Overview

The Frog sprite system groups animations into different collections. Each collection has a JSON file, called a sprite resource file, to specify the names and details of the individual animations. The typical location for miscellaneous sprite resource files is the Sprites subfolder of the graphics path for the platform's FileSystem. For example, on the PC, they would go in "FileSystems\PC\Graphics\Sprites". In that example, if you had a collection of animations for the main game, you might have a "FileSystems\PC\Graphics\Sprites\MainGame.json" file which references png files in the same folder.

A sprite resource file consists of a JSON object where each key is the name of an animation. Normally, the value associated with that key is another JSON object with various properties. However, if an animation consists of a singe image, a single frame, and all properties left as the defaults, there is a shorter way to define the animation. Rather than providing a JSON object for the animation, just give a string for the filename, excluding the extension.

The bitmap data for a sprite animation may consist of a series of files where each file is used, in its entirety, as a frame. Bitmap data may also be provided using a single bitmap file where different parts of the picture are used for individual frames. If you want to use different parts of a single bitmap, specify the FrameSize parameter. The bitmap will then be treated as a grid where cell 1 is in the top-left corner, and each cell is the size given in FrameSize. Cells are numbered in rows from left to right and top to bottom. Beware that using the FrameSize feature can result in artifacts at the edges when rotating or scaling. This can be avoided by padding the frames with transparent pixels.

Properties

- Filename : All filenames are relative to the current graphics path, and you must exclude the extension. For instance, if you want to refer to "FileSystems\PC\Graphics\Sprites \Example.png", the appropriate filename would be "Sprites\Example". If there is only one frame, and you aren't using the FrameDurations property, this gives the filename of the image to use for the sprite, excluding the extension. Otherwise, this gives the name of a folder containing a numbered sequence of bitmaps beginning at 1. The filenames should contain three digits, including leading zeros if needed. For example, if there were three frames in the animation, the folder for the sequence would contain 001.png, 002.png, and 003.png. You will probably never need more than 3 digits, but if it comes up, just don't include any leading zeros in numbers greater than 999. For example, 1000.png.
- FrameCount : The number of frames in the animation. This is mutually exclusive with ImageSequence. (Default: 1)
- FrameDurations : JSON array of numbers for the durations of individual frames, in milliseconds. This is mutually exclusive with FrameRate. (Default: defer to FrameRate)
- FrameRate : The frame rate of the animation in frames per second. This is mutually exclusive with FrameDurations. (Default: 30)
- FrameSize : If this is defined, the image specified by Filename will be treated as a grid of frames of this size. For now, all the frames must come from a single image file if this parameter is used. (Default: not used)
- ImageSequence : JSON array of numbers that specifies which images should be used for the individual frames. This is mutually exclusive with FrameCount. (Default: defer to FrameCount)
- KeepBitmapData : On some platforms, like those that use OpenGL, we don't need to keep the bitmap data to display sprite animations after they've been loaded. In those cases, the

bitmap data is automatically released to save memory. However, if you need to keep the bitmap data in memory for some other reason, just set this to true. This is a less efficient use of memory, and it may prevent the underlying Images from being reference-counted, so only use this if necessary. (Default: false)

- Offset : Shift the position of the animations by this amount. This allows you to use something other than the top-left corner as the origin for the animation. For example, if the dimensions of the animation were 8×32, and you wanted the origin to be at the center of the animation, the correct offset would be "-4|-16". This is mutually exclusive with Origin. (Default: "0|0")
- Origin : This allows you to use something other than the top-left corner as the origin for the animation. For example, if the dimensions of the animation were 8×32, and you wanted the origin to be at the center of the animation, the correct value would be "4|16". If you specifically wanted to use the center of the animation as the origin, you could specify "Center" instead. Other offset shortcuts include "TopLeft", "Top", "TopRight", "Left", "Right", "BottomLeft", "Bottom", and "BottomRight". This property is mutually exclusive with Offset. (Default: Defer to Offset)
- PlayType : How the frames of the sprite should be animated. For now, this can be "OneShot", "Loop", or "PingPongLoop". If the type is "OneShot", it will play the animation from beginning to end and stop on the last frame. If the type is "Loop", it will repeatedly play the animation from beginning to end. If the type is "PingPongLoop", it will repeatedly play the animation from beginning to end and from end to beginning. (Default: "Loop")

Example

FileSystems\PC\Graphics\Sprites\Moose.json

```
{
   "Idle":
   {
      "Filename": "Sprites/Moose/Idle",
      "FrameCount": 11,
      "FrameRate": 20,
      "PlayType": "Loop",
      "Offset": "-110|-110"
   },
   "Lose":
   {
      "Filename": "Sprites/Moose/Lose",
      "ImageSequence": [1, 2, 3, 2, 3, 2],
      "FrameRate": 20,
      "PlayType": "OneShot",
      "Offset": "-110|-110"
   },
   "Win":
   {
      "Filename": "Sprites/Moose/Win",
      "ImageSequence":  [  1,   2,    3,   2],
      "FrameDurations": [250, 500, 1000, 500],
      "PlayType": "OneShot",
      "Offset": "-110|-110"
   },
   "Example": "Sprites/Moose/SingleFrameExample"
}
```