

Objects in Cocos2d

Scenes (cocos2d::Scene)

- Scenes define different modes of your game. You might have a main menu in one scene and the actual game in another. Cocos2d-X makes it really easy to run, swap, push, pop and transition between scenes.
- Every Cocos2d-X app needs at least one scene. Scenes are invisible nodes in Cocos2d-X. You can't actually see a scene, but you can see most of its children, like sprites, buttons, backgrounds, etc.

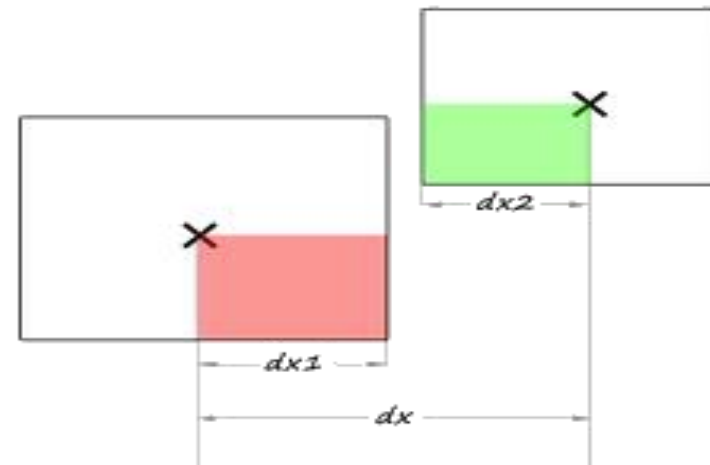
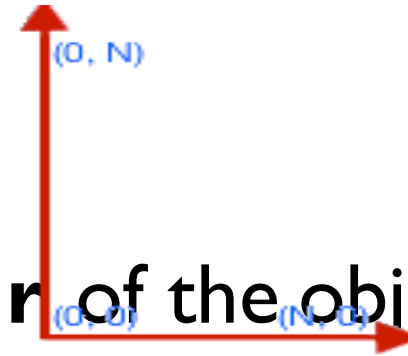


Example code: creating a scene

```
bool SpriteDragger::applicationDidFinishLaunching()  
{  
    // create a generic scene  
    auto scene = Scene::create();  
  
    // add the layer  
    auto layer = new MyLayer;  
    scene->addChild(layer, 1);  
    layer->release(); // addChild() retained so we release  
  
    // run the first scene  
    Director::getInstance()->runWithScene(scene);  
    return true;  
}
```

A quick note on coordinates

- The origin is the **lower left-hand corner** of the screen
- The **position** of an object is the **center** of the object



Layers (cocos2d::Layer)

- A **layer** (*CCLayer*) is where a user interacts with Cocos2d objects
- A layer handles touch and accelerometer events
- A layer contains other layers and sprites

Layers (cocos2d::Layer)

Layers are another type of invisible Cocos2d-Xnode. You can have multiple layers in a scene, though for the most part you'll only need one layer.



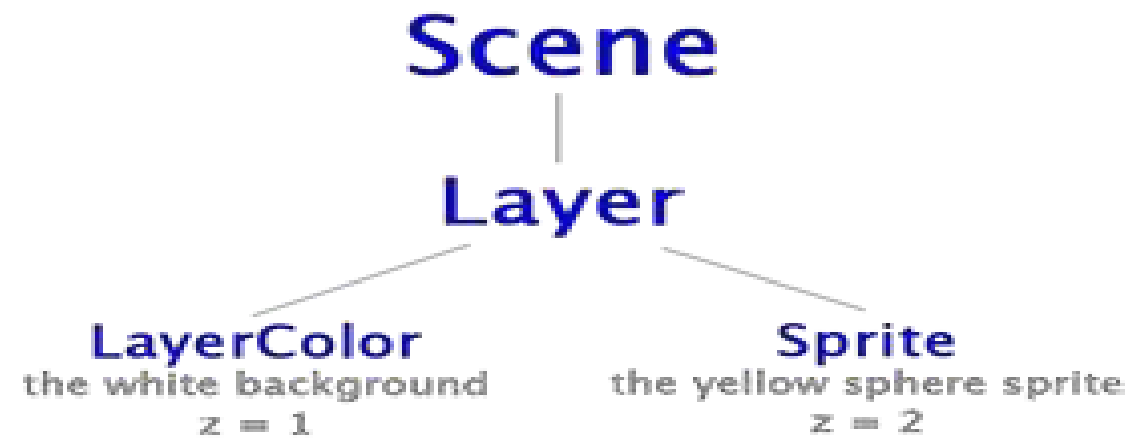


Nodes (cocos2d::Node)

- Layers, as well as scenes, sprites and everything else in Cocos2d-X, derive from Node.
- Nodes give every element in Cocos2d-X a common functionality. All nodes have a position, for example, which can be set in order to move the node. Nodes also have anchor points, cameras, z order, rotation, scaling, flipping, skewing and basic on/off visibility.
- A node has one possible parent and many possible children. This organizes all Cocos2d-X elements into a hierarchy. The scene is at the top of the hierarchy, then comes the layer, then sprites, backgrounds, etc.

Nodes (cocos2d::Node)

- Since a node's position is relative to its parent, moving a parent node (such as a layer) moves all of the children nodes as well. This helps to simplify the mathematics for your game. A complex system can be thought of in smaller, more simple chunks.
- You can think of nodes like the leaves, branches, and root of a tree. The root is a scene. The branches and leaves are layers and nodes.



Sprites

- A sprite (CCSprite) is an active object in your game
 - 2 dimensional object
 - Has an image (not generated)
 - Can be animated, touched, moved, etc.

Sprites and Anchor Points

- Once you have your images, you have to figure out where you want to place it. Note that in Cocos2D the bottom left corner of the screen has coordinates of (0,0) and the x and y values increase as you move to the upper right.
- When you set the position of an object, the position is relative to the center of the sprite you are adding. So if you wanted your player sprite to be aligned with the left edge of the screen horizontally, and vertically centered:
- For the x coordinate of the position, you'd set it to $[\text{player sprite's width}]/2$.
- For the y coordinate of the position, you'd set it to $[\text{window height}]/2$.

Sprites and Anchor Points

`(0, winSize.height)`

`(winSize.width, winSize.height)`

player anchor point



`x coordinate: player.contentSize.width / 2`
`y coordinate: winSize.height / 2`

`(0, 0)`
`50, 0`

`(winSize.width, 0)`



Sprites and Anchor Points

- The sprite is also given an anchor point which nudges the sprite upward.
- You can think of an anchor point like the nail which sticks a piece of rectangular wood onto a wall. After nailing the rectangle to the wall, you can spin it around on the nail. Imagine moving the nail to a different position within the wood. You can still rotate the rectangle, but it will rotate around a different center point.
- Sprites are given a default anchor point of $(0.5, 0.5)$, which represents the center of the sprite. To move the anchor point to the lower left of the sprite, you use an anchor point of $(0.0, 0.0)$. The upper right is $(1.0, 1.0)$.

Cocos2d-X Actions

- Actions are an easy way to kick off a bunch of transformations, animations or events. Because Cocos2d-X handles all the timing and updating of the action, it is completely hassle-free. You just run the action and forget all about it.
- You can Sequence actions so they happen one after the other. You can Spawn actions so they all happen at once. You can ScaleBy, FadeBy, RotateBy and a whole lot more. You can even call functions with CallFunc



Example code:Cocos2d-X Actions

- `// stop any existing actions and reset the scale`
- `this->sprite->stopAllActions(); this->sprite->setScale(1.0f);`
- `// animate letting go of the sprite`
- `this->sprite->runAction(Sequence::create(ScaleBy::create(0.125f, 1.111f),
ScaleBy::create(0.125f, 0.9f), nullptr`
- `));`