[ScrollMagic](http://docs.google.com/index.html)

* [Classes](http://docs.google.com/classes.list.html)
  + [Controller](http://docs.google.com/ScrollMagic.Controller.html)
  + [Scene](http://docs.google.com/ScrollMagic.Scene.html)
* [Events](http://docs.google.com/events.list.html)
  + [add](http://docs.google.com/ScrollMagic.Scene.html#event:add)
  + [change](http://docs.google.com/ScrollMagic.Scene.html#event:change)
  + [destroy](http://docs.google.com/ScrollMagic.Scene.html#event:destroy)
  + [end](http://docs.google.com/ScrollMagic.Scene.html#event:end)
  + [enter](http://docs.google.com/ScrollMagic.Scene.html#event:enter)
  + [leave](http://docs.google.com/ScrollMagic.Scene.html#event:leave)
  + [progress](http://docs.google.com/ScrollMagic.Scene.html#event:progress)
  + [remove](http://docs.google.com/ScrollMagic.Scene.html#event:remove)
  + [shift](http://docs.google.com/ScrollMagic.Scene.html#event:shift)
  + [start](http://docs.google.com/ScrollMagic.Scene.html#event:start)
  + [update](http://docs.google.com/ScrollMagic.Scene.html#event:update)
* [Plugins](http://docs.google.com/mixins.list.html)
  + [GSAP](http://docs.google.com/animation.GSAP.html)
  + [Velocity](http://docs.google.com/animation.Velocity.html)
  + [addIndicators](http://docs.google.com/debug.addIndicators.html)
  + [jQuery](http://docs.google.com/framework.jQuery.html)

Plugin: GSAP

This plugin is meant to be used in conjunction with the Greensock Animation Plattform.

It offers an easy API to trigger Tweens or synchronize them to the scrollbar movement.

Both the lite and the max versions of the GSAP library are supported.

The most basic requirement is TweenLite.

To have access to this extension, please include plugins/animation.gsap.js.

Source:

* [plugins/animation.gsap.js](http://docs.google.com/plugins_animation.gsap.js.html), [line 8](http://docs.google.com/plugins_animation.gsap.js.html#sunlight-1-line-8)

### Requires

* [GSAP ~1.14.x](http://greensock.com/gsap)

### Scene Constructor Extension

#### new ScrollMagic.Scene**(options)**

Every instance of ScrollMagic.Scene now accepts an additional option.

See [ScrollMagic.Scene](http://docs.google.com/ScrollMagic.Scene.html) for a complete list of the standard options.

##### Parameters:

| Name | Type | Argument | Description |
| --- | --- | --- | --- |
| options | object | <optional> | Options for the Scene. The options can be updated at any time. Properties  | Name | Type | Argument | Default | Description | | --- | --- | --- | --- | --- | | tweenChanges | boolean | <optional> | false | Tweens Animation to the progress target instead of setting it.  Does not affect animations where duration is 0. | |

Source:

* [plugins/animation.gsap.js](http://docs.google.com/plugins_animation.gsap.js.html), [line 53](http://docs.google.com/plugins_animation.gsap.js.html#sunlight-1-line-53)

##### Example

var scene = new ScrollMagic.Scene({tweenChanges: true});

### Scene Control Methods

#### Scene.removeTween**(reset) → {Scene}**

Remove the tween from the scene.

This will terminate the control of the Scene over the tween.

Using the reset option you can decide if the tween should remain in the current state or be rewound to set the target elements back to the state they were in before the tween was added to the scene.

##### Parameters:

| Name | Type | Argument | Default | Description |
| --- | --- | --- | --- | --- |
| reset | boolean | <optional> | false | If true the tween will be reset to its initial values. |

Source:

* [plugins/animation.gsap.js](http://docs.google.com/plugins_animation.gsap.js.html), [line 293](http://docs.google.com/plugins_animation.gsap.js.html#sunlight-1-line-293)

##### Returns:

Parent object for chaining.

{ Scene }

##### Example

// remove the tween from the scene without resetting it  
scene.removeTween();  
  
// remove the tween from the scene and reset it to initial position  
scene.removeTween(true);

#### Scene.setTween**(TweenObject, duration, params) → {Scene}**

Add a tween to the scene.

If you want to add multiple tweens, add them into a GSAP Timeline object and supply it instead (see example below).

If the scene has a duration, the tween's duration will be projected to the scroll distance of the scene, meaning its progress will be synced to scrollbar movement.

For a scene with a duration of 0, the tween will be triggered when scrolling forward past the scene's trigger position and reversed, when scrolling back.

To gain better understanding, check out the [Simple Tweening example](http://docs.google.com/examples/basic/simple_tweening.html).

Instead of supplying a tween this method can also be used as a shorthand for TweenMax.to() (see example below).

##### Parameters:

| Name | Type | Description |
| --- | --- | --- |
| TweenObject | object | string | A TweenMax, TweenLite, TimelineMax or TimelineLite object that should be animated in the scene. Can also be a Dom Element or Selector, when using direct tween definition (see examples). |
| duration | number | object | A duration for the tween, or tween parameters. If an object containing parameters are supplied, a default duration of 1 will be used. |
| params | object | The parameters for the tween |

Source:

* [plugins/animation.gsap.js](http://docs.google.com/plugins_animation.gsap.js.html), [line 193](http://docs.google.com/plugins_animation.gsap.js.html#sunlight-1-line-193)

##### Returns:

Parent object for chaining.

{ Scene }

##### Example

// add a single tween directly  
scene.setTween(TweenMax.to("obj"), 1, {x: 100});  
  
// add a single tween via variable  
var tween = TweenMax.to("obj"), 1, {x: 100};  
scene.setTween(tween);  
  
// add multiple tweens, wrapped in a timeline.  
var timeline = new TimelineMax();  
var tween1 = TweenMax.from("obj1", 1, {x: 100});  
var tween2 = TweenMax.to("obj2", 1, {y: 100});  
timeline  
 .add(tween1)  
 .add(tween2);  
scene.addTween(timeline);  
  
// short hand to add a TweenMax.to() tween  
scene.setTween("obj3", 0.5, {y: 100});  
  
// short hand to add a TweenMax.to() tween for 1 second  
// this is useful, when the scene has a duration and the tween duration isn't important anyway  
scene.setTween("obj3", {y: 100});

### Scene Parameters (getter / setter)

#### Scene.tweenChanges**(newTweenChanges) → {boolean}**

**Get** or **Set** the tweenChanges option value.

This only affects scenes with a duration. If tweenChanges is true, the progress update when scrolling will not be immediate, but instead the animation will smoothly animate to the target state.

For a better understanding, try enabling and disabling this option in the [Scene Manipulation Example](http://docs.google.com/examples/basic/scene_manipulation.html).

##### Parameters:

| Name | Type | Argument | Description |
| --- | --- | --- | --- |
| newTweenChanges | boolean | <optional> | The new tweenChanges setting of the scene. |

Source:

* [plugins/animation.gsap.js](http://docs.google.com/plugins_animation.gsap.js.html), [line 65](http://docs.google.com/plugins_animation.gsap.js.html#sunlight-1-line-65)

##### Fires:

* Scene.change,event: when used as setter

##### Returns:

* get - Current tweenChanges option value.{ boolean }
* set - Parent object for chaining.{ Scene }

##### Example

// get the current tweenChanges option  
var tweenChanges = scene.tweenChanges();  
  
// set new tweenChanges option  
scene.tweenChanges(true);

© Jan Paepke 2015

Documentation generated by [JSDoc 3.5.5](https://github.com/jsdoc3/jsdoc) using a customized version of the [DocStrap template](https://github.com/terryweiss/docstrap).