概述

2014-2-24

Cocos2d-x还有一种动作就是动画类CCAnimate，要实现CCAnimate，还需要定义CCAnimation等类。

1. CCAnimationCache类是一个单例，用于缓冲所有的动画和动画帧，其使用如下所示：

CCAnimationCache::shareAnimationCache()->addAnimation(animation, “dance”);

CCAnimationCache \*animCache = CCAnimationCache::shareAnimationCache();

CCAnimation \*normal = animCache->animationByName(“dance”);

首先通过shareAnimationCache函数可以获得动画缓冲CCAnimationCache，通过addAnimation函数往动画缓存中加入动画，并给动画命名，然后通过动画缓存调用animationByName函数传入动画名就可以获得相应的动画。

2）和精灵帧CCSpriteFrame类似，动画帧CCAnimationFrame是单张的图片，也可以通过精灵帧定义。使用如下：

CCSpriteFrame \*frame = (CCSpriteFrme\*)pObj;

CCAnimationFrame\* animFrame = new CCAnimationFrame();

animFrame->initWithSpriteFrame(frame, 1, NULL);

这里initWithSpriteFrame函数就是通过精灵帧CCSpriteFrame初始化动画帧CCAnimationFrame的，第一参数是精灵帧CCSpriteFrame，第二个参数是动画帧的延迟时间（有时动画的某一帧需要延迟，使用该参数定义就可以），最后一个参数是CCDictionary类型的信息。

3 ）CCAnimation就是动画，存储一个动画动作需要的所有帧，可以通过CCAnimation动画定义CCAnimate动画动作，代码如下：

sprite->runAction(CCRepeateForever::create(CCAnimate::create(animation)));定义之后传入runAction就可以执行动画。

AnimationCache::AnimationCache()

{

CCSpriteFrameCache \*frameCache = CCSpriteFrameCache::sharedSpriteFrameCache();

frameCache->addSpriteFramesWithFile("animations/grossini.plist");

frameCache->addSpriteFramesWithFile("animations/grossini\_gray.plist");

frameCache->addSpriteFramesWithFile("animations/grossini\_blue.plist");

//

// create animation "dance"

//

CCArray\* animFrames = CCArray::createWithCapacity(15);

char str[100] = {0};

for(int i = 1; i < 15; i++)

{

sprintf(str, "grossini\_dance\_%02d.png",i);

CCSpriteFrame \*frame = frameCache->spriteFrameByName(str);

animFrames->addObject(frame);

}

CCAnimation \*animation = CCAnimation::createWithSpriteFrames(animFrames, 0.2f);

// Add an animation to the Cache

CCAnimationCache::sharedAnimationCache()->addAnimation(animation, "dance");

//

// create animation "dance gray"

//

animFrames->removeAllObjects();

for(int i = 1; i < 15; i++)

{

sprintf(str, "grossini\_dance\_gray\_%02d.png",i);

CCSpriteFrame \*frame = frameCache->spriteFrameByName(str);

animFrames->addObject(frame);

}

animation = CCAnimation::createWithSpriteFrames(animFrames, 0.2f);

// Add an animation to the Cache

CCAnimationCache::sharedAnimationCache()->addAnimation(animation, "dance\_gray");

//

// create animation "dance blue"

//

animFrames->removeAllObjects();

for(int i = 1; i < 4; i++)

{

sprintf(str, "grossini\_blue\_%02d.png",i);

CCSpriteFrame \*frame = frameCache->spriteFrameByName(str);

animFrames->addObject(frame);

}

animation = CCAnimation::createWithSpriteFrames(animFrames, 0.2f);

// Add an animation to the Cache

CCAnimationCache::sharedAnimationCache()->addAnimation(animation, "dance\_blue");

CCAnimationCache \*animCache = CCAnimationCache::sharedAnimationCache();

CCAnimation \*normal = animCache->animationByName("dance");

normal->setRestoreOriginalFrame(true);

CCAnimation \*dance\_grey = animCache->animationByName("dance\_gray");

dance\_grey->setRestoreOriginalFrame(true);

CCAnimation \*dance\_blue = animCache->animationByName("dance\_blue");

dance\_blue->setRestoreOriginalFrame(true);

CCAnimate \*animN = CCAnimate::create(normal);

CCAnimate \*animG = CCAnimate::create(dance\_grey);

CCAnimate \*animB = CCAnimate::create(dance\_blue);

CCSequence \*seq = CCSequence::create(animN, animG, animB, NULL);

// create an sprite without texture

CCSprite \*grossini = CCSprite::create();

CCSpriteFrame \*frame = frameCache->spriteFrameByName("grossini\_dance\_01.png");

grossini->setDisplayFrame(frame);

CCSize winSize = CCDirector::sharedDirector()->getWinSize();

grossini->setPosition(ccp(winSize.width/2, winSize.height/2));

addChild(grossini);

// run the animation

grossini->runAction(seq);

}

首先分别定义三个动画传入动画缓存中，通过动画缓存获得三个动画，然后定义动画动作并运行。