M

Plus

DOCUMENTATION

SUPPORT: otiose-entertainment@hotmail.com

IN EDITOR SIMULATIONS
Edit while Simulating
Dynamic Reaction to Changes
Start /Stop single Objects or all

EASY TO USE
No Scripting neccessary
Premade Prefabs
Documantation
Several Example Scenes

FULL EDITOR INTEGRATION
Custom Inspectors
Scene View Handles
Multi Object Editing
Gizmo Drawing

A LOT OF PARAMATERS
Customize Movements
Dynamic Reactions

6 BASIC MOVEMENT TYPES
Linear, Sine, Cicular, Bezier Curve
Follow Target, Perlin Noise

STACK ALL ANIMATIONS
Infinite Stacks - Infinite possibilities
Use parent position / rotation

VERSATILE BEZIER CURVE EDITOR
Closed and Open Curves
Scale and Rotate
Precise Handles

3 BASIC ROTATION TYPES
Linear, Look at Target, Bezier Nodes
chain to movement time

DIFFERENT LOOP TYPES
None, Bounce, Loop
Phase Shift
Percentage of Path

5 TWEEN TYPES
None, Linear, Sine
Smoothstep x2 and x3
Interval Option

CONTENTS

Thank you for using MovementPlus!
Please leave a comment on the Asset Store Page!

In case of problems you can get in touch via the
support Email: otiose-entertainment@hotmail.com

You may also request new Features!

1. PACKAGE STRUCTURE

→ **Documentation**

Congratulations! You already found that...

→ **Editor**

All the scripts for custom editors are stored here. Nothing you really need to care about...

→ **Examples**

This folder contains useful example scenes to get started! You can inspect all the components to learn how MovementPlus can be used.

→ GeneralExample

Shows all the basic movement types.

→ Many more...

→ **Materials**

Contains basic materials used in the example scenes.

→ **Prefabs**

Useful Prefabs ready to be dropped into you scene!

→ BezierCurve

A Bezier Curve Object with 2 points.

→ MovingObject

A cube with the MovementController script attached to it.

→ CameraFlight

Use this to create top notch camera flights. Smooth movement and rotation of the camera already set up.

→ **Resources**

Images used in the custom editors. Just leave it there.

→ **Scripts**

All scripts you can attach to GameObjects.

→ BezierCurve

Use the Prefab instead!

→ BezierPoint

Use the Prefab instead!

→ MovementController

One script for all movement and rotation types!

→ SpeedController

Combine this with the MovementController to tween between velocities of a moving object.

→ SpeedController

A bonus script to control the scene-view camera! Attach this to any GameObject and the camera uses its position and rotation. But beware, this may cause some glitches...

2. MOVEMENT CONTROLLER

2.1. GENERAL PARAMETERS

→ **Gizmo Color**

Color of all Gizmos drawn in the scene-view related to the MovementController.

→ **Use Position as Offset**

Should the animation play at the local space zero position or should the object position be used as an offset?

Note: This does not affect Follow Target and Bezier Curve types.

→ **Time Scale**

Individual time scale for this object. Change the animation speed / velocity using the time scale parameter. To easily tween the time scale you should use the SpeedController script.

→ **Animate Position / Animation Type**

Individual time scale for this object. Change the animation speed / velocity using the time scale parameter. To easily tween the time scale you should use the SpeedController script.

→ **Animate Rotation / Animation Type**

Individual time scale for this object. Change the animation speed / velocity using the time scale parameter.

2.2. LINEAR

→ **Position A and B**

Marking the end points of the linear movement.

→ **% of Path**

Only use a percentage of the path to move on.

→ **Loop Type**

How should the object loop after reaching the path end?

→ (None) Do not loop.

→ (Bounce) Revert Movement Direction.

→ (Loop) Begin again from the start point.

2.3. SINE

→ **Position A and B**

Marking the end points of the linear movement.

→ **Phase Shift**

Shift the starting point of the animation.

2.4. CIRCULAR

→ **Center**

The center point of the circle.

→ **Radius**

The Radius of the circle.

→ **Zenith Angle**

Change the rotation of the circle using the zenith and azimuth angles.

→ **Azimuth Angle**

Change the rotation of the circle using the zenith and azimuth angles.

→ **% of Path**

Only use a percentage of the path to move on.

→ **Loop Type**

How should the object loop after reaching the path end?

→ (None) Do not loop.

→ (Bounce) Revert Movement Direction.

→ (Loop) Begin again from the start point.

→ **Phase Shift**

Shift the starting point of the animation.

2.5. FOLLOW TARGET

→ **Target Transform**

Transform of the Object to follow.

→ **Hardness**

How delayed / damped is the follow movement?

Use 0 for no follow and 1 for instant follow.

→ **Velocity Approach**

Use a more customizable follow type system. (Suited for long distances)

→ **Maximum Velocity and Acceleration**

The maximum follow velocity and acceleration outside of the snap range.

→ **Snap Range and Snap Hardness**

The range at which the follow animation gets non linear and its hardness.

2.6. BEZIER CURVE

→ **Curve Object**

The Bezier Curve Object (you can use the Bezier Curve prefab to start)

→ **% of Path**

Only use a percentage of the path to move on.

→ Constant Velocity

Always maintain a constant velocity or slow down on shorter Path parts?

Note: Constant velocity is not implemented yet!

→ **Loop Type**

How should the object loop after reaching the path end?

→ (None) Do not loop.

→ (Bounce) Revert Movement Direction.

→ (Loop) Begin again from the start point.

→ **Phase Shift**

Shift the starting point of the animation.

2.7. PERLIN NOISE

→ **Amplitude**

3D Amplitude of the random movement. Defines the maximum range.

→ **Synchronized**

All objects that are synchronized will use the same noise region.

2.8. LINEAR ROTATION

→ Bound to Movement

Should the rotation use the same time and loop type as the movement?

Note: Unbound rotation is not implemented yet!

→ **Angles**

The degrees of all three axis the object will rotate.

2.9. LOOK AT TARGET

→ **Target Transform**

Transform of the Object to look towards.

→ **Hardness**

How delayed / damped is the rotation?

Use 0 for no follow and 1 for instant follow.

2.10. BEZIER NODES

You need to use Bezier curve as the movement animation type. The objects rotation will interpolate between the rotation of the curve nodes linearly. If you need a smooth rotation and want the object to look along the path use the CameraFlight prefab instead.

2.11. LOOK ALONG CURVE

You need to use Bezier curve as the movement animation type.

→ **Curve Prediction**

Distance to a point on the curve the object should look at.

→ **Hardness**

How delayed / damped is the rotation?
Use 0 for no follow and 1 for instant follow.

3. BEZIER CURVE

To set up bezier curves you should use the BezierCurve prefab.

→ **Curve Color**

Color of the curve in the scene-view.

→ **Resolution**

Draw quality in the scene-view. Does not affect the actual movement.

→ **Curve Length**

Displays the length of the bezier curve.

→ **Closed**

Is the curve a closed loop or does it have two end points?

→ **Precise Handles**

Draw Precise Handles in the editor. Use this in combination with Unity's rotation tool to rotate bezier nodes.

→ **Curve Points**

The list of all curve points. You can change the position of nodes or add and delete specific ones.

4. SPEED CONTROLLER

Combine this with the MovementController to tween between velocities of a moving object.

→ **Toggle (Button)**

Calls the toggle function, changes between the neutral and target speed using the defined tween type. Can be automated using the interval option.

→ **Neutral Speed**

Time scale of the MovementController in the neutral state.

→ **Target Speed**

Time scale of the MovementController in the target state.

→ **Tween Type**

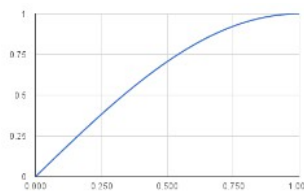Tween type to use when changing between neutral and target state.
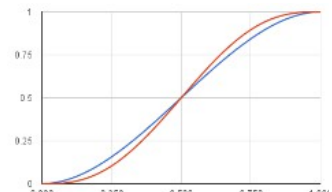
→ None
→ Linear
→ Sine
→ Smoothstep x2
→ Smoothstep x3

Sine

Smoothstep x2 and x3

→ **Tween Duration**

Duration of the tween in seconds. Not possible if Tween Type None.

→ **Interval**

Automatically timed toggles or manual toggles?

→ **Neutral Duration**

Duration in seconds to stay in the neutral state if interval option is true.

→ **Target Duration**

Duration in seconds to stay in the target state if interval option is true.

5. BONUS

5.1. PREFABS

Useful Prefabs ready to be dropped into you scene!

→ **BezierCurve**

A Bezier Curve Object with 2 points.

→ **MovingObject**

A cube with the MovementController script attached to it.

→ **CameraFlight**

Use this to create top notch camera flights. Smooth movement and rotation of the camera already set up.

The system uses two objects moving along a bezier curve. One object is the rotation target of the camera (second object). The objects have different shape shifts to control the points to look at along the path. You can also tweak the rotation hardness of the camera's MovementController.

5.2. SCENEVIEW CAMERA CONTROLLER

A bonus script to control the scene-view camera! Attach this to any GameObject and the camera uses its position and rotation. But beware, this may cause some glitches...

→ **Active**

If true, the scene-view camera cannot be controlled with user input. Instead it uses the objects position and rotation.

Note: You can still zoom in and out.

5.3. MOVEMENT PLUS EDITOR WINDOW

You can use the Editor Window to start and stop all of the Movement Plus objects in your scene at once. Simply open the window through `Tools → MovementPlus` and you are already set up.