

## Using Color QuickDraw

If your application uses color in straightforward ways, it will probably execute without change in a direct pixel environment.

For most other applications, the main concern is in the creation and use of special-purpose pixel map and graphics device records. By using the offscreen graphics world routines described in the separate section entitled, **Graphics Devices Manager**, you will find such tasks far easier than before.

If you must work with color grafPort, pixel map, and graphics device records in ways beyond the scope of the offscreen graphics world routines, the following guidelines may aid you in adapting to the **Color QuickDraw** direct pixel environment:

- Don't draw directly to the screen. Create your own offscreen graphics world, (see **Graphics Devices Manager**) and use the **CopyBits**, **CopyMask**, or **CopyDeepMask** routine.
- Don't directly change the fgColor or bkColor field of a grafPort and expect them to be used as the pixel values. **Color QuickDraw** recalculates these values for each device. If you really want to draw in an index instead of a color, use a palette with explicit colors, as described in the **Palette Manager**. For device-independent colors, use the **RGBForeColor** and **RGBBackColor** procedures.
- Fill out all the fields in a new pixel map record. The **NewPixMap** function returns a pixel map record that is cloned from the pixel map record pointed to by theGDevice. If you don't want a copy of the main screen's pixel map record—for example, you want one that is a different depth—then you must fill out more fields than just pixelSize: you must fill out the pixelType, cmpCount, and cmpSize fields. Set pmVersion to 0 when initializing your own pixel map record. For future compatibility you should also set packType, packSize, planeBytes, and pmReserved to 0.
- Don't clone a graphics device record's pixel map record. Instead, use the **NewPixMap** function or the **CopyPixMap** procedure. If you must create or manually clone a pixel map record, make sure to set the pmVersion field and other unused fields to 0 for future compatibility.
- Fill out all the fields of a new graphics device record. When creating an offscreen graphics device record by calling **NewGDevice** with the mode parameter set to -1, you must fill out the fields of the graphics device record (for instance, gdType) yourself. If you want a copy of an existing graphics device record, then copy the gdType field from it. If you explicitly want an indexed device, then set gdType to 0.
- Don't assume a pixel map record has a color table. A direct pixel map record need not have a color table. For compatibility, a direct pixel map record should have a dummy pmTable handle that points to a color table header with a seed value equal to cmpSize \* cmpCount, and the color table's ctSize field should be set to 0.

Again, all these problems are simplified if you use the offscreen graphics world routines.