



MAKE
SCHOOL

COCOS2D-X

2D Game Engine written in C++

Based on Objective-C engine, Cocos2d-iPhone

Open Source

Front-ends for C++, Javascript, and Lua

Cocos2d-x JS written in Javascript, can publish to web

TECHNOLOGY

No need to write
OpenGL code

Instead use **Node**,
Sprite, etc. to *define*
scene content



MODULES

Input

Rendering

Animations

Physics

Action System (**Action** subclasses)

Physics Engine (**PhysicsWorld**, **PhysicsBody**)

Node Library (**Node**, **Text**, **Sprite**)

NODE - THE ROOT CLASS

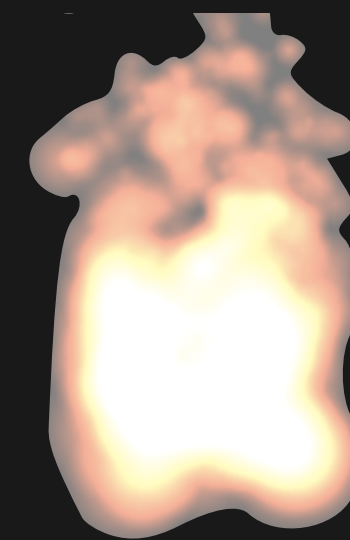
Each **Node** subclass displays a different kind of content



Sprite

Hello World

Text



ParticleSystem

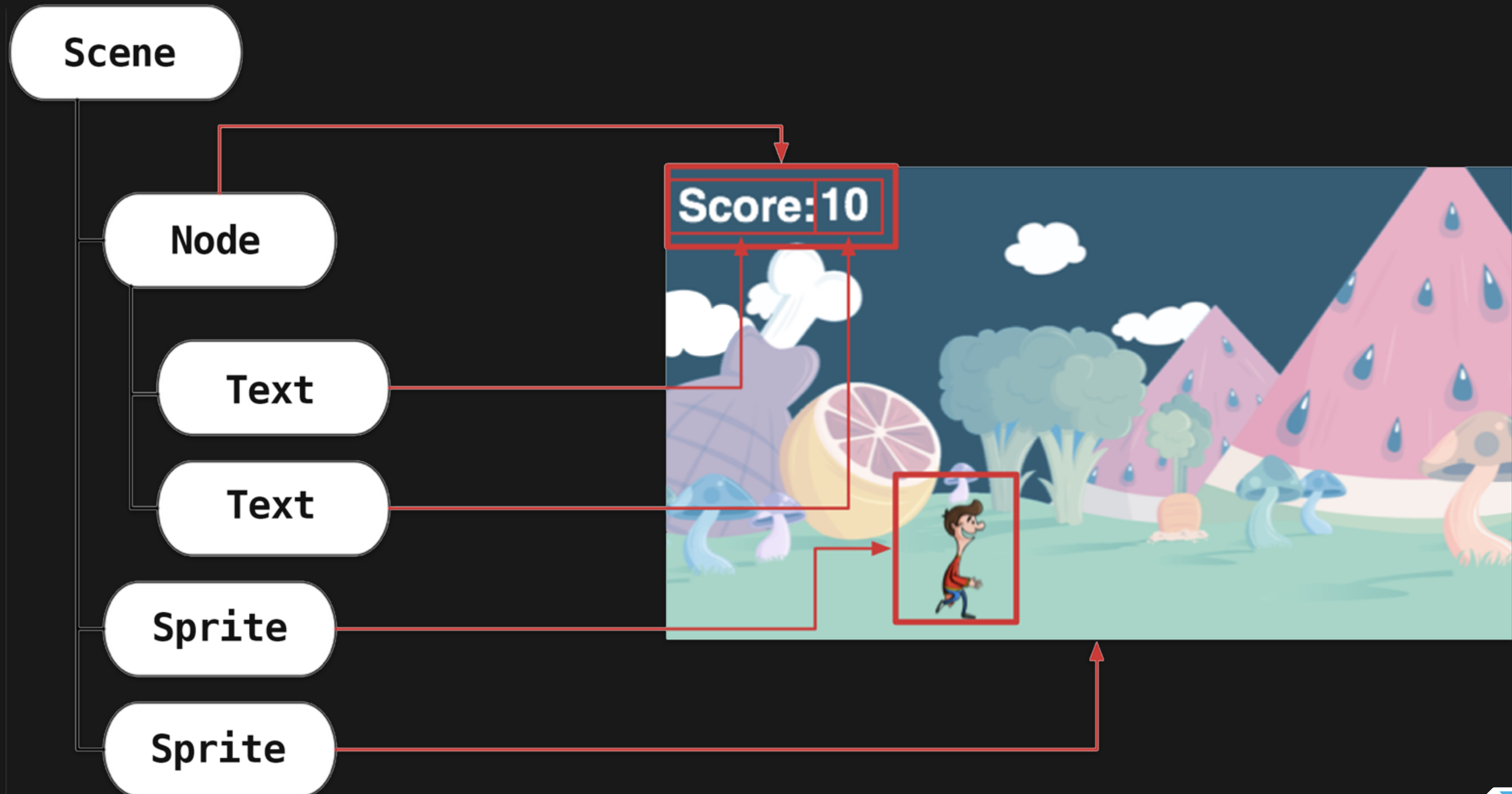
SCENE GRAPH IN COCOS2D-X

Hierarchy of different **Node** subclasses

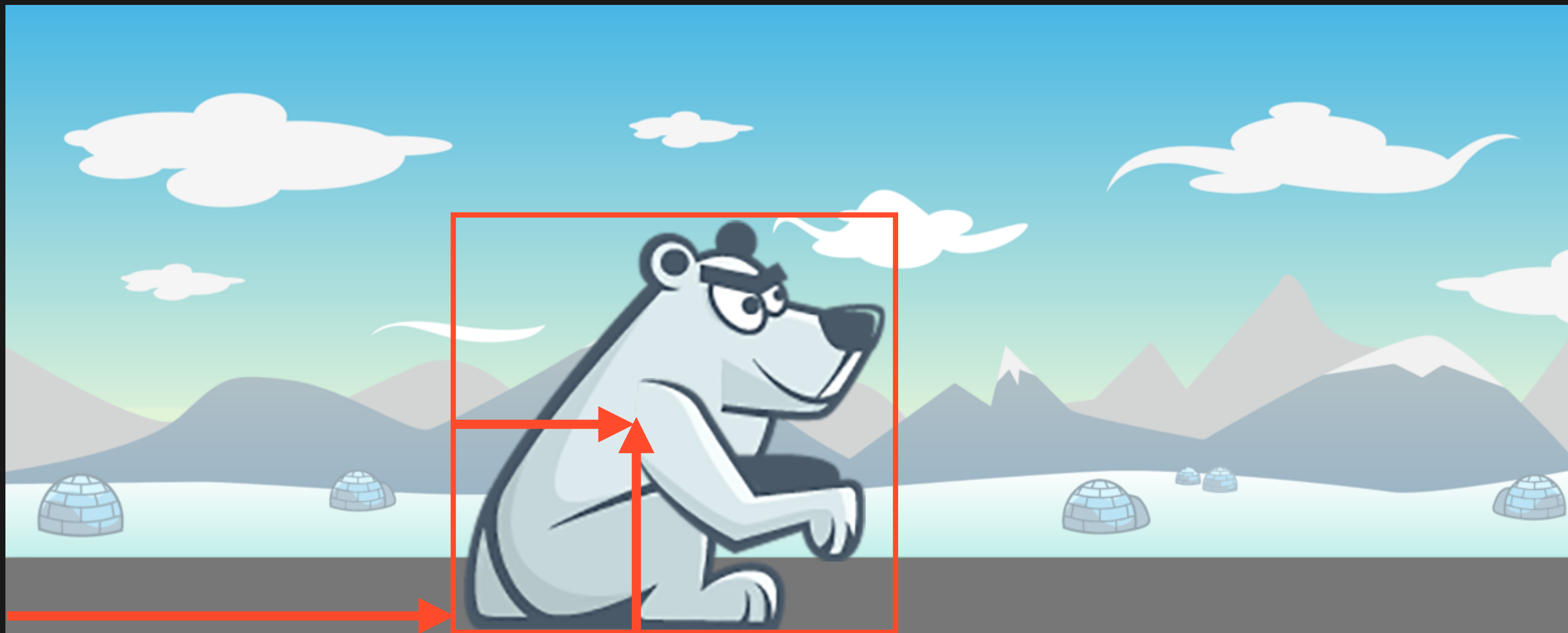
Scene is the **root node**

Every node can have children

SCENE GRAPH IN COCOS2D-X

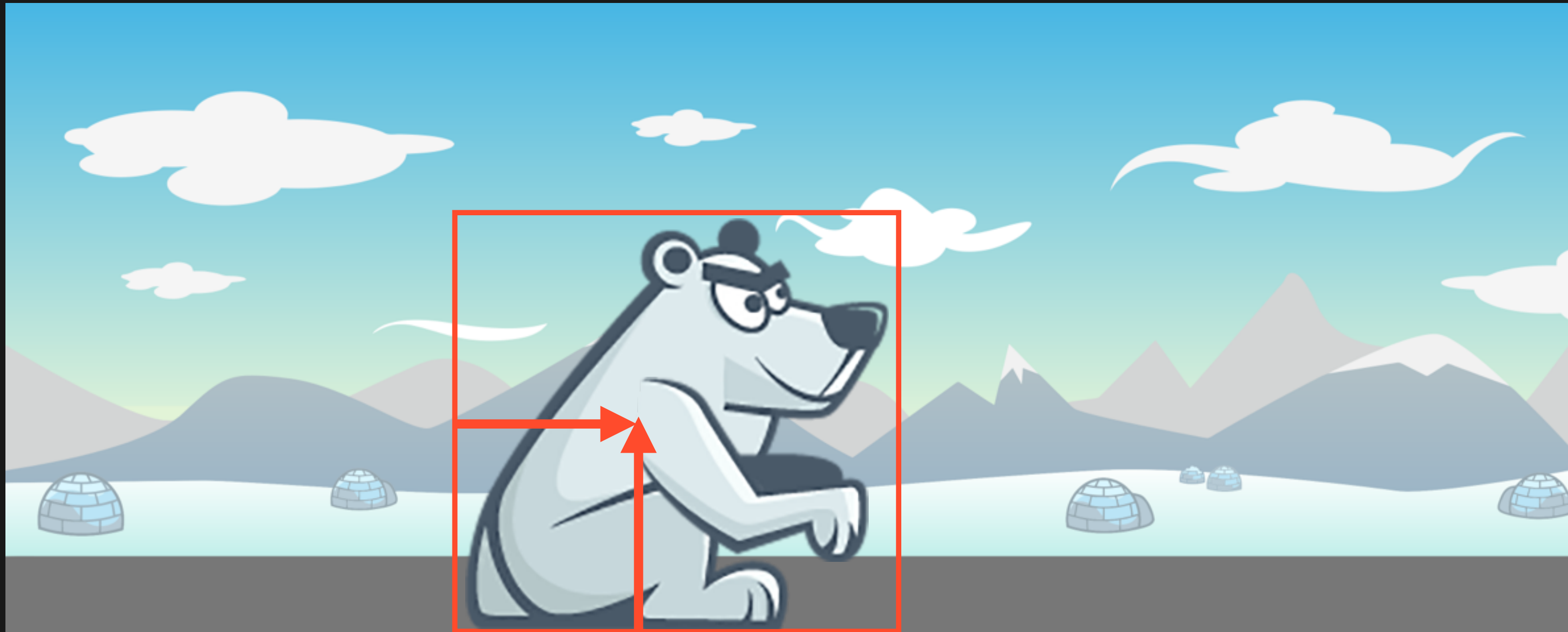


CHILDREN ARE PLACED RELATIVE TO THEIR PARENT



```
background->addChild(bear);  
bear->setPosition(100.0f, 0.0f)  
bear->addChild(bearArm)  
bearArm->setPosition(40.0f, 50.0f)
```


CHILDREN MOVE WITH THEIR PARENTS



IMPORTANT NODE PROPERTIES

Property	Default Value
<code>position</code>	<code>(0.0f, 0.0f)</code>
<code>scale</code>	<code>1.0f</code>
<code>rotation</code>	<code>0.0f</code>
<code>anchorPoint</code>	<code>(0.5f, 0.5f)</code>
<code>contentSize</code>	<code>(0.0f, 0.0f)</code>
<code>visible</code>	<code>true</code>

NODE FEATURES

Can contain other nodes

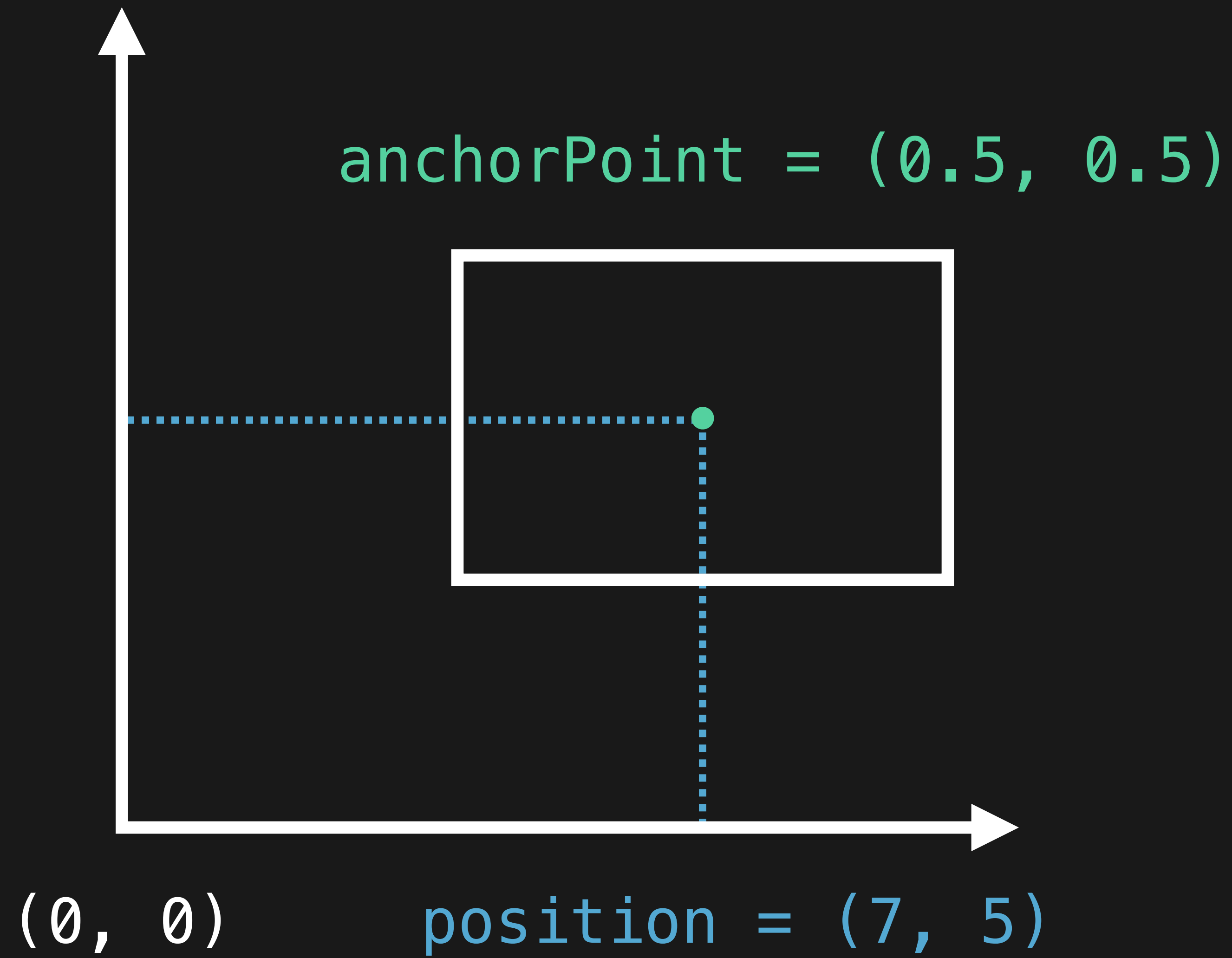
`addChild()` `removeChild()` `removeFromParent()`

Can schedule timed callbacks

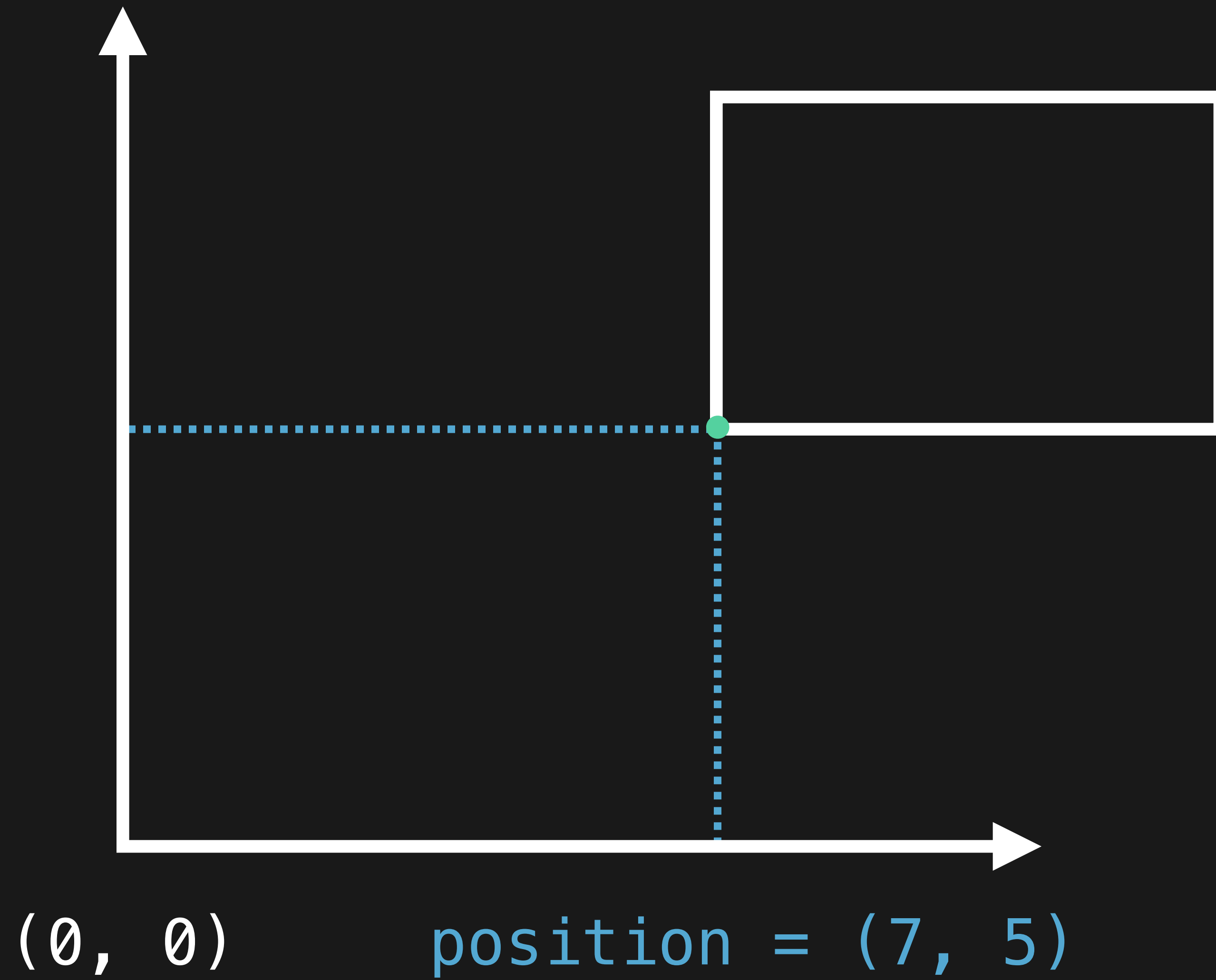
`scheduleUpdate()` `schedule()` `unschedule()`

Can execute actions

`runAction()` `stopAction()` `stopAllActions()`

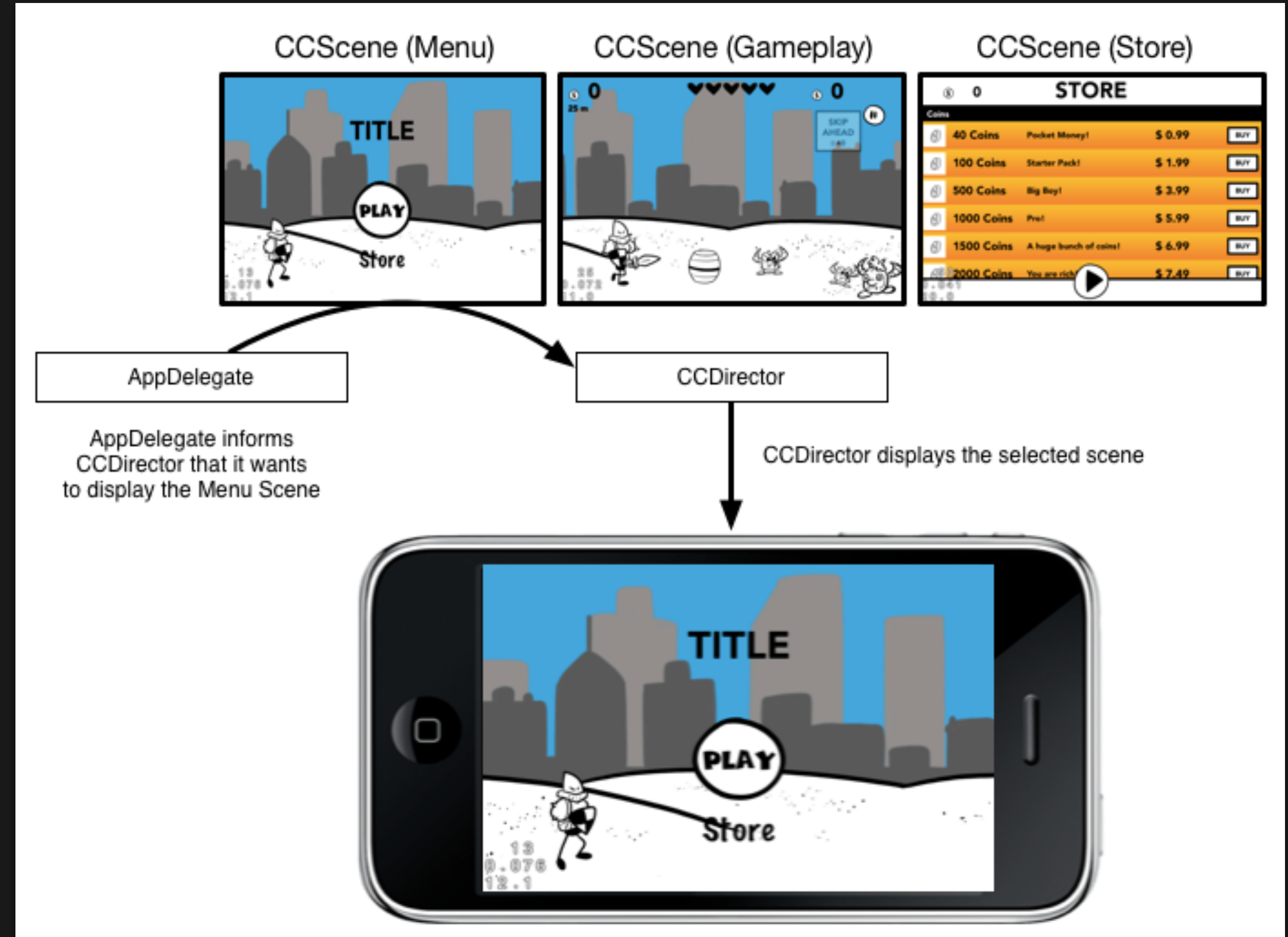


`anchorPoint = (0.0, 0.0)`



SWITCHING SCENES

Director chooses which scene is currently displayed



DIRECTOR IMPORTANT METHODS

`pushScene()`

Switch to a new scene but preserve the old one

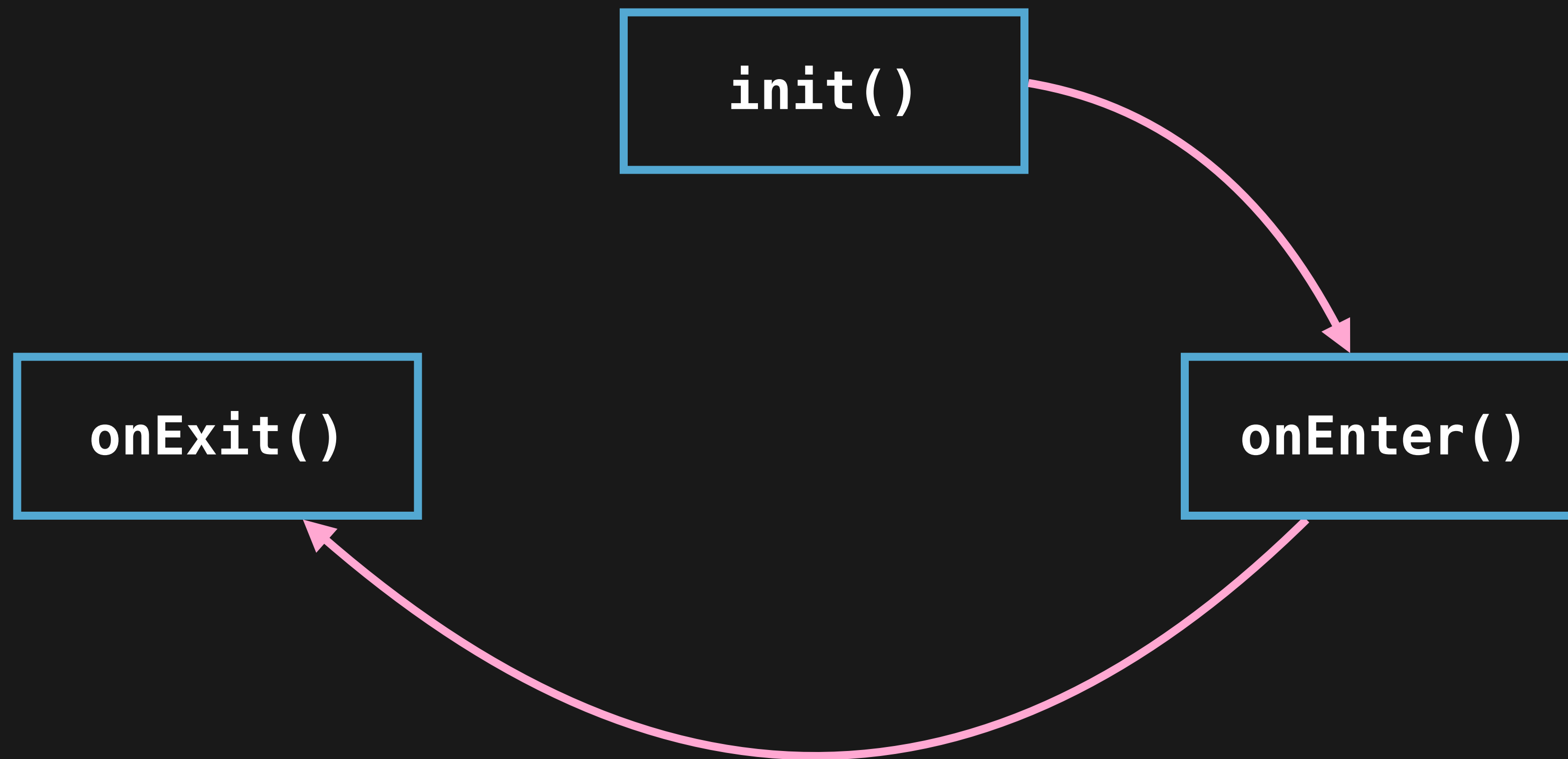
`popScene()`

Go back to the last scene before `popScene()` was called

`replaceScene()`

Switch to a new scene and delete the old one

NODE LIFECYCLE



UPDATE METHOD

Update method is called every frame

Can be used for manual animations

Also useful for continually running game logic

```
// start update method  
this->scheduleUpdate();
```

```
// stop update method  
this->unscheduleUpdate();
```

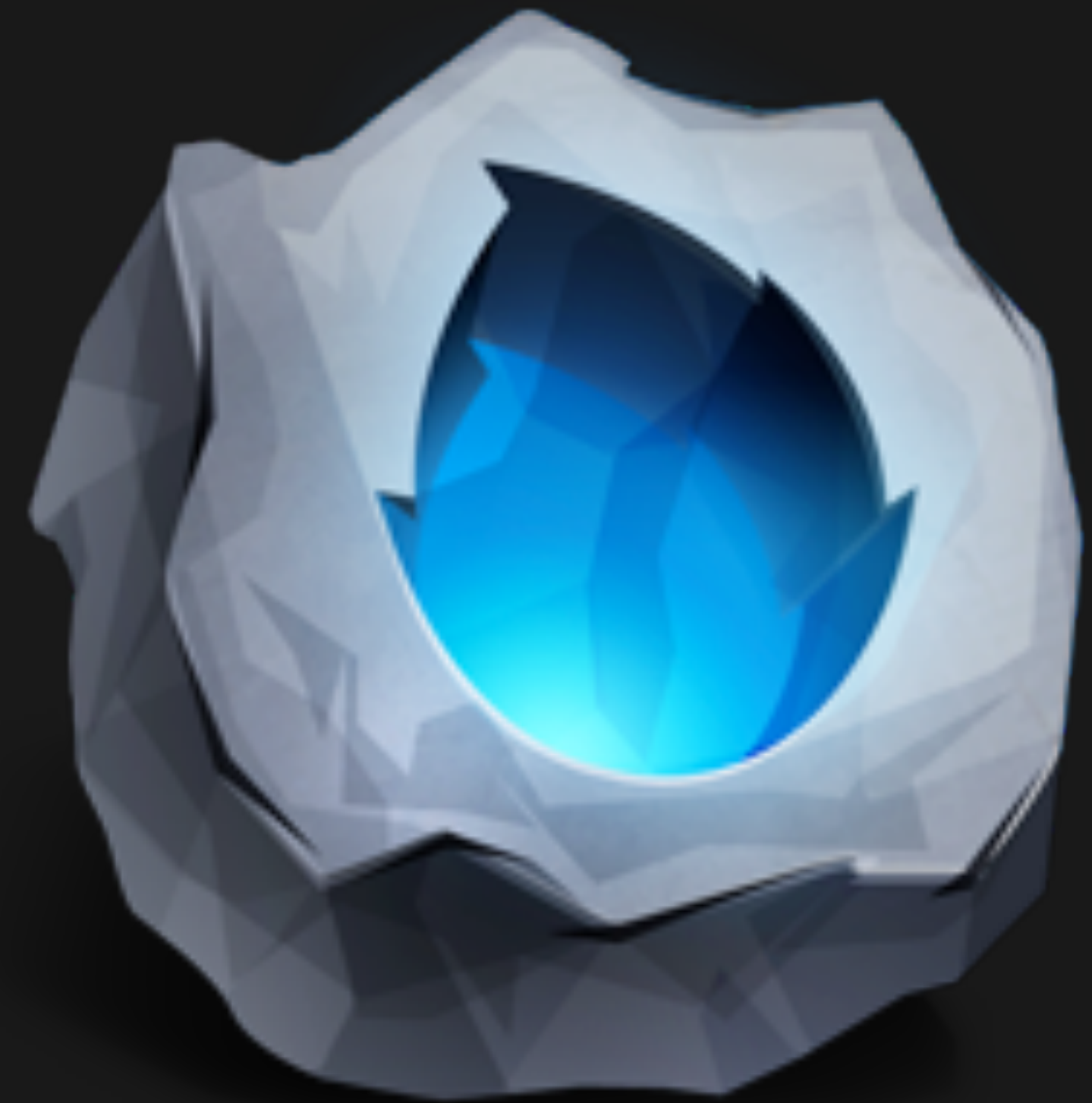
```
void MainScene::update(float dt)  
{  
    // dt is time (in seconds)  
    // since last time update was called  
  
    // if game is running 60fps  
    // dt will be .0166666  
}
```

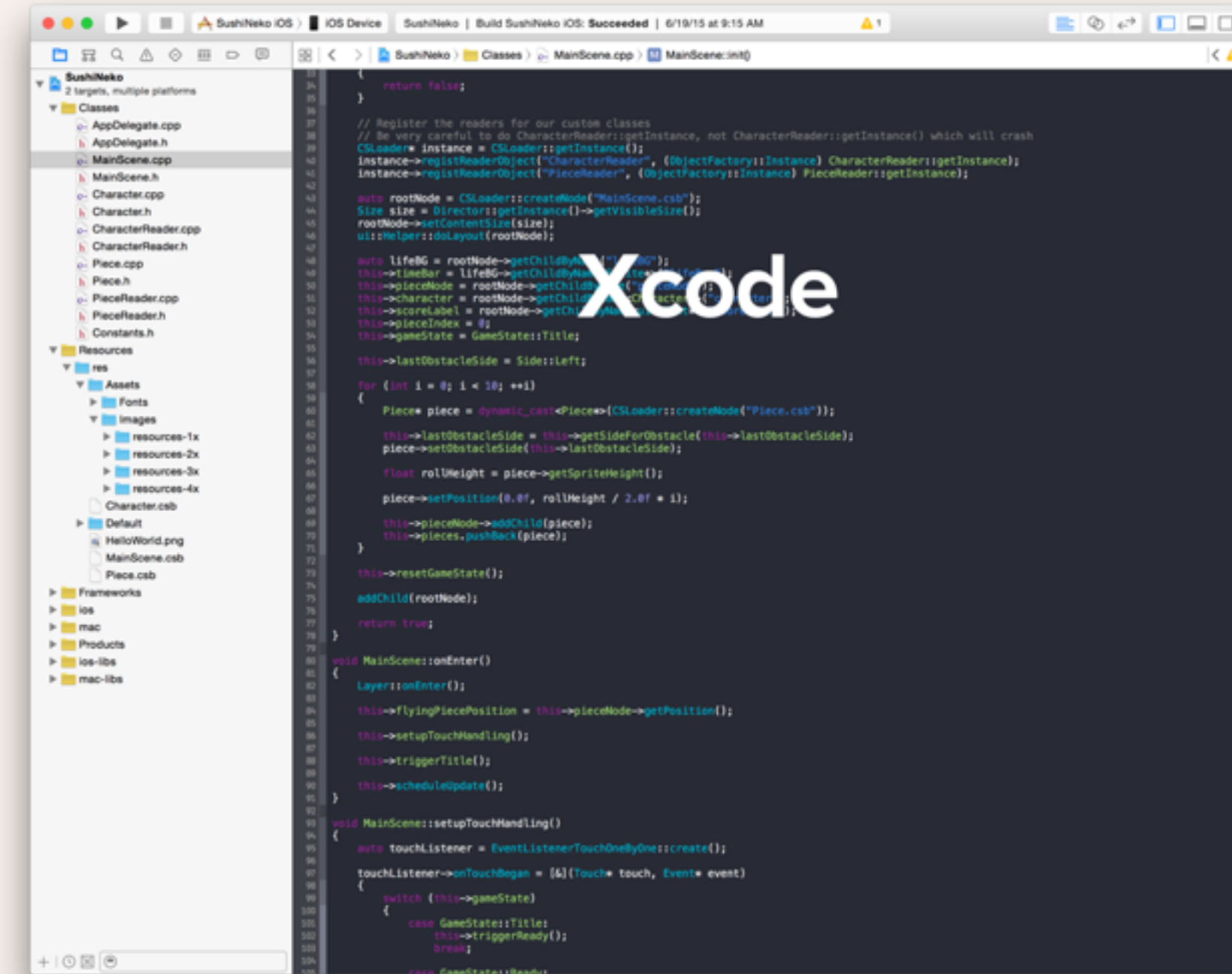
COCOS STUDIO

Cocos2D-X visual game editor

Closed source :(

Reduces the amount of code



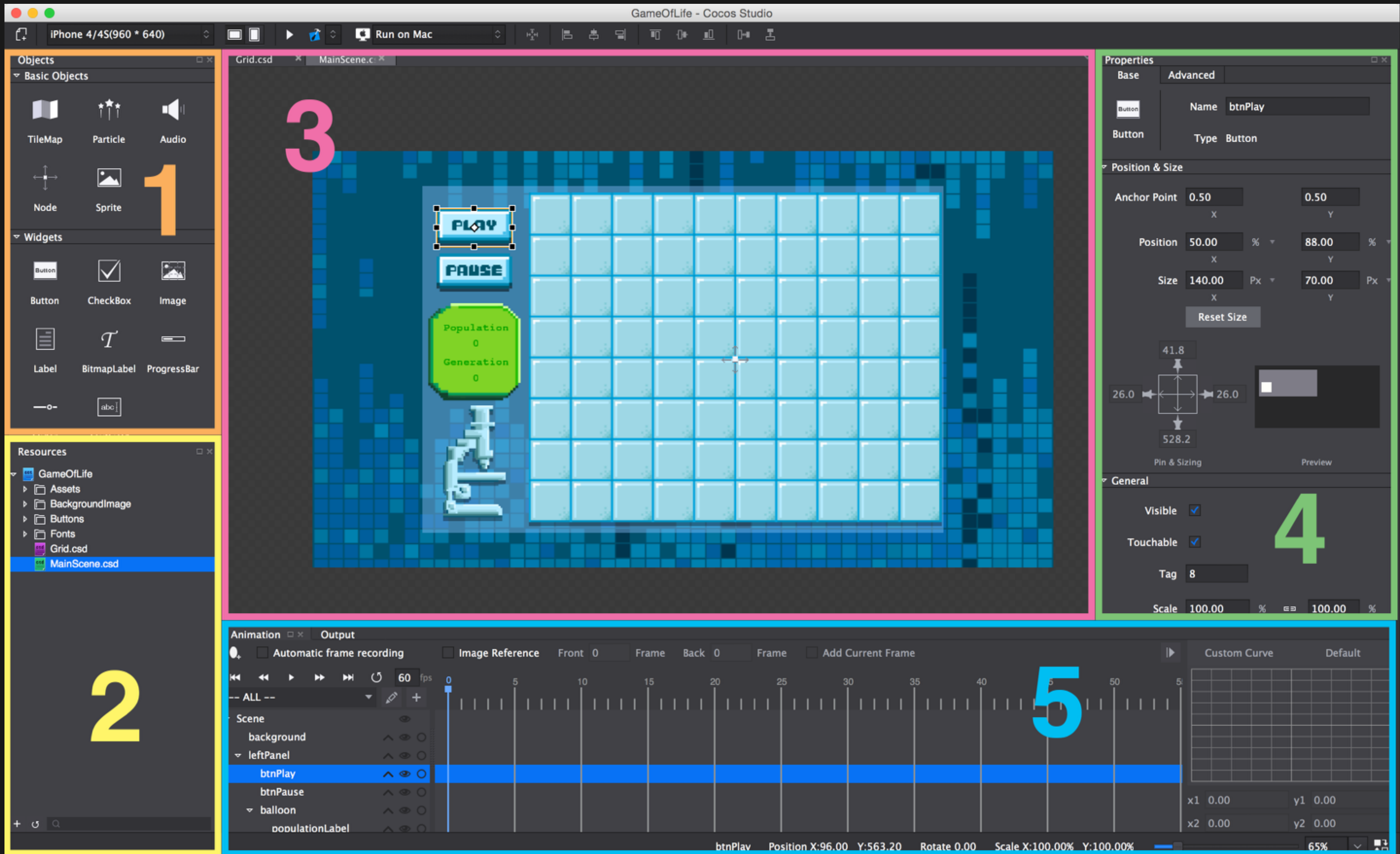


.csd file
.csd file
resources



.csb file
.csb file
resources

CSLoader
Scene
Node




CSD TYPES


New File


File Name


Scene


Progress Type











Scene

Layer

Node

SpriteSheet

3D Scene

Description

A scene is a container that contains game elements. Scenes can be used to create levels or menus.

Size

640

Px

960

Px

Width

Height

Cancel

New

CODE CONNECTIONS

Custom Classes - Tell Cocos Studio to initialize a class that you wrote

Name - Give your Cocos Studio objects names, so that you can access them in code

<http://cocos2d-x.org/programmersguide/>

<http://www.cocos2d-x.org/reference/native-cpp/V3.6/index.html>



MAKE
SCHOOL