About SpriteKit Coordinate Systems

Learn how a node conforms to its coordinate systems.

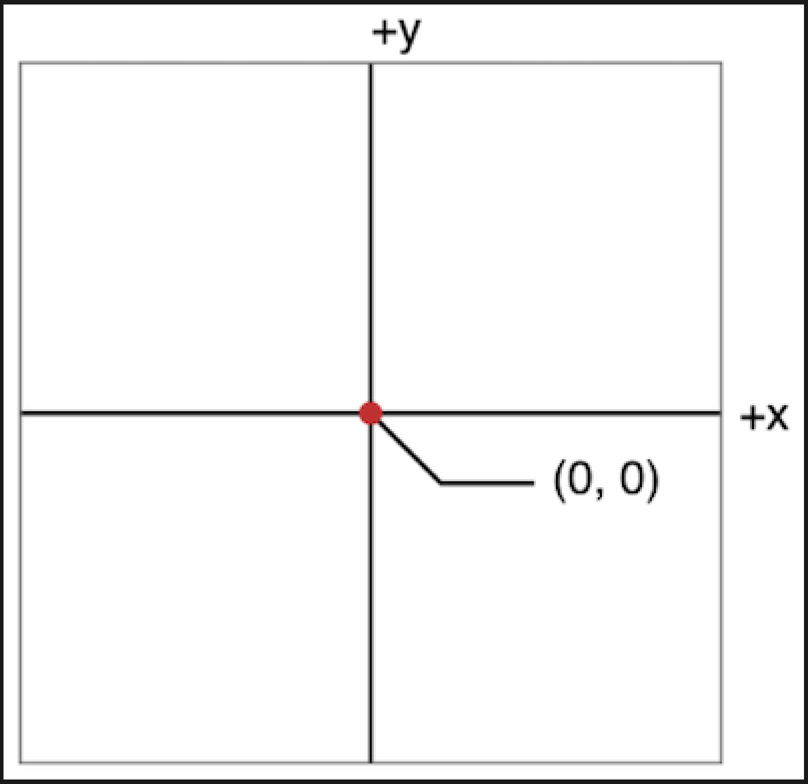
Overview

When a node is placed in the node tree, its position property places it within a coordinate system provided by its parent. SpriteKit uses the same coordinate system on both iOS and macOS.

Points as Position Units

When a node is placed in the node tree, its position property places it within a coordinate system provided by its parent. SpriteKit uses the same coordinate system on both iOS and macOS. Figure 1 shows the SpriteKit coordinate system. Coordinate values are measured in points, as in UIKit or AppKit; where necessary, points are converted to pixels when the scene is rendered. A positive x coordinate goes to the right and a positive y coordinate goes up the screen.

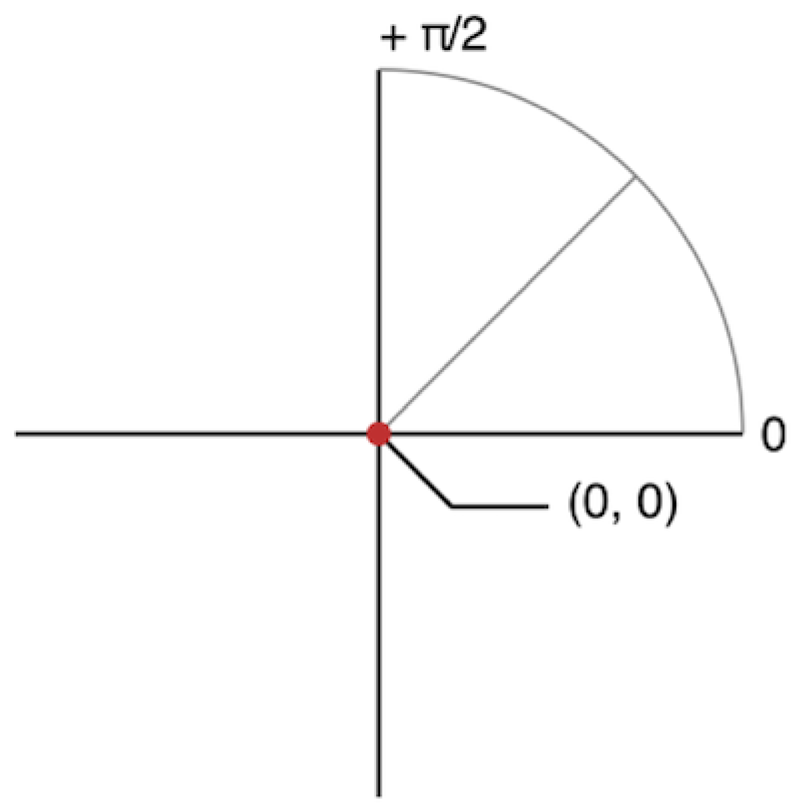
FIGURE 1 SPRITEKIT COORDINATE SYSTEM



Polar Coordinate Rotation

SpriteKit also has a standard rotation convention. shows the polar coordinate convention. An angle of 0 radians specifies the positive x axis. A positive angle is in the counterclockwise direction.

FIGURE 2 POLAR COORDINATE CONVENTIONS (ROTATION)



Nodes are rotated by setting their zRotation property to the required angle in radians. If you prefer to work in degrees, the following code shows how you can write an extension to CGFloat that converts between the two. The example in Listing 1 rotates spriteNode by 30 degrees counterclockwise.

```
extension CGFloat {
    func degreesToRadians() -> CGFloat {
        return self * CGFloat.pi / 180
    }
}

let rabbitTexture = SKTexture(imageNamed: "rabbit.png")

let spriteNode = SKSpriteNode(texture: rabbitTexture)

spriteNode.zRotation = CGFloat(30).degreesToRadians()
```