NAME

MPIL_Request_set_name − LAM/MPI-specific function to set a string name on an MPI_Request

SYNOPSIS

#include <mpi.h>
int
MPIL_Request_set_name(MPI_Request req, char *name)

INPUT PARAMETERS

req - MPI_Request (handle)
name - Name

NOTES

The *name* must be a null-terminated string. It is copied into internal storage during this call.

ERRORS

If an error occurs in an MPI function, the current MPI error handler is called to handle it. By default, this error handler aborts the MPI job. The error handler may be changed with *MPI_Errhandler_set* ; the predefined error handler *MPI_ERRORS_RETURN* may be used to cause error values to be returned (in C and Fortran; this error handler is less useful in with the C++ MPI bindings. The predefined error handler *MPI::ERRORS_THROW_EXCEPTIONS* should be used in C++ if the error value needs to be recovered). Note that MPI does *not* guarantee that an MPI program can continue past an error.

All MPI routines (except *MPI_Wtime* and *MPI_Wtick*) return an error value; C routines as the value of the function and Fortran routines in the last argument. The C++ bindings for MPI do not return error values; instead, error values are communicated by throwing exceptions of type *MPI::Exception* (but not by default). Exceptions are only thrown if the error value is not *MPI::SUCCESS* .

Note that if the *MPI::ERRORS_RETURN* handler is set in C++, while MPI functions will return upon an error, there will be no way to recover what the actual error value was.

MPI_SUCCESS
- No error; MPI routine completed successfully.

MPI_ERR_ARG
- Invalid argument. Some argument is invalid and is not identified by a specific error class. This is typically a NULL pointer or other such error.

SEE ALSO

MPIL_Request_get_name

LOCATION

mpil_rsetname.c