



# NTNU

Norwegian University of  
Science and Technology

## **Introduction to Kongull** **or** **Kongull for Dummies**

Henrik R. Nagel  
Bjørn Lindi  
NTNU IT

# Outline

- **Hardware**
- Software
- Shell Basics
- File Transfer
- Compilers
- Make Files
- Batch Jobs
- Additional Information

# Kongull



# Login and IO nodes

- 5 HP DL385 G6 servers, each with
  - 2 AMD 2431 2.4 GHz 6-core processors
  - 24 Gb ram
  - 2 HP 300Gb 10k SAS system disks
  - 1 HP P410/512MB SAS/SATA RAID controller
  - 1 HP NC522SFP+ Dual Port 10 GbE network card
  - 1 HP FC2242SR 4 Gb PCIe Dual Port fiber channel adaptor (for storage)
- The central switch is coupled to the NTNU network with 2x10 Gb fiber channels.

# Compute Nodes

- 44 HP DL165 G6 servers, each with
  - 2 AMD 2431 2.4 GHz 6-core processors
  - 48 Gb ram
  - 149 Gb SAS 15K system disk
- 49 HP DL165 G6 servers, each with
  - 2 AMD 2431 2.4 GHz 6-core processors
  - 24 Gb ram
  - 149 Gb SAS 15K system disk

# Network

- Network layout is a “Fat Tree”
- Each login/io server is connected to a central switch with 10 Gb Ethernet
- Each compute node is connected to a rack switch with 1 Gb Ethernet

# File System

- The file system today is NFS
- We will soon switch to GPFS

# Outline

- Hardware
- **Software**
- Shell Basics
- File Transfer
- Compilers
- Make Files
- Batch Jobs
- Additional Information

# Software Availability

- In general, the availability of software is guided by:
  - known needs
  - request to [support-kongull@hpc.ntnu.no](mailto:support-kongull@hpc.ntnu.no)
  - functionality that we are familiar with
  - functionality that we need for our own work
- 2 different software package systems:
  - RPM
  - Modules
- Information about new software:  
<http://docs.notur.no/ntnu/kongull.hpc.ntnu.no>

# RPM

- RPM is RedHat's package system and is used on RedHat/Fedora, SUSE Linux, CentOS, etc.
- Overview over available packages: `rpm -qa`
- Information about a package: `rpm -qi <name>`
- We have ~1200 RPMs installed, while there are ~2100 packages installed on a Fedora 12 Linux computer.

# Modules

- The module system allows multiple versions of the same software packages to be installed simultaneously on Kongull.
- Basic commands:
  - module avail
  - module list
  - module load
  - module display

# Exercise

- Login to `kongull.hpc.ntnu.no` with `ssh`.
- See what packages are available through the modules system (`module avail`).
- Load a software package (`module load`).
- See what happened, when the software package was loaded (`module display`).

# Outline

- Hardware
- Software
- **Shell Basics**
- File Transfer
- Compilers
- Make Files
- Batch Jobs
- Additional Information

# Shell Basics

- <http://www.swc.scipy.org/lec/shell01.html>
- <http://www.swc.scipy.org/lec/shell02.html>
- Log in to Kongull and try the commands

# Outline

- Hardware
- Software
- Shell Basics
- **File Transfer**
- Compilers
- Make Files
- Batch Jobs
- Additional Information

# SCP: Secure Copy

```
scp <sourcefile> <destfile>
```

```
scp host:<sourcefile> <destfile>
```

```
scp user@host:<sourcefile> <destfile>
```

```
scp <sourcefile> host:<destfile>
```

```
scp <sourcefile> user@host:<destfile>
```

- Syntax is like `cp`
  - `-r` flag to recursively copy directories
- `man scp` for more options

# Text Files

- The New line character(s):
  - Windows/Dos 013, 010
  - Apple Macintosh OS 9 and earlier 013
  - UNIX (incl. Linux), Apple OS X 10 and higher 010
- Text conversion commands:
  - dos2unix
  - unix2dos
  - mac2unix

# Binary Files

- Kongull is a little endian computer while Njord is a big endian computer
  - Little endian      least significant byte occupies lowest memory
  - Big endian        most significant byte occupies lowest memory
- Example: 34,677,374 (dec.) = 02 11 22 7E (hex.)

	Byte 1	Byte 2	Byte 3	Byte 4
Little Endian	7E	22	11	02
Big Endian	02	11	22	7E

# Exercise

- Copy a Windows text file to Kongull.
- Use the command “file” to check it.
- Convert it to UNIX format using the “dos2unix” command.
- Recheck the file.

# Outline

- Hardware
- Software
- Shell Basics
- File Transfer
- **Compilers**
- Make Files
- Batch Jobs
- Additional Information

# Compilers

- FORTRAN, C and C++
  - Intel
  - Portland
  - Gnu

# Intel Compilers

- Version 11.1 is currently installed
- Load with “module load intel/compilers”
- Overview over options:
  - `icc -help | less`
  - `ifort -help | less`
- Detailed information:
  - `man icc`
  - `man ifort`

# Optimization

- O0 Disables optimization
- O1 Optimize for maximum speed, but disable some optimizations which increase code size for a small speed benefit
- O2 Optimize for maximum speed (default)
- O3 Optimize for maximum speed and enable more aggressive optimizations
- fast Enable -xHOST -O3 -ipo -no-prec-div -static

# Portland Compilers

- Version 8.0-3 is currently installed
- Version 10.3 will soon be available (waiting for license file)
- Load with “module load pgi”
- Overview over options:
  - `pgcc -help | less`
  - `pgf90 -help | less`
- Detailed information:
  - `man pgcc`
  - `man pgf90`

# Optimization

- O0 Disables optimization. A basic block is generated for each language statement.
- O1 **Local optimization**. Scheduling of basic blocks is performed. Register allocation is performed.
- O2 **Global optimization**. -O1 plus additional global optimization (default).
- O3 **Aggressive global optimization**. -O2 plus more aggressive movement of code and scalar replacement optimizations that may not be profitable.
- O4 Same as -O3.

# Exercise

- Write and compile a small test program on Kongull.

```
#include <iostream>
int main ()
{
    std::cout << "hello, world!\n";
    return 0;
}
```

# Outline

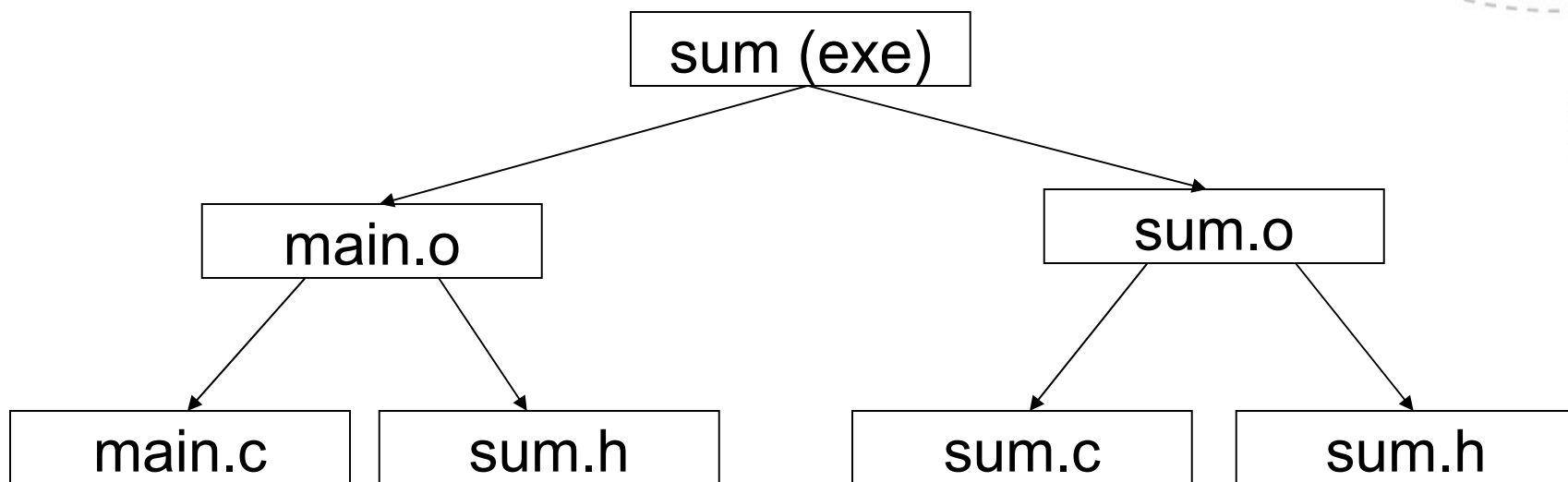
- Hardware
- Software
- Shell Basics
- File Transfer
- Compilers
- **Make Files**
- Batch Jobs
- Additional Information

# Rule Syntax

```
main.o: main.c sum.h  
    gcc -c main.c
```

} Rule

# An example program



# The Makefile

```
CC      = gcc
OBJS    = main.o sum.o

sum: $(OBJS)
    $(CC) -o sum $(OBJS)
main.o: main.c sum.h
    $(CC) -c main.c
sum.o: sum.c sum.h
    $(CC) -c sum.c

clean:
    rm -f sum $(OBJS) *~ *.bak
```

# A Makefile for a Fortran Program

```
FF      = ifort
OBSJS   = func.o global.o main.o sub.o

test: $(OBSJS)
        $(FF) -o $@ $(OBSJS)

%.o: %.f90
        $(FF) -c $<

clean:
        rm -f test *~ *.o *.mod *.bak

# Dependencies
func.o: global.o
sub.o: global.o
main.o: global.o func.o sub.o
```

# Exercise

- Write a make file for your test program.

# Outline

- Hardware
- Software
- Shell Basics
- File Transfer
- Compilers
- Make Files
- **Batch Jobs**
- Additional Information

# Job Scheduling

- OpenPBS – the batch system
- Maui – the job scheduler
- Gold – the accounting
  
- all parties should be able to utilize their fair share of the resources
- underutilized resources may be used by all users

# Queues

Queue	Resources	Availability	Note
bigmem	4 GB / core	IPT, SFI IO and Sintef Petroleum	
normal	2 GB / core	All Users	
optimist	All idle resources	All Users	A job may be killed by a job submitted to normal or bigmem. Checkpointing must be implemented to start from the point where it got killed.

# Job Scripts

- Look at the web-page for kongull for an example:

<http://docs.notur.no/ntnu/kongull.hpc.ntnu.no/batch-job-scheduleing>

# Submitting Jobs

## Command

qsub

qdel <jobid>

showq and qstat

checkjob <jobid>

qsig -sNULL <jobid>

diagnose -n

## Purpose

to submit a job to the queuing system

to delete a job that has been submitted

to get information about the queue situation

detailed information about a submitted job, especially when a job doesn't start

kill jobs that refuse to be killed with qdel

for detailed information about all the compute nodes

# Exercise

- Create a PBS job script for your test program and submit it.

# Outline

- Hardware
- Software
- Shell Basics
- File Transfer
- Compilers
- Make files
- Batch Jobs
- **Additional Information**

# Contact Us

- Ask for help: [support-kongull@hpc.ntnu.no](mailto:support-kongull@hpc.ntnu.no)
  - Also send in requests for software.
  - Request and viewpoints regarding documentation are also welcome.
- NTNU's HPC group offers support to Kongull users

# Notur Resources

- You can also use supercomputers in Norway:
  - <http://www.notur.no/hardware/>

# More information

- See our web-page:
  - <http://docs.notur.no/ntnu>