mpiexec

Run an MPI program

Synopsis

```
mpiexec args executable pgmargs [ : args executable pgmargs ... ]
```

where `args` are command line arguments for `mpiexec` (see below), `executable` is the name of an executable MPI program, and `pgmargs` are command line arguments for the executable. Multiple executables can be specified by using the colon notation (for MPMD - Multiple Program Multiple Data applications). For example, the following command will run the MPI program `a.out` on 4 processes:

```
mpiexec -n 4 a.out
```

The MPI standard specifies the following arguments and their meanings:

-n <np>
: Specify the number of processes to use

-host <hostname>
: Name of host on which to run processes

-arch <architecture name>
: Pick hosts with this architecture type

-wdir <working directory>
: cd to this one *before* running executable

-path <pathlist>
: use this to find the executable

-soft <triplets>
: comma separated triplets that specify requested numbers of processes (see the MPI-2 specification for more details)

-file <name>
: implementation-defined specification file

-configfile <name>
: file containing specifications of host/program, one per line, with # as a comment indicator, e.g., the usual mpiexec input, but with ":" replaced with a newline. That is, the configfile contains lines with -soft, -n etc.

Additional arguments that are specific to the MPICH implementation are discussed below.

Note that not all of these parameters are meaningful for all systems. For example, the `gforker` version of `mpiexec` creates all processes on the same system on which it is running; in that case, the `\-arch` and `\-host` options are ignored.

The colon character (`:`) may be used to separate different executables for MPMD (multiple program multiple data) programming. For example, to run the program `ocean` on 4 processes and `air` on 8 processes, use:

```
mpiexec -n 4 ocean : -n 8 air
```

MPICH-Specific Arguments

Many of the implementations of process managers in MPICH support the following arguments to `mpiexec`:

-np <num>

A synonym for the standard `\-n` argument

-env <name> <value>
Set the environment variable `<name>` to `<value>` for the processes being run by `mpiexec`

-envnone
Pass no environment variables (other than ones specified with other `\-env` or `\-genv` arguments) to the processes being run by `mpiexec`. By default, all environment variables are provided to each MPI process (rationale: principle of least surprise for the user)

-envlist <list>
Pass the listed environment variables (names separated by commas), with their current values, to the processes being run by `mpiexec`.

-genv <name> <value>
The `\-genv` options have the same meaning as their corresponding `\-env` version, except they apply to all executables, not just the current executable (in the case that the colon syntax is used to specify multiple execuables).

-genvnone
Like `\-envnone`, but for all executables

-genvlist <list>
Like `\-envlist`, but for all executables

-usize <n>
Specify the value returned for the value of the attribute `MPI_UNIVERSE_SIZE`.

-l
Label standard out and standard error (`stdout` and `stderr`) with the rank of the process

-maxtime <n>
Set a timelimit of `<n>` seconds.

-exitinfo
Provide more information on the reason each process exited if there is an abnormal exit

Environment variables for mpiexec

The following environment variables are understood by some versions of `mpiexec`. The command line arguments have priority over these; that is, if both the environment variable and command line argument are used, the value specified by the command line argument is used.

MPIEXEC_TIMEOUT
Maximum running time in seconds. `mpiexec` will terminate MPI programs that take longer than the value specified by `MPIEXEC_TIMEOUT`.

MPIEXEC_UNIVERSE_SIZE
Set the universe size

MPIEXEC_PORT_RANGE
Set the range of ports that `mpiexec` will use in communicating with the processes that it starts. The format of this is `<low>:<high>`. For example, to specify any port between 10000 and 10100, use `10000:10100`.

MPICH_PORT_RANGE
Has the same meaning as `MPIEXEC_PORT_RANGE` and is used if `MPIEXEC_PORT_RANGE` is not set.

MPIEXEC_PREFIX_DEFAULT
If this environment variable is set, output to standard output is prefixed by the rank in `MPI_COMM_WORLD` of the process and output to standard error is prefixed by the rank and the text `(err)`; both are followed by an angle bracket (>). If this variable is not set, there is no prefix.

MPIEXEC_PREFIX_STDOUT
Set the prefix used for lines sent to standard output. A `%d` is replaced with the rank in `MPI_COMM_WORLD`; a `%w` is replaced with an indication of which `MPI_COMM_WORLD` in MPI jobs that involve multiple `MPI_COMM_WORLD`s (e.g., ones that use `MPI_Comm_spawn` or `MPI_Comm_connect`).

MPIEXEC_PREFIX_STDERR
Like `MPIEXEC_PREFIX_STDOUT`, but for standard error.

Return Status

`mpiexec` returns the maximum of the exit status values of all of the processes created by `mpiexec`.