ANL/MCS-TM-ANL-01/x Rev x

Installation and User's Guide to MPICH, a Portable Implementation of MPI Version 1.2.7 The globus2 device for Grids

by

Contents

Abstract

MPI (Message-Passing Interface) is a standard specification for message-passing libraries. MPICH is a portable implementation of the full MPI-1 specification for a wide variety of parallel and distributed computing environments. MPICH contains, along with the MPI library itself, a programming environment for working with MPI programs. The programming environment includes a portable startup mechanism, several profiling libraries for studying the performance of MPI programs, and an X interface to all of

  - Miscellaneous new `MPI_Info` and `MPI_Datatype` routines.

- `MPICH`

we developed for the NEC SX-4 [7]. The ch_shmem

instead.)

This will configure MPICH for the default device; this is usually the appropriate choice. Section 4.1 discusses the options that can be given to configure toto MPICH.

The output of configure is piped to tee; this program both writes the output to the file specified by its argument (here 'c.log') and tstandard output. If you have trouble with the configure or make step, the file 'c.log' will help identify any problems.

3.e MPICH:

```
% make |& tee make.log
```

This may take a while, depending on the load on your system and on your file server, it may take anywhere from a few minutes to an hour or more.

2.3 Running examples

1.Optional) Build and run a simple test program:

```
% cd examples/basic
% make cpi
% ../../bin/mpirun -np 4 cpi
```

At this point you have run an MPI program on your system. If you have trouble, see Section 6.

2.Optional) Put the distribution through its complete acceptance test (See Section 4.11 for how to do this).

3.Optional) If you wish to install MPICH

At

2.4 Sample MPI programs

The MPICH distribution contains a variety of sample programs, which are located in the MPICH source tree. Most of these will work with any MPI implementation, not just MPICH.

examples/baaic contains a few short programs in Fortran, C, and C++ for testing the simplest features of MPI.

examples/test contains multiple test directories for the various parts of MPI. Enter "make testing" in this directory to run our suite of function tests.

examples/perftest Performance benchmarking programs. See the script runmpptest for information on how to run the benchmarks. These are relatively sophisticated.

```
mpicc -o foo foo.o
mpif77 -o foo foo.o
mpicxx -o foo foo.o
mpif90 -o foo foo.o
```

-mpianim Build version that generates real-time animation.

These are described in more detail in Section

to launch your MPI application, (b) you have already acquired your Globus ID and security credentials, (c) your Globus ID has been registered on all machines(d) you have a valid (unexpired) Globus proxy. See `http://www.globus.org` for information on those topics.

Every `mpirun` commurit under the `globus2` devicesubmits a Globus *Resource Specification Language Script*, or simply *RSL script*, to a Globus-enabled grid of computers. Each RSL script is composed of one or more RSL *subjobs*, typically one subjob for each machine in theute749(Y)82(y)26thesi6theyywn.itin scrirg in.-.16 -817.79 3.21us.org

The number appearing at the end of each line is optional (default=1). It specifies the maximum number of nodes that can be created in a single RSL subjob on each machine. `mpirun` uses the `-np` specification by "wrapping around" the machines file. For example, using the machines file above `mpirun -np 8` creates an RSL with a single subjob with 8

```
(arguments=" 123 456")
```

MPMD program to a SPMD (single program multiple data, not to be confused with single

gdb where possible. There are five debugger scripts; ddd, gdb, xxgdb, ddd, and totalview.

3.6.6 Debugging MPI programs with TotalView

TotalView

Note also that if you use the MPI-2 function `MPI_Comm_set_name` on a communicator, TotalView will display this name whenever showing information about the communicator, making it easier to understand which communicator is which.

3.7 Log and tracefile tools

The MPE libraries that are distributed with `MPICH` contain several libraries for either logging information about the execution of each MPI call to a file for later analysis or for tracing each call as it occurs. These libraries may be accessed by simply providing a command line argument to the compilation scripts; further, this needs only be done when linking the program. For example, to create a log file of a program such as `cpi`, the following steps are needed:

```
mpicc -c cpi.c
mpicc -o cpi -mpilog cpi.o
```

The log file will be written to a file with the name '`cpi.clog`' or '`cpi.slog`', depending on the value of the environment variable `MPE_LOG_FORMAT`

3.9 Performance measurements

The ‘mpich/examples/perftest

Figure 1: Sample output from `mpptest`

4.1 Configure options

The configure script documents itself in the following way. If you type

```
configure -usage
```

In addition, `configure` makes use of environment variables such as `MAKE`, `CC`, `FC`, `CFLAGS`, and `FFLAGS`.

Selecting `-debug` can be helpful during debugging, but can slow down performance. `-nodebug` should be used for debugged production code.

In general, `-nothreads` should be used (the Globus2 device is not multithreaded). You

```
mpicc ...
```

or

```
mpicc -config=gcc ...
```

would case `mpicc` to load '`mpich-gcc.conf`' and use the appropriate definitions.

4.4 Fortran Compilers

`MPICH` provides support for both Fortran 77 and Fortran 90. Because `MPICH` is implemented in C, using `MPICH` from Fortran can sometimes require special options. This section discusses some of the issues. Note that `configure` tries to determine the options needed to support Fortran. You need the information in this section only if you have problems. Section 4.4.6 discusses how to support multiple Fortran compilers (e.g., `g77` and `pgf77`) with a single `MPICH` installation.

4.4.1 What if there is no Fortran compiler?

The `configure` program should discover that there is no Fortran compiler. You can force `configure`

4.5 C++

The C++ support in MPICH has been provided by Indiana University (this group was

4.7 File System Issues

Most users do not need to worry about file systems. However, there are two issues: using NFS (the Network File System) with MPI-IO and using NFS with some automounters. These issues are covered in the next two sections.

4.7.1 NFS and MPI-IO

To use MPI-IO multihost on NFS file systems, NFS should be version 3, and the shared NFS directory must be mounted with the "no attribute caching" (noac) option set (the directory cannot be automounted). If NFS is not mounted in this manner, the following error could occur:

```
MPI_Barrier: Internal MPI error: No such file or directory
File locking messages
```

I order to reconfigure NFS to handle MPI-IO properly, the following sequence of steps are
(win4d)-(r333(o)63(nod)-(prop)-2m333(ersiod)-r04(25(6(t:ons.)]TJE=31.122.963.813.541temse)-6(reconrmsiod)-

4.8 Building MPICH

Once configure has determined the features of your system, all you have to do now is

```
make
```

This will clean all the directories of previous object files (if any), compile both profiling and non-profiling versions of the source code, including Romio and the C++ interface, build all

The man pages will have been copied with the installation, so you might want to add
‘

respectively. Alternately, set the prefix to ‘/usr/local/mpich-1.2.7/

executable staging. These features are provided by using services supplied by the Globus toolkit: see `http://www.globus.org` for details.

The `globus2` device requires that special servers be running on the computers where processes are to be created. In our discussion of how to use the `globus2` device, we assume that weare using the `globus2` device on a collection of machines on whichvarious Globus

To create a new message catalog, copy the file 'mpich.En_US.msg' to 'mpich.mylanguage.msg' and translate the entries. The value of 'mylanguage' should match the ones used for your system; for example, 'mpich.De_DE.msg' for German. Many systems put their NLS files in '/usr/lib/nls/msg'; you can also check the value of the environment variable NLSPATH on your system. Note that some systems provide the routines and programs to support NLS, but do not make use of it and do not provide a initial NLSPATH value.

For emacs users, check the Emacs info under "

patches that may fix your problem. After that, check Section 6.3 for common problems. If you still can't find a solution to your problem, submit a bug report and we will try to help you.

6.2 Submitting bug reports

Send any problem that you can not solve by checking this section to `mpi-bugs@mcs.anl.`

Fortran. Check the output of the configure step for any error messages or warnings about building the Fortran libraries. If you do not require Fortran, reconfigure MPICH using the configure option `--disable-f77` and remake MPICH

```
checking that the compiler f77 runs... no
Fortran compiler returned non-zero return code
Output from test was
f2ctmp_conftest.f:
   MAIN main:
```

A:

3. Q: During the configure step, messages like

   ```
   /homes/me/mpich/configure: 134956160: Permission denied
   ```

   sometimes appear. What is wrong?

   A: This is a bug in the Linux (actually GNU) sh shell. The shell is attempting to create a file with the name ‘

(but note that configure builds a new 'farg.f' from 'farg.f.in' each time that it is run).

MPICH now attempts to determine the correct names of the routines to access the command line. If you find that MPICH fails to determine the names correctly, please send a bug report to mpi-bugs@mcs.anl.gov

2. Q: The build fails for the ch_p4 device when using the Compaq C compiler.
   A: There is an incompatibility with the system include files (not the

```
sigset (code)
sighold (code)
*** Error code 1
```

A: You need to add the link option `-lV3`. The `ch_p4` device uses the System V signals on the HP; these are provided in the 'V3' library.

6.10.3 LINUX

1. Q: When linking a Fortran program, I get

```
Linking:
foo.o(.data+0x0): undefined reference to 'pmpi_wtime_'
```

A: This is a bug in the `pgf77` compiler (which is itself a workaround for a bug in the LINUX

If this works, then consider editing the value of LDFLAGS in the compiler scripts (e.g., mpicc) to include this option.

Unfortunately, each compiler has a different way of passing these arguments to the linker, and each linker has a different set of arguments for specifying the

A: This means that MPICH was built with the xlC compiler but that some of the machines ii7(h5(yc)27ours)-364‘t

4. Q: When running on an IBM SP, my job generates the message

```
Message number 0031-254 not found in Message Catalog.
```

   and then dies.

   A: If your user name is eight characters long, you may be experiencing a bug in the IBM POE environment. The only fix at the time this was written was to use an account whose user name was seven characters or less. Ask your IBM representative about PMR 4017X (poe with userids of length eight fails) and the associated APAR IX56566.

6.12 Programs fail at startup

6.12.1 General

1. Q:

(b) Use the secure server (`serv_p4`). See the discussion in the Users Guide.

(c) Redirect all standard output to a file. The MPE routine `MPE_IO_Stdout_to_file`

or it might compute

$$((a + b) + (c + d)) + ((e + f) + (g + h)),$$

where $a, b, \ldots$ are the values of

```
  Z()  { cerr << "*Z" << endl; }
  ~Z() { cerr << "+Z" << endl; }
};

Z z;

int main(int argc, char **argv) {
  MPI_Init(&argc, &argv);
  MPI_Finalize();
}
```

when running with the ch_p4 MPI˙Finalip4

Then edit the upshot script to use this shorter name instead. This may require root access, depending on where you put the link.

(b) Create a regular shell program containing the lines

```
#! /bin/sh
/usr/local/tk3.6/bin/wish -f /usr/local/mpi/bin/upshot
```

(with the appropriate names for both the 'wish' and 'upshot' executables).

Appendices

A Frequently Asked Questions

The web page www.mcs.anl.gov/mpi/mpich/faq.html contains answers for many fre-

- Create a file .rhosts in your home directory
-

services, you must edit the files ‘/etc/xinetd.d/rsh’ and ‘/etc/xinetd.d/rlogin’. Here is the rsh file as it looks by default:

```
# default: on
# description: The rshd server is the server for the rcmd(3) routine and, \
#       consequently, for the rsh(1) program.  The server provides \
#       remote execution facilities with authentication based on \
```

```
# Firewall configuration written by lokkit
# Manual customization of this file is not recommended.
# Note: ifup-post will punch the current nameservers through the
#       firewall; such entries will *not* be listed here.
:input ACCEPT
:forward ACCEPT
:output ACCEPT
-A iCEPT
-A iCEPT
-A iCEPT
-A iCEPT
-A iCEPT
-A iCEPT
-A iCEPT
-A iCEPT
-A iCEPT
```

```
#
# New rules for rlogin/rsh traffic, incoming or outgoing
#
-A input -p tcp -s 0/0 -d 0/0 513 -b -j ACCEPT
-A input -p tcp -s 0/0 -d 0/0 514 -b -j ACCEPT
#
# End of new rules
#
-A input -p tcp -s 0/0 -d 0/0 0:1023 -y -j REJECT
-A input -p tcp -s 0/0 -d 0/0 2049 -y -j REJECT
-A input -p udp -s 0/0 -d 0/0 0:1023 -j REJECT
-A input -p udp -s 0/0 -d 0/0 2049 -j REJECT
-A input -p tcp -s 0/0 -d 0/0 6000:6009 -y -j REJECT
-A input -p tcp -s 0/0 -d 0/0 7100 -y -j REJECT
```

```
-A input -p tcp -s 0/0 -d 0/0 22 -b -j ACCEPT
#
# End of new rules
#
-A input -p tcp -s 0/0 -d 0/0 0:1023 -y -j REJECT
-A input -p tcp -s 0/0 -d 0/0 2049 -y -j REJECT
-A input -p udp -s 0/0 -d 0/0 0:1023 -j REJECT
-A input -p udp -s 0/0 -d 0/0 2049 -j REJECT
-A input -p tcp -s 0/0 -d 0/0 6000:6009 -y -j REJECT
-A input -p tcp -s 0/0 -d 0/0 7100 -y -j REJECT
```

At this point users on remote systems should be able to ssh into the machine, but they will still need a password.

Users should set up a private/public authentication key pair in order for ssh to operate without passwords. This process is documented in the installation guide, but a summary of the steps for RH7.2 will be included here.

First run the "ssh-keygen -t rsa" application to create the private/public key pair. By default this will create the files '~/.ssh/id_rsa' and '~/.ssh/id_rsa.pub'. Use a password.

Next place the public key ('~.ssh/id_rsa.pub') in the file '~/.ssh/authorized_keys'. If more than one machine is going to be used, then this key must be put in the '

The two default rules of interest are the following:

the problem is probably not with MPI. Instead, check for program bugs including

```
mpif77 -Wno-globals mycode.f
```

An alternative is to use a Fortran 90 or Fortran 95 compiler with the MPI module instead of the ‘`mpif.h`

MPICH was designed to provide an implementation of the MPI standard that could replace the proprietary message-passing systems on the massively parallel computers of the day, such as the Intel Paragon, IBM SP, and TMC CM5. MPICH used an early version of the abstract device interface (ADI), based on the Chameleon [14] portability system, to provide

```
[-f90libpath=LIBRARY_PATH_SPEC_FORMAT_FOR_F90]
[-lib=LIBRARY] [-mpilibname=MPINAME]
[-mpe_opts=MPE_OPTS]
[-make=MAKEPGM ]
[-memdebug] [-ptrdebug] [-tracing] [-dlast]
[-listener_sig=SIGNAL_NAME]
[-cross]
[-adi_collective]
[-automountfix=AUTOMOUNTFIX]
[-noranlib] [-ar_nolocal]
[-rsh=RSHCOMMAND] [-rshnol]
[-noromio] [-file_system=FILE_SYSTEM]
[-p4_opts=P4_OPTS]
```

fstype can be any combination of nfs, ufs,pfs (intel), piofs (IBM), hfs (HP), sfs (NEC), andxfs (SGI), c

The option '-pkt_size=LENGTH' allows you to choose the message length at which the ADI (Abstract Device Interface) switches from its short to long message format. LENGTH must be positive.

The option '-adi_collective' allows the ADI to provide some collective operations in addition to the basic point-to-point operations. Currently, most systems do not support this option (it is ignored) and on the others it has not been extensively tested.

Options for Experts:

The option '-memdebug' enables extensive internal memory debugging code. This should be used only if you are trying to find a memory problem (it can be used to help find memory problems in user code as well). Running programs with the option '-mpidb memdump' will produce a summary, when 'MPI_Finalize' is called, of all unfreed memory allocated my MPI. For example, a user-created datatype that was not later freed would be reported.

The option '-tracing' enables tracing of internal calls. This should be used only for debugging the MPICH implementation itself.

The option '-dlast' enables tracing of the most recent operations performed by the device. These can be output when a signal Tno3ke SIGINT), error, or call to a special routine occurs. There is a performance penalty for this option, but it can be very useful for implementors attempting to debug problems.

Sample Configure Usage:

To make for running on Sun's running Solaris with ch_p4 as the device,
 and with the installrell).memory to theCurell).me:ice,

```
        ch_nc     (native nCUBE calls, requires -arch=ncube),
        ch_cmmd   (native TMC CM-5 CMMD calls)
These are no longer distributed with the MPICH distribution.

Known architectures include (case is important)
```

```
        file <machine-file name>. This is a list of all available
        machines; use -np <np> to request a specific number of machines.
-np <np>
        specify the number of processors to run on
-nodes <nodes>
        specify the number of nodes to run on (for SMP systems;
        only ch_mpl device supports this)
-nolocal
        don’t run on the local machine (only works for ch_p4 jobs)
-all-cpus, -allcpus
        Use all available CPUs on all the nodes.
-all-local
        Run all processes on the master node.
-exclude <list>
        Exclude nodes in a colon delimited list.
-map <list>
        Use the colon delimited list to specify which rank
```

```
mpirun -arch sun4 -np 2 -arch rs6000 -np 3 program.%a
```

If instead the executables are in di□erent directories; for example, ‘/tmp/me/sun4’ and ‘/tmp/me/rs6000’, then the command is

```
mpirun -arch sun4 -np 2 -arch rs6000 -np 3 /tmp/me/%a/program
```

It is important to specify the architecture with `-arch` *before* specifying the number of processors. Also, the *first* `arch`

```
cd tk3.6
patch -p 1 < ../tk3.6p1patch
cd ..
```

(Note that the instructions say to use `patch -p`; newer versions of patch require an argument and the correct vai2.TJ/F71n this case is one; other versions of patch will want `-p1`

1. Q: I got the following messages when I tried to build on the Paragon:

References

[14] William D. Gropp and Barry Smith. Chameleon parallel programming tools users manual. Technical Report ANL-93/23, Argonne National Laboratory, Argonne, IL, March 1993.