

A Color Printing Overview (Color LaserJet, 5, 5M, DeskJet)

Introduction

This appendix provides an overview of the way color is used in the HP Color LaserJet, Color LaserJet 5, 5M, DeskJet 1200C, and DeskJet 1600C printers. It previews the remaining appendices, which describe the specific details of Hewlett-Packard color printing.

Note

The features described in these appendices are a superset of those supported by the HP Color LaserJet, Color LaserJet 5, 5M and DeskJet printers. Some features are supported on one or more printers, but not necessarily on all of them. See the *PCL 5 Comparison Guide* for specific feature support for each printer.

Processing a color document involves specifying a palette or palettes, and then using the colors within the current palette to print. For non-raster printing, items such as text, rules, and vectors are simply printed in the currently active color, which is specified using the Foreground Color command or Select Pen command if in HP-GL/2. For raster printing, the color of each pixel is specified as either a direct color specification, or as an index into the palette, depending on the pixel encoding mode.

The PCL Print Model determines how color is applied to the page. The printed result can vary in background and texture depending on the source transparency mode, pattern transparency mode, and selected logical operation (ROP). Besides the pre-defined shading and patterns, users can define new monochrome or multicolor patterns.

When printing color pages, a user can choose one of several color modes, depending on the desired results. Each color mode has a palette associated with it. *Simple Color Mode* provides a palette of fully saturated colors whose colors are similar to those of a plotter's pen colors. The palette is nonprogrammable, and is intended for simple printing of items such as bar and pie charts. For applications requiring different or more specific colors, the printer offers the *PCL Imaging* and *HP-GL/2 Imaging Modes*. The palette colors in these two modes can be modified to provide the desired result.

When choosing color for a particular application, the Color LaserJet printer provides *device-dependent* and *device-independent* color (the DeskJet 1200C and 1600C support only device-dependent color). *Device-independent* color provides accurate color matching based on an absolute color standard. It is preferred when users want a precise color to match the output from another device or to match the color on an existing page.

Besides providing device-independent color for precise color matching, the HP printers can modify color to compensate for various characteristics. The Color LaserJet printer supports the following methods of modifying color (the DeskJet 1200C and 1600C printers support halftone algorithms and gamma correction, but not color lookup tables or the Viewing Illuminant command).

- *Halftone render algorithms* determine how colors are rendered using the printers available colors. Halftone algorithms can be used to change apparent resolution, change the texture of images, reduce the number of colors, and change a color image to monochrome.
- *Color lookup tables* can remap colors to compensate for various differences in input data, such as unwanted color casts caused by unbalanced photographic light sources.
- *Gamma correction* provides a way to adjust for color differences in display monitors so that the display more closely matches the printed output.
- Since the appearance of colors changes under different viewing light sources, the *Viewing Illuminant* command allows the application to modify output color based on the light source used to view the printed page.

Color Concepts

This section describes some of the concepts and terminology used in this chapter, such as palettes, raster vs. non-raster color, device-independent vs. device-dependent color, black and white references, color selection, pixel encoding, and color modes.

Palettes

A palette is a collection of colors that are selected by their index numbers. You can create your own palette or choose from one of several fixed palettes. Although only one palette is active at any time, all palettes are assigned ID numbers and can be stored in the printer for later selection using the ID number. They can be deleted when desired. Palettes can also be saved (pushed) to a stack and later retrieved (popped) when needed.

Raster Mode

Raster mode is entered explicitly by the Start Raster command ($\text{\textasciicircum{E}c*r\#A}$) or implicitly by a Transfer Raster command ($\text{\textasciicircum{E}c*b\#V}$, $\text{\textasciicircum{E}c*b\#W}$). Raster mode is exited explicitly by an End Raster command ($\text{\textasciicircum{E}c*rC}$) or implicitly by a non-raster command.

Raster Color vs. Non-Raster Color

Palettes are used differently depending on whether the printer is in raster mode.

- In non-raster mode, the palette is always used for color selection. The color of text or patterns is specified using the Foreground Color command ($\text{\textasciicircum{E}c*v\#S}$).
- In raster mode, the palette is only used for indexed color selection; it is not used for direct color selection. (Indexed and direct color selection are explained later in this section.)

Device-Dependent vs. Device-Independent Color

Device-dependent color spaces are relative to the device's ability to produce specific colors. For example, if red is specified in a device-dependent color space, two different printers will combine the same amounts of cyan, magenta, yellow, and black toner to produce the color, but the results will be different because of the different properties of the toner.

Device-independent color is specified absolutely, in a color coordinate system that is independent of any device. For example, if red is specified in a device-independent color space, two printers will always produce the same result, even though they may need to combine different amounts of cyan, magenta, yellow, and black toner. Printers that produce device-independent color are calibrated to precise color standards.

Black and White References

Device-dependent color specifications are based upon an arbitrary range of values for each primary color component. The range endpoints for each color component are called black and white references for that component. Colors relative to these predefined limits are derived by specifying the amount of each component.

For the Device RGB color space, the maximum limit is called the white reference and the minimum limit is called the black reference. Regardless of the number chosen, the white reference represents the maximum value of a primary color that a device can produce, and the black reference represents the minimum value of that primary color. For example, if 100 is chosen as the white reference for red in the RGB color model, it represents the reddest red the device can produce. If 10 were chosen instead, then 10 would represent the same red.

Example:

Assuming the white and black references are set as follows:

Scenario 1

White Reference	Black Reference
red = 63	red = 0
green = 63	green = 0
blue = 63	blue = 0

Scenario 2

White Reference	Black Reference
red = 63	red = 4
green = 127	green = 0
blue = 31	blue = 0

Using these reference values, 50% blue for scenario 1 is 31, and 50% blue for scenario 2 is 15.

Color Selection

The HP color printers offer two methods for selecting colors:

- Indexed selection
- Direct selection

In indexed selection, colors are chosen using their palette index numbers. For non-raster mode, the palette index number is specified using the Foreground Color command. In raster mode, the data bit combination for each pixel forms an index number. The example below shows how the index numbers for an 8-color palette are specified:

Three-bit combinations:	0 1 0 1 0 1 0 1
	0 0 1 1 0 0 1 1
	0 0 0 0 1 1 1 1
Palette index number:	0 1 2 3 4 5 6 7

The number of colors in the palette dictates the number of bits per pixel of raster data required to specify an index number. For example, to specify 256 colors you need to send 8 bits of raster data per pixel ($2^8 = 256$).

In direct selection, colors are specified using the proportions of their primary components. For example, using a 24-bit-per-pixel representation, the color specified by (0xff, 0xf0, 0x00) for red, green, and blue would print a slightly red-tinted yellow. A palette is not used for direct selection.

Pixel Encoding

Colors are encoded in a row of raster data using either *plane* or *pixel* format. In planar format, all the pixels in a row are partially specified by one plane (bit) before the next plane is sent. In pixel format, each pixel is fully specified before sending the next pixel.

Encoding by Plane

Planar encoding uses successive data planes, each providing one bit for each pixel in a row. Each plane builds upon the preceding planes until the pixels in a row are fully defined. A pixel is not fully defined until it has received all the planes for that row.

The planes in a row form index numbers that define a pixel by selecting a palette entry. For example, an 8-entry palette requires 3 planes ($2^3 = 8$). The underlined bits below compose the index of the color of the third pixel in the first row.

$E_C * b \# V$	row 1	plane 1 (red)	b1	b1	<u>b1</u>	b1	b1	b1	...
$E_C * b \# V$		plane 2 (grn)	b2	b2	<u>b2</u>	b2	b2	b2	...
$E_C * b \# W$		plane 3 (blue)	b3	b3	<u>b3</u>	b3	b3	b3	...
$E_C * b \# V$	row 2	plane 1 (red)	b1	b1	b1	b1	b1	b1	...

Encoding by Pixel

When encoding by pixel, each pixel is fully specified before any bits are sent for the next pixel. For example, if four bits are needed to define a pixel, then every group of four bits in the data stream defines a pixel. The underlined (c4 . . . c1) group below defines the second pixel in the first row.

$E_C * b \# W$ row 1 b4 b3 b2 b1 c4 c3 c2 c1 . . .
 $E_C * b \# W$ row 2 b4 b3 b2 b1 . . .

The table below shows the four PCL options for selecting colors and encoding color raster data.

	Planar Encoding	Pixel Encoding
Indexed Selection	Indexed planar	Indexed pixel
Direct Selection	Direct planar	Direct pixel

Color Modes

There are four PCL 5 color modes:

- Black and White mode
- Simple Color mode
- PCL Imaging mode
- HP-GL/2 Imaging mode

All four modes create a palette. The palettes used in the Black and White mode and Simple Color mode are not modifiable. You can, however, modify the palettes in the PCL Imaging and HP-GL/2 Imaging modes.

You can use more than one mode on the same page. For example, you could enter the Simple Color mode to print a headline and a bar chart, PCL Imaging mode to print a raster photograph, and Black and White mode to print some body text. Each mode is described in more detail in Appendix B, “Using Color Modes.”

Device-Dependent Color Spaces

The following PCL commands can alter color processing for device-dependent color spaces:

- Rendering Algorithm ($E_C^*t\#J$)
- Gamma Correction ($E_C^*t\#I$)
- Color Lookup Tables ($E_C^*l\#W$)
- Configure Image Data ($E_C^*v\#W$)
- Simple Color ($E_C^*r\#U$)
- Monochrome Print Mode ($E_C\&b\#M$)

Color lookup tables or gamma correction (which are mutually exclusive) can modify the mapping of input to output.

Device-Independent Color Spaces

The following PCL commands can alter color processing for device-independent color spaces:

- Rendering Algorithm ($E_C^*t\#J$)
- Gamma Correction ($E_C^*t\#I$)
- Color Lookup Tables ($E_C^*l\#W$)
- Configure Image Data ($E_C^*v\#W$)
- Viewing Illuminant ($E_C^*i\#W$)
- Monochrome Print Mode ($E_C\&b\#M$)

Device-independent color spaces are supported under the following conditions:

- 1 The Configure Image Data command ($E_C^*v\#W$) configures the current palette and specifies a device-independent color space.
- 2 The Render Algorithm command ($E_C^*t\#J$) is set to one of the following algorithms (see “Halftone Render Algorithms” in Appendix D for details):
 - Continuous tone detail (E_C^*t0J , E_C^*t16J)
 - Continuous tone smooth (E_C^*t15J , E_C^*t17J)
 - Continuous tone basic (E_C^*t18J , E_C^*t19J)
 - Device best dither (E_C^*t3J , E_C^*t5J)
 - Error Diffusion (E_C^*t4J , E_C^*t6J)

- Cluster Ordered Dither (E_C^*t7J , E_C^*t8J)
- Ordered Dither (E_C^*t11J , E_C^*t12J)

Color processing reverts to device-dependent processing if the render algorithm is changed from one of the above. This is because extensive device characterization is necessary to achieve device-independence—calibration must be based on known parameters that affect the device's color gamut. Render algorithms such as Snap to Primaries (E_C^*t1J), Snap Black to White and Colors to Black (E_C^*t2J), or User-Defined Halftone (E_C^*t9J) either limit the number of colors available, or are undefined to the extent that their performance is not as precise. These algorithms, therefore, produce device-dependent results. Device-independent color is again generated if the render algorithm changes to one of the 5 listed above and the color space has not changed.

Device-Independent Color

The PCL language characterizes color as either device-dependent or device-independent. Both categories encompass many color spaces, each with unique characteristics.

Device-Dependent Color

Device-dependent color is relative to the device's inherent characteristics. For example, the colors produced by plotters are relative to the color of the installed plotter pens. Pen color varies considerably, especially as pens wear out, changing the color of the output. Likewise, for monitor screens, the red, green, and blue screen phosphors determine the colors produced. Fully saturated colors can vary greatly between screens. For printers, the color produced on a page depends on the printer's subtractive inks or toner (cyan, magenta, yellow, and black).

When using device-dependent color, devices receiving relative color specifications for the same color frequently do not produce the same color. For example, a monitor's saturated red may be different than a plotter's. In short, the same color page may appear considerably different on different devices.

The HP color printers provide device-dependent color specified using either the Device RGB or CMY color spaces.

Device-Independent Color

In contrast with device-dependent color, device-independent color is based on an absolute color standard—the tristimulus values of human vision. The device, whether a printer or otherwise, is calibrated to match an independent color specification. The color specification is translated in such a way that the resultant color is independent of the device. Examples of color spaces based on absolute standards include Kodak Photo YCC, CIE $L^*a^*b^*$, YUV, and the proposed YCrCb. Each is a transform from tristimulus CIE XYZ space.

With proper calibration, any device can provide a transform from device-independent color space to the device's own color space, producing output from different devices that have the same color appearance. For example, if a monitor's parameters are known (gamma, gain, chromaticity coordinates for each primary, and the white point), the monitor's RGB pixel information can be transformed into device-independent color.

The Color LaserJet printers provide device-independent color specified using either the CIE $L^*a^*b^*$, Colorimetric RGB, or Luminance-Chrominance color spaces.

HP Color LaserJet 5 and 5M printers also provide device independent color specified using either the CIE $L^*a^*b^*$, Colorimetric RGB or Luminance-Chrominance color spaces. However, HP Color LaserJet 5 and 5M printers are somewhat restricted in what they provide. Details of this can be found in Appendix D.

Color Matching

When attempting to match color produced by different devices, it is important to know the difference between true color matching and appearance matching.

Proper device calibration can achieve true color matching, so that a side-by-side comparison of a printed page with the monitor on which the page was designed will show an exact match. However, true color matching is only satisfactory when using the monitor as a viewing reference. Viewed away from the screen, the printed page may appear flat and unsaturated because printers and monitors have different dynamic ranges. For example, black on the screen appears gray when compared to printed black, which is unacceptable if the intent is pure black. Likewise, the white produced on a monitor screen appears yellow or blue when compared to a white sheet of paper. True color matching would require that gray be printed in the black areas and colored dots be printed in the white areas.

Color Appearance Matching

Color appearance matching goes beyond true color matching by including adjustments for the dynamic ranges of the devices, so the user's intent is maintained. For example, the white areas of a page shown on a monitor display screen would be printed as white on a printed page because the user specified white, even though the screen cannot duplicate a white that truly matches white paper. Although the printed color does not exactly match screen color, color appearance does match, which is what users usually want. To maximize user satisfaction, the PCL language uses appearance matching when rendering device-independent color.

Color Lookup Tables

Color lookup tables, which provide additional control of the printed output, are transformations that map input data into a new output color range based upon point-by-point conversions.

Overhead transparencies provide one example of a good use for color lookup tables. Let's say a page is printed on plain paper and it matches the user's expectations. When printing the same document on overhead transparency film, the resulting image looks unsaturated and flat. To compensate, the user can send a color lookup table to increase color saturation without changing composition (for example, using the CIE $L^*a^*b^*$ color space to increase the a^* and b^* parameters in equal amounts).

Color lookup tables can also be used to adjust data from a Kodak CD-ROM, which uses the Photo YCC device-independent color space. The gamma correction table is complex and cannot be described by the traditional logarithmic expression. However, since the data can be mapped into new data values via tables, the user can provide a gamma correction table that essentially describes the complex correction factors.

Color lookup tables can be used to "neutral-balance" an image. For example, an underwater photograph produces a severe bluish cast when printed. The user can eliminate that cast from the image by providing a color lookup table that subtracts some color portion from each of the primaries.

Illumination Models

Illumination sources have different spectral distributions, causing colors to appear differently under one light source compared to another. For example, printed colors that look normal in natural sunlight shift in hue when viewed under fluorescent and tungsten lighting. The PCL language allows the user to compensate for the differences in viewing illumination using the Viewing Illuminant command. It allows the user to select different illuminations.

Processing Color Documents

To process a color page, PCL provides ways of specifying and modifying color so that the printed result appears as the user desires. This section provides a conceptual overview of the process.

Non-Raster Color vs. Raster Color

All color portions of a page consist of either:

- Page Marking Primitives (non-raster data)
- Color Raster Data

Page Marking Primitives

Non-raster data consists of HP-GL/2 and PCL page marking primitives such as glyphs, rules, polygons, circles, and vectors. Page marking primitives contain no color information about the image. They merely mark the page with attributes assigned to the current working environment (for example, colors, patterns, logical operation modes, etc.). Page marking primitives act as stencils through which color “paint” is poured, forming a homogeneous pattern.

Page marking primitives print in the currently specified color, which is specified using the Foreground Color command. For example, if you specify the color blue using the Foreground Color command, and then send some text to the printer, the text will be printed blue.

Color Raster Data

Unlike page marking primitives, each pixel of a color raster image contains color information. A color raster pixel may be defined by either:

- Palette Entry Indices
- Direct Color Specifications

User-defined color patterns are a form of color raster, but each pixel of a user-defined color pattern can be defined only by palette entry indices, not by direct color specifications.

Color Processing Functions

Given these two color uses, page marking primitives and color raster data, color processing must:

- Convert color attributes to an internal representation that can be poured through the page marking stencil onto the destination via some logical operation.
- Convert multiple-bit-per-pixel color raster to an internal representation that can be merged into the destination via some logical operation.

Color processing must have access to the following state variables, which indicate the form and attributes by which the two color groups are generated.

- Halftone (rendering algorithm)
- RGB gamma correction
- Device-dependent color lookup tables for each of the three primaries

Appendix B describes in more detail how color raster data is specified.

