AniSprite

Documentation
HTML Version

1. Interface

1. Scale size of sprite sheet preview.

2. Pop out preview to a dockable window.

3. Sprite preview

4. Toggle for animating on start.

5. Frame rate of animation.

6. Set play direction and looping.

7. Start frame of animation (blue highlight) and start and end frames of whole animation from sheet (red highlight).

8. List of all the frames added from sheet. White outline in preview (yellow highlight when frame selected).

9. Saved animations list.

2. First Animation

2.1 Loading Frames

For the first animation you will need a sprite sheet or a series of sprites. For this we will use a sprite sheet. (It can be found under the Example folder in the AniSprite download).


These sprites are 16 x 16 pixels. After splitting up the sheet in the sprite editor and placing a sprite in the scene add the AniSprite component. This can be done either by dragging and dropping the AniSprite script in the *AniSprite > Scripts* folder or by clicking 'Add Component' in the inspector and navigating the menu to *AniSprite > AniSprite.*
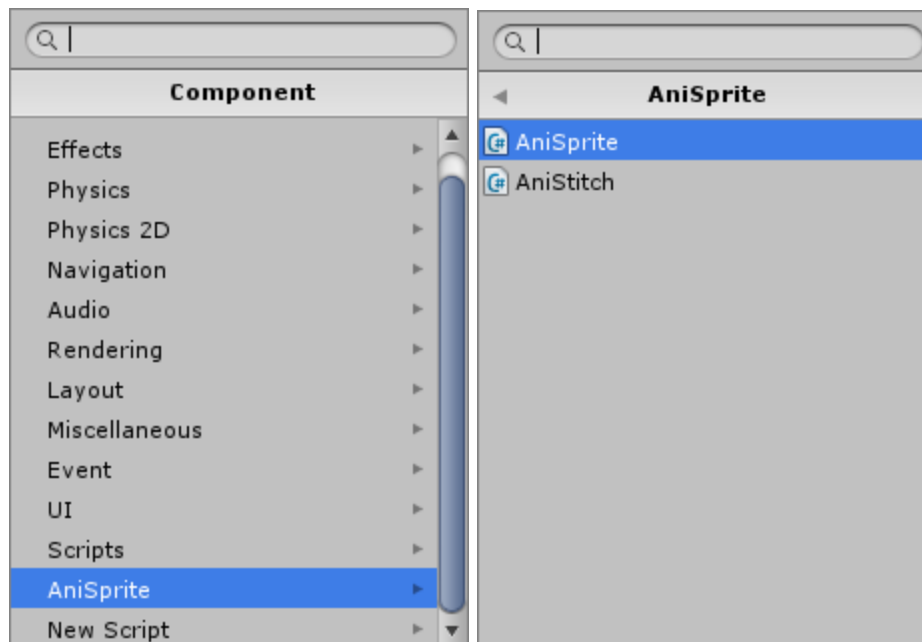

Now the AniSprite custom editor should show up. If you haven't already go and take a look at the interface section of this document. From here we need to add all the frames that we want for the animation. Go to the fold out named *'Sheet'* and add new frame and drag in the sprites

that are needed for the animation. If need all the sprites from a sheet add one sprite first then click the *'Load All Sprites'* button.
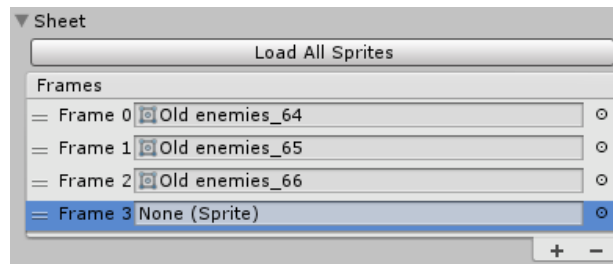
As the sprites get loaded the preview for the sprites should show now white outlines for the sprites that have been added. You will see that the first frame is also highlighted blue. Blue is the start frame for the animation. When you add a new AniSprite component it should start at 0. It is important to note that this is the programmer's 0 where 0 is the first element, hence our first frame (this means the last frame will be the number of frames minus 1, eg. 10 frames in all, last frame is 9).
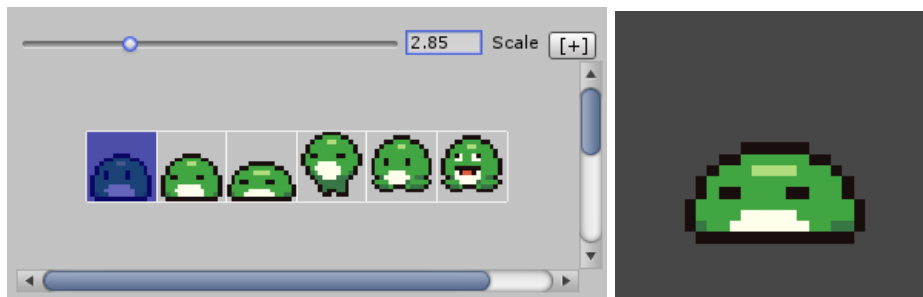
If all the frames you have added to AniSprite is the whole animation all you are done! Set the frames per second and play the scene. The animation should run through all the frames you added.

2.2 Splicing Frames

This last step is to show how to define which frames will be used for the animation. If you have more frames than the animation wanted you need to select which ones to play. For this we will animate the first row for jumping. This is frames 1 through 3 (remember we are starting at 0 so if counted the frames it would be 2 and 4). So in the frames box set the start frame to 1 and then the start and end splice to 1 and 3. You will see now that the frames we have selected are highlighted with red. Finally set the frames per second or FPS to 4 and press play. You should have a looping jump animation. All settings can be tweaked while the animation is running.

3. Saves

AniSprite makes it easy for setting up a single animation. However for most games certain objects need to switch between animations. A common one being the player character for a platformer which can jump, run or just stand still. With AniSprite once an animation is defined you can save it to then play it again later in the game.

3.1 Making Saves

We are going to use the same sprite sheet as used in the First Animation section of this document using again the green frog.

Here the jump animation frames are set since we can see that they are highlighted. To save the animation click on the big *'Save Current Animation'* button directly below the animation

options. Once done a new save will be made in the AniSaves list in the '*Saves*' foldout (default it is open).
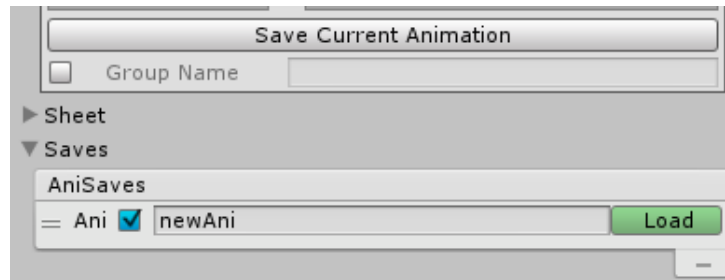

Each save in the list can be named and you can quickly toggle looping for the animation. This animation we would name 'Jump'. Names are not case sensitive so Jump, jump and JUMP are all the same animation for this frog. A full set of animations would look like the following (0 Stand, 1 - 2 Idle, 1 - 3 Jump, 4 Fall, 5 Die)

Now when pressing the load button next to each save the saved animation will show in the preview and all the options will show the settings for the animation excluding FPS (generally FPS needs to be the same for the animations of an object. If you need to change the frame rate between animations set the FPS via script or see AniStitch). To test the saves play the scene and while an animation is playing press load. See the scripting reference to see how to call saves from code.

3.2 Groups

Groups are a way to organize saves. This is also a way to have switchable states in your animation. To use grouped saves you first have to toggle the option to save in a group. Then you will need to type in a name for the group. Now when you save an animation it will be grouped under that name. See the scripting reference to see how to call grouped saves.
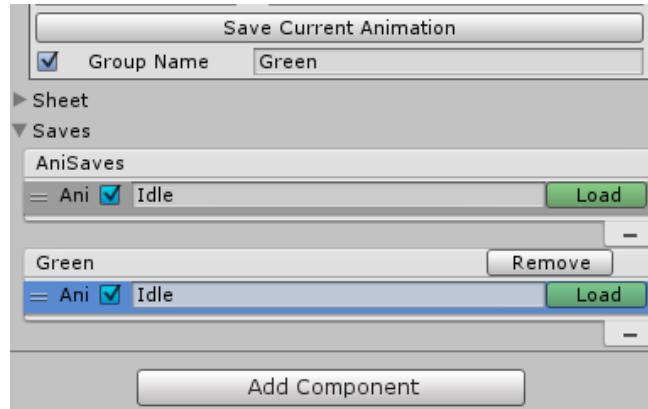
4. AniStitcher

AniStitcher is an add-on component for AniSprite. AniStitcher allows you to link together saved animations made in the main AniSprite component. To make a sequence you type in the name of the animation you want to play, the number of times to play and the frame rate at which to play the animation at. You can also either loop the whole sequence or play it for a certain number of times. To play a sequence you will have to reference the AniStitcher component from a script. However if you have a reference to AniSprite already you can call the function PlayStitch() and it will get the reference and play the stitch for you. See the scripting reference for more.
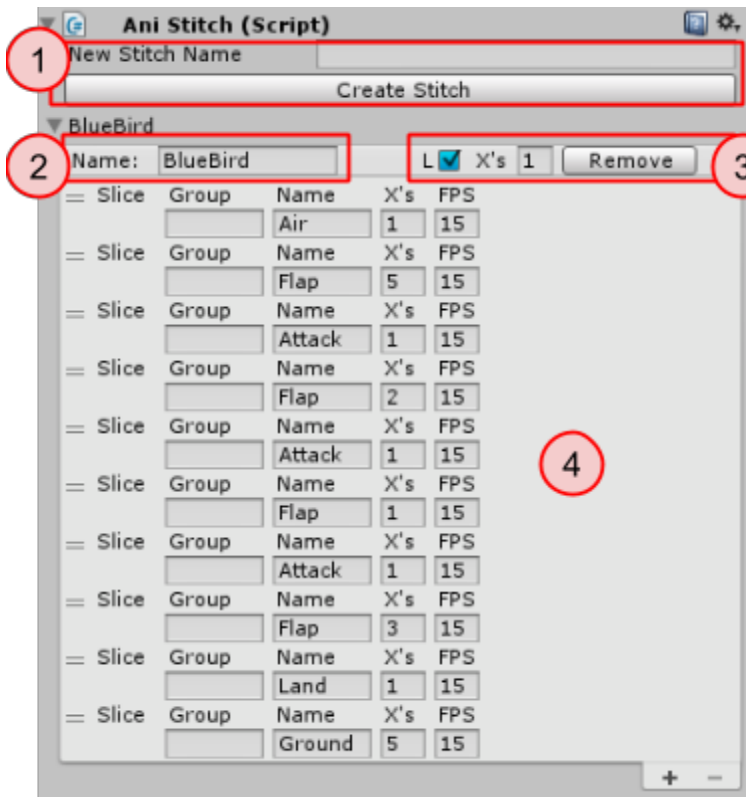
1. Textfield and button for making a new stitch.

2. Name of stitch and editable.

3. Sequence options. Left to right: Loop toggle, time to play, remove sequence.

4. Animations to play in the stitch.

5. Script Reference

5.1 AniSprite

These are the public properties and functions for AniSprite. Most of the properties can be set (or are automatically set) through the custom editor.

```
public List<Sprite> sprites = new List<Sprite>();
public bool  animatedOnStart = true;
public int FPS;
public int startFrame;
public int currentFrame;
public bool  paused;
public bool loop;
public int spliceStartFrame;
public int spliceEndFrame;
public event EventAni onComplete;
public string currentAniName;
public List<AniSave> saves = new List<AniSave>();
public List<AniGroup> groups = new List<AniGroup>();
public AniStitch stitcher
```

Plays the animation after being stopped. The animation will play from the start frame if a save is provided.
```
public void  Play ()
public void  Play ( int startFrame )
public void  Play ( string aniName )
public void  Play ( string aniGroup, string aniName  )
```

Stops all animation.
```
public void  Stop ()
```

Change the play direction.
```
public void  PlayForward ()
public void  PlayBackward ()
```

Step forward to the next or previous frame (normally used when the animation is stopped).
```
public void  StepForward ()
public void  StepBackwards ()
```

Set frame will stop and set the animation to the indexed frame or start of animation.
```
public void  SetFrame ( int frame )
public void  SetFrame (string aniName)
public void  SetFrame (string aniGroup, string aniName)
```

Access the AniStitch Component if one exists on this gameObject and plays the animation.
```
public void PlayStitch(string stitchName)
```

Loads an animation and it's settings but does not play it. Use the Play() function instead.
```
public void  LoadSave ( string name )
public void  LoadSave ( string groupName, string aniName )
```

5.2 AniStitch

These are the public properties and functions for AniSprite. Most of the properties can be set (or are automatically set) through the custom editor.

```
public List<AniCompound> patchworks = new List<AniCompound>();
public event EventAni onComplete;
```

Gets the reference to the AniSprite component.
```
public AniSprite aniSprite
```

Plays the stitch with the matching name.
```
public void Play(string name)
```