Version 1.1

Easy Sprites Animation it's an animation system designed mostly to be used on games that require character animation or objects that have unique sprites and its animation changes sprites in a given time unit, for example retro games or action games.

The principal advantages of Easy Sprites Animation in comparison to Unity Animation System are the option of set an arbitrary time period between frames, instead of the need to define key-frames with fixed time distances, the ability to loop to a specific key-frame, the possibility to activate events from any frame in the editor and then easily receive them on script, and others.

Besides the previous features of Easy Sprites Animation, the main reason to use this system is that its easier to define and use the animations, and the advantages on editor and scripting level that are given compared to the basic system. The editor it's simple but complete, with drag and drop in all its functions, complete UNDO, and various functionalities to perfectly achieve the maximum of its potential.

The scripting part include events for all the important changes, and it's very easy to use.

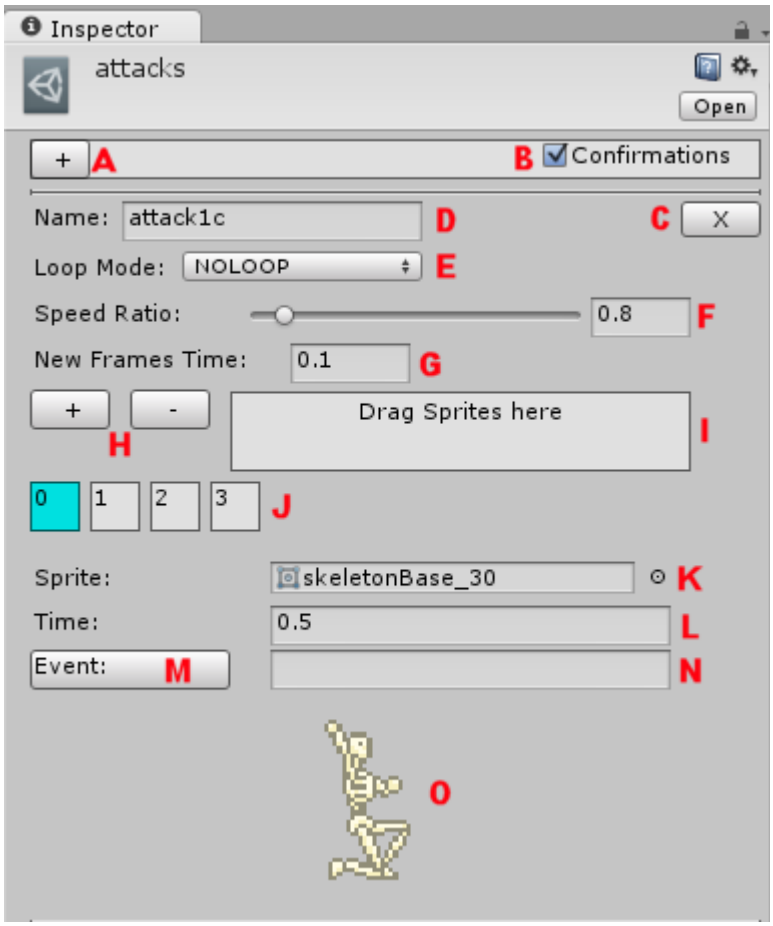
Tutorial

First Steps

Easy Sprites Animation uses 2 elements: SpriteAnimationAsset (ScriptableObject) and SpriteAnimation (Monobehaviour component).

The SpriteAnimationAsset object contains all the animation data and can be edited on the inspector.

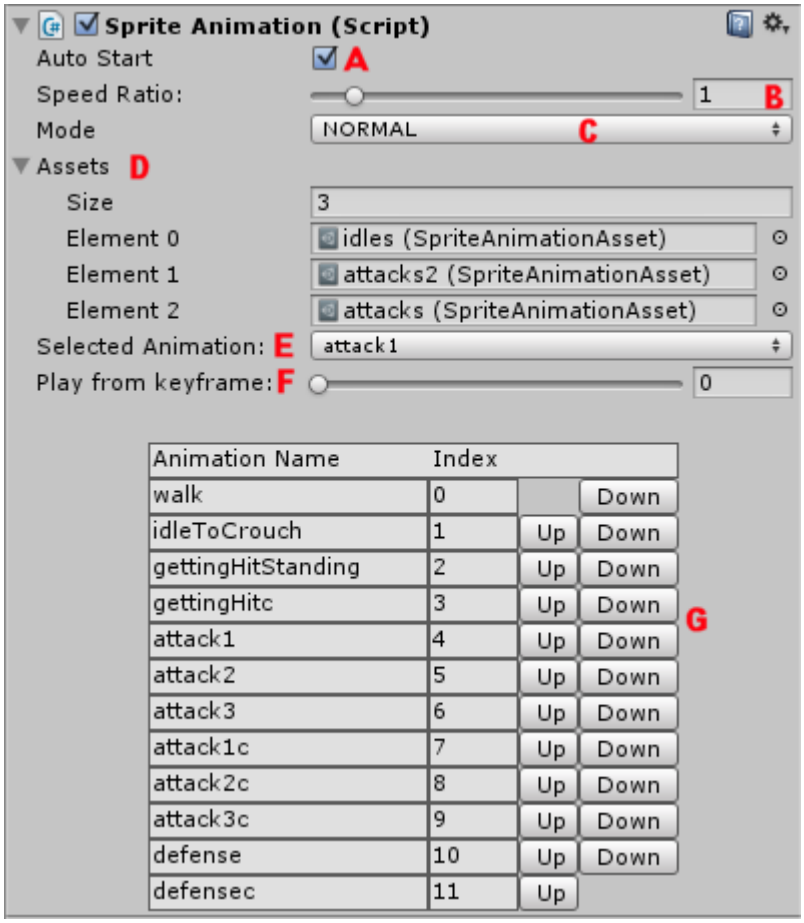
SpriteAnimationAsset Inspector items description:



A – Adds new animation.
B – Toggle confirmations.
C – Delete the animation.
D – Animation name.
F – Speed ratio for the animation. It multiplies the time scale of the game and the speed ratio of the animation component.
G – The time between frames value that the added new frames will have.
H – Adds and Removes key-frames
I – Drag zone to drag sprites from the project window.
J – Key-frames. Click them to change the values of K, L, N, M and O, reflecting the selected key-frame values. They can be interchanged using drag and drop.
K – Sprite for selected key-frame.
L – Time between selected key-frame and the next (or the end of the animation).
M – Event toggle. Activate it to make the animation send a **OnKeyFrameEvent**.
N – The string sended with the **OnKeyFrameEvent** if enabled.
O – Preview of the selected key-frame’s sprite.

The SpriteAnimation component has to be attached to the GameObject that has the animations. It requires a SpriteRenderer component.

SpriteAnimation Inspector items description:



A – Toggle autoStart for animations, affecting when the GameObject awake, then the animation changes from script or when it changes from the animations list.
B – Speed ratio for the component. It multiplies the time scale of the game and the speed ratio of the animation from the asset.
C – Toggles between modes Normal and TimeScale Independent.
D – Animation assets. You can drop different SpriteAnimationAssets in this list and it will update the animation list for the component with all the available animations.
F – Sets a start keyframe for the animation, overriding the asset behaviour. Useful to make Idle animations that look different in several characters in the screen at the same time.
G – The animation list, reorderable with the Up and Down buttons. Reordering the list updates the index of the selected animation if any.

First steps video tutorial

Scripting

SpriteAnimation component this defined events :

OnStartAnimation (SpriteAnimation, SpriteAnimationData)
Called when the animation starts (but not if it's looping).
OnAnimationStartLoop (SpriteAnimation, SpriteAnimationData)
Called when the animation loops (but not the first time).
OnFinishAnimation (SpriteAnimation, SpriteAnimationData)
Called when the animation finish.
OnStopAnimation (SpriteAnimation, SpriteAnimationData)
Called if the animation is stopped.
OnFinishOrStopAnimation (SpriteAnimation, SpriteAnimationData)
Called when the animation finish or is stopped.
OnKeyFrameEvent (SpriteAnimation, SpriteAnimationData, int, string)
Subscribe to this event to receive the custom keyframe events defined in the SpriteAnimationAsset inspector.

These are the public Methods:

SetCurrentAnimation(int idx) / SetCurrentAnimation (string animName)
Set the current animation as the idx or the name of the animation according to the component animations list. If the instance's autoStart setting is enabled, it will play the animation too.

Play (int idx) / Play (string animName) / Play ( )
Plays the animation assigned by SetCurrentAnimation or by the parameter.

Stop ( )
Stops the current animation if playing.

Scripting and Events video tutorial