**Building Slicer from Source**

The next step in the research phase is to build Slicer using the source code. For building the software from source code a few steps given in the developer’s manual have to be followed. But before going into the building process there are certain requirements or pre-requisites. These are essential components that are required on the computer before the build process can be started.

# Pre-requisites

For building the software in windows the following components or programs are necessary

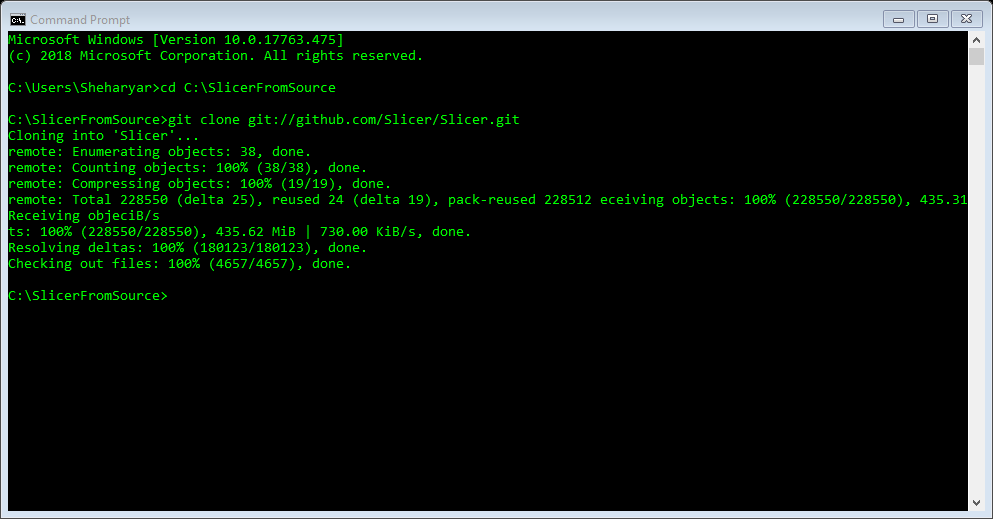
* CMake version 3.13.4 or later
* Git version 1.7.10 with all its components including bash.
* Subversion (SVN) client that has command line tools
* Qt5 version 5.10 or older along with *qtscript* and *qtwebengine*
* Visual studio 14 2015 64-bit version with component *Programming languages / Visual C++ / Common Tools for Visual C++ 2015* installed which is not enabled by default.

# Build Process

The build process is actually quite complex but it can become easy if the following steps are followed

1. **Cloning the git hub repository**

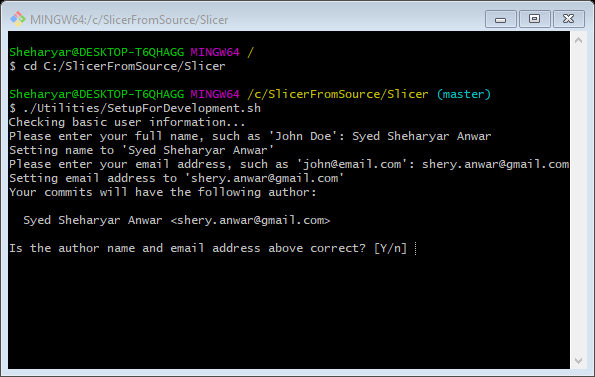
The first step in building slicer is to clone the git hub repository from the git hub link. It has to be noted that directly downloading the git hub project will not work since programs like CMake will not take it as a valid git file. So, to clone the repository use the command prompt. In order to do this, simply setup the path of the folder where it is intended to clone the repository using the command **cd desired path** and then use this command to clone **git clone git://github.com/Slicer/Slicer.git**



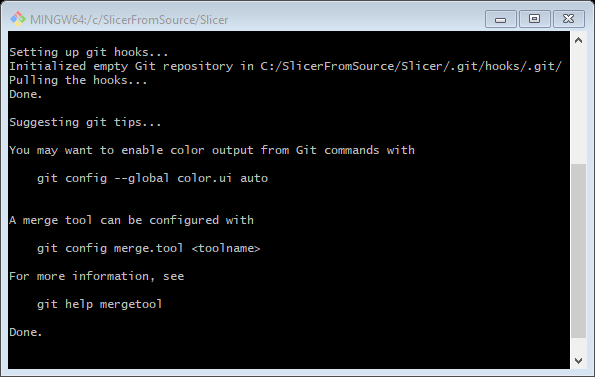
When these steps are followed the git hub repository will be cloned in the folder that was selected which in this case is SlicerFromSource. However, this command also generates a sub folder by the name of slicer inside the destination folder.

1. **Setting up the development environment**

The next step in building slicer is the setup of development environment. To setup the development environment the command to be used is **./Utilities/SetupForDevelopment.sh.** However, it has to be kept in mind that this command will not work on command prompt and has to be entered in a bash shell such as git bash or any other git client with a graphical user interface. Git bash is available with the git setup. In the bash shell first, it is needed to setup the path of the folder where the repository was cloned. Then use the above-mentioned command. After pressing enter it will ask the user to enter full name and email address. Once username and email address are set it will display them and ask if they are correct. If the information is correct type y and press enter otherwise type n and press enter and type the information again.



Once enter is pressed it will start setting up git hooks. Once this script is run these hooks are pulled and placed in the .git/hooks directory to install them.



1. **Configure git svn bridge**

The next step in this regard is the configuration of git svn bridge to ensure the mapping with svn revision. This step is only for core developers who have svn access while others may simply skip it. The command that are used for this configuration are given below

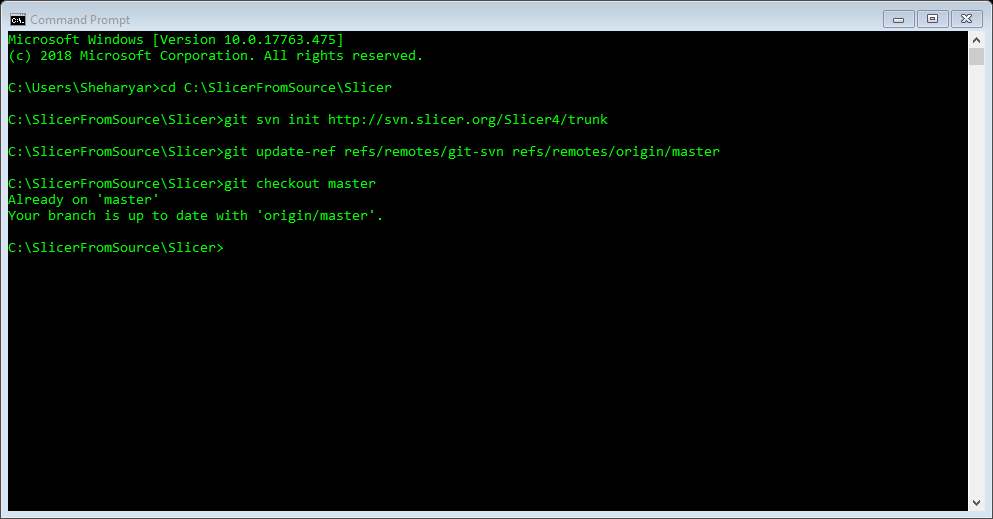
**git svn init http://svn.slicer.org/Slicer4/trunk**

**git update-ref refs/remotes/git-svn refs/remotes/origin/master**

**git checkout master**

**git svn rebase**

where, at first the user might have to setup the repository clone path, if not set already.



1. **Configure and generate solution files**

Once the svn client configuration is done the next step is to configure and generate the slicer solution files. There are two ways to configure and generate slicer solution files, one is to use CMake while the other is to use command prompt. If using CMake

* Set up the source directory as the folder where the repository of slicer was cloned.
* In the same directory create a new folder by the name of build and select this folder or directory as the build destination.
* Select the compiler as Visual studio 14 2015 Win 64. Remember that without specifying Win 64 the software takes it as Win 32 by default.
* Add *Qt5\_DIR* variable pointing to Qt5 folder such as *C:\Qt\5.10.0\msvc2015\_64\lib\cmake\Qt5*: click Add entry button, set Name to *Qt5\_DIR*, Type to *PATH*, and Value to the Qt5 folder.
* Configure
* Click generate and then close CMake

If CMake is not being used the same process can be done using command prompt with a few simple commands. In command prompt

* Create a folder where the generated files will be stored. This can be done by the command of *mkdir C:\SlicerFromSource\Build\SuperBuild.* This makes a new directory by the name of SuperBuild inside the original directory.
* Set up destination as this newly created directory using command *cd /d* *C:\SlicerFromSource\Build\SuperBuild.*
* Use command *"C:\Program Files\CMake\bin\cmake.exe" -G "Visual Studio 14 2015 Win64" -DQt5\_DIR:PATH=C:\Qt\5.10.0\msvc2015\_64\lib\cmake\Qt5 C:\SlicerFromSource\Slicer.* This command will generate the solution files.
* Finally apply command *"C:\Program Files\CMake\bin\cmake.exe" --build . --config Release* to build the solution files. In this step the build process can be carried out in two configurations either release or debug. For debug mode simply replace the word Release by Debug in the above-mentioned command. This step takes a lot of time since it configures all the libraries and tools (in our case it took around 14 hours).

1. **Build using Visual studio**

Once the configuration and generation process is done the final step is then to build these solution files using a compiler. To build the files follow these steps

* Open visual studio.
* In visual studio open project *Slicer-SuperBuild\Slicer.sln* from the path *C:\SlicerFromSource\Build\SuperBuild\Slicer-build\Slicer.sln.*
* Select the build configuration as Release.
* Build.

This will start the building process. It will take some time. Once the build process is done successfully it means that slicer has been built from source and installed on the computer. Slicer is completely installed and ready to be used on the system.

1. **Run Slicer**

This build process will have generated an executable file of slicer in the build directory. Run slicer from the path C:\SlicerFromSource\Build\SuperBuild\Slicer-build\Slicer.exe