Compiling SpArcFiRe into Standalone Executable

Step 1: Preparation

The "main" file, or the first function that gets called, is *findClusterArcsServer.m*. In order to compile for deployment, make sure that *all* functions that get called by *findClusterArcsServer.m* reside in the same folder or it will not compile correctly (when compiling, Matlab automatically locates files for you that are used by *findClusterArcsServer.m;* however, they need to be in the same folder). Another possible way is to have the rest of the files in subfolders and the main file to be at the root.

Step 2: Load Matlab Module and Compiler Version

Log into the openlab servers. After logging in, load the Matlab module by entering the command `module load matlab`. Then run the command `matlab`. You are now in the Matlab module and able to execute Matlab commands. You should see the following:

In order to compile Matlab code into an executable, make sure that you have the Matlab compiler installed. To check if the compiler installed, enter the command `ver` into the Matlab command line:

If you see "MATLAB Compiler" in the result, you're good to go.

Step 3: Convert C++ file to MEX file

In the SpArcFiRe source code, there is a C++ source file named *doHacClustering.cpp.* This file needs to be converted into a MEX file before compiling. MEX files are Matlab's solution for allowing non-Matlab code to be invoked from within Matlab. To build this MEX file, enter the following command:

```
mex doHacClustering.cpp
```

This will produce a file named *doHacClustering.mexa64* in the current folder. Make sure this file is included in the same folder as the other files to be compiled.

Step 4: **Compile**

To compile into an executable, enter the following command:

```
mcc -mv findClusterArcsServer.m
```

In the command above, mcc is the command to compile Matlab functions for deployment; the -m means create a "main" program, a standalone executable. -v requests verbose output. The resulting executable will be a file named *run_findClusterArcsServer.sh* and gets stored in the same folder you are currently in.


```
Removing: '/tmp/06ac1MJignL_27329.auth'.
Removing: '/tmp/06ac1Tlb90N_27329.auth'.
Removing: '/tmp/06ac13Uh3EQ_27329.auth'.
Removing: '/tmp/06ac1cSJYiT_27329.auth'.
Generating file "/home/araceg3/sparcfire/matlab/readme.txt".
Generating file "run_findClusterArcsServer.sh".
>>
```

Step 5: Run

To run the new executable, first exit Matlab by entering the command `exit`. Navigate to the folder in which the executable is located and enter the following command:

```
./run_findClusterArcsServer.sh /pkg/matlab/7.14_r2012a NONE
```

The middle argument "/pkg/matlab/7.14_r2012a" should be your own path to where you have Matlab installed. To find this, enter the command `matlabroot` into the Matlab module and it will display the path. The third argument "NONE" is the argument passed to the function.


```
$ ./run_findClusterArcsServer.sh /pkg/matlab/7.14_r2012a NONE
----------------------------------------
Setting up environment variables
---
LD_LIBRARY_PATH is .:/pkg/matlab/7.14_r2012a/runtime/glnxa64:/pkg/matlab/7.14_r2012a/bin/glnxa64:/pkg/matlab/7.14_r20
/jre/glnxa64/jre/lib/amd64/native_threads:/pkg/matlab/7.14_r2012a/sys/java/jre/glnxa64/jre/lib/amd64/server:/pkg/matl
lient:/pkg/matlab/7.14_r2012a/sys/java/jre/glnxa64/jre/lib/amd64
Warning: No display specified.  You will not be able to display graphics on the screen.
server run ID is 7360417341034800
No fit settings specified. Using defaults.
18-Mar-2015 17:37:06
server run ID: 7360417341034800
settings recorded to 7360417341034800_settings.txt

>>
```