

## Bounds Settings

This dialog resets parameters that determine size-related control features. Changes made here affect only the FWND in memory. Use "Save All to disk" to make changes permanent.

**Bounds** - The maximum area occupied by the control (= the standard "controlRect" associated with Macintosh controls).

**Indent** - The top, left, bottom, and right indent from the Bounds rectangle (a "margin") within which the content of the control is to be drawn. Most ViewIt controls support use of an indent, but standard CDEF-based controls do not. Note that if the top or left indent is negative, then the content is clipped at the top or left by that amount. This can be useful, for example, when drawing a PICT created by a drawing program that always adds "slop" to the top and left sides of copied PICTs.

**Content** - The minimum and maximum size of the control's content area. These values are used by ViewIt to limit resizing of the control, and to support "hand scrolling" when the content area is larger than the control Bounds. In some cases, it is preferable to have the content area "stretch" to fit the Bounds as the control is resized. To make this happen, set "Max V" and/or "Max H" to zero. Some ViewIt controls support use of the Content rectangle and others ignore it, always drawing their content to fit the Bounds (- Indent) area.

**Hand Icon** - If the control's "Max V" or "Max H" Content area is  $> 0$ , then this icon can be checked to enable "hand scrolling" of the control's content (= dragging the content with a hand cursor). This support for "hand scrolling" is built into ViewIt, and works best with static-type controls that do not support their own scroll bars. A static control linked to a large PICT, for example, works well with hand scrolling, but a HelpCt control does not since the latter already has built-in support for scroll bar-based scrolling.

The control driver's on-line help will usually indicate whether hand scrolling can be used with its controls. Also note that if a hand-scrollable control is not a "static" control, then the OPTION key must be pressed to hand scroll, since clicks in such controls usually have another meaning.

**Frame** - The thickness of the control's bounding rectangle. Most ViewIt controls support drawing of a frame, but standard CDEF controls do not.

**Round** - Can be optionally used by drivers to support the drawing of rounded control frames.

**Add/Attach** - Special size- or frame-related options directly supported by ViewIt (w/o help from control drivers):

**Add Ring** - Draws a 3-pixel wide ring around the control. Works best with standard default buttons. Note that adding a ring does not make the control the default control (this is supported by a separate option in the Control dialog).

**Add Shadow** - Adds a 1-pixel shadow to the right and bottom sides of the control. Looks best with isolated button or pop-up menu controls that also have a 1-pixel frame.

**Attach Right** - Forces ViewIt to keep the right edge of the control equal to the right edge of the content area of its parent view or window.

**Attach Bottom** - Forces ViewIt to keep the bottom edge of the control equal to the bottom edge of the content area of its parent view or window.

The "Attach Right" and "Attach Bottom" options affect ViewIt's support for zooming and growing windows. When control's are "attached" and are not of a fixed size (Min H  $\neq$  Max H or Min V  $\neq$  Max V), then the parent view or window cannot be made larger or smaller than the limits of the attached controls. If, for example, a text-editing control is attached to the right and bottom edges of its parent view, and this view is attached to the right and bottom edges of the window, then ViewIt will not allow the window to be made smaller or larger than the limits imposed by the text-editing control.

The "Attach Right" and "Attach Bottom" options can also be used to support views or controls that "float" with the parent window or view edge. If an attached control is of a fixed size (Min H = Max H or Min V = Max V), then the control is moved in the corresponding dimension without changing its size as its parent view or window is resized. Thus such controls also remain "attached", but do so by floating rather than stretching.