NSRulerView Client Example

This project demonstrates many of the interactions between an NSRulerView and its client view. It should give you an idea how to go about this when creating a view subclass that really does something.

The principal class, RectsView, displays colored rectangles that the user can select and drag around. The RectsView puts markers in the horizontal and vertical rulers showing the placement of the rectangle; manipulating these markers changes the size of the selected rectangle. Removing a marker deletes the selected rectangle, and clicking in the ruler area creates a new rectangle. The user can also lock a rectangle down, so that the markers can't be moved.

RectsView.m defines a lot of methods, but these are the ones primarily related to working with NSRulerViews:

METHOD	ACTION
rulerView:shouldMoveMarker:	Denies move if no selection or if selection is locked.
rulerView:willMoveMarker: toLocation:	Limits minimum rectangle size and updates display; try changing this to snap

	markers to whole units.
rulerView:didMoveMarker:	Refreshes ruler display.
rulerView:shouldRemoveMarker:	Denies remove of no selection or if selection is locked.
rulerView:didRemoveMarker:	Deletes the selected rectangle and refreshes the display.
rulerView:shouldAddMarker:	Returns YES, always allowing new rectangles to be added.
rulerView:willAddMarker: atLocation:	Returns the proposed location unchanged; try changing this to snap markers to whole units.
rulerView:didAddMarker:	Creates a new rectangle.
rulerView:handleMouseDown:	Adds a new marker to be tracked.
rulerView:willSetClientView:	Does nothing.