
xSGE Physics Framework Documentation

Release 0.12

onpon4

December 03, 2016

1	xsge_physics Classes	3
1.1	xsge_physics.Collider	3
1.2	xsge_physics.Wall	5
1.3	xsge_physics.SolidLeft	5
1.4	xsge_physics.SolidRight	6
1.5	xsge_physics.SolidTop	6
1.6	xsge_physics.SolidBottom	7
1.7	xsge_physics.Solid	7
1.8	xsge_physics.Slope	8
1.9	xsge_physics.SlopeTopLeft	8
1.10	xsge_physics.SlopeTopRight	9
1.11	xsge_physics.SlopeBottomLeft	10
1.12	xsge_physics.SlopeBottomRight	11
1.13	xsge_physics.MobileWall	11
1.14	xsge_physics.MobileColliderWall	13
	Python Module Index	15
	Index	17

Contents

- *xSGE Physics Framework*
 - *xsg physics Classes*
 - * *xsg physics.Collider*
 - *xsg physics.Collider Methods*
 - *xsg physics.Collider Event Methods*
 - * *xsg physics.Wall*
 - * *xsg physics.SolidLeft*
 - *xsg physics.SolidLeft Event Methods*
 - * *xsg physics.SolidRight*
 - *xsg physics.SolidRight Event Methods*
 - * *xsg physics.SolidTop*
 - *xsg physics.SolidTop Event Methods*
 - * *xsg physics.SolidBottom*
 - *xsg physics.SolidBottom Event Methods*
 - * *xsg physics.Solid*
 - *xsg physics.Solid Event Methods*
 - * *xsg physics.Slope*
 - * *xsg physics.SlopeTopLeft*
 - *xsg physics.SlopeTopLeft Event Methods*
 - * *xsg physics.SlopeTopRight*
 - *xsg physics.SlopeTopRight Event Methods*
 - * *xsg physics.SlopeBottomLeft*
 - *xsg physics.SlopeBottomLeft Event Methods*
 - * *xsg physics.SlopeBottomRight*
 - *xsg physics.SlopeBottomRight Event Methods*
 - * *xsg physics.MobileWall*
 - *xsg physics.MobileWall methods*
 - * *xsg physics.MobileColliderWall*

xSGE is a collection of extensions for the SGE licensed under the GNU General Public License. They are designed to give additional features to free/libre software games which aren't necessary, but are nice to have.

xSGE extensions are not dependent on any particular SGE implementation. They should work with any implementation that follows the specification.

This extension provides an easy-to-use framework for collision physics. This is especially useful for platformers, though it can also be useful for other types of games.

Note: This collision system supports plain rectangle-based collision detection ONLY. Attempting to use precise or ellipse collision detection will not work as you expect, and may often not even work at all. This is because implementing support for such mask-based collision detection would be guaranteed to be slow, unreliable, and needlessly complicated; it's not worth it.

If you must use precise collision detection, consider doing so with a separate object that follows the appropriate *Collider* object.

XSGE_PHYSICS CLASSES

xsgge_physics.Collider

```
class xsgge_physics.Collider(x, y, z=0, sprite=None, visible=True, active=True,
                             checks_collisions=True, tangible=True, bbox_x=None, bbox_y=None,
                             bbox_width=None, bbox_height=None, regulate_origin=False, collision_ellipse=False, collision_precise=False, xvelocity=0, yvelocity=0,
                             xacceleration=0, yacceleration=0, xdeceleration=0, ydeceleration=0,
                             image_index=0, image_origin_x=None, image_origin_y=None, image_fps=None, image_xscale=1, image_yscale=1, image_rotation=0,
                             image_alpha=255, image_blend=None, image_blend_mode=None)
```

Class for objects which have physics interactions.

Note: This class depends on use of `Collider.move_x()` and `Collider.move_y()` to handle physics interactions. `event_update_position()` uses these methods, so speed attributes will work properly, but changing `x` and `y` manually will not cause any physics to occur.

nonstick_left

nonstick_right

nonstick_top

nonstick_bottom

These attributes are used by certain wall classes to exclude the object from “sticking” to the wall. See the documentation for the wall classes for more information.

Default value: `False`

xsgge_physics.Collider Methods

`Collider.move_x(move, absolute=False, do_events=True, exclude_events=())`

Move the object horizontally, handling physics.

Arguments:

- `move` – The amount to add to `x`.
- `absolute` – If set to `True`, the distance moved horizontally is absolute, i.e. will not be reduced as a result of vertical movement caused by slopes. Otherwise, any vertical movement caused by slopes will result in a reduction of horizontal movement.
- `do_events` – Whether or not physics collision events should be executed when appropriate.

- `exclude_events` – A set, list, or tuple of wall objects which should not cause collision events to be executed if collided with.

`Collider.move_y` (*move*, *absolute=False*, *do_events=True*, *exclude_events=()*)

Move the object vertically, handling physics.

Arguments:

- `move` – The amount to add to `y`.
- `absolute` – If set to `True`, the distance moved vertically is absolute, i.e. will not be reduced as a result of horizontal movement caused by slopes. Otherwise, any horizontal movement caused by slopes will result in a reduction of vertical movement.
- `do_events` – Whether or not physics collision events should be executed when appropriate.
- `exclude_events` – A set, list, or tuple of wall objects which should not cause collision events to be executed if collided with.

`Collider.get_left_touching_wall` ()

Return a list of `SolidRight` objects whose right sides are touching the left side of this object.

`Collider.get_right_touching_wall` ()

Return a list of `SolidLeft` objects whose left sides are touching the right side of this object.

`Collider.get_top_touching_wall` ()

Return a list of `SolidTop` objects whose top sides are touching the bottom side of this object.

`Collider.get_bottom_touching_wall` ()

Return a list of `SolidBottom` objects whose bottom sides are touching the top side of this object.

`Collider.get_left_touching_slope` ()

Return a list of `SlopeTopRight` and `SlopeBottomRight` objects whose right sides are touching the left side of this object.

`Collider.get_right_touching_slope` ()

Return a list of `SlopeTopLeft` and `SlopeBottomLeft` objects whose left sides are touching the right side of this object.

`Collider.get_top_touching_slope` ()

Return a list of `SlopeBottomLeft` and `SlopeBottomRight` objects whose bottom sides are touching the top side of this object.

`Collider.get_bottom_touching_slope` ()

Return a list of `SlopeTopLeft` and `SlopeTopRight` objects whose top sides are touching the bottom side of this object.

xsg physics.Collider Event Methods

`Collider.event_physics_collision_left` (*other*, *move_loss*)

Called when the left side of the collider collides with a wall or slope in the sense of the physics system, rather than in the sense of SGE collision detection.

Arguments:

- `move_loss` – The amount of movement that was prevented by the collision in pixels. For example, if the object would have moved 6 pixels, but only moved 2 pixels as a result of this collision, this value will be 4. This can be used to undo such a reduction in movement.

See the documentation for `sge.dsp.Object.event_collision` () for more information.

`Collider.event_physics_collision_right` (*other, move_loss*)

Called when the right side of the collider collides with a wall or slope in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `xsgc_physics.Collider.event_physics_collision_left()` for more information.

`Collider.event_physics_collision_top` (*other, move_loss*)

Called when the top side of the collider collides with a wall or slope in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `xsgc_physics.Collider.event_physics_collision_left()` for more information.

`Collider.event_physics_collision_bottom` (*other, move_loss*)

Called when the bottom side of the collider collides with a wall or slope in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `xsgc_physics.Collider.event_physics_collision_left()` for more information.

xsgc_physics.Wall

```
class xsgc_physics.Wall (x, y, z=0, sprite=None, visible=True, active=True, checks_collisions=True,
    tangible=True, bbox_x=None, bbox_y=None, bbox_width=None,
    bbox_height=None, regulate_origin=False, collision_ellipse=False,
    collision_precise=False, xvelocity=0, yvelocity=0, xacceleration=0,
    yacceleration=0, xdeceleration=0, ydeceleration=0, image_index=0,
    image_origin_x=None, image_origin_y=None, image_fps=None, im-
    age_xscale=1, image_yscale=1, image_rotation=0, image_alpha=255,
    image_blend=None, image_blend_mode=None)
```

Base class for all wall objects that `Collider` objects interact with in some way. It is functionally identical to its parent class, `sge.dsp.Object`.

xsgc_physics.SolidLeft

```
class xsgc_physics.SolidLeft (x, y, z=0, sprite=None, visible=True, active=True,
    checks_collisions=True, tangible=True, bbox_x=None, bbox_y=None,
    bbox_width=None, bbox_height=None, regulate_origin=False, colli-
    sion_ellipse=False, collision_precise=False, xvelocity=0, yvelocity=0,
    xacceleration=0, yacceleration=0, xdeceleration=0, ydeceleration=0,
    image_index=0, image_origin_x=None, image_origin_y=None, im-
    age_fps=None, image_xscale=1, image_yscale=1, image_rotation=0,
    image_alpha=255, image_blend=None, image_blend_mode=None)
```

Class for walls which stop movement of `Collider` objects from the top.

xsgc_physics.SolidLeft Event Methods

`SolidLeft.event_physics_collision_left` (*other, move_loss*)

Called when the left side of the wall collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgc_physics.SolidRight

```
class xsgc_physics.SolidRight(x, y, z=0, sprite=None, visible=True, active=True,
                              checks_collisions=True, tangible=True, bbox_x=None,
                              bbox_y=None, bbox_width=None, bbox_height=None, regulate_origin=False,
                              collision_ellipse=False, collision_precise=False, xvelocity=0,
                              yvelocity=0, xacceleration=0, yacceleration=0, xdeceleration=0,
                              ydeceleration=0, image_index=0, image_origin_x=None,
                              image_origin_y=None, image_fps=None, image_xscale=1,
                              image_yscale=1, image_rotation=0, image_alpha=255,
                              image_blend=None, image_blend_mode=None)
```

Class for walls which stop movement of *Collider* objects from the right.

xsgc_physics.SolidRight Event Methods

SolidRight.**event_physics_collision_right** (*other*, *move_loss*)

Called when the right side of the wall collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgc_physics.SolidTop

```
class xsgc_physics.SolidTop(x, y, z=0, sprite=None, visible=True, active=True,
                             checks_collisions=True, tangible=True, bbox_x=None, bbox_y=None,
                             bbox_width=None, bbox_height=None, regulate_origin=False,
                             collision_ellipse=False, collision_precise=False, xvelocity=0,
                             yvelocity=0, xacceleration=0, yacceleration=0, xdeceleration=0,
                             ydeceleration=0, image_index=0, image_origin_x=None,
                             image_origin_y=None, image_fps=None, image_xscale=1,
                             image_yscale=1, image_rotation=0,
                             image_alpha=255, image_blend=None, image_blend_mode=None)
```

Class for walls which stop movement of *Collider* objects from the top.

xsgc_physics.SolidTop Event Methods

SolidTop.**event_physics_collision_top** (*other*, *move_loss*)

Called when the top side of the wall collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgc_physics.SolidBottom

```
class xsgc_physics.SolidBottom(x, y, z=0, sprite=None, visible=True, active=True,
    checks_collisions=True, tangible=True, bbox_x=None,
    bbox_y=None, bbox_width=None, bbox_height=None,
    regulate_origin=False, collision_ellipse=False, collision_precise=False,
    xvelocity=0, yvelocity=0, xacceleration=0, yacceleration=0, xdeceleration=0,
    ydeceleration=0, image_index=0, image_origin_x=None, image_origin_y=None,
    image_fps=None, image_xscale=1, image_yscale=1, image_rotation=0,
    image_alpha=255, image_blend=None, image_blend_mode=None)
```

Class for walls which stop movement of *Collider* objects from the bottom.

xsgc_physics.SolidBottom Event Methods

```
SolidBottom.event_physics_collision_bottom(other, move_loss)
```

Called when the bottom side of the wall collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgc_physics.Solid

```
class xsgc_physics.Solid(x, y, z=0, sprite=None, visible=True, active=True, checks_collisions=True,
    tangible=True, bbox_x=None, bbox_y=None, bbox_width=None,
    bbox_height=None, regulate_origin=False, collision_ellipse=False, collision_precise=False,
    xvelocity=0, yvelocity=0, xacceleration=0, yacceleration=0, xdeceleration=0,
    ydeceleration=0, image_index=0, image_origin_x=None, image_origin_y=None, image_fps=None,
    image_xscale=1, image_yscale=1, image_rotation=0, image_alpha=255,
    image_blend=None, image_blend_mode=None)
```

Inherits *SolidLeft*, *SolidRight*, *SolidTop*, and *SolidBottom*. Meant to be a convenient parent class for walls that should stop movement in all directions.

xsgc_physics.Solid Event Methods

```
Solid.event_physics_collision_left(other, move_loss)
```

Called when the left side of the wall collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

```
Solid.event_physics_collision_right(other, move_loss)
```

Called when the right side of the wall collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

```
Solid.event_physics_collision_top(other, move_loss)
```

Called when the top side of the wall collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

`Solid.event_physics_collision_bottom` (*other, move_loss*)

Called when the bottom side of the wall collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgc_physics.Slope

```
class xsgc_physics.Slope(x, y, z=0, sprite=None, visible=True, active=True, checks_collisions=True,
    tangible=True, bbox_x=None, bbox_y=None, bbox_width=None,
    bbox_height=None, regulate_origin=False, collision_ellipse=False,
    collision_precise=False, xvelocity=0, yvelocity=0, xacceleration=0,
    yacceleration=0, xdeceleration=0, ydeceleration=0, image_index=0,
    image_origin_x=None, image_origin_y=None, image_fps=None, im-
    age_xscale=1, image_yscale=1, image_rotation=0, image_alpha=255,
    image_blend=None, image_blend_mode=None)
```

Base class for all slopes. It is functionally identical to its parent class, `Wall`.

xsgc_physics.SlopeTopLeft

```
class xsgc_physics.SlopeTopLeft(x, y, z=0, sprite=None, visible=True, active=True,
    checks_collisions=True, tangible=True, bbox_x=None,
    bbox_y=None, bbox_width=None, bbox_height=None,
    regulate_origin=False, collision_ellipse=False, colli-
    sion_precise=False, xvelocity=0, yvelocity=0, xaccelera-
    tion=0, yacceleration=0, xdeceleration=0, ydeceleration=0,
    image_index=0, image_origin_x=None, image_origin_y=None,
    image_fps=None, image_xscale=1, image_yscale=1, im-
    age_rotation=0, image_alpha=255, image_blend=None, im-
    age_blend_mode=None)
```

A parent class for slopes which point in some direction upwards and to the left.

Slopes of this type go from the bottom-left corner to the top-right corner of the bounding box.

xsticky_top

If set to `True`, a collider that moves to the left while touching the top side of the slope will attempt to keep touching the top side of the slope by moving downward, unless the collider's `nonstick_bottom` value is `True`.

Default value: `False`

ysticky_left

If set to `True`, a collider that moves upward while touching the left side of the slope will attempt to keep touching the left side of the slope by moving to the right, unless the collider's `nonstick_right` value is `True`.

Default value: `False`

xsgc_physics.SlopeTopLeft Event Methods

`SlopeTopLeft.event_physics_collision_left` (*other, move_loss*)

Called when the left side of the slope collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

`SlopeTopLeft.event_physics_collision_top` (*other, move_loss*)

Called when the top side of the slope collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgc_physics.SlopeTopRight

```
class xsgc_physics.SlopeTopRight(x, y, z=0, sprite=None, visible=True, active=True,
    checks_collisions=True, tangible=True, bbox_x=None,
    bbox_y=None, bbox_width=None, bbox_height=None,
    regulate_origin=False, collision_ellipse=False, collision_precise=False,
    xvelocity=0, yvelocity=0, xacceleration=0, yacceleration=0, xdeceleration=0,
    ydeceleration=0, image_index=0, image_origin_x=None, image_origin_y=None,
    image_fps=None, image_xscale=1, image_yscale=1, image_rotation=0,
    image_alpha=255, image_blend=None, image_blend_mode=None)
```

A parent class for slopes which point in some direction upwards and to the right.

Slopes of this type go from the top-left corner to the bottom-right corner of the bounding box.

xsticky_top

If set to `True`, a collider that moves to the right while touching the top side of the slope will attempt to keep touching the top side of the slope by moving downward, unless the collider's `nonstick_bottom` value is `True`.

Default value: `False`

ysticky_right

If set to `True`, a collider that moves upward while touching the right side of the slope will attempt to keep touching the right side of the slope by moving to the left, unless the collider's `nonstick_left` value is `True`.

Default value: `False`

xsgc_physics.SlopeTopRight Event Methods

`SlopeTopRight.event_physics_collision_right` (*other, move_loss*)

Called when the right side of the slope collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

`SlopeTopRight.event_physics_collision_top` (*other, move_loss*)

Called when the top side of the slope collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgc_physics.SlopeBottomLeft

```
class xsgc_physics.SlopeBottomLeft(x, y, z=0, sprite=None, visible=True, active=True,
    checks_collisions=True, tangible=True, bbox_x=None,
    bbox_y=None, bbox_width=None, bbox_height=None,
    regulate_origin=False, collision_ellipse=False, collision_precise=False,
    xvelocity=0, yvelocity=0, xacceleration=0, yacceleration=0,
    xdeceleration=0, ydeceleration=0, image_index=0, image_origin_x=None,
    image_origin_y=None, image_fps=None, image_xscale=1, image_yscale=1,
    image_rotation=0, image_alpha=255, image_blend=None, image_blend_mode=None)
```

A parent class for slopes which point in some direction upwards and to the left.

Slopes of this type go from the top-left corner to the bottom-right corner of the bounding box.

xsticky_bottom

If set to `True`, a collider that moves to the left while touching the bottom side of the slope will attempt to keep touching the bottom side of the slope by moving upward, unless the collider's `nonstick_top` value is `True`.

Default value: `False`

ysticky_left

If set to `True`, a collider that moves downward while touching the left side of the slope will attempt to keep touching the left side of the slope by moving to the right, unless the collider's `nonstick_right` value is `True`.

Default value: `False`

xsgc_physics.SlopeBottomLeft Event Methods

`SlopeBottomLeft.event_physics_collision_left` (*other, move_loss*)

Called when the left side of the slope collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

`SlopeBottomLeft.event_physics_collision_bottom` (*other, move_loss*)

Called when the bottom side of the slope collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgge_physics.SlopeBottomRight

```
class xsgge_physics.SlopeBottomRight (x, y, z=0, sprite=None, visible=True, active=True,
    checks_collisions=True, tangible=True, bbox_x=None,
    bbox_y=None, bbox_width=None, bbox_height=None,
    regulate_origin=False, collision_ellipse=False, col-
    lision_precise=False, xvelocity=0, yvelocity=0, xac-
    celeration=0, yacceleration=0, xdeceleration=0, yde-
    celeration=0, image_index=0, image_origin_x=None,
    image_origin_y=None, image_fps=None, image_xscale=1,
    image_yscale=1, image_rotation=0, image_alpha=255,
    image_blend=None, image_blend_mode=None)
```

A parent class for slopes which point in some direction upwards and to the right.

Slopes of this type go from the bottom-left corner to the top-right corner of the bounding box.

xsticky_bottom

If set to `True`, a collider that moves to the right while touching the bottom side of the slope will attempt to keep touching the bottom side of the slope by moving upward, unless the collider's `nonstick_top` value is `True`.

Default value: `False`

ysticky_right

If set to `True`, a collider that moves downward while touching the right side of the slope will attempt to keep touching the right side of the slope by moving to the right, unless the collider's `nonstick_left` value is `True`.

Default value: `False`

xsgge_physics.SlopeBottomRight Event Methods

`SlopeBottomRight.event_physics_collision_right` (*other, move_loss*)

Called when the right side of the slope collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

`SlopeBottomRight.event_physics_collision_bottom` (*other, move_loss*)

Called when the bottom side of the slope collides with a collider in the sense of the physics system, rather than in the sense of SGE collision detection. See the documentation for `sge.dsp.Object.event_collision()` for more information.

xsgge_physics.MobileWall

```
class xsgge_physics.MobileWall (x, y, z=0, sprite=None, visible=True, active=True,
    checks_collisions=True, tangible=True, bbox_x=None,
    bbox_y=None, bbox_width=None, bbox_height=None, regu-
    late_origin=False, collision_ellipse=False, collision_precise=False,
    xvelocity=0, yvelocity=0, xacceleration=0, yacceleration=0, xdecel-
    eration=0, ydeceleration=0, image_index=0, image_origin_x=None,
    image_origin_y=None, image_fps=None, image_xscale=1,
    image_yscale=1, image_rotation=0, image_alpha=255, im-
    age_blend=None, image_blend_mode=None)
```

A parent class for walls and slopes that can move. When an object of this class moves, it “pushes” any appro-

priate colliders, as a real wall might be expected to do.

Note: For classes derived from this class to be useful, they also need to inherit one or more of the other wall classes. Objects of this class that are not also objects of other classes will naturally not collide.

Note: This class depends on use of *MobileWall.move_x()* and *MobileWall.move_y()* to handle physics interactions. *event_update_position()* uses these methods, so speed attributes will work properly, but changing *x* and *y* manually will not cause any physics to occur.

push_left

If set to `True`, the wall will push any colliders collided with as a result of the wall's movement to the left. Otherwise, the wall will pass through such colliders.

Default value: `True`

push_right

If set to `True`, the wall will push any colliders collided with as a result of the wall's movement to the right. Otherwise, the wall will pass through such colliders.

Default value: `True`

push_up

If set to `True`, the wall will push any colliders collided with as a result of the wall's movement upwards. Otherwise, the wall will pass through such colliders.

Default value: `True`

push_down

If set to `True`, the wall will push any colliders collided with as a result of the wall's movement downwards. Otherwise, the wall will pass through such colliders.

Default value: `True`

sticky_left

If set to `True`, any colliders touching the left side of the wall will move along with it, regardless of the direction of movement, with the exception of colliders whose `nonstick_right` values are `True`.

Default value: `False`

sticky_right

If set to `True`, any colliders touching the right side of the wall will move along with it, regardless of the direction of movement, with the exception of colliders whose `nonstick_left` values are `True`.

Default value: `False`

sticky_top

If set to `True`, any colliders touching the top side of the wall will move along with it, regardless of the direction of movement, with the exception of colliders whose `nonstick_bottom` values are `True`.

Default value: `False`

sticky_bottom

If set to `True`, any colliders touching the bottom side of the wall will move along with it, regardless of the direction of movement, with the exception of colliders whose `nonstick_top` values are `True`.

Default value: `False`

xsge_physics.MobileWall methods

MobileWall.**get_stuck_colliders**()

Return a list of *Collider* objects which are “stuck” to this wall (i.e. will move along with the wall regardless of direction).

MobileWall.**move_x**(*move*)

Move the wall horizontally, handling physics.

Arguments:

- move* – The amount to add to *x*.

MobileWall.**move_y**(*move*)

Move the wall vertically, handling physics.

Arguments:

- move* – The amount to add to *y*.

xsge_physics.MobileColliderWall

```
class xsge_physics.MobileColliderWall(x, y, z=0, sprite=None, visible=True, active=True,
    checks_collisions=True, tangible=True, bbox_x=None,
    bbox_y=None, bbox_width=None, bbox_height=None,
    regulate_origin=False, collision_ellipse=False, collision_precise=False,
    xvelocity=0, yvelocity=0, xacceleration=0, yacceleration=0,
    xdeceleration=0, ydeceleration=0, image_index=0, image_origin_x=None,
    image_origin_y=None, image_fps=None, image_xscale=1,
    image_yscale=1, image_rotation=0, image_alpha=255,
    image_blend=None, image_blend_mode=None)
```

A parent class for mobile walls that are also colliders. See the documentation for *xsge_physics.Collider* and *xsge_physics.MobileWall* for more information.

Note: Due to the way movement is implemented in this class, it is not safe to move this object during its collision events. Doing so may cause colliders that should be stuck to this object to become un-stuck, or cause this object to pass through colliders it should be pushing.

X

xsgc_physics, 1

C

Collider (class in `xsge_physics`), 3

E

`event_physics_collision_bottom()` (`xsge_physics.Collider` method), 5

`event_physics_collision_bottom()` (`xsge_physics.SlopeBottomLeft` method), 10

`event_physics_collision_bottom()` (`xsge_physics.SlopeBottomRight` method), 11

`event_physics_collision_bottom()` (`xsge_physics.Solid` method), 7

`event_physics_collision_bottom()` (`xsge_physics.SolidBottom` method), 7

`event_physics_collision_left()` (`xsge_physics.Collider` method), 4

`event_physics_collision_left()` (`xsge_physics.SlopeBottomLeft` method), 10

`event_physics_collision_left()` (`xsge_physics.SlopeTopLeft` method), 8

`event_physics_collision_left()` (`xsge_physics.Solid` method), 7

`event_physics_collision_left()` (`xsge_physics.SolidLeft` method), 5

`event_physics_collision_right()` (`xsge_physics.Collider` method), 4

`event_physics_collision_right()` (`xsge_physics.SlopeBottomRight` method), 11

`event_physics_collision_right()` (`xsge_physics.SlopeTopRight` method), 9

`event_physics_collision_right()` (`xsge_physics.Solid` method), 7

`event_physics_collision_right()` (`xsge_physics.SolidRight` method), 6

`event_physics_collision_top()` (`xsge_physics.Collider` method), 5

`event_physics_collision_top()` (`xsge_physics.SlopeTopLeft` method), 8

`event_physics_collision_top()` (`xsge_physics.SlopeTopRight` method), 9

`event_physics_collision_top()` (`xsge_physics.Solid` method), 7

`event_physics_collision_top()` (`xsge_physics.SolidTop` method), 6

G

`get_bottom_touching_slope()` (`xsge_physics.Collider` method), 4

`get_bottom_touching_wall()` (`xsge_physics.Collider` method), 4

`get_left_touching_slope()` (`xsge_physics.Collider` method), 4

`get_left_touching_wall()` (`xsge_physics.Collider` method), 4

`get_right_touching_slope()` (`xsge_physics.Collider` method), 4

`get_right_touching_wall()` (`xsge_physics.Collider` method), 4

`get_stuck_colliders()` (`xsge_physics.MobileWall` method), 13

`get_top_touching_slope()` (`xsge_physics.Collider` method), 4

`get_top_touching_wall()` (`xsge_physics.Collider` method), 4

M

MobileColliderWall (class in `xsge_physics`), 13

MobileWall (class in `xsge_physics`), 11

`move_x()` (`xsge_physics.Collider` method), 3

`move_x()` (`xsge_physics.MobileWall` method), 13

`move_y()` (`xsge_physics.Collider` method), 4

`move_y()` (`xsge_physics.MobileWall` method), 13

N

`nonstick_bottom` (`xsge_physics.Collider` attribute), 3

`nonstick_left` (`xsge_physics.Collider` attribute), 3

`nonstick_right` (`xsge_physics.Collider` attribute), 3

`nonstick_top` (`xsge_physics.Collider` attribute), 3

P

`push_down` (`xsge_physics.MobileWall` attribute), 12

push_left (xsge_physics.MobileWall attribute), 12
push_right (xsge_physics.MobileWall attribute), 12
push_up (xsge_physics.MobileWall attribute), 12

S

Slope (class in xsge_physics), 8
SlopeBottomLeft (class in xsge_physics), 10
SlopeBottomRight (class in xsge_physics), 11
SlopeTopLeft (class in xsge_physics), 8
SlopeTopRight (class in xsge_physics), 9
Solid (class in xsge_physics), 7
SolidBottom (class in xsge_physics), 7
SolidLeft (class in xsge_physics), 5
SolidRight (class in xsge_physics), 6
SolidTop (class in xsge_physics), 6
sticky_bottom (xsge_physics.MobileWall attribute), 12
sticky_left (xsge_physics.MobileWall attribute), 12
sticky_right (xsge_physics.MobileWall attribute), 12
sticky_top (xsge_physics.MobileWall attribute), 12

W

Wall (class in xsge_physics), 5

X

xsge_physics (module), 1
xsticky_bottom (xsge_physics.SlopeBottomLeft attribute), 10
xsticky_bottom (xsge_physics.SlopeBottomRight attribute), 11
xsticky_top (xsge_physics.SlopeTopLeft attribute), 8
xsticky_top (xsge_physics.SlopeTopRight attribute), 9

Y

ysticky_left (xsge_physics.SlopeBottomLeft attribute), 10
ysticky_left (xsge_physics.SlopeTopLeft attribute), 8
ysticky_right (xsge_physics.SlopeBottomRight attribute), 11
ysticky_right (xsge_physics.SlopeTopRight attribute), 9