Freeing returns control of the rest of the argument list (e.g., the data buffer address and array arguments).

MPI operations are available in one or more of these forms: blocking, nonblocking, and persistent.

Blocking operation For a **blocking operation**, all four stages are combined in a single procedure call (as shown in Figure 2.1 and defined in Section 2.4.2).

Figure 2.1: State transition diagram for blocking operations

Nonblocking operation For a **nonblocking operation**, the initialization and starting stages are combined into a single nonblocking procedure call and the completion and freeing stages are combined into a separate, single procedure call, which can be blocking or nonblocking (as shown in Figure 2.2 and defined in Section 2.4.2).

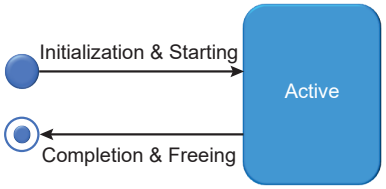

Figure 2.2: State transition diagram for nonblocking operations

Persistent operation For a **persistent operation**, there is a separate procedure for each of the four stages (as shown in Figure 2.3 and defined in Section 2.4.2). Each of these procedures may be blocking or nonblocking.

For a partitioned send operation, an additional call to activate each partition of the send buffer (see Section 4.2.1) is required to finish the starting stage. For a partitioned receive operation, before the operation is complete the user is allowed to access a partition of the output buffer after verifying that it has arrived (see Section 4.2.2).

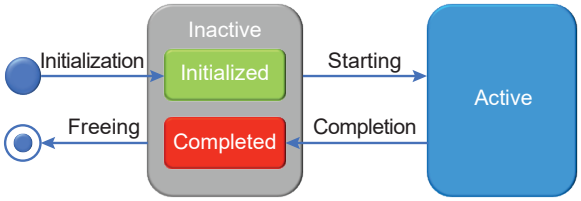

Figure 2.3: State transition diagram for persistent operations

Additionally, an MPI operation can be collective or noncollective.