[When a failure prevents the MPI implementation from successfully completing a point-to-point communication, the communication is marked as completed with an error of class MPI_ERR_PROC_FAILED. Future point-to-point communication with the same process on this communicator must also return MPI_ERR_PROC_FAILED.

The completion of a nonblocking receive from MPI_ANY_SOURCE can return one of the following three error codes due to process failure. MPI_SUCCESS is returned if the receive was able to complete despite the failure. MPI_ERR_PROC_FAILED indicates that the request has been matched with the send, but cannot complete succesfully due to the failure at the sender. MPI_ERR_PENDING indicates that while a process has failed, the request is still pending and can be continued. To acknowledge a failure and discover which processes failed, the user should call MPI_COMM_FAILURE_ACK.]

1. Introduction

2. General case

3. ANY_SOURCE

4. Hint Failure_Ack

An MPI implementation raises the following error classes to notify users that a point-to-point communication operation could not complete successfully because of the failure of involved processes:

- MPI_ERR_PENDING indicates, for a non-blocking communication, that the communication is a receive operation from MPI_ANY_SOURCE and no matching send has been posted, yet a potential sender process has failed. Neither the operation nor the request identifying the operation are completed. Note that the same error class is also used in status when another communication raises an exception during the same operation (as defined in Section 3.7.5).
- In all other cases, the operation raises an exception of class MPI_ERR_PROC_FAILED to indicate that the failure prevents the operation from following its failure-free specification. If there is a request identifying the point-to-point communication, it is completed. Future point-to-point communication with the same process on this communicator must also raise MPI_ERR_PROC_FAILED.

Advice to users.

To acknowledge a failure and discover which processes failed, the user should call MPI_COMM_FAILURE_ACK (as defined in Section 17.3.1).

(*End of advice to users.*)