

Compiling Matlab code for cluster use

From NUTMEGwiki

- This is copied from the UCSF BIL twiki, so some is UCSF specific, but hopefully most can generalise to other clusters/setups.

---++ First step is to get an account. This is not the same as your mrsc/radiology account. Ask Jeff Block (preferably in a helpdesk request <http://radhelp.ucsf.edu>) for an account.

Now that you have an account, and can log in, you can see that there is a whole wiki devoted to the cluster in general, which you can access at: <https://salilab.org/qb3cluster/> and the username and passwd are 'qb3' and 'qb3wiki'

The info below is meant to supplement that wiki and some info specific to running compiled matlab code / CTF programs.

---++ Compiling matlab code for qb3 In your bil home directory, have ~/bin/i686 and ~/bin/x86_64 and ~/bin/MCR directory with specific matlab versions

Have a function like mc: (which you run from your local computer by calling mcq, see below)

```
<verbatim>
```

```
1. !/bin/csh
```

```
setenv OUTPUTDIR /home/johannaz/qb3/bin cd $OUTPUTDIR`arch` source  
~/bin/MatlabSetup.sarangbin
```

```
foreach ii ($*) /netopt/share/matlab_versions/matlab_R14p3/bin/mcc -mv -R -nojvm -R -nojit -R  
-nodisplay -a /home/johannaz/nutmeg_NUTS/beamformers/nut_Default_Beamformer.m -p  
/home/johannaz/hdata/meg/ -p /home/johannaz/matlab/em/ -nocache $ii
```

```
switch (`arch`) case i686
```

```
/netopt/share/matlab_versions/matlab_R14p3/toolbox/compiler/deploy/glnx86/extractCTF $ii.ctf  
breaksw
```

```
case x86_64
```

```
/netopt/share/matlab_versions/matlab_R14p3/toolbox/compiler/deploy/glnxa64/extractCTF $ii.ctf  
breaksw
```

```
endsw end </verbatim>
```

and mcq: (run from location of where you normally open matlab, probably your home directory).

```
<verbatim>
```

```
1. !/bin/csh
```

```
~/bin/mc $* & ssh wildfire ~/bin/mc $* </verbatim>
```

If any of these cause errors, especially with not finding lib files correctly, you might need to modify your `~/matlab/R14SP3/mbuildopts.sh` file. Also, seahorse is another x86_64 machine to try instead of wildfire (especially if power is out at mission bay - it's been known to happen!)

---++ Using CTF programs on the qb3 cluster For SAMcov and SAMsrc, it works to simply copy these programs to your qb3 home directory. When you call them in the qsub scripts, make sure to specify their correct path.

---++ Uploading files to qb3 The QB3 cluster currently can't see your files on our `/data/research_meg*` or your bil home directories, so you must move the files needed to run the jobs to the cluster, run the job, then transfer everything back.

It might seem best to create a directory structure locally that will mirror the directories on the cluster that you will create, which will making syncing the files easier.

Rather than just `=ssh username@chef.compbio.ucsf.edu=`, you want to be careful of syncing files. so use a script like:

```
<verbatim> rsync --progress --exclude=bin/MCR --exclude=sim_out -avue ssh /data/research_meg2/jayZstuff/qb3/ chef.compbio.ucsf.edu:/netapp/home/johannaz/ ssh -X johannaz@chef.compbio.ucsf.edu rsync --progress --exclude=bin/MCR --exclude=.ssh --exclude=nosync --exclude=qlogs --exclude=.matlab --exclude=CTF_v5 -avue ssh chef.compbio.ucsf.edu:/netapp/home/johannaz/ /data/research_meg2/jayZstuff/qb3/ </verbatim>
```

Remember that your file names and contents must not show any patient or subject identities! If you do have identifiers in your directories you can proceed as follows:

```
* Rename the files you need to transfer to qb3. You can rename CTF datasets with the =changeDsName= command.
* Create a separate folder (e.g. =mkdir qb3=) that contains subfolders with the initials of all your subjects.
* In these subfolders, create symbolic links to the actual anonymous data of the corresponding subject: =ln -s /data/research_meg2/jayZstuff/qb3/subject_initials/subject_initials*.h5 qb3/subject_initials/subject_initials*.h5=
* Then you can use a script as follows to transfer the linked data to your qb3 home directory:
```

```
<verbatim> rsync --progress --exclude='*.hist' -rLtvue ssh /data/research_meg5/blabla/qb3/ chef.compbio.ucsf.edu:/netapp/home/aguggis/ </verbatim>
```

```
* Note that you must exclude all =*.hist= files in CTF datasets, because they contain patient identifiers.
* To get SAM files (*.svl) back to your directories, use the following script:
```

```
<verbatim> rsync --progress --exclude='*.1' --exclude='*,*to*ms' --exclude='*,*to*ms*Hz' --exclude='*.hist' -rKtvue ssh chef.compbio.ucsf.edu:/netapp/home/aguggis/ /data/research_meg5/blabla/qb3/ </verbatim>
```

---++ qsub scripts for qb3 Once you are in chef, the compiled matlab files should be copied over (if you have any), but they still need to go through the "CTF Extract" process which creates the `*mcr` directory. On chef, source `~/bin/MatlabSetup.csh`, go to x86_64 directory. If the `*mcr` directory exists, remove it first. Then run your executable w/out inputs. Then from chef, ssh to xeonint, and do the same but from the i686 directory, for 32bit.

Your qsub scripts should have extra flags like

```
<verbatim>
```

1. `$ -l ibm32=true`
2. `$ -l opt64=true`
3. `$ -l intel64=false`
4. `$ -l h_rt=01:15:00`
5. `$ -j y`
6. `$ -r y`

7. \$ -q medium.q
8. \$ -S /bin/csh

</verbatim>

Submit qsub jobs same as you would on the bil cluster.

---+++ Thank you to Sarang for all these files/scripts/tips!

-- Main.JohannaZumer - 10 Oct 2006 -- Main.AdrianGuggisberg - 10 Aug 2007

Retrieved from

"http://nutmeg.berkeley.edu/index.php?title=Compiling_Matlab_code_for_cluster_use"

- This page was last modified 08:53, 28 July 2009.
- Content is available under Attribution-Noncommercial-Share Alike 3.0 Unported.