SliderCellFine

INHERITS FROM SliderCell: ActionCell : Cell : Object

REQUIRES HEADER FILES SliderCellFine.h

DEFINED IN Acme Gizmos, version 1.0

CLASS DESCRIPTION

The SliderCellFine is used to implement the SliderDualActing Gizmo™ Control. See Class SliderDualActing for details. The **trackRect** is the rectangle inside which tracking occurs (i.e. the interior of the bezeled area).

INSTANCE VARIABLES

Inherited from Object	Class	isa;
Inherited from Cell	char	*contents;
	id	support;
	struct _cFlags1	cFlags1;
	struct _cFlags2	cFlags2;
Inherited from ActionCell	int	tag;
	id	target;
	SEL	action;
Inherited from SliderCell	double	value;
	double	maxValue;
	double	minValue;
	NXRect	trackRect;
Declared in SliderCellFine	id	textPal;
	struct _scfFlags	scfFlags;
	double	altStep;
	double	defaultValue;
	double	defaultMax;
	double	defaultMin;

textPal	The controlView's text field: cached for speed
altStep	The value step when alt key is depressed during drag
defaultValue	The original and reset default value
defaultMax	The starting and reset default maximum
defaultMin	The starting and reset default minimum

METHOD TYPES

Initializing	+new - finalInit
Setting SliderCell Values	- increment - decrement - checkValue:
Setting SliderCell Defaults	- setAltStep: whole: default: - setMax:allowHigher:min:allowLower: - setDefault:
Tracking the Mouse	- continueTracking:at:inView:
Archiving	- read: - write:

CLASS METHODS

new

+ **new**

Creates a default SliderCellFine. The **value** is 50.0, the **minValue** is 0.0, and the **maxValue** = 100.0 and the SliderCellFine does not send upAction continually [although it might send its action continuously: see SliderCell].

INSTANCE METHODS

checkValue:

- (double) **checkValue:**(double)*val*

Returns a validated value based on **defaultMax** and **defaultMin**.

continueTracking:at:inView:

- (BOOL)**continueTracking:**(const NXPoint *)*lastPoint*
  at:(const NXPoint *)*currentPoint*
  inView:*controlView*

Overridden to add fine positioning with the Alternate and Shift keys, resetting to the defaults with the Command key, and making float-formatted fields with the Alternate and Shift keys.

increment

- **increment**

Increases the cell's value by **altStep**. Called by SliderDualActing.

isDecimal

- (BOOL) **isDecimal**

Returns YES if **textPal** is showing fractional numbers

decrement

- **decrement**

Reduces the cell's value by **altStep**.Called by SliderDualActing.

read:

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

Reads the SliderCellFine from the typed stream *stream*.

setAltStep:whole:default:

- **setAltStep:**(double) *aDouble* **whole:** (BOOL)*flag* **default:**(double) *aDouble2*

Sets the value by which SliderCellFine 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:**(double)*aDouble*

Sets the default value of the SliderCellFine to *aDouble*. This is the value that the slider will return to if Command-clicked.

setMax:allowHigher:min:allowLower:

- **setMax:**(double)*max* **allowHigher:**(BOOL)*hi* **min:**(double)*min* **allowLower:**(BOOL)*lo***;**

Sets the default maximum and minimum values of the SliderCellFine to *max* and *min* respectively. The booleans determine whether a value entered into the text field which is higher than **defaultMax** or lower than **defaultMin** can change the **minValue** and **maxValue** of the slider. If *hi* is YES, for example, then the user is free to type in any larger number and then the slider will reset its maximum value. Command-click restores the original settings.

setMaxValue:

- **setMaxValue:**(double)*aDouble*

Sets the maximum value of the SliderCell.

setTextPal:

- **setTextPal:**textField

Sets the textField instance.

write:

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

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