

Manifold User's Guide

Kenneth L. Smith <ksmith@gravity.psu.edu>

Revision: 1.4
November 6, 2002

Usage of our computer resources is regulated by University policy AD-20.
(<http://guru.psu.edu/policies/AD20.html>)

1 Introduction

Manifold is a 40-node cluster, built on the Intel® Xeon™ architecture, to be used for Numerical Relativity simulations. This document is designed to aid a new user of Manifold in understanding the layout of the machine and hopes to get him or her up and running with minimal fuss.

2 Accounts

2.1 Obtaining an account

See Pablo Laguna <pablo@astro.psu.edu> or Bernd Brüggemann <bruegman@gravity.psu.edu> to obtain an account on Manifold.

2.2 Access

All access into Manifold from the outside will be via `ssh`. All other access is disabled. Internally, the server is known as “head” and all nodes are named sequentially, “node001”, “node002”, etc. Access between the master and slave nodes is provided by `rsh`, although `ssh` is also available.

Those with accounts on any of the NR workstations may also access the contents of the home and bulk directories across NFS (see below).

3 Hardware

This section will detail the hardware of the machine (just in case anyone is interested)

4 Structure

4.1 `/home/manifold`

Your home directory will be `/home/manifold/user`. This directory is NFS shared to all nodes, and to NR workstations. It will be backed up (but as of 30 Oct 2002, it is not). Please observe the policy that your home directory

be used to hold the files and directories deemed necessary - i.e. those which you would mind losing in the event of a catastrophe. **Simulations should NEVER be run from within your home directory.**

4.2 /bulk/manifold

In your home directory you will find a symbolic link “bulk”. This is a link to `/bulk/manifold/user`. This filesystem corresponds to our terabyte (1 TB) RAID array, and is also NFS shared to all nodes and NR workstations. This partition will never be backed up, but the redundancy inherent to the RAID (Level 5) means that data stored here should not be lost. All simulations should be started somewhere under your bulk directory.

4.3 /usr/nrlocal

All NR-related application and packages are installed in `/usr/nrlocal`. It is here that you may find the Lahey/Fujitsu Fortran compiler, the Intel compilers for Linux, Maple, OpenOffice 1.0, and so on.

4.4 /usr/beowulf

Cluster-specific applications, packages, and libraries are found in `/usr/beowulf`. Most important here is the MPICH (<http://www-unix.mcs.anl.gov/mpi/mpich/>) set of executables and libraries necessary to run parallel code on Manifold. The use of MPICH to compile and run MPI code will be explained further below.

5 MPI (Message Passing Interface)

MPI is a widely-accepted library standard for message-passing. Two of the most common implementations of MPI are MPICH (<http://www-unix.mcs.anl.gov/mpi/mpich/>) and LAM (<http://www.lam-mpi.org>). Because of its support for Myrinet, Manifold uses MPICH.

5.1 Compiling and Linking

In `/usr/beowulf`, you will find a subdirectory for each variant of MPICH and the C++/Fortran compilers available. The convention is that the version of MPICH compiled for a regular Ethernet interface is named `mpich.compiler`¹ and the version compiled for the Myrinet interface is named `mpich-gm.compiler`¹. Suffice it to say that most users will want to use the Myrinet version; the Ethernet is there for debugging.

The available options for *compiler* are:

`gcc` : GNU Compiler Collection v2.96 (`gcc,g++,g77`)

`lahey` : Lahey/Fujitsu Fortran Express v6.0 F77/F90/F95 compiler (1f95) w/ `gcc/g++` for C/C++

`intel` : Intel Fortran and C/C++ compiler v6.0 (`ifc,icc`)

Ex. Suppose you wish to compile with the MPICH libraries for Myrinet, using the Intel compiler. Then you may have something like the following, but this is purely schematic²:

¹Actually, `/usr/beowulf/mpich[-gm].compiler` is a symbolic link to `/usr/beowulf/mpich-gm.X.YY.Z.compiler` where X.YY.Z is the current version of MPICH (Currently 1.2.4 for MPICH and 1.2.4..8a for MPICH-GM). The user should always use the directory without the version information as it is guaranteed to always point at the correct up-to-date version.

```

MPICH_DIR      = /usr/beowulf/mpich-gm.intel
MPICH_LIB_DIR  = ${MPICH_DIR}/lib
MPICH_INC_DIR  = ${MPICH_DIR}/include
GM_LIB_DIR     = /usr/gm/lib
LIB_DIR        = ${LIB_DIR} ${MPICH_LIB_DIR}
LIBS           = ${LIBS} -lmpich -lgm

```

For the Cactus users in the crowd, there are a few “config” files available. These can be found on the web at <http://www.astro.psu.edu/nr/computers/manifold/share/cactus-cfg> or on Manifold in `/usr/beowulf/share/cactus`

5.2 Running with mpirun

You can run code in parallel by using a variant of the “mpirun” command. As part of the login scripts, `/usr/beowulf/bin` is added to your path. In `/usr/beowulf/bin`, you’ll find several shell scripts useful for performing common tasks on the cluster such as issuing a command on all nodes, or seeing which nodes are up. You’ll also find there executables named `empirun.compiler` and `mmpirun.compiler`. The ‘e’ or ‘m’ indicates Ethernet or Myrinet interfaces. As before, the *compiler* depends upon your choice. You should use the same interface and compiler choice to run your code as you used to compile it.

Ex. Adding to the earlier example, you compiled your code with the MPICH libraries for Myrinet, using the Intel compiler. To run your code, use⁴:

```
{/usr/beowulf/bin/}mmpirun.intel -np X mycode
```

Options for the `mpirun` command may be found at <http://www.astro.psu.edu/nr/computers/manifold/man/mpich-www/www1/mpirun.html> or by running any `(e|m)mpirun.compiler` with the `-h` argument.

5.3 MPICH Documentation

The full online documentation provided by MPICH is available for reference on the Numerical Relativity website at <http://www.astro.psu.edu/nr/computers/manifold/man/mpich-www>.

6 OpenPBS (Portable Batch System)

6.1 Description

The Portable Batch System (<http://www.openpbs.org>) is a “flexible batch queuing and workload management system”. It will control the finer aspects of negotiating the users’ requests for resources and attempt to ensure that the resources are used to the maximum extent possible. While compilation and interactive jobs may be run on certain designated nodes (currently just ‘head’), the remaining nodes will only be available via the queuing system.

6.2 Queues

At the moment, there is only a single execution queue “default”. This will change within a relatively short time once a policy has been found. At this time, when submitting your jobs, you may either specify the queue explicitly as “default”, or omit any queue request.

²The gm libraries will be necessary to use the Myrinet interface

³By “config” files, we mean those which one uses to set compiler, library, and path variables before configuration via the syntax `(g)make <config-name>-config options=<config-file>`.

⁴As `/usr/beowulf/bin` has been added to your path, specifying it explicitly is superfluous.

6.3 Job Submission

6.3.1 PBS Batch Script

The most common way a user will submit a job is by using a batch script. A PBS batch script is just a regular shell script with PBS commands embedded as comments.

See the example file <http://www.astro.psu.edu/nr/computers/manifold/share/example.pbs> (which is also available on Manifold in `/usr/beowulf/share/`). The options which one can specify in a PBS script are the same as those one can specify to the submission program, `qsub`, on the command line. Please refer to ‘`man qsub`’ or the online version provided at

<http://www.astro.psu.edu/nr/computers/manifold/man/pbs/qsub.1.html> for further information and options.

As a quick reference, here are a few of the more commonly used options:

<code>-l nodes=x:ppn=y</code>	Specify the number of nodes <code>x</code> and the number of processors per node that you want for your job
<code>-l walltime=hh:mm:ss</code>	Specify that you expect your job to last for <code>hh:mm:ss</code> . It’s best that you make this estimate as realistic as possible for efficient scheduling.
<code>-q queueName</code>	Specify to which queue you wish to submit your job
<code>-j oe</code>	Specify that you wish to (j)oin stdout and stderr into one file
<code>-M user@domain.name</code>	Specify to what address email notifications will be sent
<code>-m be</code>	Specify that the user above will be sent an email at the (b)eginning and (e)nd of the job.

6.3.2 `qsub`’ing a job

Once you have tailored a PBS batch script to your specific application, you may submit it to the queue via the command:

```
qsub batch-script
```

You will immediately receive feedback to stdout with the name of your job:

```
jobid.manifold.astro.psu.edu
```

where `jobid` is a unique integer identifier attached to your job for the extent of its execution. Unless you specified the ‘`-k`’ option in your batch script, the output from stdout and stderr will not be available until the completion of your jobs.

6.3.3 Once a job is running...

Much like sending a job to a print queue, you’ll find that for some jobs you’ll want to kill them, others you’d just like to monitor, some should be modified *in situ*. PBS offers you the ability to perform these tasks with the collection of ‘`q`’ commands:

`qalter` Alter a job’s attributes.

`qdel` Delete a job.

`qhold` Place a hold on a job to keep it from being scheduled for running.

qmove Move a job to a different queue or server.
qmsg Append a message to the output of an executing job.
qrerun Terminate an executing job and return it to a queue.
qrls Remove a hold from a job.
qselect Obtain a list of jobs that met certain criteria.
qsig Send a signal to an executing job.
qstat Show status of PBS batch jobs.

6.4 Graphical tools

For the more graphically inclined out there, essentially all of the above tasks from job submission to job alteration can be performed with two graphical tools (written in Tcl/Tk). Between them, `xpbsmon` is the most useful for general use to see what nodes are available and what jobs are currently running. You can typically think of it as an alternative to `qstat`.

`xpbs` GUI front end to PBS commands
`xpbsmon` GUI for displaying, monitoring the nodes/execution hosts under PBS

6.5 PBS Documentation

The man pages for the user-level commands of PBS have also been provided on the Numerical Relativity website at <http://www.astro.psu.edu/nr/computers/manifold/man/pbs>.