Installing MPI in Linux

Abu Saad Papa

This document describes the steps used to install MPICH2, the MPI-2 implementation from Argonne National Laboratory in UNIX (Fedora Core 4) based system. Most of the steps followed here, have been explained in MPICH2 Installer's Guide which is the source of this document.

We need the following prerequisites

1) The tar file mpich2-1.0.5p3.tar.gz (which can be obtained from http://www-unix.mcs.anl.gov/mpi/mpich2/)
2) A C compiler (gcc is sufficient)
3) A Fortran compiler if Fortran applications are to be used (g77 is sufficient)

Both the C and Fortran compiler are present in Fedora Core 4 by default.

Step 1. Create a directory MPI(we can use any name) in the home directory.

```
$ cd $HOME
$ mkdir MPI
```

Step 2. Unpack the tar file.

```
$ tar xfz  mpich2-1.0.5p3.tar.gz
```

The directory MPI will now contain a sub-directory mpich2-1.0.5p3.

Step 3. Choose an installation directory (the default is /usr/local/bin)

```
$ mkdir mpich2-install
```

Step 4. Choose a build directory

```
$ mkdir mpich2-1.0.5
```

Now the MPI directory will contain three sub-directories namely mpich2-1.0.5p3, mpich2-1.0.5 and mpich2-install.

Step 5. Configure MPICH2, specifying the installation directory and running the configure script in the source directory.

```
$ cd $HOME
$ cd MPI/mpich2-1.0.5
$/home/you/MPI/mpich2-1.0.5p3/configure -prefix=/home/you/MPI/mpich2-install
```

For other configure options please refer the MPICH2 Installer's Guide

Step 6. Build MPICH2

```
$ make
```

Step 7. Install the MPICH2 commands.

$ make install

Step 8. Add the bin directory to your path.

$ export PATH=/home/you/MPI/mpich2-install/bin:$PATH

(It is better to add this line in .bash_profile file present in the home directory so that this path gets permanently added once we reboot the system.

$ cd $HOME
$ vi **.bash_profile**
Then append the above command of step 8.)

We can check that everything is in order at this point by doing

$ which mpd
$ which mpicc
$ which mpiexec
$ which mpirun

All should refer to the commands in the bin subdirectory of our install directory.

The MPI has been successfully installed now. We can follow the same steps to install MPI in other machines.

We will have to follow these steps to form a cluster using MPI.

Step 1. We must have a valid host name for each system. The host name localhost.localdomain is not accepted by MPI. We can change the host name by

$ vi **/etc/sysconfig/network**

Step 2. Now we must add the IP address of home machine and other machines in /etc/hosts

$vi **/etc/hosts**

We will get a file like this

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       localhost.localdomain   localhost
```

Here we can add the IP address and hostname's of other machines. Remember to put the home machine at the top. The file /etc/hosts after appending looks like this.

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       localhost.localdomain   localhost
172.16.8.75     system1
172.16.8.76     system2
172.16.5.46     system3
172.16.5.43     system4
```

After these two steps please type the following command else the system will take a lot of time to boot.

$ chkconfig sendmail off

Step 3. Create a file consisting of a list of machine names, one per line. Name this fille **mpd.hosts**.

```
$ cd $HOME
$ vi mpd.hosts
```

These three steps should be followed in all the machines.

The users before starting MPI programs must follow the steps given below.

Step 1. On each system, only once we have to execute these commands.

```
$ touch .mpd.conf
$ chmod 600 .mpd.conf
$ vi .mpd.conf
```

and add

MPD_SECRETWORD=<_____> any word without spaces, which should be same for each user in all the systems.

We should do this so that one user daemon should not communicate with daemons of other users. This mechanism is a must, without which your MPI interface would not start.

Step 2. To coordinate the activities between different systems for each user, each user will be running a daemon named 'mpd' on each system. Among them one daemon will be awarded as Master Daemon, which to be initiated at first. Then other mpd daemons of other nodes are started later as will be shown in step 3.

```
$ mpd &
$ mpdtrace -l
```

It gives hostname_portno (IP address) as the output. Note down the portno, and have to provide to other mpd daemons, which to be run on other systems.

Step 3. Login into other systems, and start mpd daemons as such

```
$ mpd -h <hostname of master daemon> -p <portno> &
```

Step 4. Run

```
mpdtrace -l
```

to check how many systems are included in our loop. It differs for every user, because every user has to launch his/her respective daemons on their chosen nodes.

Step 5. Check your personal multi-processor environment by

```
mpiexec -n 4 /bin/hostname
```

We can now write our MPI programs and Compile any program using MPI constructs by

```
mpicc -o <object file> <source code>.c
```

Execute them on our formulated MPI environment by

```
mpiexec/mpirun -np <i> <object file>
```

Please also refer to the documents MPICH2 Installer's Guide and User's Guide available at http://www-unix.mcs.anl.gov/mpi/mpich/