**Installing Chaste on Windows 8.1, with Visual Studio 13**

**TROUBLESHOOTING NOTES**

In the file *cmake/thirdpartylibs/Cmakelists.txt* I made the following changes:

1. Always allowing Cygwin to auto-install seems to work best. Just comment out the decision statement.
2. Change the address on line 45 to <http://cygwin.com/setup-x86.exe> (the old file has moved)
3. If you’ve decided to use Visual Studio 13’s compiler, called msvc-12.0, change the boost build command on line 243 to:

b2 –toolset = msvc-12.0 –build-type = complete stage … etc.

N.B:

Some problems arise trying to compile Chaste’s dependencies using

msvc-12.0. This is because the Visual Studio 13 compiler tries to adhere to the new C++11 standard; as a result it has slightly different behaviour from earlier compiler versions. I recommend either sticking with msvc-11.0, or altering the makefile to download newer versions of Chaste’s dependencies, which are more likely to have been patched to work with this new compiler. (Don’t use Boost 1.56 though – its serialization library has a documented problem with msvc-12.0. Boost 1.55 should be fine.)

IF you decide to stick with the makefile as is, and use msvc-12.0, you will probably need to do the following:

1. Edit

*thirdpartylibs/downloads/parmetis/parmetis[version]/headers/gk\_arch.h*

*thirdpartylibs/downloads/metis/metis[version]/GKlib/gk\_arch.h*

Comment out the block between

#ifdef \_MSC\_ …. #endif

which defines the macro INFINITY and the function rint(). These arenow defined elsewhere and leaving this block in leads to a redefinition error.

1. Certain VTK-5.8 files no longer include the correct headers, since std:min and std:max have been moved to <algorithms> in C++11, and std:greater has been moved to <functional>. The following files are affected:

*infovis/vtkAdjacencyMatrixToEdgeTable.cxx* (include algo and func)

*infovis/vtkNormaliseMatrixVectors.cxx* (include algo)

*infovis/vtkPairwiseExtractHistogram.cxx* (include algo)

*views/vtkParallelCoordinatesRepresentation.cxx* (include algo)

*charts/vtkChartXY.cxx*  (include algo)

*charts/vtkControlPointsItem.cxx* (include algo)

1. In addition, the following VTK-5.8 files also need changing for compatibility with C++11:

*io/vtkEnsightBinaryReader(Gold).cxx*

Comment out all the Fortran exception blocks occurring between lines 3925 and 4119[[1]](#footnote-1). Could lead to silent failures if you are a Fortran user, so if you like you can try and figure out why these blocks are causing errors.

*rendering/vtkMapArrayValues.cxx*

At line 92, remove the bit between <…> from the call to make-pair; templating is no longer required or allowed here.

*common/vtkOStreamWrapper.cxx*

Line 60 is no longer acceptable for some reason. Replacing it with:

(ostream &a){ vtkOStreamWrapper b = vtkOStreamWrapper(a);

(this->ostr) << b;

return \*this;}

seems to work for me.

1. With these changes, you should be able to get far enough to produce a Visual Studio solution file called chaste in your build directory. At this point it’s easier to open up that solution in Visual Studio and try building chaste bit by bit, addressing any errors as they occur. If you right click ALLBUILD and go to see dependencies, then build order, it will show you in what project files to attempt to build first (start with global).

Errors you may encounter at this point:

* Visual Studio may fail to recognise all functions in the cstdlib. This happens because there is a file called cstdlib in the cxxtest folder of the Chaste source that seems to confuse visual studio 13. To check whether this file is being referenced instead of the “real” cstdlib, right click on one of the include statements in the code and select “open document” to see exactly what is being referenced.

To resolve this issue you can rename the cstdlib file in cxxtest to something else, and also rename it in the only two files where it’s referenced: *cxxtest/sample/mock/mock\_stdlib* and *cxxtest/sample/mock/real\_stdlib.*

* An error finding files of the form libboost\_....lib indicates that one of the boost libraries failed to build or was not built with the correct compiler. I had to rebuild boost manually for some reason to make the filesystem and serialization libraries appear. Just go into

*third\_party\_libs/boost/boost[version]* and run:

*bootstrap.bat*

followed by the b2 –toolset=msvc-12.0 … etc command copied from line 243 of the makefile. Ensure you give correct paths for your install.

* For more general linking errors, right click on a project and check in properties/ configuration properties/ C++ menu to ensure the Additional Include Directories option has all the include folders listed from your third\_party\_libs/install folder. If it’s a test project you’re having problems with, also check under properties/ configuration properties/ Linker to see that the Additional Library Directories are all there.

1. Lines 3925, 3944, 4001,4008, 4025,4048,4055,4072,4095,4102 and 4119 [↑](#footnote-ref-1)