MiscHalftoneColor

Inherits From: MiscColor

Conforms To: NXTransport

Declared In: misckit/MiscHalftoneColor.h

Class Description

The MiscHalftoneColor class implements a halftoned color. You can choose the amount of halftoning and the various PostScript screen parameters to specify to the Postscript interpreter how the color's halftone should be rendered. Besides the methods documented here, this class also overrides several of the MiscColor methods so that halftones can be properly supported on the display and while printing various separations.

The halftone amount decides how much ink is used to print the color on the printed page. A value of zero is no ink, while a value of 1.0 means the ink is printed solidly over the painted area. Any value in between will make use of a halftone screen, which defines some pattern which should be applied while painting. (This simulates

shades of a color.) You can determine the MiscHalftoneColor's amount of halftoning with the ±**halftone** method. The ±**setHalftone:** method allows you to change this value. If necessary, you can compare the halftone amounts of two MiscHalftoneColors via the ±**isEqualHalftoneAmount:** method.

In order to provide the proper color rendering on the screen, the MiscHalftoneColor class implements a filter in the ±**filterColor:** method. This method assumes that the halftone screen used is linear. Thus, the filter will return a color that starts at the base ink shade as defined by the input color and fades to white as the halftone amount decreases. If you need a non-linear filter to more accurately portray the effects of the halftone screens you are using, then you will need to subclass the MiscHalftoneColor and implement a properly calibrated filter function. Note that halftoning is by nature device dependant. As a result, a subclass of MiscHalftoneColor with the proper ±**filterColor:** method for each type of supported output device is really needed. This class at least provides a framework for you to begin with.

This implementation of MiscHalftoneColor does not yet have full support of halftone screens in it. At the moment it supports the amount of halftoning, but not halftone screen patterns and phase angles. Future implementations will add the necessary functionality to support the additional parameters.

Instance Variables

float **percent**;

percent	The color's halftone amount (range of 0.0 to 1.0).

Method Types

Halftoning	± filterColor: ± halftone ± isEqualHalftoneAmount: ± setHalftone:

Instance Methods

filterColor
- (NXColor)**filterColor:**(NXColor)*input*

Returns an NXColor whose color depends upon the input color and the amount of halftoning specified by the receiver's *percent* instance variable. The returned color is a linear interpolation between *input* and NX_COLORWHITE. When *percent* is 0.0, white is returned; when *percent* is 1.0 *input* is returned.

halftone
- (float)**halftone**

Returns the amount of halftoning, as stored in *percent*.

See also: -**isEqualHalftoneAmount:**, -**setHalftone:**

isEqualHalftoneAmount
- (BOOL)**isEqualHalftoneAmout:***anObject*

If *anObject* responds to the ±**halftone** message and returns a value equal to *percent*, this method returns YES. Otherwise, this method returns NO.

See also: -**halftone**, -**setHalftone:**

setHalftone
- **setHalftone:**(float)*amount*

Sets the amount of halftoning, stored in *percent*, to *amount*. Returns *self*.

See also: -**halftone**, -**isEqualHalftoneAmount:**