**1. Get the code**

* Register for the wiki (<http://forge.ipsl.jussieu.fr/nemo/wiki/Users>) by sending an email containing your username (5 characters minimum length) to [nemo@forge.ipsl.jussieu.fr](mailto:nemo@forge.ipsl.jussieu.fr);
* Get the conformation email then reset your password;

**2. Compile NEMOGCM for a specific experiment**

* Find the bash file build\_sisu.bash and make a copy

>> cp build\_sisu.bash my\_build\_sisu.bash

* Insert your user name in my\_build\_sisu.bash and save

$USER=”YOUR\_USER\_NAME”

* Execute the bash file

>> sh my\_build\_sisu.bash

* Type your password on the screen and type yes when asked. Now the bash file make the changes in certain configuration files and compile them (please read the bash file and the instructions in the [wiki](http://forge.ipsl.jussieu.fr/nemo/wiki/Users/ModelInstall#Setupyourarchitectureconfigurationfile) if you want to know what has been done)

Now you have built an executable, i.e. nemo.exe, for the experiment GYRE\_PISCES

**3. Run an experiment**

* Copy the executable to the experiment directory first.

The test case is GYRE\_PISCES

>> cp $TMPDIR/MY\_GYRE/BLD/bin/nemo.exe .

* Then
* For a short parallel run with 4 nodes

>>aprun –n 4 ./nemo.exe

* For real experiment submit jobs through SLURM batch queue system. Change job names etc. in bash\_job.sh first

>> sbatch bash\_job.sh

* Monitor your job (optional; you can try either of them)

>>squeue –u YOUR\_SISU\_USER\_NAME

or

>>tail –f YOUR\_SLURM\_JOB\_FILE

or

>>tail –f solver.stat

**4. Clean nemo build**

* Clean a bad configuration

>> ./makenemo –c YOUR\_CONFIG clean\_config

* Uninstalling (Clean up the whole thing?)

>> ./makenemo -n YOUR\_BUILD clean

(e.g. >> ./makenemo ./makenemo -t $TMPDIR -m XC40-SISU -n MY\_GYRE\_XIOS clean)

**5. Using XIOS2.0**

* you need to add new keys in .fcm file, e.g. the line ‘key\_nosignedzero key\_xios2.0’ in the end of the line in file cpp\_GYRE\_XIOS.fcm when doing GYRE\_XIOS experiment

>> sed -i 's/$/ key\_nosignedzero key\_xios2.0/' GYRE\_XIOS/cpp\_GYRE\_XIOS.fcm

**5. Experiment ORCA2\_LIM3**

* download data
* add keys
* copy \*xml files in GYRE\_XIOS/EXP00 to ORCA2\_LIM3/EXP00

**6. Run output**

* Find the errors and run statutes in ocean.output
* Use ncview (now it is only in Petteri’s directory) to check the results

>> cd /homeappl/home/puotila/bin

>> export HDF5\_DISABLE\_VERSION\_CHECK=1

>> ./ncview /YOUR\_OUTPUT\_DIR/#\_5d\_00010101\_00011230\_grid\_T.nc &

(

**6. Debugging**

* Check ocean.output