PixMap structure

#include <Quickdraw.h>

typedef struct **PixMap {**

		Size	Offset	Description
Ptr	baseAddr;	4	0	Address of start of bitmap data
short	rowBytes;	2	4	Bytes per row in the bitmap data
Rect	bounds;	8	6	Coordinates imposed over the bitmap data
short	pmVersion;	2	14	Version of Color QuickDraw
short	packType;	2	16	Format of packing algorithm
long	packSize;	4	18	Image's size after compression
fixed	hRes;	4	22	Horizontal resolution in pixels per inch
fixed	vRes;	4	26	Vertical resolution (hres=vres=72)
short	pixelType;	2	30	Pixel image storage format
short	pixelSize;	2	32	Bits per pixel (always a power of 2)
short	cmpCount;	2	34	Color components per pixel
short	cmpSize;	2	36	Logical bits per RGBColor component
long	planeBytes;	4	38	Offset in bytes to next plane
CTabHandle	pmTable;	4	42	Handle to color table
long	pmReserved;	4	46	Reserved for future expansion
} PixMap;		50		

typedef PixMap ***PixMapPtr**;
typedef PixMap ****PixMapHandle**;

Field descriptions

rowBytes The restriction that the value of rowBytes be less than $2000 has been relaxed: rowBytes must be less than $4000. The value must be even, and for best performance it should be a multiple of 4.

pmVersion The value of pmVersion is normally 0. If pmVersion is 4, **Color QuickDraw** treats the pixel map's baseAddr field as being 32-bit clean. (All other flags are private.) Most applications never need to set pmVersion.

pixelType Direct pixel values are specified by a pixelType field value of RGBDirect, or 16. In a pixel map of the graphics device record for a direct device, the pixelType field is set to the constant RGBDirect when the screen depth is set.

pixelSize Pixel sizes must be powers of 2. The original **Color QuickDraw** supported pixel sizes of 1, 2, 4, and 8 bits; with direct pixels, pixel sizes may also be 16 and 32 bits.

cmpCount With indexed pixels, each pixel is a single value representing an index in a color table, and therefore the cmpCount field of a pixel map record is 1- the index is the single component. With direct pixels, each pixel contains three components, one integer each for the intensities of red, green, and blue, so cmpCount is 3. Other values are undefined.

cmpSize For an indexed pixel value, which has only one component, the value of cmpSize is the same as the value of pixelSize: 1, 2, 4, or 8. A 32-bit pixel consists of three components (red, green, and blue values) of 8

THINK Reference © 1991-1992 Symantec Corporation

bits each. Since cmpCount * cmpSize (3 * 8 = 24) is less than the value of pixelSize, 8 bits in the pixel are not part of any component. These bits are unused: **Color QuickDraw** sets them to 0 in any image it creates. If presented with a 32-bit image-for example, in the **CopyBits** procedure-it passes whatever bits are there. (Generally, therefore, your application should clear image memory to 0 before creating a 32-bit image.)

A 16-bit pixel consists of three components of 5 bits each. This leaves an unused high-order bit, which **Color QuickDraw** sets to 0.

Color QuickDraw expects that the sizes of all components are the same, and that cmpCount * cmpSize is less than or equal to pixelSize.

In each direct pixel, the pixel value is the concatenation of the red, green, and blue components, where red is in the most significant bits and blue is in the least significant bits. The entire direct pixel is right aligned; unused bits occupy the highest-order bits.

THINK Reference © 1991-1992 Symantec Corporation