

Editable Text

Editable text controls must be of type "Editable", and can be optionally linked to an STR# resource. When linked to an STR# resource, the string list is used to support "parameter" text that is read just once when the control is initialized. See "Static Text" for a further discussion of parameter text.

Operation

BaseCt uses TextEdit routines to create and display editable text controls. The editable control's TEHandle can be found in cHiData (after calling GetCtl), and used with TextEdit toolbox calls. The TE record is an old-style record, meaning that multiple styles are not supported within the same editable text control.

Editable text controls support the standard menu items Cut (13), Copy (14), Paste (16), Clear (18), and Select All (19). These menu items can be made available through menu controls in same window, or, for modeless windows, through menus in the main menu bar.

Options

The following bit values can be added to VarCode to set editable text control options:

1 = always select all of text in control when control becomes selected. This option mimics standard dialog behavior which selects all text in editable text items as the user tabs between items.

2 = limit number of characters entered to size of linked string variable. This option only works when the editable text item is linked to a program string variable (cDataType < 0 - see "Data Links" in the ViewIt Guide for more info on data linking). Note that this only serves as an input aid for users, since the program never gets back (from GetVal) a string larger than the size of its linked string variable.

4 = always reselect text in currently selected control after SetVal is executed (otherwise the text is only reselected if changed by SetVal).

8 = do not allow editing of the text (read only). This is useful when you need a text item that is selectable but not editable. A selected, read-only text item, for example, could be used to display and edit the current style settings for an object drawn by the program.

User-Defined Styles

If the "Uses FSSC" option is checked in the Control dialog, then the editable text item supports Utilt's Font, Size, Style, and Color menus. This means that the user will be able to change the control's text font, size, style, or color using the FSSC menus (which can be made available as hierarchical menus from any main or control menu - try the "Style Picker" FCTL example control).

The font, size, style, and color chosen by the user can be retrieved from fRec after calling GetCtl (in cTxFont, cTxSize, cTxFace, cColors). To get the content color from the associated cColors table, use, Facelt(nil,GetFgC,0,-2,ord(cColors),2);

where "2" is the part number corresponding to the control's content, and the color is returned in uRGB. Also note that for color selection to work properly, the "content" color of the editable control must be set at least once using ViewIt's Style menu in edit mode.

One way of saving the current text font, size, style, and color settings associated with editable controls in a ViewIt window is to update the associated FWND on disk. This is easily done by using GetWVC to update the FWND in memory, followed by SavWnd to write the updated FWND to disk.

Data Linking

Editable text controls have "values" equal to a Pascal string containing up to 255 characters (see "Data Links" in the ViewIt Guide for more info on data linking). As with static text, this makes it easy to have editable text items in ViewIt windows that are directly linked to integer, real, or string program variables. BaseCt also does elementary filtering of typed characters based on the data type of the linked program variable. For example, if editable text is linked to an integer, then non-numbers other than "+" or "-" cannot be entered (they just beep).

Limitations

Although editable text items act as single controls, the "GetCtl..." and "SetCtl..." toolbox calls do not get or set the control's state since no use is made of the control's "contrlValue", "contrlMin", or "contrlMax".

All limitations of Apple's TextEdit apply (such as 32K per text block). The global hilite color is always

used whether that option is checked or not. No support for control colors other than frame, body, & content. No support for hand scrolling. Must have solid body.