Collective communications

MPI Broadcast

The same data is sent from the root to all processes in the communicator

In C:

```
MPI_Bcast(void *buffer, int count, MPI_Datatype data_type, int root, MPI_Comm comm);
```

In Fortran:

```
call MPI_Bcast(buffer, count, data_type, root, comm, error)
```

MPI Scatter

Different data is sent to each process in the communicator

In C:

```
MPI_Scatter(void *sendbuffer, int sendcount, MPI_Datatype senddata_type,
            void *recvbuffer, int recvcount, MPI_Datatype recvdata_type,
            int root, MPI_Comm comm);
```

In Fortran:

```
call MPI_SCATTER(sendbuffer, sendcount, senddata_type,
                 recvbuffer, recvcount, recvdata_type,
                 root, comm, error);
```

Collective communications

Example: code demonstrating Broadcast subroutine

Look for “**broadcast.c**”

C example:

```
# include <mpi.h>
Int main (int argc, char *argv[])
{
        int rank;
        double param;
        MPI_Init(&argc, &argv);
        MPI_Comm_rank(MPI_COMM_WORLD, &rank);
        if(rank==2) param=23.0;
        MPI_Bcast(&param,1,MPI_DOUBLE,2,MPI_COMM_WORLD);
        printf(“P:%d after broadcast parameter is %f\n”,rank,param);
        MPI_Finalize();
}
```

Collective communications

Example: code demonstrating Broadcast subroutine

Look for “**broadcast.f**”

Fortran example:

```
program BROADCAST
include ‘mpif.h’
integer error, rank, size
real param
call MPI_INIT(error)
call MPI_COMM_RANK(MPI_COMM_WORLD, rank, error)
if(rank.eq.5) param=23.0
call MPI_BCAST(param,1,MPI_REAL,5,MPI_COMM_WORLD,error)
print*,”P:”, rank, “after broadcast param is “, param
call MPI_FINALIZE(error)
end
```

Collective communications

Example: code demonstrating Broadcast subroutine

Running “broadcast.c” :

```
mpirun -np 4 ./broadcast.exe
```

Output:

```
P:0 after broadcast parameter is 23.000000
P:2 after broadcast parameter is 23.000000
P:1 after broadcast parameter is 23.000000
P:3 after broadcast parameter is 23.000000
```