**Software and server access – UNC Users**

Step 1: Get a dogwood account

Getting an Account

You can visit the [Onyen Services](https://onyen.unc.edu/cgi-bin/unc_id/services) page, https://its.unc.edu/onyen-services/, then click on the **Subscribe to Services** button and select **Dogwood** account through Onyen Services you may see the following message

You are ineligible for the following services for the reasons cited:

KillDevil Cluster: missing '(LIVE|EXCH)' prerequsite.

In this case, you should send an email to [research@unc.edu](mailto:research@unc.edu) requesting an account on Dogwood. In the email, please do indicate that you were not able to subscribe for a Dogwood account through Onyen Services and include the following information in your request:

* Onyen
* Your “@email.unc.edu” email address
* Full name
* Campus address
* Campus phone number (if any) and number where you can be reached while running jobs
* Department you are affiliated with (the one relevant to the work you will do on Dogwood).
* Faculty sponsor’s (PI) name (and onyen if known) if you are not a faculty member
* A description of the work you expect to do on Dogwood

You will receive an email notification once your account has been created. When requesting a Dogwood account, do not request to be added to a group. If you know you need to be added to a group, request that in an email to [research@unc.edu](mailto:research@unc.edu).

Step 2: Get a terminal

**On a Mac**

Use your spotlight finder in Mac and search for terminal and open it. You will then want to type ssh onyen@dogwood.unc.edu

**On Windows**

Probably the best program to use is Putty. You can download it here:

<http://www.putty.org>

You will want to log into dogwood.unc.edu using port 22 with your onyen and password.

Step 3: VPN installations and clients to login off campus

Please follow the step-by-step instructions here:

<https://help.unc.edu/help/vpn-installation-and-clients/>

Step 4: SFTP (moving files between local computer and killdevil)

I recommend using filezilla to move your files between your computer and the cluster.

<https://filezilla-project.org>

Step 5: Download and install Visit Visualization

<https://wci.llnl.gov/simulation/computer-codes/visit/downloads>

Step 6: Make sure you have a text editor that won’t add random hidden characters (like Word)

I like Notepad++

<https://notepad-plus-plus.org>

And Sublime

<https://www.sublimetext.com>

Step 7: Make sure you have MATLAB installed

**Running IBAMR/IBFE on super computers!**

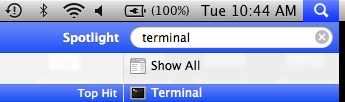
(…or how to run simulations on Dogwood)

adapted from Nicholas Battista, Aug. 2014

**Introduction:**

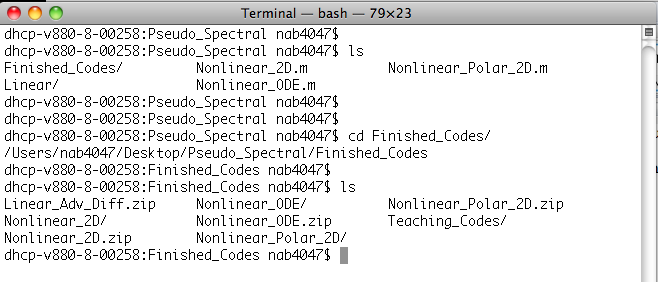
Running simulations on the clusters is a very valuable tool in scientific computing research, especially when the code is parallelized and has the ability to run on multiple processors. This allows the code to be partitioned into parts that can all be run simultaneously and thereby finish running theoretically a lot sooner than if it was being run on a serial machine.

**Part I: Loggin’ in, submittin’ jobs, checkin’ in on simulations, and all that jazz…**

1. First I will begin this tutorial by giving a step-by-step guide to logging into Dogwood and where to actually submit jobs and run simulations.
2. The first thing is that we have to *remotely* log into Dogwood through the terminal. (MacOSX or Linux platforms) or using Putty (Windows).
3. Use your spotlight finder in Mac and search for terminal and open it.
4. Once the terminal is opened, we can begin searching through directories (folders) and files through your local personal computer, i.e., it is a way to search through your computer without using a mouse, but moreover you can create, copy, move, remove files, and open programs.
5. The following are a few helpful commands you can use in the terminal (or command-line).



1. For example I’ll illustrate a few of these commands below,



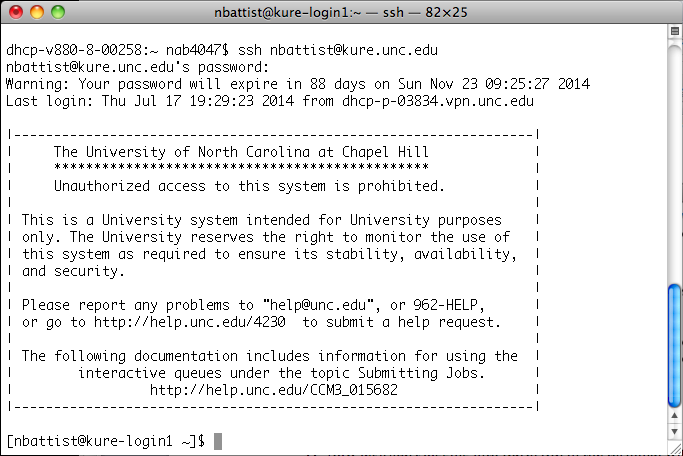




1. Now let’s start logging into Dogwood. In the terminal window, type

ssh [your onyen]@dogwood.unc.edu

and click enter. Now you will be prompted to enter the password associated with your UNC onyen.



1. Now you should be at the screen listed above. Congratulations - you are now remotely connected to a supercomputer! If you type “pwd” you should be listed in a similar directory to the one below, but with your associated onyen.

Macintosh HD:Users:nab4047:Desktop:Screen shot 2014-08-26 at 11.17.47 AM.png

1. A note for remote users: You will need to use a VPN client to access Dogwood if you are working off campus. You can obtain it here:

https://software.sites.unc.edu/shareware/#v

(Note: If you have the Cisco client already installed for another connection, it may not let you switch between the two.) You will also need DUO (dual step authentication.) You can connect it to your home phone, but your best bet is the app for the IOS or Android. See their respective app stores.

1. It does not show it, but you are currently logged into the *login node* on the cluster. This is essentially a processor that all users use when they first log into Dogwood. Now this is very important, but repeat after me – **WE DO NOT RUN SIMULATIONS ON THE LOGIN NODE ☺** To run simulations we must change directories to our *netscr* directory, as seen below. To change directories type:

cd /21dayscratch/scr/o/n/onyen/

1. In your home directory you also have access to ms (mass storage). You can permanently store data there, up to 1 TB. Do not run programs out of this directory.

To access Mass Storage from Dogwood, type:

cd ~/ms

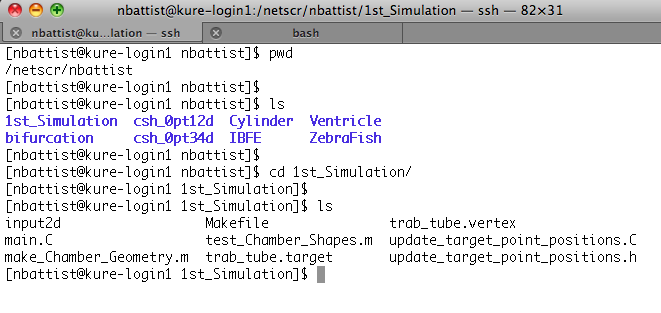
Any files in the scratch space that you wish to save, can be moved to the mass storage preferably in tar or zip format.

1. Change directory to 21dayscratch. Please be aware that whatever you put in this directory is erased after 21 days, it is temporary! Now that we are in the correct directory for *submitting jobs* (i.e., submitting a simulation request to the Dogwood queue to run our code for us), we can begin running codes!
2. It is now time to learn how to submit jobs to run. Well let’s not get too excited, we have a little more work to do before we can do that. Namely, we must copy the simulation we wish to run to Dogwood to actually be able to run it! We will (cleverly) run a simulation called *1st\_Simulation*. (Please see the associated .zip file). It is a 2D IB example of a plate stabbing a rubber band.
3. Back on your local machine, open another terminal window. Change directories until you are on your Desktop/IBAMR\_ON\_DOGWOOD/ folder.
4. Now that we are in the right folder, */Desktop/IBAMR\_ON\_DOGWOOD/* , we can begin to copy the simulation folder we want to our *netscr/[onyen]* directory on Dogwood. Type the following command

scp –r 1st\_Simulation [your onyen]@dogwood.unc.edu: /21dayscratch/scr/o/n/onyen

The *scp* stands for secure copy. The *–r* stands for a recursive flag, which means it is going to copy everything in the folder with tell it and all of the subfolders that folder contains. Next the *1st\_Simulation* tells the system what folder or file you wish to copy, in this case, the folder called *1st\_Simulation* that contains the required files to run a simulation using IBAMR. Then finally the [your onyen]@dogwood.unc.edu: /21dayscratch/scr/o/n/ tells the system where you wish to copy the file to. In this case it says you want to copy the file to a remove computer called *[your onyen]@dogwood.unc.edu* and it says to copy the file in a specific location on that computer called /21dayscratch/scr/o/n*/[your onyen]*. Note, the colon, : , is necessary for telling it a specific place.

1. After you type that command and press “enter”, you will be prompted for your UNC onyen password. Enter it. And you should see:



1. Now go back to your other terminal, the one which is connected to Dogwood. You should be in your /21dayscratch/scr/o/n*/[your onyen]* directory. Try typing *ls* to see what folders are now in your directory and the *1st\_Simulation* directory should be present. (Note: in the picture below there are more in mine since I have more folders in my personal *netscr/nbattist* directory.)
2. Next we must load the ibamr module. Please type the following in the command window

[lam9@dogwood-login1 769\_examples]$ module add ibamr/2018-03

Now check that you have the right modules loaded by typing

[lam9@dogwood-login1 769\_examples]$ module list

Currently Loaded Modules:

1) openmpi\_2.0.3/gcc\_4.8.5 2) ibamr/2018-03

You can read more about modules and commands on dogwood here:

https://help.unc.edu/help/getting-started-on-dogwood/

1. Now we are ready to begin our journey of “making” *main2d* (or *main3d*) executables and submitting our jobs to be run on Dogwood! We will not worry about all the files and what their purpose is now – we will just assume they are all necessary files to run a simulation using IBAMR.
2. Now we will “make” the executable. Simply type

make clean

make main2d

and wait for the code to finish running. This is essentially compiling our code, making sure there are not syntax mistakes, and connecting all necessarily bits of software to one another.

Note: There will what appears to be a ton of information that gets printed to your terminal screen. What is important is for it all to end without saying it has found any error. Furthermore, if there is a warning present, that is fine for our purposes (hopefully).

1. Now if you type *ls* you will see a few files have been added – main2d, main.o, stamp-2d, and update\_target\_point\_positions.o. For our considerations, we don’t really need to pay much attention to these files as long as the *make main2d* went smoothly and finished compiling.
2. Now it’s finally that time- let’s submit our first job to the Dogwood queue! We will submit a job using sbatch. You can check the status of your submitted SLURM jobs with the command “squeue -u ” (note squeue shows jobs from all users, provide your onyen to just show your jobs). If you need to kill/end a running job, use the “scancel” command: scancel [JobID]
3. We will need to create a bash script using our favorite editor. If you don’t have a favorite editor, use nano (for now).

nano example.sh

The script contains job submission options followed by application commands. Please enter the following into your script as an initial example (Note: that each SBATCH switch below has two ‘-‘ characters, not one):

#!/bin/bash

#SBATCH --job-name=first\_slurm\_job

#SBATCH --ntasks-per-nod=16

#SBATCH --time=1:00:00

#SBATCH --mem=1000

echo "Hello SLURM"

Save your file and exit nano. Submit your job using the sbatch command:

sbatch example.sh

You have created a script, example.sh that will ask for 1 core and 100MB of memory for 1 minute. It will name the job “first\_slurm\_job” and run the echo command. If no Slurm options are specified then the defaults are:

-p general = “general” partition

-n 1 = 1 CPU processor

--mem-per-cpu=4800 = memory per CPU in MB

-t 60 or -t 1:00:00 or -t 0-01:00 or --time=1:00:00 = 60 minutes or 1 hour run time.

-t 1-0 for 1 day

1. Here is a script specific for an IBAMR simulation.

------------------------------------------

#!/bin/bash

#SBATCH --job-name=first\_ibamr\_job

#SBATCH --ntasks-per-node=16

#SBATCH --time=2:00:00

#SBATCH --mem=1000

#SBATCH --partition=debug\_queue

#SBATCH --output IBFE2D.out

mpirun ./main2d input2d

Note that there are a number of partitions to which you can submit. Some have minimum numbers of ntasks and run times. You can look up the information here:

<https://its.unc.edu/research-computing/techdocs/dogwood-partitions-and-user-limits/>

-----------------------------------------------------

When submitting, it is important for to jot down the JOB ID#, and keep a good record of when the simulation was submitted as well as a little bit about the simulation you are running (i.e., what simulation are you running, what is the Re, what is the time-step, the viz-dump interval, etc).

1. All that is left is to wait! Sort of. There are a few things we can do while it’s running

We can continually check to see if the simulation is still running by typing *squeue -uonyen* in the terminal window when we are logged into Dogwood. Note that we do not have to be in the same folder to see if a simulation is still running. What we will need is just the JOB ID#.

1. We can also check other ways to see how the simulation is doing, i.e., how far it is into the simulation, etc. This time you MUST be in the folder containing the information for the specific simulation you are interested in.

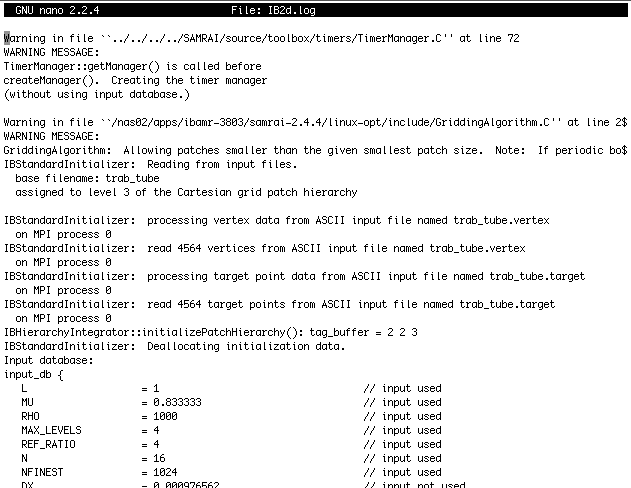
There are two ways to see how the simulation is doing. The easiest solution is to type *ls* and see that there should be a file called *IB2d.log*. This log file prints out information for each time-step such as what the current simulation time and time-step is, how many iterations it takes for the Stokes solver, the CFL number, etc. All you will need to keep track of is seeing which iteration and time-step it is on. 

**22.** Now to open the IB2d.log file we can use one of many editors. The editors I traditionally use are either *vi* or *nano*. I will illustrate an example of both and say the commands to scroll through the file with each editor now.

First let’s start with *nano*. To open the *IB2d.log* file with *nano* just type

nano IB2d.log

into the terminal window and hit *enter*. This will bring you to a screen that looks like the following,



You can scroll through the *IB2d.log* file using the ‘arrow’ keys on your keyboard. Doing this method you can essentially go character by character across a line or up and down one single line at a time.

Scrolling through the entire *log* file this way would take forever. Luckily for us there are very helpful commands in *nano* for scrolling and editing text (it is afterall a text editor.)

If you notice at the bottom of the editor there are keyboard shortcuts (useful commands) to use in *nano*. Macintosh HD:Users:nab4047:Desktop:Screen shot 2014-08-26 at 5.06.35 PM.png

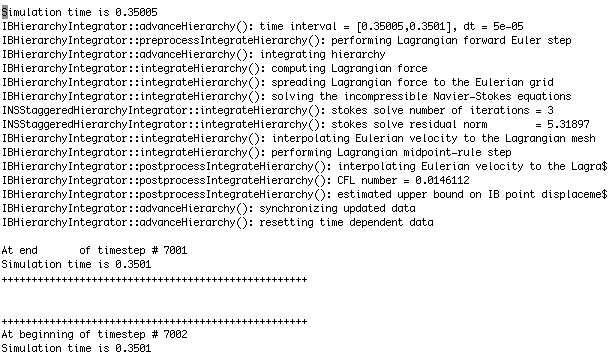
To scroll a page at the time you hit the following buttons on your keyboard together,

Ctrl + v : scroll down a page

Ctrl + y : scroll up a page

Ctrl + w + v : scroll to the bottom of the file

Ctrl + w + y : scroll to the top of the file

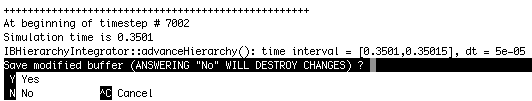
For instance if you want to scroll to the bottom of the file to see how far it’s gone you press *ctrl + w + v*  and then will probably see something similar to the following picture

Now the last thing we have to do in *nano* after we’ve seen how much of the simulation has finished running, is exit. I’ll make a quick note here to say we can of course change text files in *nano*, i.e., actually using the editor to, you know, edit something, but for our purposes here we just want to see what is in the *IB2d.log* file.

To exit simply press

*ctrl + x*

and you may be prompted by something that asks if you wish to save the modified buffer, as seen below,



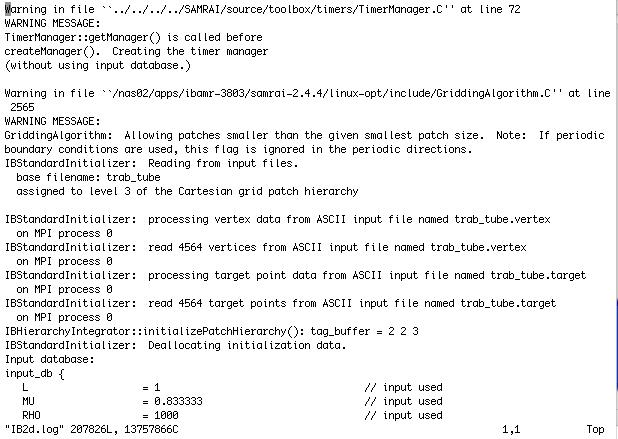
simply type ***n***to not save any changes you have made.

**24.** Likewise, we could also check the *IB2d.log* file using a different editor called ***vi*** or ***vim***. It is more of a personal preference to use whichever editor you feel more comfortable with.

To open the *IB2d.log* file with ***vi*** type in the terminal,

vi IB2d.log

and the following screen should appear (it may take a few moments for the editor to open the log file), which looks strikingly similar to the screen we saw in *nano.* IT SHOULD BE – it is the same log file after all.



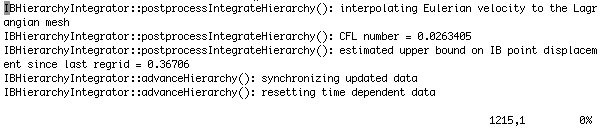
The first thing you might note is that the ***vi*** editor does not list a bunch of keyboard shortcuts at the bottom. Now as has been described to me before by some computer science graduate students, ***vi*** and ***vim*** are the more cool editor to use because they are more cryptic, professional, and “cool”. Not sure what they mean by that, but luckily for us, there are a bunch of keyboard shortcuts – they just are not as conveniently located at the bottom of the editor such as is the case with *nano.*

If you type a ‘ : ‘ followed by a number, i.e.,

:1215

and hit *enter* it will scroll to line 1215 in the file. If there are not 1215 lines in the file, it will scroll to the last line. This is how you can scroll to the bottom pretty easily, simply type “ : “ and a very large number after it.





To make a change in ***vi*** you may have noticed you cannot simply open the file and start typing away. You must first tell the ***vi*** editor to go into *write* mode. To do this hit the *esc* key. You can now write all you wish.

To exit the file there are a few commands you can do

*esc* then *:wq* [exit with writing to the file]

*esc* then *:*q [exit without writing to the file if nothing was changed]

*esc* then *:q!* [exit without writing to the file, even if characters were changed but don’t want to save them]

**25.** Now that you can check on simulations to see how far they have run, you can write down the time and how far the simulation has run. Other than that, sit back, submit more jobs to the queue on Dogwood, and wait to do the analysis on the simulations.

**Part II: Getting data off of Dogwood**

1. Once the simulation has finished running you will want to copy all the data from the cluster to either your local machine or external hard drive (off of your local machine).

I want to stress TWO things at this point. The first is that you will want to copy all the data (basically the whole simulation folder) shortly after (at least within a week) of the simulation finishing. This is because although Dogwood has a lot of storage in the *scr* folder, it recycles files if they have not been updated or changed for a while – meaning, it will delete everything after some time. It is good practice to try to copy all the files within a few days at most of the simulation finishing.

The second thing is that you will want to store these files on an external hard drive. Most of these simulations typically will range in sizes between 500mb and 4gb of data or more. That will fill your local machine’s hard drive pretty fast.

1. Now to actually copy the files from the cluster to your local machine (or external hard drive), it will be very similar to how we copied the folder *1st\_Simulation* to the *21dayscratch/scr/o/n/[your onyen]* directory to begin with.

Before we copy anything, go back to the other terminal that is pointing at the */Desktop/IBAMR\_ON\_DOGWOOD/* folder. (Or open another terminal window and *cd* to that directory for some fun practice in the Linux/Unix environment).

1. To copy the files off of Killdevil, type

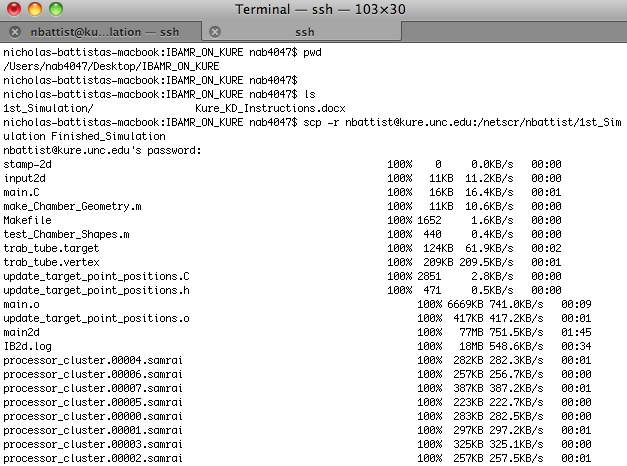
scp –r [your onyen]@dogwood.unc.edu:/21dayscratch/scr/o/n/[your onyen]/1st\_Simulation Finished\_Simulation

Like before, the *scp* is protocol to secure copy a file or folder. The *–r* tells the system to recursively copy all the folders and files contained within the folder we’re telling it to copy. The next part [your onyen]@dogwood.unc.edu:/21dayscratch/scr/o/n/[your onyen]/1st\_Simulation tells the system what directory you wish to copy. Finally the last part *Finished\_Simulation* tells the system what to rename the directory where you are copying it to.

You will be prompted to enter your UNC onyen password and then it should start copying away!

**NOTE:** You do not have to wait until the simulation has finished running to copy the data if you wish to check to see if the simulation is performing what you actually want it to.

1. You will likely see something that starts to resemble the following picture,



1. This process will take a few minutes before everything copies into the IBAMR\_ON\_DOGWOOD folder. Once the copying is complete, we can now begin to visualize the simulation!

Please see lecture 1 for how to visualize this data!