

CAPH.WUI.ANI.TRANSFERANIMATION

TransferAnimation represents a class that provides translate effect for widgets. This is one of the six basic types of animation.

Contents

Constructor
TransferAnimation

Methods
TransferAnimation
add
remove
getList
clone
equals

Constructor

TransferAnimation

Description		
Create a TransferAnimation.		
Parameters		
obj	Object	A widget created by user.
option	Object	The option include below properties * x : (Number) indicates the X coordinate of position of widget after animation. * y : (Number) indicates the Y coordinate of position of widget after animation. * z : (Number) indicates the Z coordinate of position of widget after animation. * ease : (String) describes the motion tween of the batch of animations, it could be set as one of the values below - ('SineCubic.InOut', 'Linear.None', if user don't set ease, this is default. 'Quadratic.In', 'Quadratic.Out', 'Quadratic.InOut', 'Cubic.In', 'Cubic.Out', 'Cubic.InOut', 'Quartic.In', 'Quartic.Out', 'Quartic.InOut', 'Quintic.In', 'Quintic.Out', 'Quintic.InOut', 'Sinusoidal.In', 'Sinusoidal.Out', 'Sinusoidal.InOut', 'Exponential.In', 'Exponential.Out', 'Exponential.InOut', 'Circular.In', 'Circular.Out', 'Circular.InOut', 'Elastic.In', 'Elastic.Out', 'Elastic.InOut', 'Back.In', 'Back.Out', 'Back.InOut', 'Bounce.In', 'Bounce.Out', 'Bounce.InOut') * duration : (Number) describes how long the batch of animations to be performed in the meanwhile would last, the unit is millisecond(ms). - [default : 1000] * delay : (Number) describes the time duration of the batch of animations would defer starting, the unit is millisecond(ms))
Emulator Support	Y	
SDK Constraint	None	
Example		
<pre>var TransferAnimation = caph.wui.ani.TransferAnimation; var AniLoader = caph.wui.ani.AniLoader; var Box = caph.wui.widget.Box; var UIContext = caph.wui.widget.UIContext; var transfer = new TransferAnimation(); var loader = new AniLoader(); var uiContext = new UIContext(); var widget = new Box(); widget.render(uiContext); var transferOpt = {x: 260, y: 480, duration: 2000}; transfer.add(widget, transferOpt); loader.add(sync); loader.start(uiContext);</pre>		

Methods

TransferAnimation	
Description	
(Constructor) Create a TransferAnimation.	
Parameters	<div>■obj</div> <div>- Object</div> <div>- A widget created by user.</div> <div>■option (Optional)</div> <div>- Object</div> <div>- The option include below properties</div> <div>* x : (Number) indicates the X coordinate of position of widget after animation.</div> <div>* y : (Number) indicates the Y coordinate of position of widget after animation.</div> <div>* z : (Number) indicates the Z coordinate of position of widget after animation.</div> <div>* ease : (String) describes the motion tween of the batch of animations, it could be set as one of the values below</div> <div>- ('SineCubic.InOut', 'Linear.None', if user don't set ease, this is default. 'Quadratic.In', 'Quadratic.Out', 'Quadratic.InOut', 'Cubic.In', 'Cubic.Out', 'Cubic.InOut', 'Quartic.In', 'Quartic.Out', 'Quartic.InOut', 'Quintic.In', 'Quintic.Out', 'Quintic.InOut</div> <div>* duration : (Number) describes how long the batch of animations to be performed in the meanwhile would last, the unit is millisecond(ms).</div> <div>- [default : 1000]</div> <div>* delay : (Number) describes the time duration of the batch of animations would defer starting, the unit is millisecond(ms))</div>
Return	■Void
Emulator Support	Y
SDK Constraint	none
Example	

```
var TransferAnimation = caph.wui.ani.TransferAnimation;
var AniLoader = caph.wui.ani.AniLoader;
var Box = caph.wui.widget.Box;
var UIContext = caph.wui.widget.UIContext;
```

```
var transfer = new TransferAnimation();
var loader = new AniLoader();
var uiContext = new UIContext();
var widget = new Box();
widget.render(uiContext);
```

```
var transferOpt = {x:260, y:480, duration:2000};
transfer.add(widget, transferOpt);
```

```
loader.add(sync);
loader.start(uiContext);
```

add

Description

Binds the options needed for transfer animation with widget. Before binding the options, it will call the parent class " _checkAdd" method to check the validity of the options.

Parameters	<div>■obj</div> <div>- Object</div> <div>- A widget created by user.</div> <div>■option (Optional)</div> <div>- Object</div> <div>- The option include below properties</div> <div>* x: (Number) indicates the X coordinate of position of widget after animation.</div> <div>* y: (Number) indicates the Y coordinate of position of widget after animation.</div> <div>* z: (Number) indicates the Z coordinate of position of widget after animation.</div> <div>* ease : (String) describes the motion tween of the batch of animations, it could be set as one of the values below</div> <div>- ('SineCubic.InOut', 'Linear.None', if user don't set ease, this is default 'Quadratic.In', 'Quadratic.Out', 'Quadratic.InOut', 'Cubic.In', 'Cubic.Out', 'Cubic.InOut', 'Quartic.In', 'Quartic.Out', 'Quartic.InOut', 'Quintic.In', 'Quintic.Out', 'Quintic.InOut</div> <div>* duration : (Number) describes how long the batch of animations to be performed in the meanwhile would last, the unit is millisecond(ms).</div> <div>- [default : 1000]</div> <div>* delay : (Number) describes the time duration of the batch of animations would defer starting, the unit is millisecond(ms)}</div>
Return	■Void
Emulator Support	Y
SDK Constraint	none

Example

```
var TransferAnimation = caph.wui.ani.TransferAnimation;
var AniLoader = caph.wui.ani.AniLoader;
var Box = caph.wui.widget.Box;
var UIContext = caph.wui.widget.UIContext;
```

```
var transfer = new TransferAnimation();
var loader = new AniLoader();
var uiContext = new UIContext();
var widget = new Box();
widget.render(uiContext);
```

```
var transferOpt = {x:260, y:480, duration:2000};
transfer.add(widget, transferOpt);
```

```
loader.add(sync);
loader.start(uiContext);
```

remove

Description

Removes all the arguments that needed for animation on the widget.

Parameters	<div>■obj</div> <div>- Object</div> <div>- An instance of widget created by user.</div>
------------	---

Return	■Void
--------	-------

Emulator Support	Y
------------------	---

SDK Constraint	none
----------------	------

Example

```
var TransferAnimation = window.caph.wui.widget.TransferAnimation;
var animation =new TransferAnimation();
```

```
// Let's say that obj and options are given.
animation.add(obj, options);
animation.remove(obj);
```

getList

Description

Returns the array that contains widget-arguments pairs, here 'arguments' are the options needed for animation.

Parameters	■Void
Return	<div>■Array</div> <div>- The array of a object pairs,including widget and options of its' animation.</div>

Emulator Support	Y
------------------	---

SDK Constraint	none
----------------	------

Example

```
var TransferAnimation = window.caph.wui.ani.TransferAnimation;
var animation =new TransferAnimation();
```

```
// Use Animation Sub-classes....
```

```
var arr = animation.getList();
for(var i = 0; i < arr.length; i++) {
    console.log(arr);
}
```

clone

Description

Creates and returns clone object from current object, the cloned object will have the same properties and same methods with the current object.

Parameters

■Void

Return

■Object
- The cloned object.

Emulator Support

Y

SDK Constraint

none

Example

```
var TransferAnimation = window.caph.wui.widget.TransferAnimation;
var animation =new TransferAnimation();
var obj = animation.clone();
```

equals

Description

Compares the contents of two objects using strict equality, objects are considered equal if they both have the same set of properties and the values of those properties are equal.

Parameters

■Object
- Object
- The object which wants to compare with current object.

Return

■Boolean
- Indicates whether the two objects are equal.
- true : if they are equal, return true.
- false : if they aren't equal, return false.

Emulator Support

Y

SDK Constraint

none

Example

```
var TransferAnimation = window.caph.wui.widget.TransferAnimation;
var animation =new TransferAnimation();
var obj = animation.clone();
var isequal = obj.equals(animation);
```