

and does not block the process.

The events that may be monitored and the corresponding notifications are detailed in *Emeraude TCI Reference Manual*.

2.4 Sending and receiving messages

2.4.1 Sending notifications

To send a notification from a tool, use the `TCM_notify` function. The notification will normally be sent to all tools that have registered an interest in the notification, but if required you can specify that the notification is to be sent to a specific process. This would normally be a process with which this tool had already exchanged messages and whose TCI process id is therefore known; see Section 2.5.

2.4.2 Sending requests and receiving replies

TCI provides two functions for sending a request.

`TCM_request_and_wait` sends a request message and waits for the reply. This function does not return until the reply has been received from the servicer. On return, therefore, any output parameters of the message will have been filled, giving the requester its reply.

This function fails if the servicer sends a reply indicating failure, or if TCI cannot deliver the request.

`TCM_request` sends a request and returns immediately. In this case you can name a hook function that is to be called when the reply is received, to handle the message parameters. If the tool does not provide a hook function the reply will be discarded. Figure 2-3 illustrates this case. Section 2.8 gives more information about hook functions to handle replies.

Normally, TCI automatically routes the request to a tool that can handle it. If required, however, you can specify that the request is sent to a specific servicer. For example, you might wish to send a request to the same process that handled an earlier message.

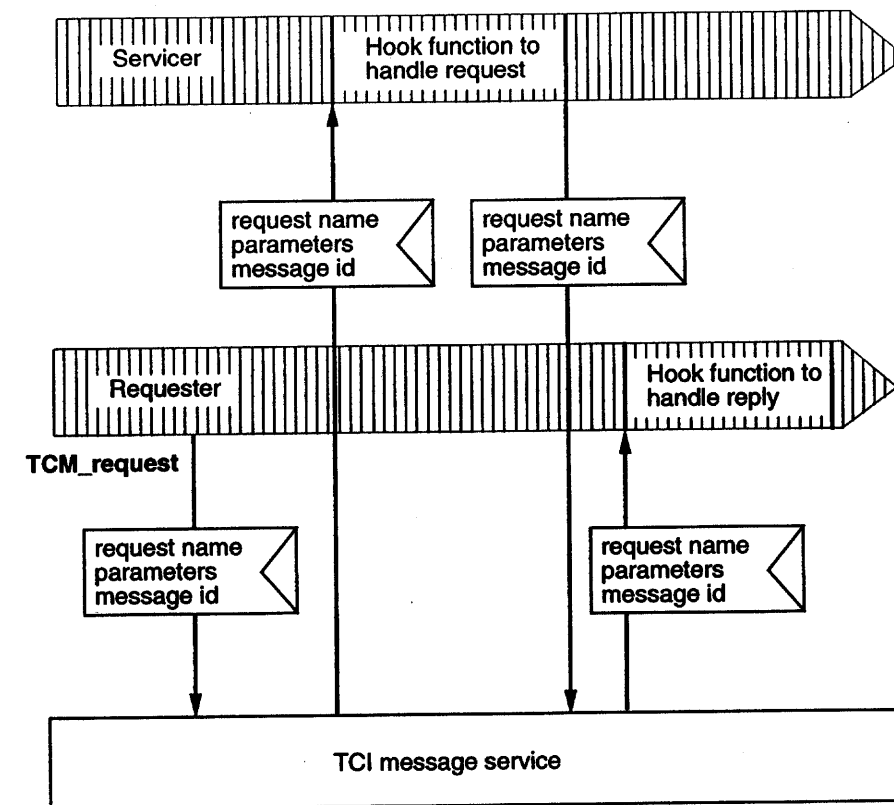


Figure 2-3. Asynchronous requests

2.4.3 Receiving messages and sending replies

If the tool was connected using the `TCM_init_X11` function, requests and notifications will be received automatically by the X11 event loop, and the appropriate hook function called.

If the tool was connected using `TCM_init` it must monitor the file descriptor returned by `TCM_init`. When this is active, indicating that a message has been sent to the tool, the tool must call the `TCM_receive` function. `TCM_receive` identifies the message and calls the appropriate hook function.

Normally, the reply to the request is sent on the return of the hook function. However, in some circumstances you may need to send the reply explicitly. You might need to send the reply either before the hook function returns (for example, because the output parameters point to values that are not global variables) or after it returns (for example, because the reply cannot be sent until the tool has obtained some input from the user). In this case, keep the message identification given to