

## BaseCt™ 2.24 Guide

BaseCt is referred to as the "basic" control driver because it supplements the standard button, check box, and radio button CDEF-based controls with other basic control types that users have come to expect in typical Mac dialogs. These additional types include static & editable text controls, scrollable lists, dials, pop-up menus, and the use of icon and picture-based controls as static items, buttons, check boxes, radio buttons, and pop-up palettes.

In addition to supporting many useful control types, BaseCt also supports the full range of style options seen in ViewIt's editing menus. These style options combine with the variety of control types to produce a bewildering number of possibilities, all of which can be applied and tested from within a running program. Some of the more sensible combinations appear in ViewIt's own dialogs, and you can always enter edit mode and copy such controls for your own use.

**IMPORTANT:** A very powerful feature of BaseCt is its support for displaying resources in controls. Be certain to read the "Resource Lists" subtopic in the "Icons, Picts, ..." topic which describes how BaseCt uses resource links and the control's min, max, and value to determine which resource to display. Later topics assume that you understand this information.

## Resource Types

The resource types that BaseCt controls often make use of are briefly described here. ResEdit can be used to create and edit these resource types.

STR# - list of Pascal strings (any size)  
TEXT - block of plain text (any size)  
MENU - template used when creating new menus  
PICT - picture (any size)  
ICON - icon (32 x 32 bits)  
SICN - list of small icons (16 x 16 bits each)  
cicn - color icon (any size)  
PAT - pattern (8 x 8 bits)  
PAT# - list of patterns (8 x 8 bits each)  
CURS - cursor (16 x 16 bits)  
acur - list of CURS resources (16 x 16 bits each)  
clut - list of RGB colors (3 x 2-byte integers each)

Note that cicn color icons are complex structures that include both a mask (like cursors) and a B&W version for use when color is not available.

## Hilited Appearance

Buttons, check boxes, radio buttons, and menu controls become "hilited" when pressed. In addition, some check boxes and radio buttons use their hilited state as their checked state. If a separate resource has not been assigned to this hilite state, then BaseCt uses one of 4 methods to draw the hilited control:

- Color Switching: If the following is true,
  - not linked to a cicn or CURS resource
  - has solid body
  - body color is not white or black
  - "Global Hilite" option is not set (in Control dialog)

then hiliting is done by redrawing the control after switching the control's body and content colors (as is done by Apple's standard buttons).

- Dimming: If linked to a cicn resource or the following is true,
  - not linked to CURS resource
  - 4096 added to VarCode
  - color in use
  - "Global Hilite" option is not set (in Control dialog)

then colors are "dimmed" (like what the System ≥ 7 Finder does to selected file icons). If not linked to a cicn, and the control is transparent, then the body color of the parent view is dimmed without affecting any other colors (the menu controls at the top of the main ViewIt Help window use this option).

- Cursors: If linked to a CURS resource, then hiliting is done by painting the cursor's mask with black (or the control's body with black if mask is not drawn). If the content color of the drawn cursor is also black,

then this is changed to white. (The "Hand" check box in the Bounds dialog is an example of this.)

- Inversion: If none of the above conditions apply, then the control's contents are simply inverted. If the "Global Hilite" option is checked in the Control dialog, then this inversion displays the System hilite color in place of the control's body color.

## Inactive Appearance

Controls can be "inactivated" via the corresponding menu item from within edit mode, or via the ActCtl command. BaseCt usually distinguishes inactive from active controls by painting inactive controls with a gray pattern using the control's background color. If, however, the program is running under System  $\geq 7$ , in a color window, at a screen depth  $\geq 4$  bits/pixel, then BaseCt will draw inactive text and simple icons (from ICON, SICN, CURS, or acur resources) using an intermediate color between the control's background and content colors (instead of painting with gray).