SliderDualActing

INHERITS FROM	Slider: Control : View : Responder : Object
REQUIRES HEADER FILES	SliderDualActing.h
DEFINED IN	Acme Gizmos, version 1.0

CLASS DESCRIPTION

SliderDualActing is a Control that combines a textField and slider as one logical object, knows how to "undo" itself, synchronize its displays, and validate and reset the slider if the user types in values higher than **maxValue** or lower than **minValue**.

INSTANCE VARIABLES

Inherited from Object	Class	isa;
Inherited from Responder	id	nextResponder;
Inherited from View	NXRect	frame;
	NXRect	bounds;
	id	superview;
	id	subviews;
	id	window;
	struct __vFlags	vFlags;
Inherited from Control	int	tag;
	id	cell;
	struct _conFlags	conFlags;
Inherited from Slider	(none)	
Declared in SliderDualActing	id	textPal;
	id	upTarget;
	SEL	upAction;
	id	undoTarget;
	int	undoPosition;
	double	lastValue;

METHOD TYPES

Initializing SliderDualActing Factory	+ initialize
Creating and initializing a SliderDualActing	+ newFrame: - setUpTarget:action: - setUpTarget:action:isContinuous

- setAltStep: whole: default:
- setMax:allowHigher:min:allowLower:

Drone TextField Methods
- setFormat:left:right:
- isDecimal

Interface Builder Methods
- sendTextAction:
- incrementDecrement:

Setting SliderDualActing Values
- setIntValue:
- seFloatValue:

Undo Methods
- undo
- setUndoTarget:position
- setLastThing:

Handling Events
- mouseDown:

CLASS METHODS

newFrame:

+ **newFrame:**(const NXRect *)*frameRect*

Creates a new SliderDualActing. Does not create the textfield or the button matrix. These must be created in IB and and the textfield connected to the **textPal** outlet. This allows you to set attributes of the textfield in IB, as well as its location, font, and other attributes.

initialize

+inialize

This method initializes the subclass of SliderCell used in implementing SliderDualActing. The default is SliderCellFine. Use this when you subclass SliderCellFine to modify the behaviour of a SliderDualActing.

INSTANCE METHODS

incrementDecrement:

- **incrementDecrement:** *sender*

The matrix of buttons sends this message to the slider.The slider increments or decrements (depending on the button's tag) by the **altStep** instance variable.

isDecimal:

- **isDecimal:**(BOOL)*flag*

Returns whether the SliderDualActing's **textPal** is formatted to show decimal places.

mouseDown:

- **mouseDown:**(NXEvent *)*theEvent*

Calls **trackMouse:inRect:ofView:** on the Slider's cell. Overridden

to set undo variable before changing the slider and to send the **upAction:** when mouse goes up.

read:

- **read:**(NXTypedStream *)*stream*

Reads the SliderDualActing from the typed stream *stream*.

sendTextAction:

- **sendTextAction:**(id)*sender*

This method is provided for the **textPal** when you connect it to the slider in IB. It sends the **upAction** to the **upTarget**, after validating the entry for exceeding limits.

setAltStep:whole:default:

- **setAltStep:**(float)*aFloat* **whole:**(BOOL)flag *default***:** (float)*aFloat2*

Sets the value by which SliderDualActing increments and decrements, whether the field is an integer value, and the default value to which the slider should be reset when command-clicked. It also initializes the slider's value to the default value.

setDefault:

- **setDefault:**(float)*aFloat*

Sets the default value of the SliderDualActing.

setFloatValue:

- **setFloatValue:**(float)*aFloat*

Sets both the slider and textfield.

setFormat:left:right:

- **setFormat:**(BOOL)*auto* **left:**(unsigned short)*aShort* **right:** (unsigned short)*aShort*

Sets the text formatting of the slider's **textPal**. It is a cover method for TextField's **setFloatingPointFormat:left:right:**, but without peeking inside the object. NOTE: this may need to be called after **appDidInit:**; if the **textPal** is not initialized when the slider gets initialized, your message will not get sent.

setIntValue:

- **setIntValue:**(int)*anInt*

Sets both the slider and textfield to *anInt*.

setLastThing:

- **setLastThing:**(int)*poundDefinedTag*

Caches value of SliderDualActing before it changes. See **Undo**.

setMaxValue:allowHigher:setMin:allowLower:

- **setMaxValue:**(float)*max* **allowHigher:**(BOOL)*hi* **setMin:**(float)*min* **allowLower:**(BOOL)*lo*

Sets the maximum and minimum value of the SliderDualActing. Sets the flags which allow user to exceed or not to exceed these values.

setUndoTarget:tag:

- **setUndoTarget:**(id)*targ* **tag:**(int)*tag*

Sets the **undoTarget** and the pound-defined tag used by the **undoTarget** if it's sent an **undo** message. By default, we handle the undo, and the tag is 0. See **undo.**

setUpAction:

- **setUpAction:**(SEL)*action*

Sets the **upAction** which gets sent to the **upTarget** on mouseUp, or if
scfFlags.sendContinuously is YES, continuously.

setUpTarget:

- **setUpTarget:**(id)*targ*

Sets the **upTarget** which will receive the action **upAction** on mouseUp, or if **scfFlags.sendContinuously** is YES, continuously while the slider is dragged.

setUpTarget:action:

- **setUpTarget:**(id)*targ* **action:**(SEL)*action*

Sets the **upTarget** and **upAction** of the SliderDualActing. **scfFlags.sendContinuously** is NO.

setUpTarget:action:isContinuous:

- **setUpTarget:**(id)*targ* **action:**(SEL)*action* **isContinuous:**(BOOL)*flag*

Sets the **upTarget** and **upAction** of the SliderDualActing. **scfFlags.sendContinuously** is set to flag.

textPal

-textPal

Returns the SliderDualActing's textField instance, **textPal**.

undo

-undo

Restores the SliderDualActing to its previous state. Sends a display method to
the NXApp mainWindow as a default. If the user has set the **undoTarget** variable, the slider will notify this target of an impending change with **setLastThing**:*last_id* before changing

the value of the slider.
The default is for the undoTarget to be self, in which case we cache the last value and respond to the undo message by restoring **lastValue**'s value.

write:

- **write:**(NXTypedStream *)*stream*

Writes the receiving SliderDualActing to the typed stream *stream*.