

10. Event Handling

Programs that use the Facelt module to handle main menu events, modeless window events, and other program-wide features must include an "event loop". This event loop is a loop that begins by calling Facelt's DoLoop command to get Facelt to pre-process low-level Macintosh events. Facelt determines whether any of these events need to be handled by the main program and, if so, returns the event as a menu or "pseudo-menu" event to the program (i.e., it takes care of everything until something needs your attention).

Event or message types are distinguished from one another by the fRec variable uMenuID. uMenuID can be either a menu ID (indicating a menu selection), a window's FWND ID (indicating a hit in a window), or the baseID of a module (indicating a message from that module). In the minimum code example, a ViewIt window is opened with NewWnd,

```
Facelt(nil,NewWnd,1000,1,0,0);
```

and events from that window are detected by checking if uMenuID is equal to the window's FWND ID (= 1000):

```
repeat
```

```
    Facelt(nil,DoLoop,0,0,0,0);
```

```
    if (fRec.uMenuID = 1000) then
```

```
        ...
```

```
until false;
```

The use of just a few simple variables like uMenuID to distinguish event types, and the automatic handling of most low-level events by Facelt and ViewIt, combine to greatly reduce the code required in Facelt-based programs.

The Facelt and ViewIt Guides provide further information about the types of events that can be returned by DoLoop. In general, it is not necessary for you to know everything about all possible events returned, but only those that you have purposely arranged to occur, such as the selection of program menu items and hits in enabled controls in ViewIt windows. This means that your event loop will often consist almost entirely of program-specific code, which makes Facelt-based programs much cleaner, simpler, and easier to understand than those produced with competing tools.