Whereas it is unlikely that opening **The .MPP Driver** will adversely affect another program, an application program should never be allowed to close **The .MPP Driver**, because another program might be using it. Under certain very specific circumstances, however, the system might close **The .MPP Driver**. The system uses **The .MPP Driver**'s **PATalkClosePrep** function to send a permission-to-close transition to each routine in the AppleTalk Transition Queue. This transition indicates that the system intends to close **The .MPP Driver** so that each routine in the queue has the opportunity to deny permission to do so.

When the operating system calls the **PATalkClosePrep** function, any routine in the AppleTalk Transition Queue that wishes to deny permission to close **The .MPP Driver** may return a pointer to a high-level language string. The high-level language string should be the name of the application that placed the entry in the queue. If any routine in the AppleTalk Transition Queue denies permission to close **The .MPP Driver**, the **PATalkClosePrep** function returns the result code closeErr.

If any routine denies the requested permission to close **The .MPP Driver**, the **AppleTalk Manager** sends a cancel-close transition to every routine in the AppleTalk Transition Queue that previously received the permission-to-close transition. The caller of the **PATalkClosePrep**function may display a dialog box informing the user that another application program is using **The .MPP Driver** and showing the name (if any) returned by the AppleTalk Transition Queue routine. The dialog box gives the user the option of canceling the request to close AppleTalk or of closing AppleTalk anyway.

If the user chooses to close AppleTalk despite the fact that an application is using it, the system calls the **MPPClose** function. AppleTalk then calls each application in the AppleTalk Transition Queue, this time informing each one that AppleTalk is about to close. In this case, your AppleTalk Transition Queue routine must prepare for the imminent closing of AppleTalk; it cannot deny permission to the **MPPClose** function.