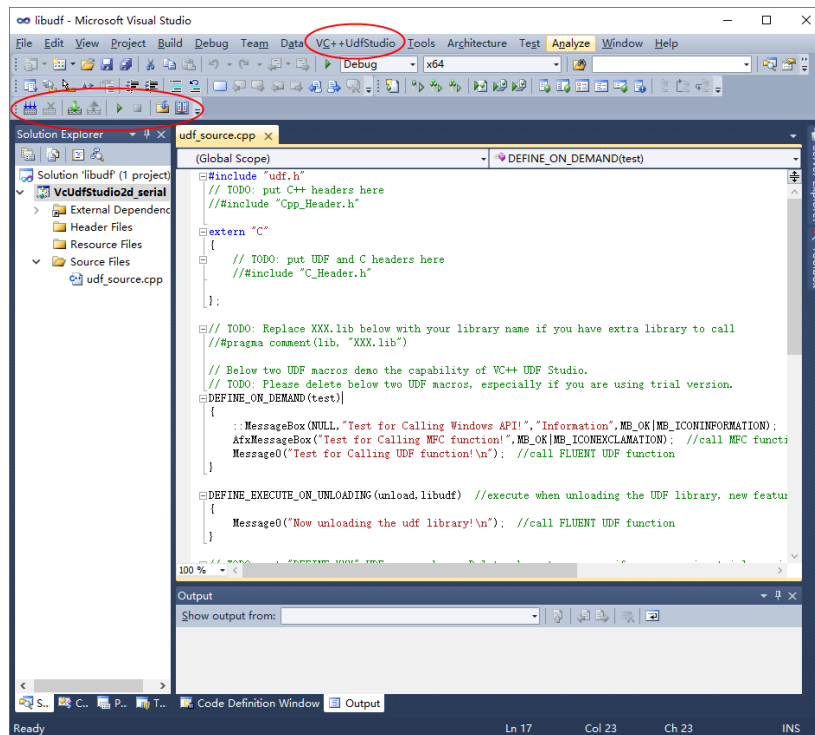
If "udf_source.cpp" file exists in the "source" folder, a warning message box will appear as below to confirm whether to overwrite it or not. Project files "*.vcproj" (VS2008) or "*.vcxproj"(VS2010 and later version) will be auto deleted when Visual Studio closes. Therefore, please never change the project settings. If you want to link additional library "XXX.lib", you can add #pragma comment(lib, "XXX.lib") in "udf_source.cpp".
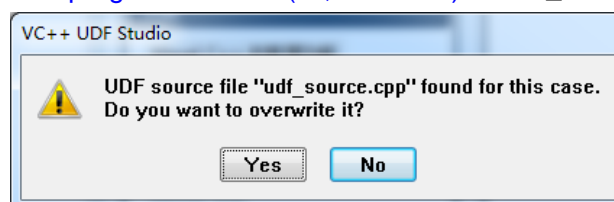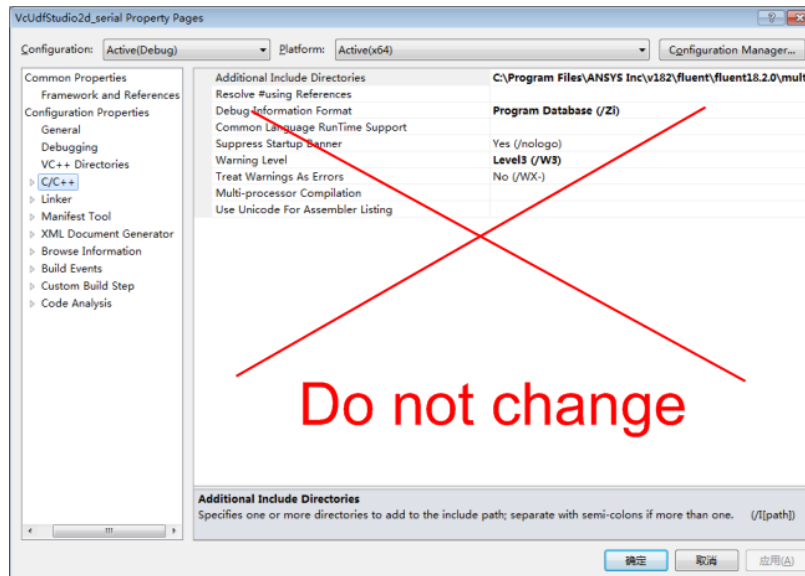
Note: All project files and intermediate folders (e.g., *.sln, *.suo, *.vcproj, *.vcxproj, *.user, *.filters, *.ncb, *.sdf, Debug folder, Release folder) <u>EXCEPT</u> "udf_source.cpp" and "udf_source.cpp.bak" will be auto deleted after Visual Studio closed.

5. Edit UDF source code. Add following test code to the end of "udf_source.cpp" file (If you are using trial version, please delete the DEFINE_ON_DEMAND and DEFINE_EXECUTE_ON_UNLOADING macros. Otherwise, macro over limitation error will be reported).
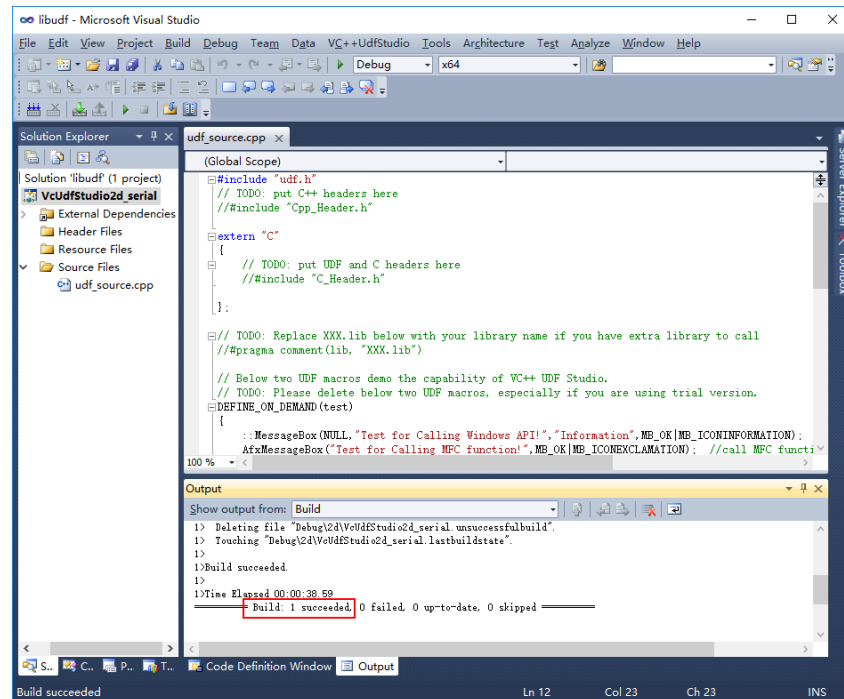
```
DEFINE_ON_DEMAND(debug)
{
    int aaa=123;
    int bbb=345;
    int ccc=aaa+bbb;
}
```
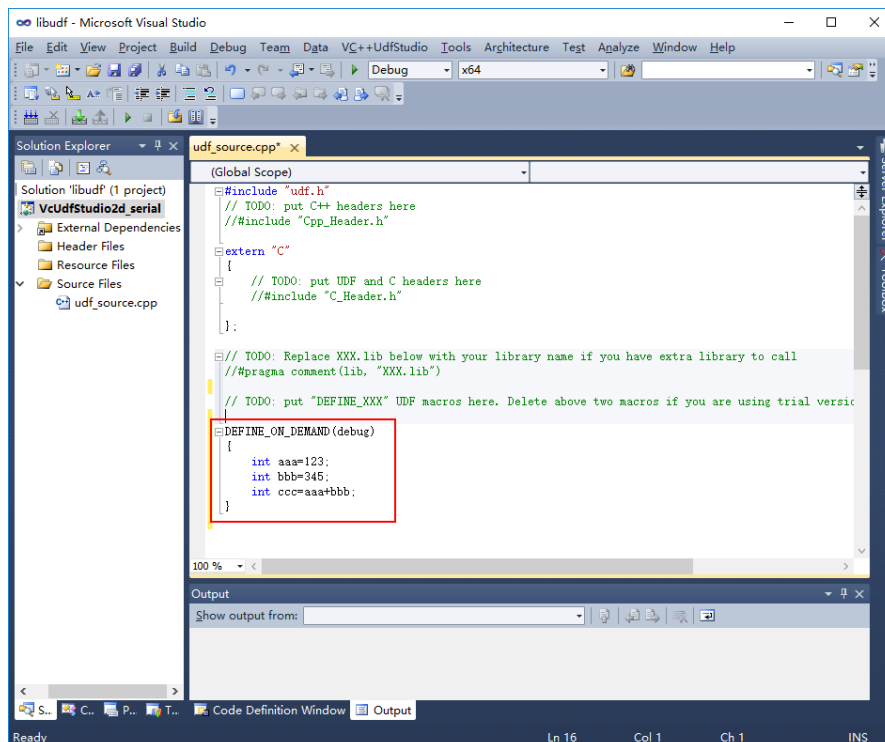
6. Click the "Build UDF library" button (or hotkey "F7") to compile the UDF source.

If no syntax errors found, then compiling will succeed (shown in below figure). You can't load or debug until the source code has been successfully built.

7. Click "Load UDF library to Fluent" button to load the library to FLUENT. Then, the FLUENT console should show message that "libudf" library is opened.
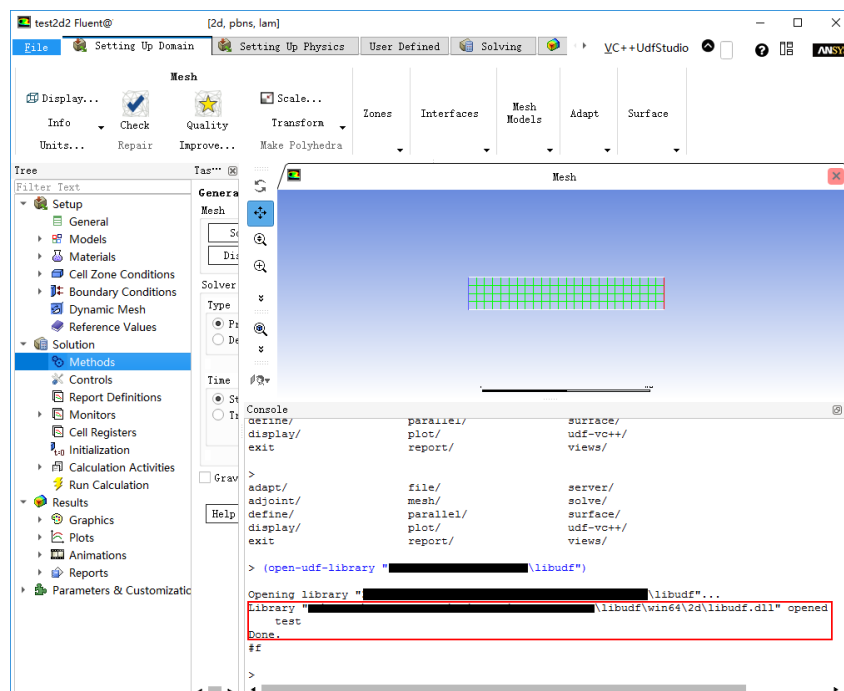
8. Set a breakpoint before "int aaa=123;" (mouse hover on this line and press hotkey "F9") and press "Start debugging UDF library" button. Of course, directing clicking this button is also OK without loading the library first. This tool will auto load the library first for you. But anyway, UDF library must be generated successfully.


9. Execute "debug::libudf" in Fluent. Visual Studio will auto stop at the breakpoint can you can see all the variable values.

Note: Please first understand when Fluent calls the macro you want to debug. If Fluent doesn't call the macro, breakpoints in the macro won't work. For example, DEFINE_SOURCE is called during iteration and DEFINE_INIT is called when initialization. If you have set a breakpoint in DEFINE_SOURCE but haven't started iteration, program won't stop at the breakpoint.
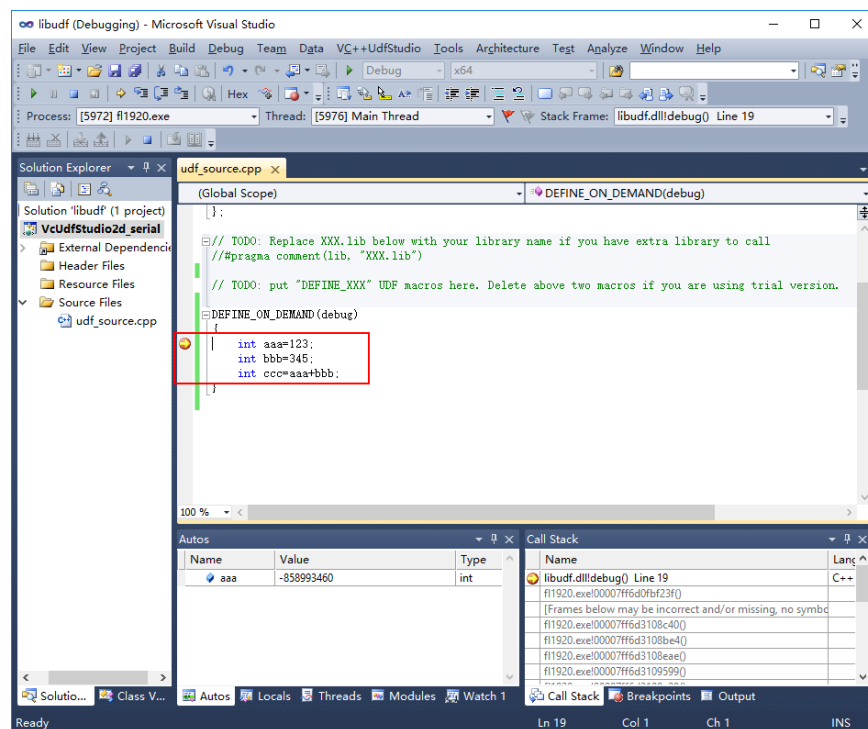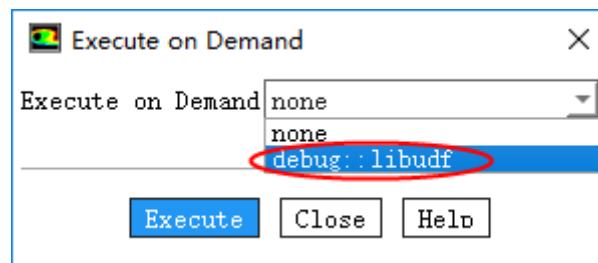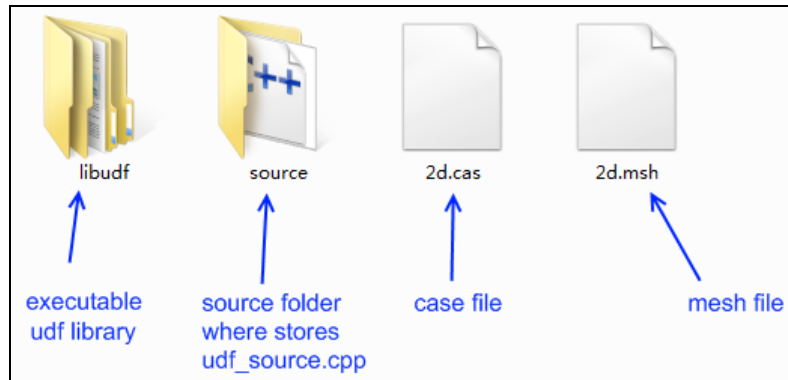

10. Step over the code line by line (or "F10" hotkey). You can observe all variable values just in the usual debugging way.
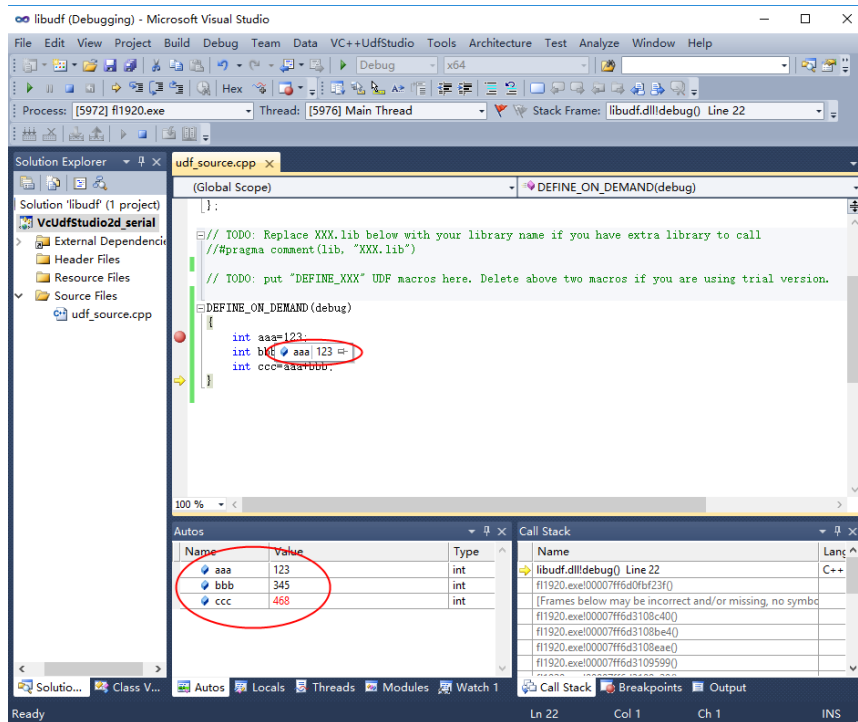
11. After all bugs removed, you can change from Debug to Release and re-compile it in release mode.
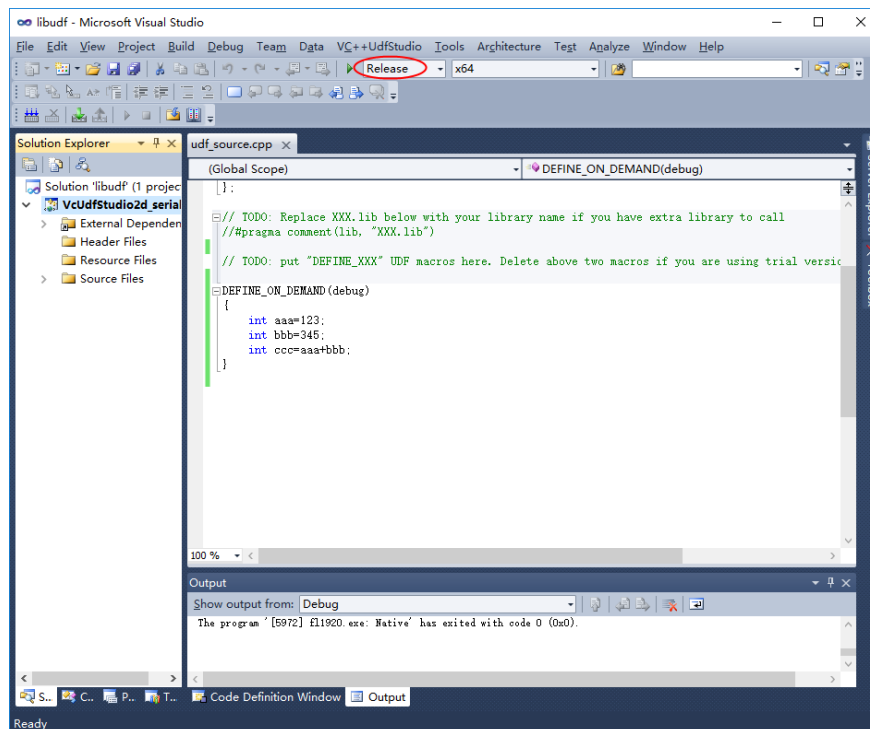
12. Now, your case folder may look like below, where "libudf" folder contains the release version of your UDF library and "source" folder stores the UDF source file "udf_source.cpp".

executable udf library → libudf

source folder where stores udf_source.cpp → source

case file → 2d.cas

mesh file → 2d.msh

13. After successfully compiling your release version of UDF library, you needn't start Fluent from VC++ UDF Studio launcher if you only want to run the case (not modify/compile UDF source). Just start Fluent in usual way and load the UDF library by "Define->User-Defined->Functions->Manage" menu and type "libudf" in the "Library Name" edit box and press "Load" button.
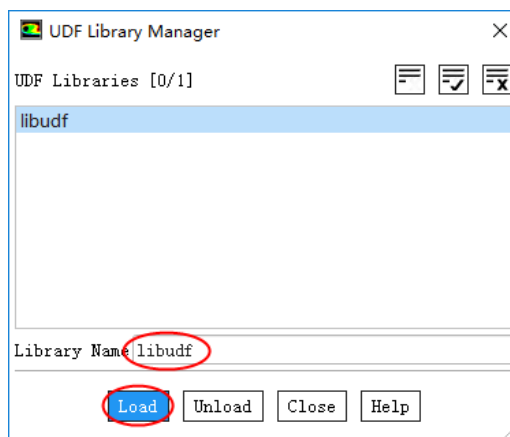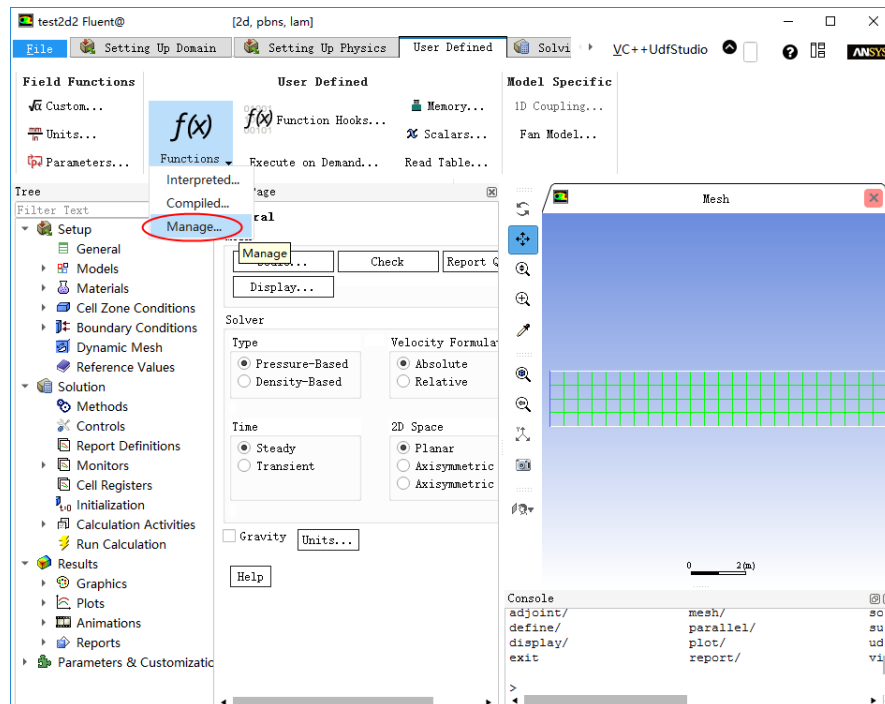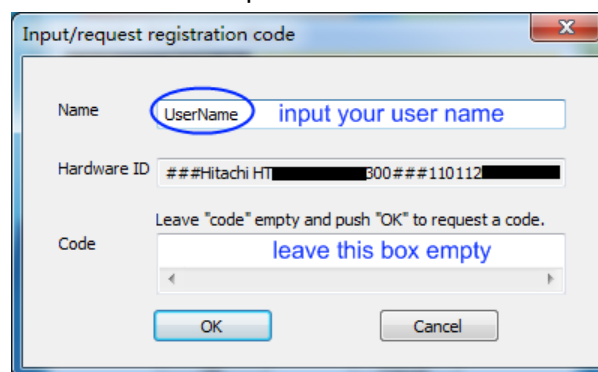
How To Register:

1. Open launcher and click "Register" menu.
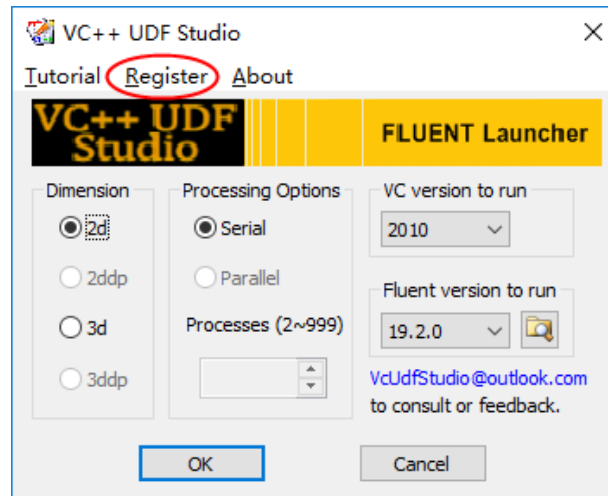
2. Input your username and leave the "Code" text box empty, then click OK. All your user name and hardware information will be put into the text file "user.ini".

3. Attach the "user.ini" in the email to vcUdfStudio@outlook.com (International) or vcUdfStudio@sohu.com (China). The content in the "user.ini" should look like below:

    Name=UserName

   *HardWare ID=###Hitachi HTSABC434CD300###110782PCDN403M7H6KM*

4. Contact vcUdfStudio@outlook.com (International) or vcUdfStudio@sohu.com (China) and pay for the software. After receiving returned email with register code, run the Launcher as administrator and click "Register" menu again. Input the user name and register code and all functions are available now.

Tips:

1. "Visual assistant" (www. wholetomato.com) is highly recommended to install, which has a lot of extended functions (such as code completion, braces matching, user-defined keyword colors).
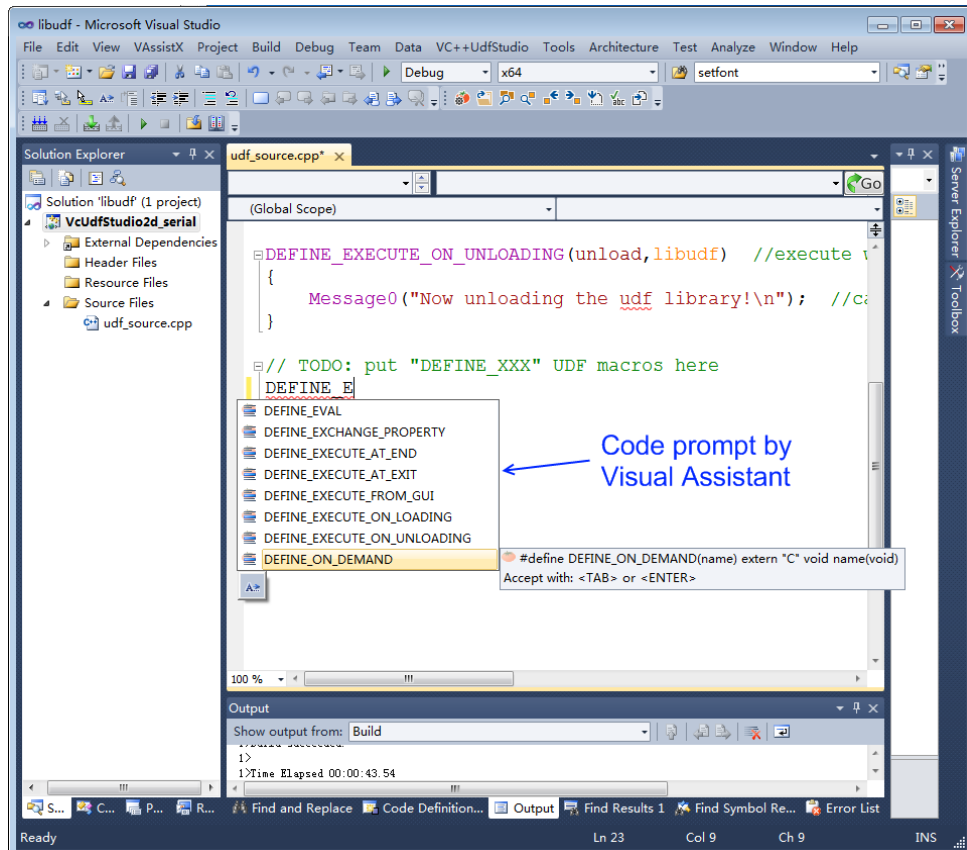
2. VC++UdfStudio menu in Fluent can be loaded or unloaded by TUI command
udf-vc++/load or *udf-vc++/unload*
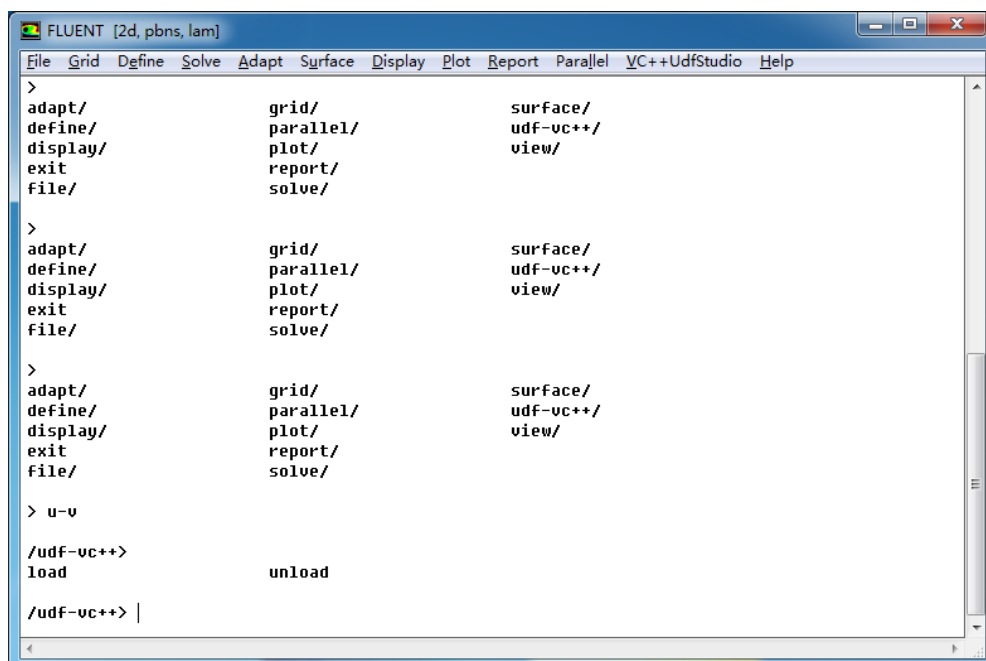
For more information, see the website at https://vcUdfStudio.bitbucket.io

To order registered version or report bugs, please contact vcUdfStudio@outlook.com (International) or vcUdfStudio@sohu.com (China).