

---

# **xSGE Path Documentation**

***Release 1.0.1***

**onpon4**

April 10, 2017



## CONTENTS

<b>1</b>	<b>xsge_path Classes</b>	<b>3</b>
1.1	xsge_path.Path . . . . .	3
1.2	xsge_path.PathLink . . . . .	4
	<b>Python Module Index</b>	<b>5</b>
	<b>Index</b>	<b>7</b>



**Contents**

- *xSGE Path*
  - *xsge\_path Classes*
    - \* *xsge\_path.Path*
      - *xsge\_path.Path Methods*
      - *xsge\_path.Path Event Methods*
    - \* *xsge\_path.PathLink*
      - *xsge\_path.PathLink Methods*

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 paths for the SGE. Paths are used to make objects move in a certain way.



## XSGE\_PATH CLASSES

### xsge\_path.Path

**class** xsge\_path.**Path** (*x, y, points=(), z=0, visible=False, tangible=False, \*\*kwargs*)

Class for paths: objects which define movement patterns for other objects. Paths are defined as a series of points for an object to follow.

This class is derived from `sge.dsp.Object` and inherits all of that class's attributes and methods.

---

**Note:** `event_step()` is used to implement path-following behavior. Keep this in mind if you derive a class from this one.

---

#### points

A list of the points that make up the path relative to the position of the path in the room, excluding the first point. Each point should be a tuple in the form (*x, y*), where *x* is the horizontal location and *y* is the vertical location. The first point is always (0, 0), which is why it is not included in this list.

### xsge\_path.Path Methods

**Path.\_\_init\_\_** (*x, y, points=(), z=0, visible=False, tangible=False, \*\*kwargs*)

Arguments set the respective initial attributes of the object. See the documentation for [Path](#) for more information.

*x, y, z, visible, tangible*, and all arguments passed to *kwargs* are passed as the corresponding arguments to the constructor method of the parent class.

**Path.follow\_start** (*obj, speed, accel=None, decel=None, loop=0*)

Cause *obj* to start following this path at the speed indicated by *speed*.

While path objects have a location within the room, this location has no bearing on how objects following the path move. Movement is determined relative to the location of the object following the path when it started, not the location of the path.

By default, the object follows the path at a constant speed. If *accel* or *decel* is set to a value other than *None*, the object will instead accelerate or decelerate, respectively, by that amount each frame on each segment of the path.

*loop* indicates the number of times the object should follow the path after it does so the first time. For example, if set to 2, the object will follow the path a total of 3 times. Set to *None* to loop indefinitely.

**Path.follow\_stop** (*obj*)

Cause *obj* to stop following this path.

## xsge\_path.Path Event Methods

`Path.event_follow_end(obj)`

Called when an object, indicated by `obj`, finishes following the path.

## xsge\_path.PathLink

```
class xsge_path.PathLink(x, y, points=(), next_path=None, next_speed=None, next_accel=None,
                        next_decel=None, next_loop=0, z=0, visible=False, tangible=False,
                        **kwargs)
```

Class for path links. Path links are just like normal paths, but can be linked to other path links or paths to form chains.

By using a chain of path links, you can cause an object to move in different ways at different points of the path. For example, you can cause the object to change its speed, or you can cause it to accelerate and decelerate only at particular points.

---

**Note:** `event_follow_end()` is used to implement path linking. Keep this in mind if you derive a class from this one.

---

### **next\_path**

The next *xsge\_path.Path* object to be followed after this one. If set to `None`, no additional paths will be followed.

### **next\_speed**

The value to pass on to the `speed` argument of the next path's *xsge\_path.Path.follow\_start()* call. If set to `None`, the next path will not be followed.

### **next\_accel**

The value to pass on to the `accel` argument of the next path's *xsge\_path.Path.follow\_start()* call.

### **next\_decel**

The value to pass on to the `decel` argument of the next path's *xsge\_path.Path.follow\_start()* call.

### **next\_loop**

The value to pass on to the `loop` argument of the next path's *xsge\_path.Path.follow\_start()* call.

## xsge\_path.PathLink Methods

```
PathLink.__init__(x, y, points=(), next_path=None, next_speed=None, next_accel=None,
                 next_decel=None, next_loop=0, z=0, visible=False, tangible=False, **kwargs)
```

Arguments set the respective initial attributes of the object. See the documentation for *Path* for more information.

`x`, `y`, `points`, `z`, `visible`, `tangible`, and all arguments passed to `kwargs` are passed as the corresponding arguments to the constructor method of the parent class.



**X**

xsge\_path, 1



## Symbols

`__init__()` (`xsge_path.Path` method), 3

`__init__()` (`xsge_path.PathLink` method), 4

## E

`event_follow_end()` (`xsge_path.Path` method), 4

## F

`follow_start()` (`xsge_path.Path` method), 3

`follow_stop()` (`xsge_path.Path` method), 3

## N

`next_accel` (`xsge_path.PathLink` attribute), 4

`next_decel` (`xsge_path.PathLink` attribute), 4

`next_loop` (`xsge_path.PathLink` attribute), 4

`next_path` (`xsge_path.PathLink` attribute), 4

`next_speed` (`xsge_path.PathLink` attribute), 4

## P

`Path` (class in `xsge_path`), 3

`PathLink` (class in `xsge_path`), 4

`points` (`xsge_path.Path` attribute), 3

## X

`xsge_path` (module), 1