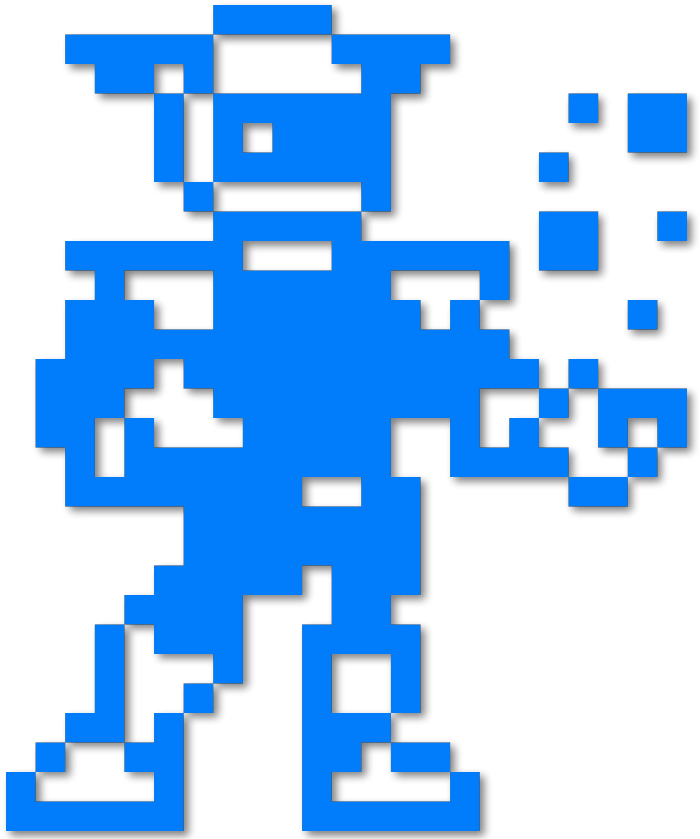


SpriteKit

by Justin Shacklette

What is SpriteKit?

- 2D sprite-based game framework

sprites **animation** **emitters**

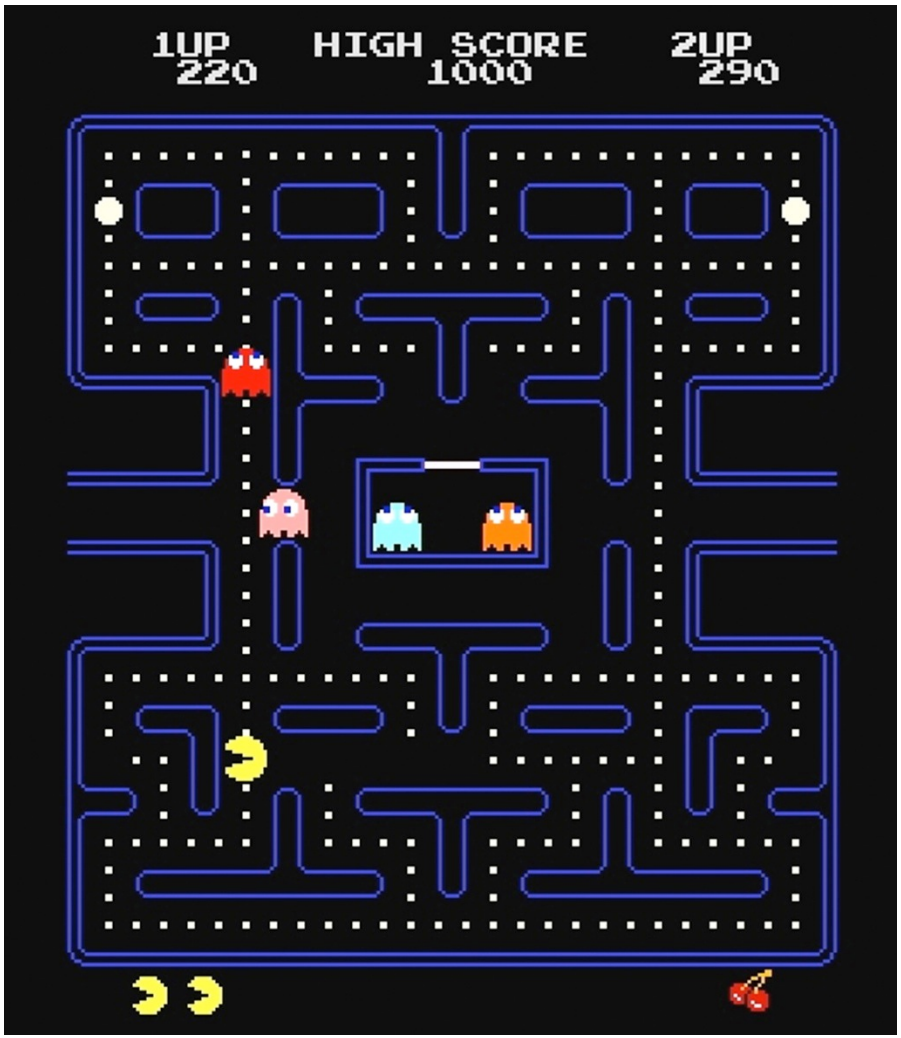
particles **physics** **actions**

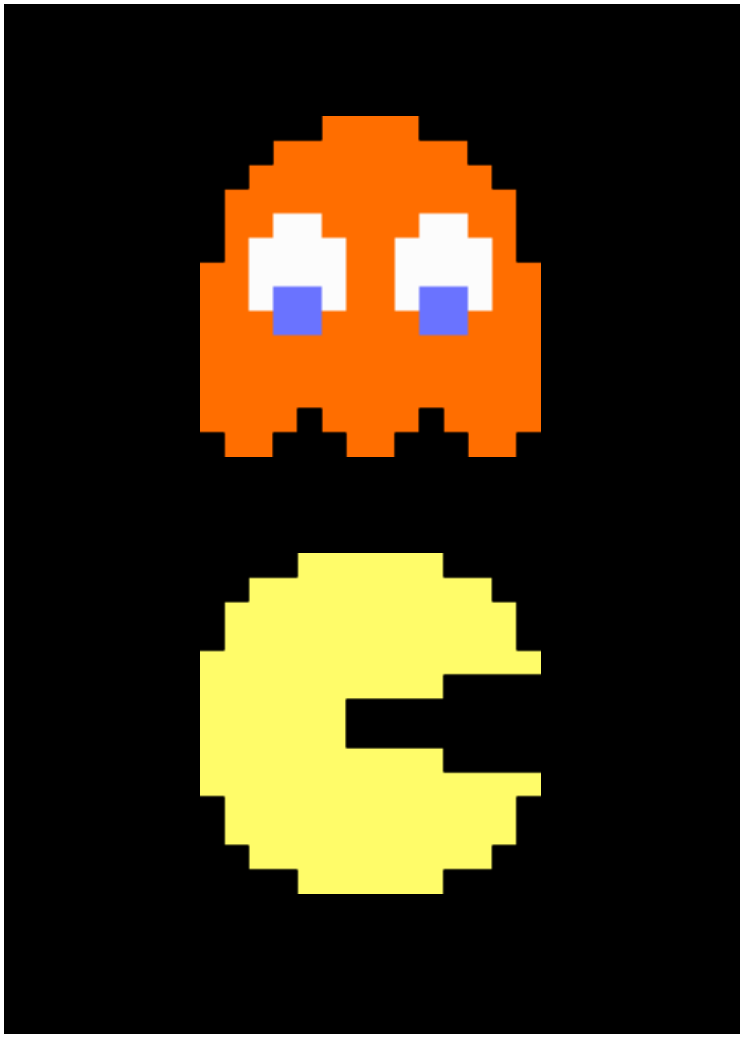
audio **events**

gestures

Sprites

- 2D bitmap images



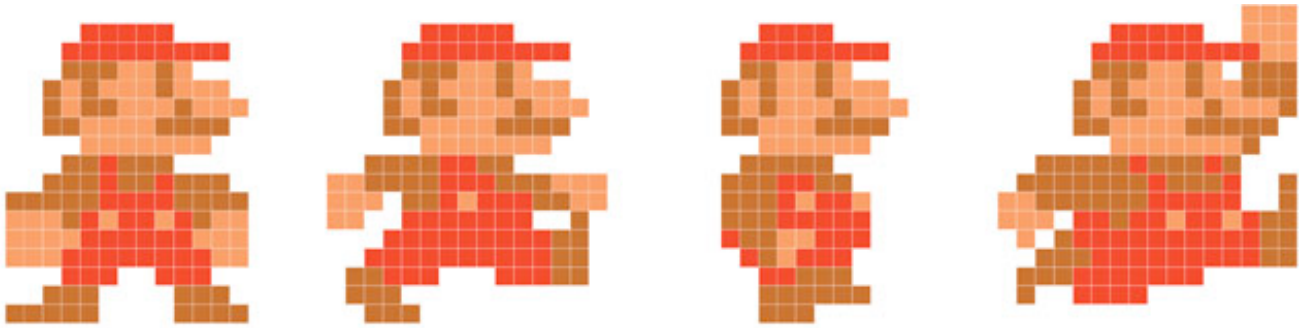


Sprites

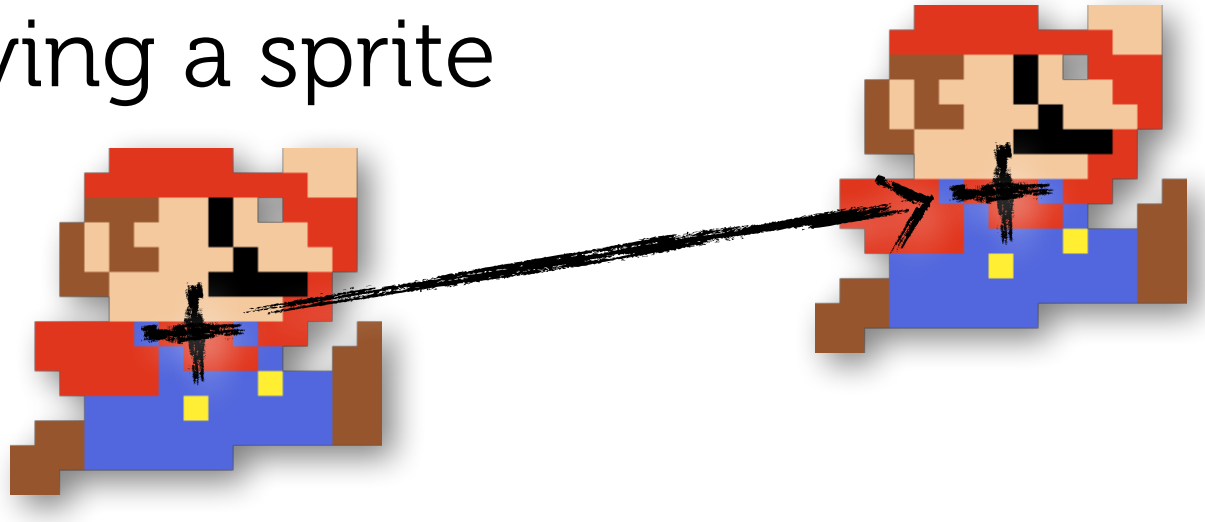
```
SKSpriteNode *sprite = [SKSpriteNode
  spriteNodeWithImageNamed:@“Pacman”];

[self addChild:box];
```

Animation

- Animating a sprite



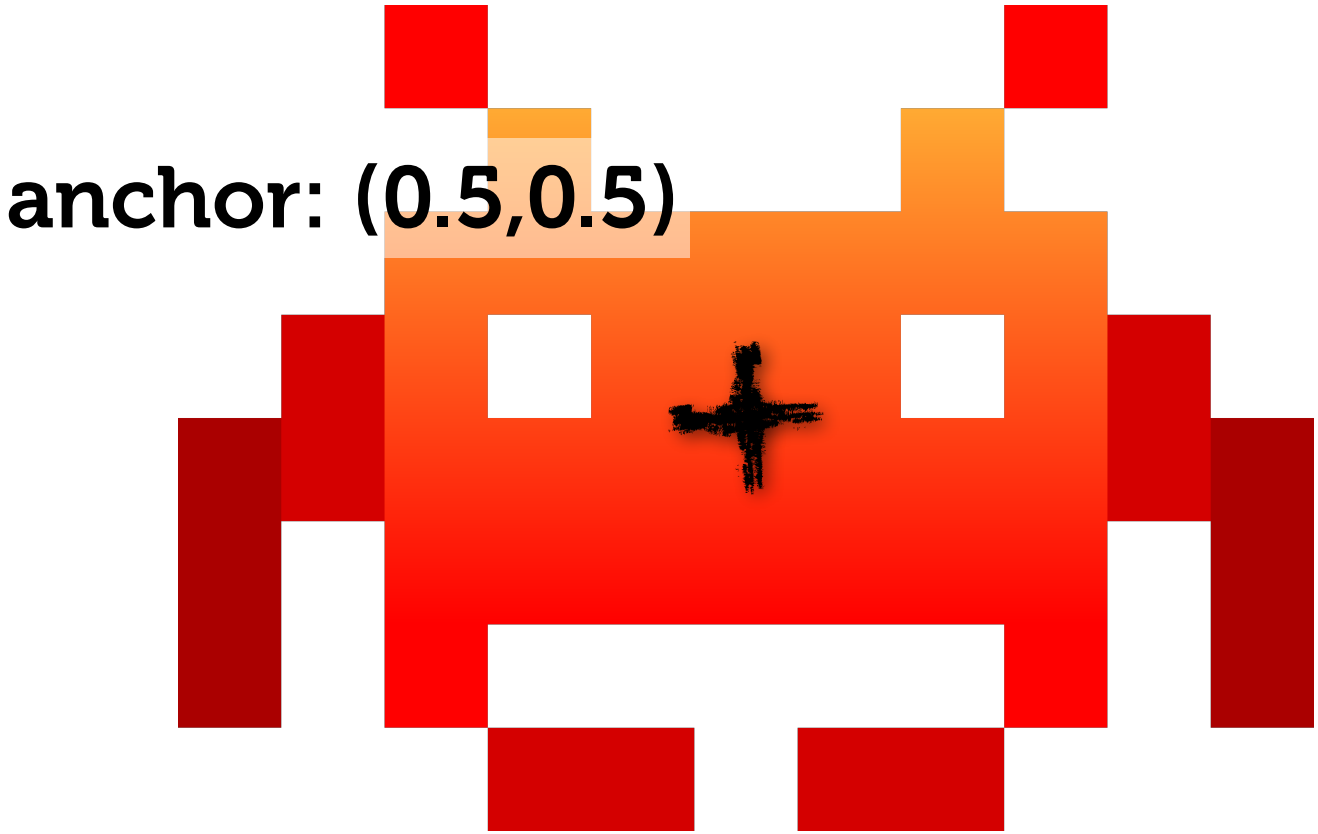
- Moving a sprite



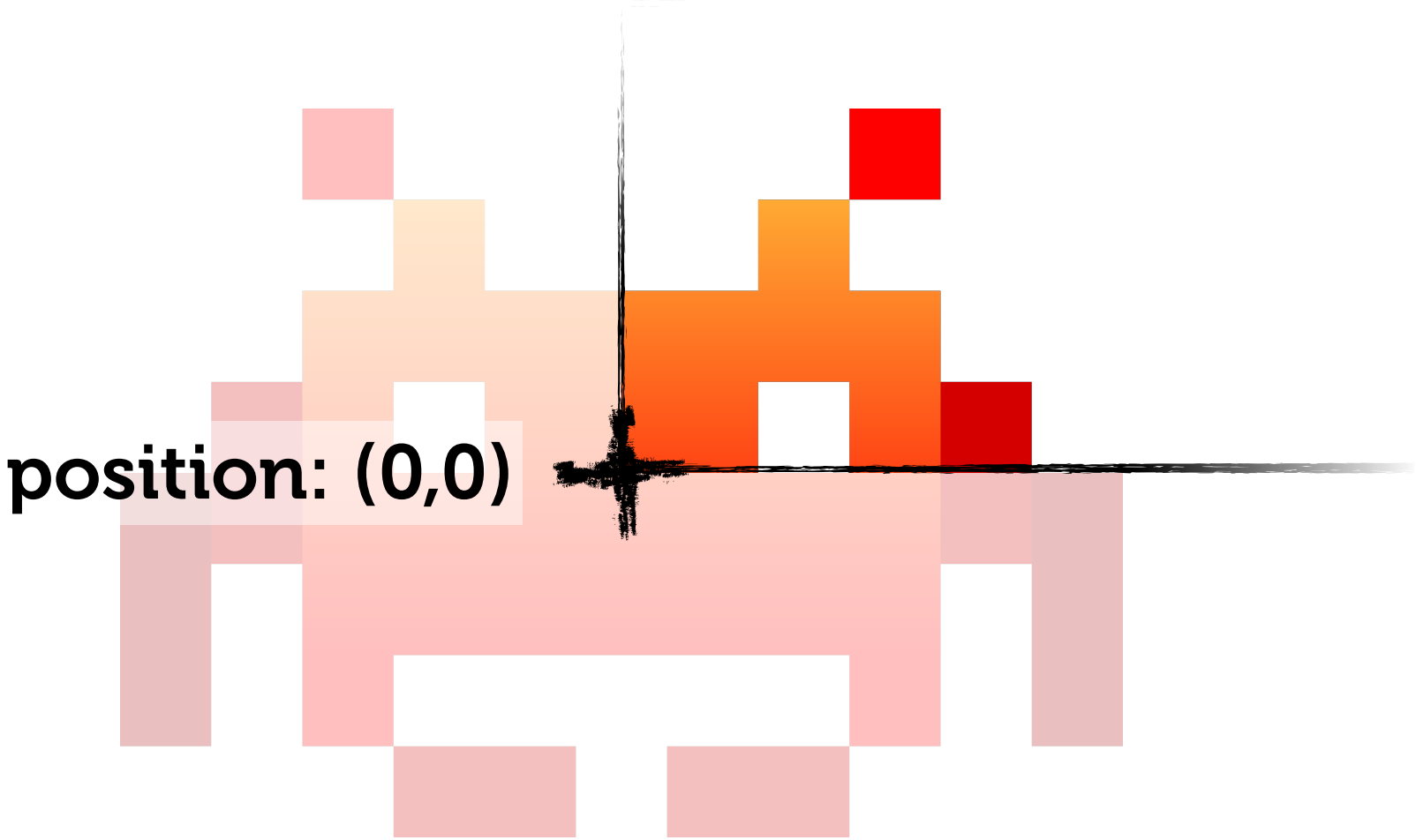
Animation

```
SKTextureAtlas *atlas =
  [SKTextureAtlas atlasNamed:@“alien”];

SKTexture *a = [atlas textureNamed:@“alien1.png”];
SKTexture *b = [atlas textureNamed:@“alien2.png”];
SKTexture *c = [atlas textureNamed:@“alien3.png”];

SKAction *anim = [SKAction
  animateWithTextures:@[a,b,c] timePerFrame:0.1f];

//moving a sprite
sprite.position = CGPointMake(
  sprite.position.x + 10, sprite.position.y + 10);
```

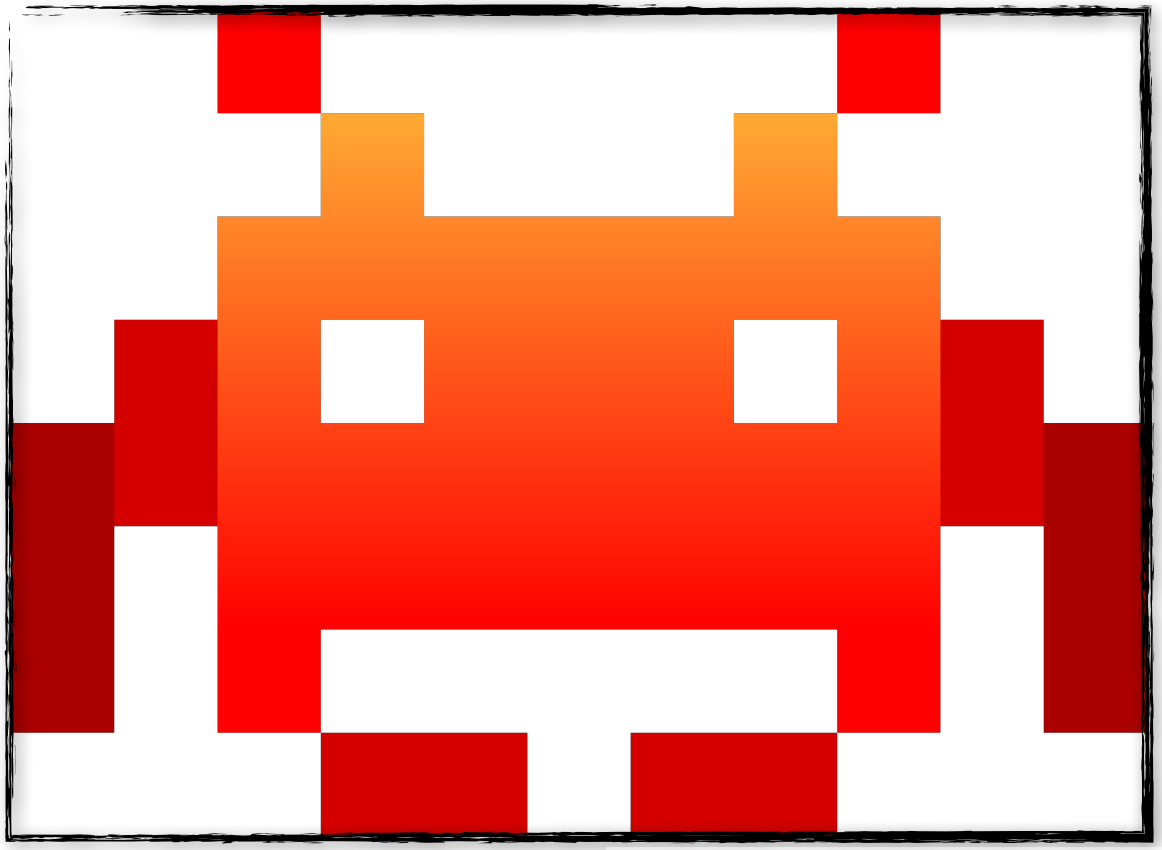
Sprite Anchor



Sprite Position



Sprite Size

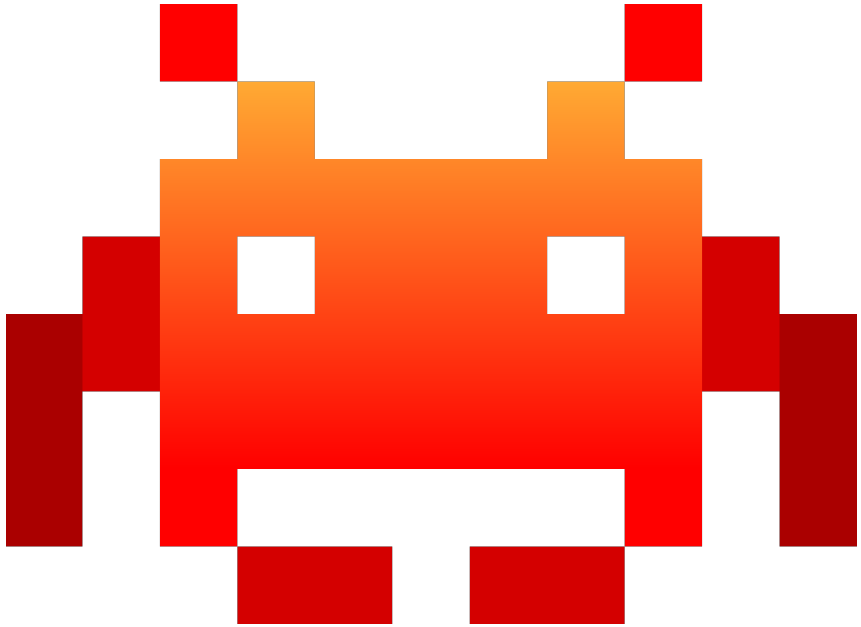


size: 44px × 32px

Sprite xScale, yScale

1.0

44px × 32px

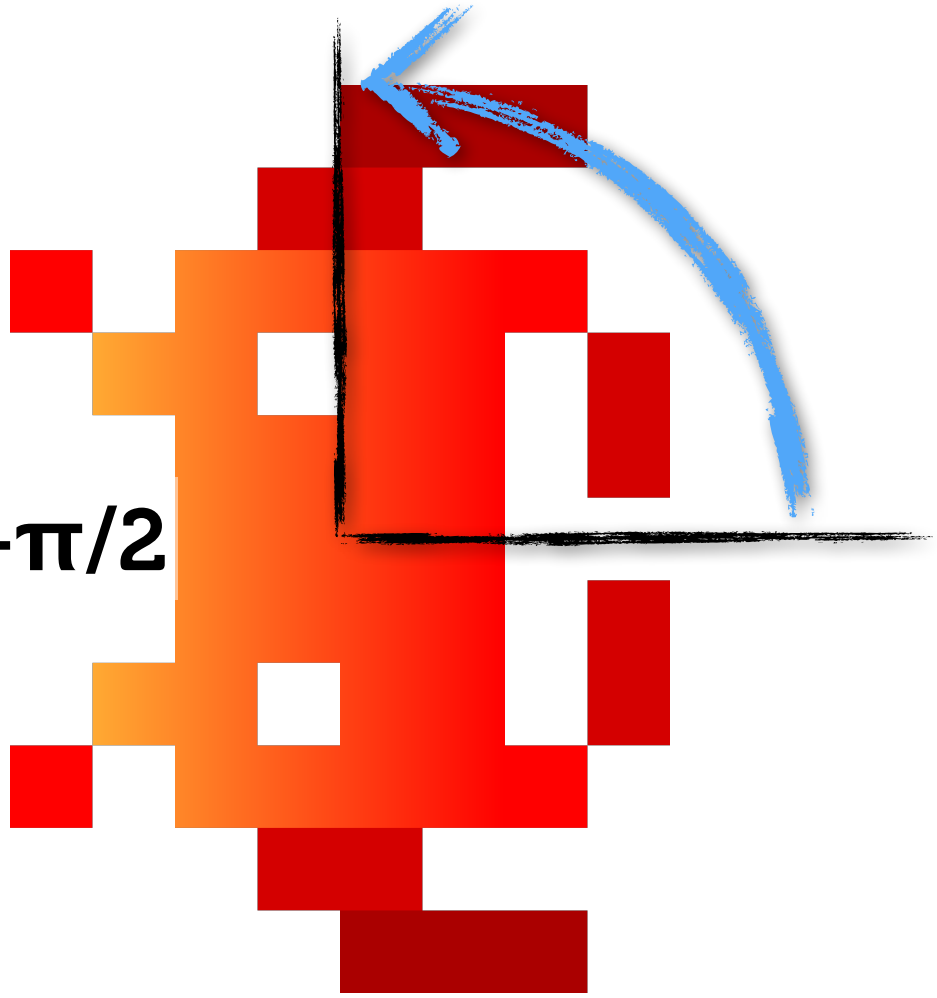


2.0

88px × 64px

Sprite zRotation

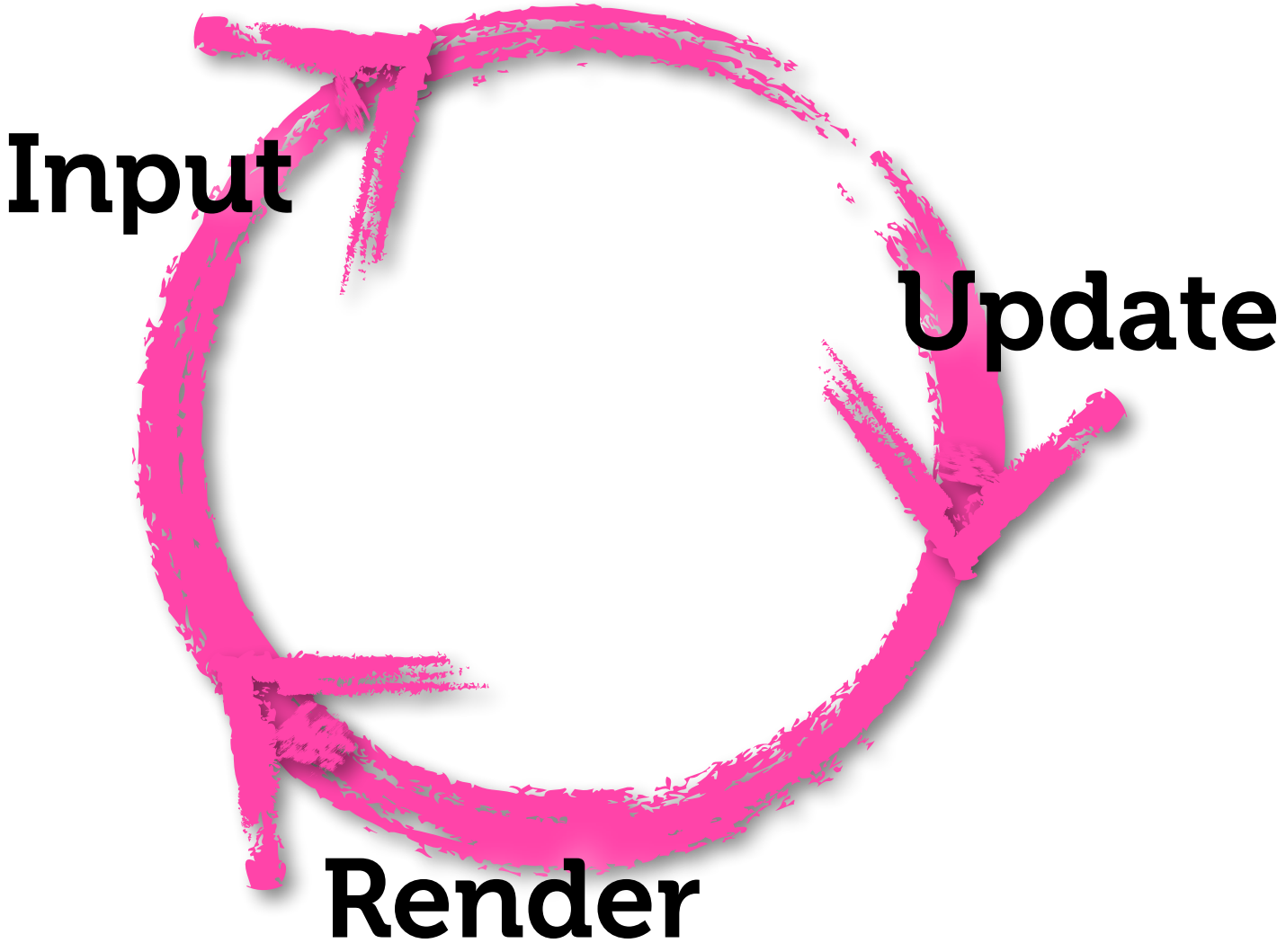
zRotation: +π/2



Sprite Properties

```
alien.anchorPoint = CGPointMake(0.5f,0.5f);

alien.position = CGPointMake(0,0);

alien.xScale = alien.yScale = 2.0;

alien.zRotation = M_PI_2;
```

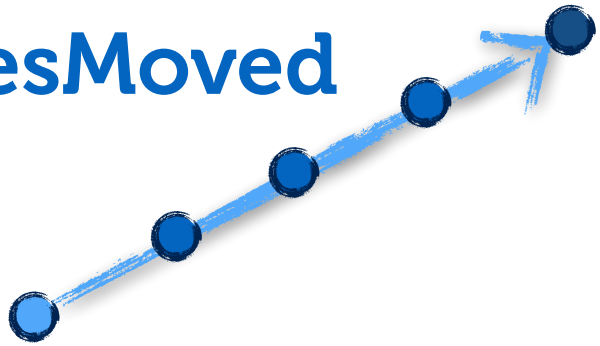
The Game Loop



Touch Events

touchesEnded

touchesMoved



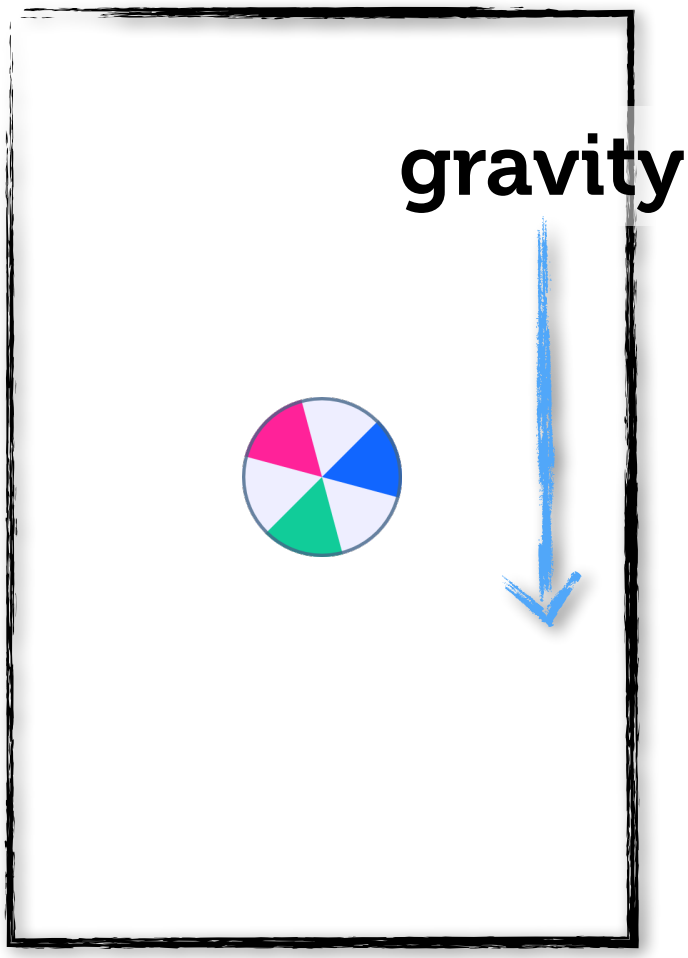
touchesBegan

(0,0) (25,25) (50,50) (75,75) (100,100)

Touch Events

```
-(void)touchesBegan:(NSSet *)touches
    withEvent:(UIEvent *)event;

-(void)touchesMoved:(NSSet *)touches
    withEvent:(UIEvent *)event;

-(void)touchesEnded:(NSSet *)touches
    withEvent:(UIEvent *)event;
```

Physics



Physics

```
//world (SKScene)
self.physicsBody = [SKPhysicsBody
  bodyWithEdgeLoopFromRect:self.frame];

//box (SKSpriteNode)
box.physicsBody = [SKPhysicsBody
  bodyWithRectangleOfSize:box.size];
box.physicsBody.dynamic = YES;

//ball (SKSpriteNode)
ball.physicsBody = [SKPhysicsBody
  bodyWithCircleOfRadius:ball.size.width/2];
ball.physicsBody.dynamic = YES;
ball.physicsBody.restitution = 0.6f;
```

Actions

- lots of actions: fade in, fade out, move, scale, rotate, wait, play sound, animate texture, colorize, alpha, follow path, execute block, invoke selector
- group: in sequence, in parallel
- repeat: for count, forever
- easing: linear, in, out, in+out

Actions

```
[SKAction fadeOutWithDuration:1.0f];
[SKAction fadeInWithDuration:1.0f];

[SKAction moveTo:CGPointMake(1,2)
  duration:1.0f];

[SKAction playSoundFileNamed:@“miss.wav”
  waitForCompletion:NO];

[SKAction removeFromParent];

[SKAction runBlock:^{…}];

[alien runAction:[SKAction sequence:
  @[miss,fade,block,remove]]];
```

The Good

- very easy to get started
- great API
- direct Xcode support
- lots of actions
- simple integrated physics
- simple particles + emitters

The Bad

- no custom drawing in OpenGL
- limited access to physics engine
- scene sizing issues
- limited extensibility
- feels beta

github.com/saturnboy

justin@saturnboy.com
@saturnboy

Links

- Official Docs

http://developer.apple.com/library/ios/documentation/GraphicsAnimation/Conceptual/SpriteKit_PG/

- Ray Wenderlich Tutorials

http://www.raywenderlich.com/42699/spritekit-tutorial-for-beginners

http://www.raywenderlich.com/45152/sprite-kit-tutorial-animations-and-texture-atlases

- more...

http://www.ymc.ch/en/ios-7-sprite-kit-my-top-5-pros-and-cons/

http://www.ymc.ch/en/ios-7-sprite-kit-setting-up-correct-scene-dimensions/