Lecture 1B – Computing Resources

- **Wopr/Vortex clusters**
  - 210 processors (vortex)
  - 12 processors (wopr)
  - Linux operating system
  - Gigabit Ethernet communication
  - Graduate student research
  - We will be using the WOPR system in this class. You should have accounts on this system by now.

- **Davistron desktop-cluster**
  - Built at UCD by students
  - 8 second-generation processors
  - Linux operating system
  - Gigabit Ethernet communication
  - System lasted approximately 5 years before becoming obsolete

Motivation for Building DAVISTRON

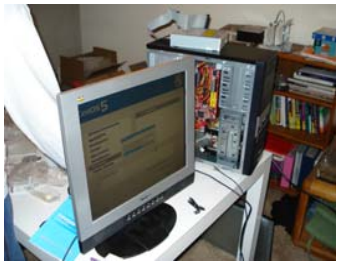
by Michael Ahlmann, Nick McGuire, and Matthew Fife
(former graduate and MAE267 students)

- **Providing a computational resource for future students**
- **Demonstrating the use of open-source solutions for CFD research**
- **Gaining a practical understanding of HPC hardware**

Original Hardware

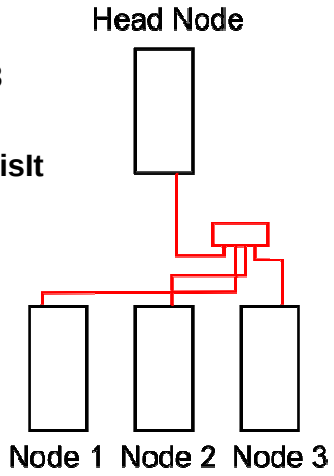
4x	**Barebones Computer with k9VGM-V Motherboard**	**$291.96 ($72.99 each)**
1x	**Samsung DVD Drive (SH-S203B)**	**$29.99**
4x	**AMD Athlon 64 X2 4000+ Brisbane 2.1 Ghz Processor (Dual Core)**	**$279.96 ($69.99 each)**
4x	**2Gb Corsair Value Select 240-Pin DDR2 Memory (PC2 5300)**	**$199.96 ($49.99 each)**
4x	**80 Gb Western Digital Caviar 7200 RPM Sata 3.0 Gb/s Hard Drive**	**$171.96 ($42.99 each)**
5x	**TRENDnet FastE Ethernet Card**	**$29.95 ($5.99 each)**
Total		**$1,003.78**

Construction



Configuration

- **Operating system :: Rocks 5.2**
- **MPI Distribution :: Open MPI 1.3**
- **Compiler :: Gnu/PGI**
- **Visualization :: ParaVIEW and VisIt**
- **Internode Communication :: Gigabit Ethernet**



Logging into WOPR

- **You will need to obtain secure shell (ssh) or putty in order to log into any of our parallel systems**
  - This software uses encryption to protect the systems and users from malicious attacks
  - WOPR is located at wopr.engr.ucdavis.edu
  - Each of you should be able to log into WOPR under your own account via the secure-shell utility (an encrypted version of telnet)

    ssh wopr.engr.ucdavis.edu
  - You can monitor the status of WOPR by going to wopr.engr.ucdavis.edu on the web
- **You will be located in your "home" directory when you first enter WOPR,**

  /home/"your username" e.g. /home/davisrl

Common Linux Commands

- **cd** :: **Go to home directory**
- **cd ./...** :: **Go to directory "..." in current folder**
- **cd ..** :: **Go up one directory from current folder**
- **cd /...** :: **Change to absolute directory "..."**
- **ls** :: **List contents of current directory**
- **ls -lh** :: **List file permissions and file size**
- **du -h** :: **List size of current directory contents**
- **df -h** :: **List available space on all installed drives**

- **Other basic Linux commands are located at smartsite under the "Additional Material" folder**
  - BasicLinuxCommands.pdf

Creating Directories, Programs, Files

- **Once you have successfully logged into WOPR, you should create your own directory in which you will create computer programs and run**

Create directory:	mkdir "codename"	where codename is something like hw1
Enter directory:	cd "codename"	set directory to hw1

- **Programs/files can be created using one of many editors. Examples include "vi", "emacs", "edit". There are advantages/disadvantages to each.**

  vi mfp.f

  emacs mfp.f

  edit mfp.f

Compiling a Fortran Program

- **Each computer vendor typically has its version of a Fortran compiler that converts the Fortran into machine language**
  - Examples include:
    - IBM – xlf90
    - SGI – f90
    - Linux – pgf90, gfortran
- **Programming, compiling, and linking a program is usually done on the "front-end" computer, i.e. WOPR**

Compiling a Fortran Program

- **We have 2 compilers on WOPR**
  - Portland Group, pgf90, pgcc, etc. (commercially available)
  - GNU, gfortran (open source)
- **These compilers are located at**
  - **Serial Execution:**
    - pgf90: /share/apps/pgi10.9/linux86/10.9//bin/
      - For pgf90, pgf95, pgcc, pgCC basic compiling (without MPI bindings)
    - gfortran without (or with) MPI bindings: /opt/openmpi/bin
      - For mpif90, mpicc, etc.
  - **Parallel Execution:**
    - pgf90 with MPI bindings: /share/apps/openmpi-pgi/bin
      - For mpif90, mpicc, etc.
    - gfortran with MPI bindings: /opt/openmpi/bin
      - For mpif90, mpicc, etc.

Compiling a Fortran Program

- **Once you have created the program, you may compile it on WOPR:**

  pgf90 –c mfp.f for the Portland Group compiler
  mpif90 –c mfp.f for the GNU compiler

- **This will create an object, mfp.o If a program consists of several objects (subroutines) along with the main program, they can be "linked" together**

  pgf90 –o mfp mfp.o, sub1.o, sub2.o…
  mpif90 –o mfp mpf.o, sub1.o, sub2.o…

- **This will create the executable "mfp" which can be run by just entering**

  ./mfp

- **However, when we go to run batch parallel jobs, I would like for you to use the SGE job submit procedure**

Using MAKEFILES

- **Compiling and Linking to create executables can be all performed in a script called a "makefile"**
- **An example makefile for "codename" has been put on smartsite under the "codes" folder**
- **If your makefile is simply named "makefile", then you can run it by typing "make"**
- **If your makefile is named something else (eg makefile_code) then you can run it by typing "make –f makefile_code"**

Sun Grid Engine (SGE)

- SGE is a job scheduling tool that allows multiple users to submit batch jobs without having to worry about running on top of each other
- Jobs are prioritized based on the number of CPU's requested and the amount of time the job has been in the queue

- qsub jobname :: submit job "job name"
- qstat -u * :: show status of all jobs for all users
- qdel :: delete job
- qhost :: show compute hosts in queue

Additional Examples

- The WOPR Example Directory

  /share/apps/Examples

  Contains a few examples of makefiles and job submission scripts. First, copy this to your home directory.

  *cp -r /share/apps/Examples /home/username/**

Serial makefile explained 1/2

```
File  Edit  View  Terminal  Help
#Sample Makefile for Bounce
#Written by Michael Ahlmann
#Written for MAE267
#August 22, 2009

#Assumptions
#       1) Prorgram is written in fortran
#       2) All files are in current directory
#       3) All files have .f extension

#Set Compiler Flags (to gnu version of openmpi)
FC = /opt/openmpi/bin/mpif90

#Set Optimization or Debug Flags
#     -fast = Full Optimizations
#     -g    = Debug Mode
OPTMZ =

#Set Compiler Flags
#       -c        = Compile Only Don't Link (Required)
CFLAGS = -c

#Set Compiler Libraries
LIBS =

#Set Program Name
PROG = plot3d_sample

#List Object Files
#       This section should include all files to be
#       compiled with the .f90 extension replaced with
#       a .o extension. Use \ at the end of each line to
#       extend across multiple lines
OBJS = datagen.o \
       plot3d.o
```

Use absolute path names to compilers when possible.

mpif90 is a wrapper compiler that points to either the gnu or pgi fortran compiler and links to the mpi libraries. It is not necessary to use mpif90 for serial programs, but it does not hurt to do so.

Program name can be whatever you want. Just make sure it is consistent with the name in your SGE run script.

Multiple object files can be listed at once.

Serial makefile explained 2/2

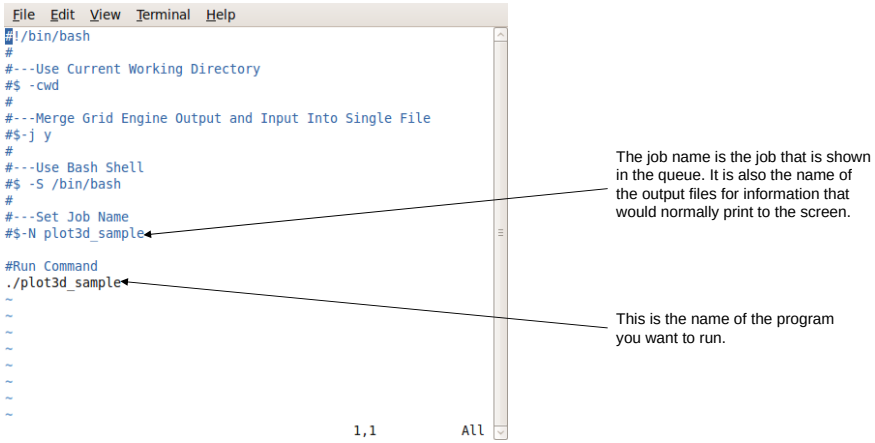
```
File  Edit  View  Terminal  Help
#List Modules (follow same rules as for objects)
MODULES =

#Command Telling Make to Compile Program
$(PROG): $(MODULES) $(OBJS)
        $(FC) $(MODULES) $(OBJS) $(OPTMZ) -o $(PROG)

#Command Telling Make to Compile Modules
$(MODULES): %.o: %.f90
        $(FC) $(CFLAGS) $(OPTMZ) $< -o $@

#Command Telling Make to Compile Object Files
$(OBJS): %.o: %.f90 $(MODULES)
        $(FC) $(CFLAGS) $(OPTMZ) $< -o $@

#Command to Clean Directory if User Desires
clean:
        rm -f *.o
        rm -f *.mod
        rm $(PROG)
        rm *.xyz
        rm *.flo
        rm plot3d_sample.o*
        rm plot3d_sample.po*
```

Module files can be listed here

This section should not need to be changed. If you are interested in the details of how these commands work, either look it up on google or e-mail questions to davisrl@ucdavis.edu

The clean command is not required, but it can be convenient to clean up files from compiling or running your code. Simply run *make clean* to invoke the code from this section.

Serial Job Submission Script



Other Useful (Free) Tools

Plotting

ParaVIEW : Good for visualization of plot3d files.
Available from www.paraview.org
VisIT : Good for visualization of CGNS files. Can visualize plot3d files, but a quick script is required (look in plot3d example folder for a sample script). Available from www.llnl.gov/visit.
TechPLOT: Good for contour or line plots and animations.
Available from Jacob in MAE.
FIELDVIEW: Good for contour, line, and animations.
Available from Jacob in MAE.

Text Editing

Textpad : Good for viewing source code on a windows computer.
Available from www.textpad.com
Text Wrangler : Good for viewing source code on an Apple computer.
Available from www.barebones.com/products/TextWrangler/
Kate : Good for viewing source code on a linux computer (install via Yum)

SSH and File Transfer

Cyber Duck : File transfer on an Apple Available from http://cyberduck.ch/
WinSCP : File transfer on Windows.
Available from www.winscp.net

Homework 1

- **Read Chapters 1- 4 in Fortran 95/2003**
- **Try out your accounts on WOPR**
  - Try logging on using Secure Shell (ssh at www.ssh.org) or Putty (www.putty.org)
  - Learn various Linux/Unix commands to navigate, etc.
    - See BasicLinuxCommands.pdf in Additional Material folder on smartsite
  - Set up directories for your homework and projects
- **Problems**
  - Problems Below Due Friday, Oct. 4
    - You can develop and execute these on the WOPR front-end computer or on the cluster nodes by using the serial job submit procedure

Problem 1

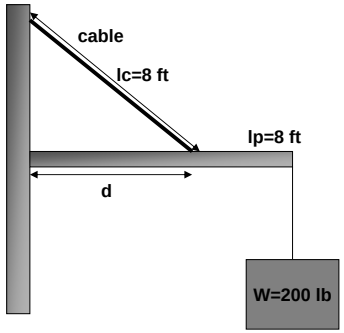
- **Period of a Pendulum**
  - The period of an oscillating pendulum T (in seconds) is given by the equation

$$T = 2\pi\sqrt{\frac{L}{g}}$$

where L is the length of the pendulum in meters and g is the acceleration due to gravity in meters per second squared. Write a Fortran program to calculate the period of a pendulum of length L. The user will specify the length of the pendulum when the program is run. Use good programming practices in your program. (The acceleration due to gravity at the Earth's surface is 9.81 m/sec^2.)

Problem 2

- **Tension on a Cable**
  - A 200 pound object is to be hung from the end of a rigid 8-foot horizontal pole of negligible weight, as shown below:



Problem 2 (cont)

The pole is attached to a wall by a pivot and is supported by an 8-foot cable that is attached to the wall at a higher point. The tension on this cable is given by the equation:

$$T = \frac{W(lc)(lp)}{d\sqrt{lp^2 - d^2}}$$

where T is the tension on the cable, W is the weight of the object, lc is the length of the cable, lp is the length of the pole, and d is the distance along the pole at which the cable is attached. Write a program to determine the distance d at which to attach the cable to the pole in order to minimize the tension on the cable. The program should calculate the tension on the cable at 0.1 foot intervals from d=1 foot to d=7 feet and should locate the position d that produces the minimum tension.