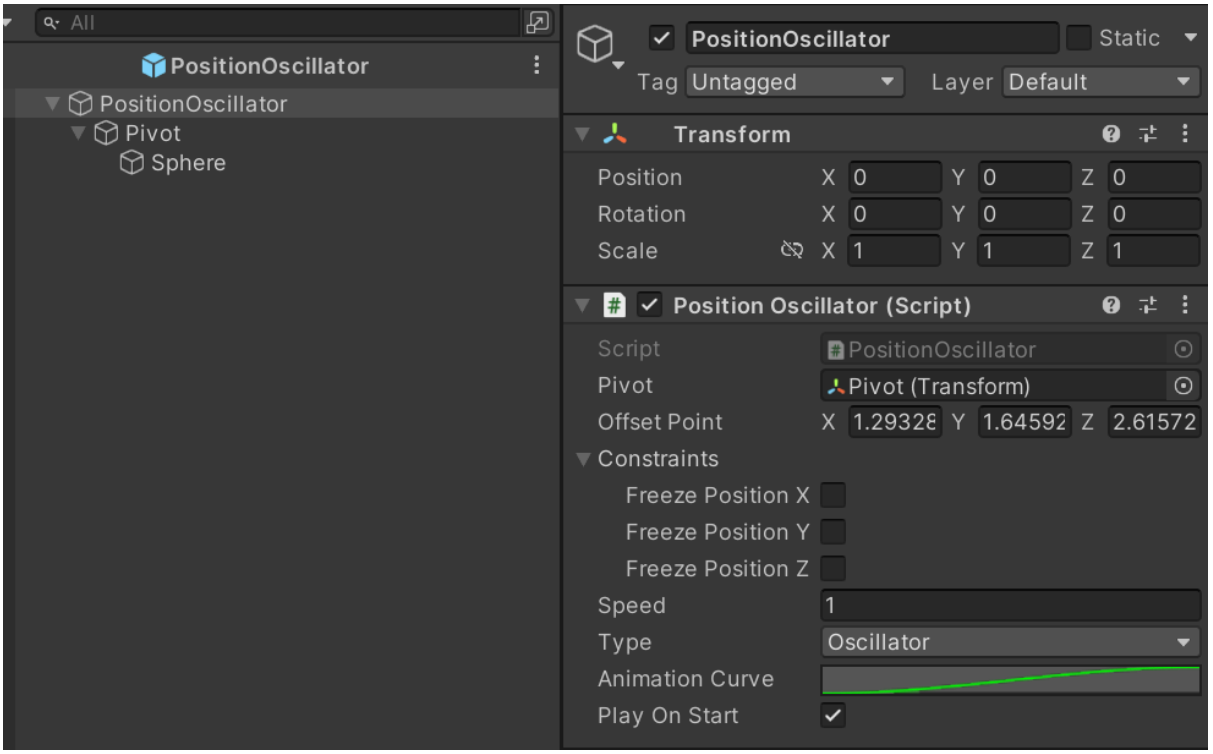
Transform Oscillator Documentation

Author: Cristian Montalvo
Git Repository: https://github.com/dismalito/unity-transform-oscillator
Portfolio: https://dismalito.github.io/

The main reason for these scripts is to have an easy way to make simple transform animation by code instead of having an animator to modify the oscillation position, scale or rotation of the transform. It provides a fast way to modify the current movement and the possibility to use animation curves to control the time and movement.

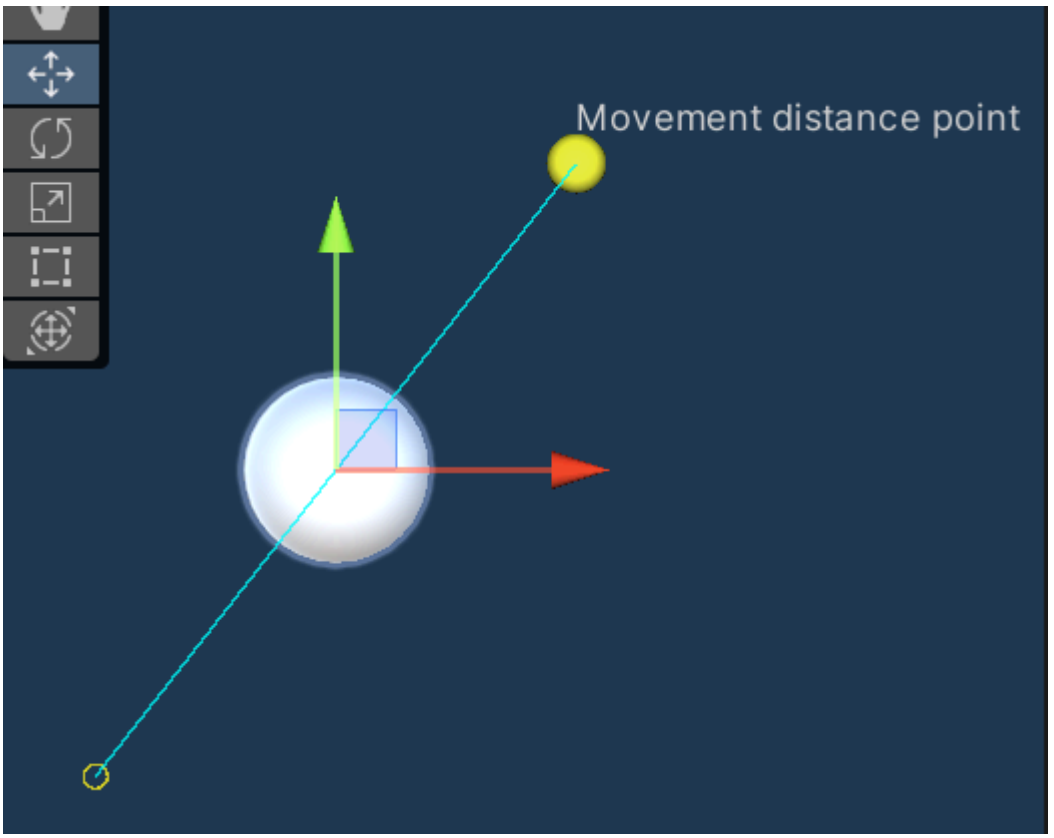
Position Oscillator



The way to use the position oscillator is to have the GameObject we want to move inside the Pivot. The script will move the Pivot according to the animation curve selected.

Script properties and options:

Pivot: reference to the transform child which will change its position by the script animation.
Offset Point: It is the target point for the movement. Take into account the movement will start from the center to the target point but it will oscillate to the opposite point which means the target point is the semi-period distance of the movement.



We can modify it manually or use a handler to move the offset point easily. As you can see, a line is drawn with the trajectory position animation from the opposite point to the offset point (target);

Constraints: to freeze the axis when we are using the handler to modify the offset point. Useful in 2D or when we only want to create an animation in an axis direction specifically.

Speed: to increase or decrease the velocity of the animation.
Type: the period type, we have two options:

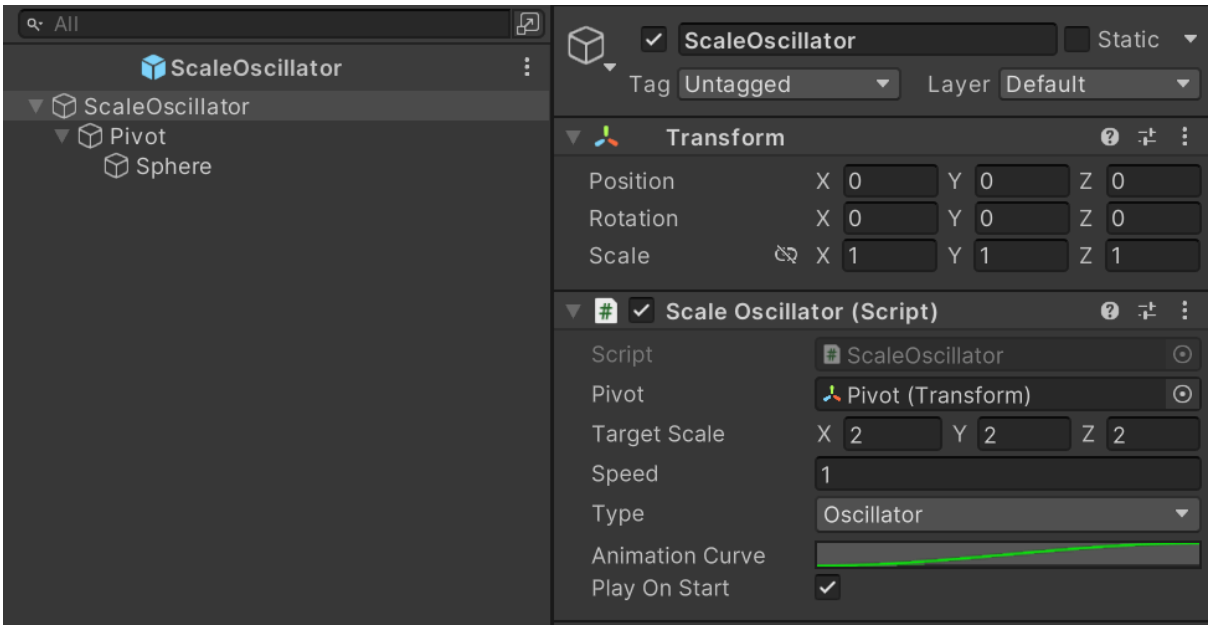
- Oscillator: it will use a triangular wave graph (pingpong), from 0 to 1 and it will return from 1 to 0, and repeat.
- Repeat: It will use a linear graph from 0 to 1 and repeat. It will create a gap between the last period frame to the new period frame.

Animation Curve: we can modify the animation movement using a simple or complex curve, this curve will evaluate following the period type (oscillator or repeat).
Play on Start: It will start the animation when the Monobehaviour.Start method is executed.

Public Methods

Play: Start the animation
Stop: Stop the animation

Scale Oscillator



The way to use the scale oscillator is to have the GameObject we want to scale as a child of Pivot. The script will scale the Pivot according to the animation curve selected.

Script properties and options:

Pivot: reference to the transform child which will change its scale by the script animation.
Target Scale: the animation will run rescaling from the original Pivot scale to the target scale.
Speed: to increase or decrease the velocity of the animation.
Type: the period type, we have two options:

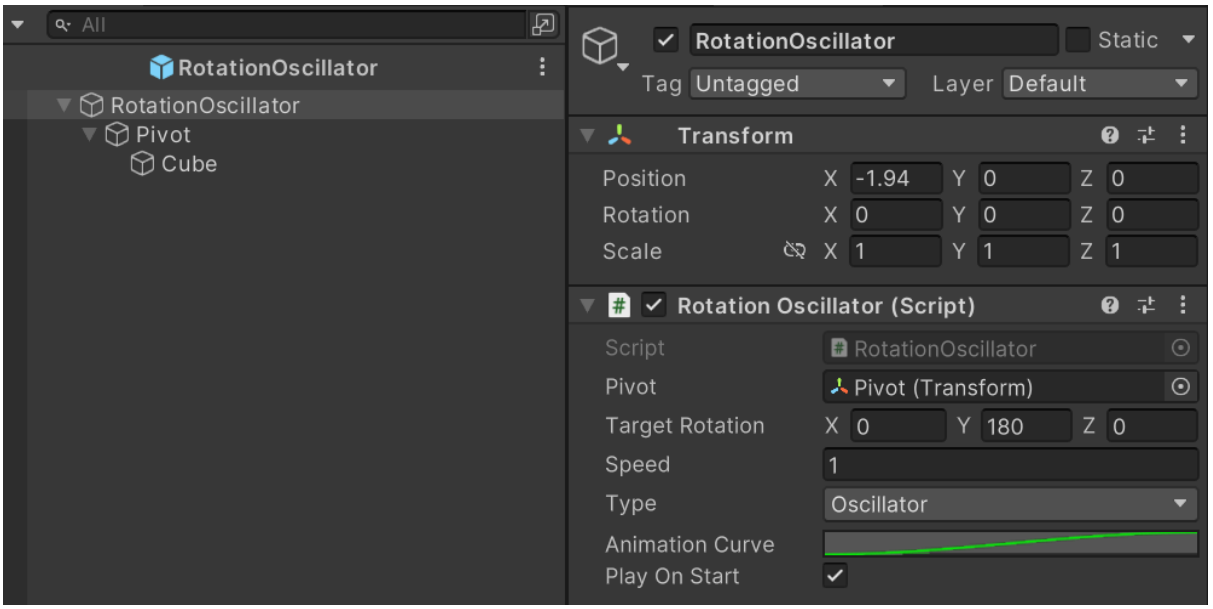
- Oscillator: it will use a triangular wave graph (pingpong), from 0 to 1 and it will return from 1 to 0, and repeat.
- Repeat: It will use a linear graph from 0 to 1 and repeat. It will create a gap between the last period frame to the new period frame.

Animation Curve: we can modify the animation scale using a simple or complex curve, this curve will evaluate following the period type (oscillator or repeat).
Play on Start: It will start the animation when the Monobehaviour.Start method is executed.

Public Methods

Play: Start the animation
Stop: Stop the animation

Rotation Oscillator



The way to use the rotation oscillator is to have the GameObject we want to scale as a child of Pivot. The script will rotate the Pivot according to the animation curve selected.

Script properties and options:

Pivot: reference to the transform child which will change its rotation by the script animation.
Target Rotation: the animation will run rotating from the original Pivot rotation to the target rotation.
Speed: to increase or decrease the velocity of the animation.
Type: the period type, we have two options:

- Oscillator: it will use a triangular wave graph (pingpong), from 0 to 1 and it will return from 1 to 0, and repeat.
- Repeat: It will use a linear graph from 0 to 1 and repeat. It will create a gap between the last period frame to the new period frame.

Animation Curve: we can modify the animation rotation using a simple or complex curve, this curve will evaluate following the period type (oscillator or repeat).
Play on Start: It will start the animation when the Monobehaviour.Start method is executed.

Public Methods

Play: Start the animation
Stop: Stop the animation