New Technical Notes
Macintosh

Developer Support

Picture Utility Q&As

Imaging

Revised by: Developer Support Center May 1993
Written by: Developer Support Center October 1990

This Technical Note contains a collection of Q&As relating to a specific topic—questions you've sent the Developer Support Center (DSC) along with answers from the DSC engineers. While DSC engineers have checked the Q&A content for accuracy, the Q&A Technical Notes don't have the editing and organization of other Technical Notes. The Q&A function is to get new technical information and updates to you quickly, saving the polish for when the information migrates into reference manuals.

Q&As are now included with Technical Notes to make access to technical updates easier for you. If you have comments or suggestions about Q&A content or distribution, please let us know by sending an AppleLink to DEVFEEDBACK. Apple Partners may send technical questions about Q&A content to DEVSUPPORT for resolution.

GetPictInfo and QuickTime compressed PICT files

Date Written: 2/24/92
Last reviewed: 8/1/92

Is it my imagination, or does GetPictInfo return a bit depth of 1 on QuickTime compressed PICT files?

___

Yep! This is what's happening: The Picture Utilities Package doesn't know of the QuickTime Compressed Pixmap opcode (0x8200), so it just skips over the opcode's data; then it finds the PacksBitRect opcode containing the black-and-white pseudo-alert that you get when you draw the picture on a machine that doesn't have QuickTime installed, and GetPictInfo reports back this alert.

Trivia: When QuickTime is installed, it displays the compressed image and then ignores the following PacksBitRect since QuickTime knows it's only the black-and-white alert.

NewPalette doesn't use CTab2Palette to create a palette

Date Written: 3/12/92

Last reviewed: 8/1/92

I’m using the Picture Utilities Package to extract the color table from a picture. After getting the color table, I use NewPalette to construct a palette from the color table (usage = tolerant, tolerance = 0). After I do this, the RGB values in the palette don’t always exactly match the RGB values in the source color table, causing my program to fail. If I use NewPalette without a source color table, and then use CTab2Palette to copy the colors over (again with usage = tolerant, tolerance = 0), the colors match exactly.

___

It turns out that NewPalette doesn’t use CTab2Palette, but copies the RGB fields in a strange way that’s causing the problems you’re seeing. NewPalette copies the high byte in each color table RGB entry into both the high byte and the low byte of the corresponding palette entry. Thus, if the color table entry for red was $F000, it becomes $F0F0. This of course makes no difference to QuickDraw since the low byte isn’t displayed, but if your program expects the low byte to match, that’s where your problem exists. CTab2Palette is different, in that it doesn’t copy the high byte into the low byte unless the pmAnimated bit is set.

The best solution for your code isn’t to compare the entire RGB value when comparing colors, but rather to compare the high byte of each RGB component separately. If this isn’t possible, the next best solution is for you to use the workaround that you’ve already discovered with CTab2Palette.

It’s unlikely that the Palette Manager is going to change in the future for something like this. In fact, we would almost call it a “feature” since other developers may even depend on it.

Spooling PixMaps to disk

Date Written: 6/10/91
Last reviewed: 10/22/91

Do you have sample code for spooling PixMaps to disk in PICT format? Should I write the PICT opcodes to the file myself?

___

Apple recommends that you do not try to write the PICT opcodes yourself. Instead, replace the PutPicProc bottleneck proc, as shown in the Color QuickDraw chapter of *Inside Macintosh* Volume V on page V-89.

Two additional samples can be found in the sample code contained in the Developer CDs. Look for: “Tools & Apps: Graphics and Imaging: PICT Stuff.” One is a program and the other is an FKEY; both dump the main screen to disk as a PICT. The FKEY is a more complete sample in the sense that it works in black and white as well as Macintosh color computers, but the other is a smaller and simpler sample.

Getting the color usage from a picture under System 6

Date Written: 6/8/92
Last reviewed: 9/15/92

Do you know how I can obtain the color table of a picture when using a system version that happens to be less than 7.0? The Picture Utilites package seems to be only implemented in System 7.0.

___

You’re correct; the Picture Utilities package is implemented only under System 7. However, it’s possible to write code to duplicate its functionality under System 6. Basically, what you want to do is parse a picture, looking at the colors used for the different objects. How you deal with the colors is up to you.

What you do is replace the QuickDraw bottlenecks in a GrafPort with procedures of your own; in all the bottlenecks for QuickDraw primitives, you can just record the current color as having been used for an object. When you get a StdBits opcode, you’ll have to parse the pixmaps, looking through the image and recording all the colors used. As a shortcut, you could just record all the colors in the color table of the pixmap, if it's an indexed pixel image. After collecting this list of colors and any information on how often they are used, it’s up to you to boil this down into useful information, depending on how you want to use it.