

How to get Gadget2 running on the SUN cluster at CHPC: The Walkthrough

Vuyani Mampofu

January 9, 2013

Before doing anything you need to create an installation directory (e.g. installations) on your /export/home/username directory. This will help you to avoid any super user conflicts. After doing that you need to type 'module load sunstudio' and 'module load clustertools' (these will help in loading openmpi and other tools required to compile Gadget). Note that PATHINSTALL refers to /export/home/username/installations.

Building Gadget

After logging in type the following commands:

- module load sunstudio
- module load clustertools

Install GSL

The Gnu Scientific Library is available as a download from the internet. To install it follow these steps:

- tar -xvf tar.gz file and cd to gsl directory
- ./configure --prefix=/PATHINSTALL
- make
- make install

Install FFTW-2.1.5

For this step you need to use FFTW-2.1.5 because version 3.xx does not work with Gadget2. Follow these steps to install:

- tar -xvf tar.gz and cd to fftw directory
- ./configure --prefix=/PATHINSTALL --enable-type-prefix --enable-mpi
- make
- make install
- make clean
- ./configure --prefix=/PATHINSTALL --enable-float --enable-type-prefix --enable-mpi

- make
- make install

Install HDF5

You need to install HDF5 version 1.6.x to avoid the latest changes in the HDF5 API. To install HDF5 you will first need to install SZIP. To install SZIP download SZIP from the internet and follow these steps:

- tar -xvf tar.gz file and cd to szip directory
- ./configure --prefix=/PATHINSTALL --enable-shared=no --enable-static=yes
- make
- make install

Now build HDF5 by following these steps.

- tar -xvf tar.gz file and cd to hdf5 directory
- ./configure --prefix=/PATHINSTALL
- make
- make install

Compile Gadget

To compile Gadget you need to edit the Makefile found in the Gadget2 directory. In the Makefile you will need to add CHPC" to the systems that are available there and comment out all the other SYSTYPES. This is what the CHPC system should look like:

```
ifeq ($(SYSTYPE),"CHPC")
CC      = mpicc
OPTIMIZE = -O3
GSL_INCL = -I/export/home/vmampofu/packages/installations/include
GSL_LIBS = -L/export/home/vmampofu/packages/installations/lib -Wl,
"-R /export/home/vmampofu/packages/installations/lib"
FFTW_INCL= -I/export/home/vmampofu/packages/installations/include
FFTW_LIBS= -L/export/home/vmampofu/packages/installations/lib
MPICHLIB =
HDF5INCL = -I/export/home/vmampofu/packages/installations/include
```

```
HDF5LIB = -L/export/home/vmampofu/packages/installations/lib -lhdf5
endif
```

In this case /export/home/vmampofu/packages/installations is where I installed all the packages for Gadget. Now you need to test if Gadget is working. To do that you need to follow the following steps:

- In the Gadget2 directory run ./Gadget2 you should see a message that says "Gadget2 reports that it is a parallel code and that the parameter file is missing". This means you need provide a parameter file.
- Create a directory in your scratch directory call it lcdm_gas (I use the lcdm_gas example tha comes with Gadget2).
- Locate the 'lcdm_gas.param' in the parameterfiles directory in Gadget2 and copy it to where the Gadget2 executable is.
- Edit the lcdm parameter file.
 - edit the InitCondFile line to specify where the initial conditions file is (use 'lcdm_gas.littleendian.dat' in the Gadget2 ICs directory).
 - edit the OutputDir line to specify the output directory (in this case its the lcdm directory you created above).
- run Gadget with mpirun -np 12 Gadget2 lcdm_gas.param. If it is running Congratulations!!!!