

Введение в Cocos2d 2



Тарас Палиенко

Содержание

- Основные требования
- Установка и настройка
- Обзор тестов и примеров
- Основы Cocos2d
- Основные действия (Actions)
- Атласы текстур и Zworptex

Основные требования

- Знание Objective-C на начальном уровне
- Понимание основ управления памятью в Objective-C
- Компьютер с Mac OS X
- Устройство с поддержкой OpenGL ES 2.0

Установка и настройка

- Загрузка
- Установка шаблонов
- Конвертация в ARC (опционально)

Загрузка

Загружаем cocos2d 2.0 по следующей ссылке:

<http://www.cocos2d-iphone.org/download>

Установка

- Открываем Terminal
- Переходим в папку с cocos2d: `cd cocos2d-iphone-2.0`
- Запускаем установку шаблонов: `./install-templates.sh -f`
- Теперь можно создать проект: Xcode - New - New Project - cocos2d v2.x, и запустить его:



Конвертация в ARC

- Собираем исходники cocos2d как Static Library без ARC
- Подключаем к проекту
- Конвертируем проект в ARC
- ???????
- PROFIT!

<http://www.learn-cocos2d.com/2012/04/enabling-arc-cocos2d-project-howto-stepbystep-tutorialguide/>

Обзор тестов и примеров

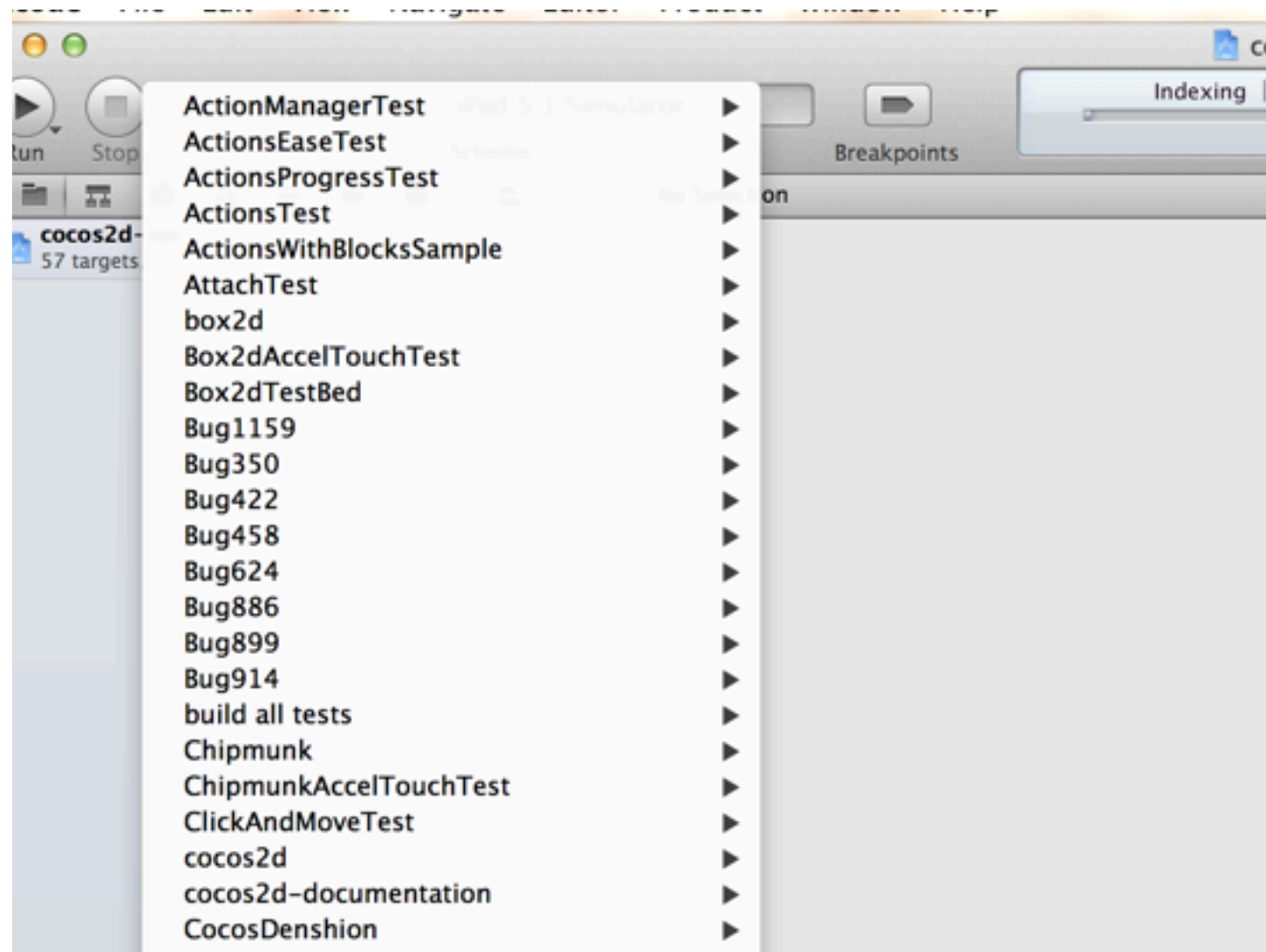
В комплекте с cocos2d идет этот большой и замечательный проект:



`cocos2d-ios.xcodeproj`

Выбор теста

Все тесты включены в проект как target'ы



Пример теста



ОснОвы Cocos2d

- Простой пример кода
- Как все устроено
- Основные классы и аналогии с UIKit

```
CCSprite *seeker1;  
CCSprite *cocosGuy;
```

```
// create and initialize our seeker sprite, and add it to this layer  
seeker1 = [CCSprite spriteWithFile: @"seeker.png"];  
seeker1.position = ccp( 50, 100 );  
[self addChild:seeker1];
```

```
// do the same for our cocos2d guy, reusing the app icon as its image  
cocosGuy = [CCSprite spriteWithFile: @"Icon.png"];  
cocosGuy.position = ccp( 200, 300 );  
[self addChild:cocosGuy];
```



Добавим движения

Запланируем метод на каждый фрейм:

```
// schedule a repeating callback on every frame  
[self schedule:@selector(nextFrame:)];
```

Перемещаем спрайт seeker1 горизонтально:

```
- (void) nextFrame:(ccTime)dt {  
    seeker1.position = ccp( seeker1.position.x + 100*dt, seeker1.position.y );  
    if (seeker1.position.x > 480+32) {  
        seeker1.position = ccp( -32, seeker1.position.y );  
    }  
}
```

Добавим обработку касаний

Включаем touch для CCLayer:

```
self.isTouchEnabled = YES;
```

Перемещаем спрайт cocosGuy в точку касания:

```
- (void)ccTouchEnded:(UITouch *)touch withEvent:(UIEvent *)event {  
    CGPoint location = [self convertTouchToNodeSpace: touch];  
  
    [cocosGuy stopAllActions];  
    [cocosGuy runAction: [CCMoveTo actionWithDuration:1 position:location]];  
}
```

<https://github.com/tar500/cocos2d-seeker>

Как все устроено

- Игра состоит из сцен (CCScene)
- Сцены состоят из слоев (CCLayer)
- Слои отлавливают касания (touches) и содержат спрайты (CCSprite)
- Сценами управляет CCDirector

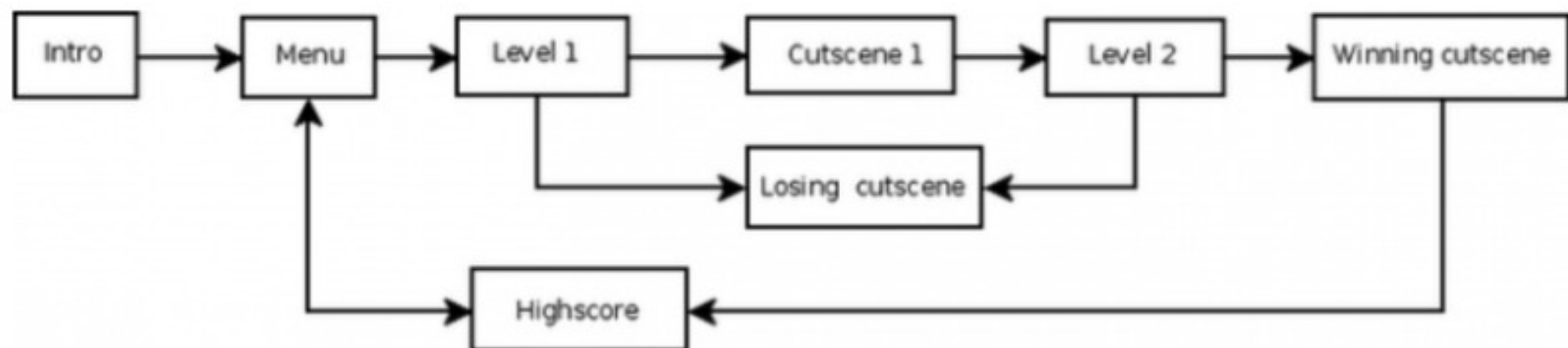
Сцены (CCScene)

- Это отдельные экраны в приложении
- Между сценами можно делать анимированные переходы (см. TransitionsTests в тестовом проекте)

Запуск первой сцены:

```
[[CCDirector sharedDirector] runWithScene: [HelloWorld scene]];
```

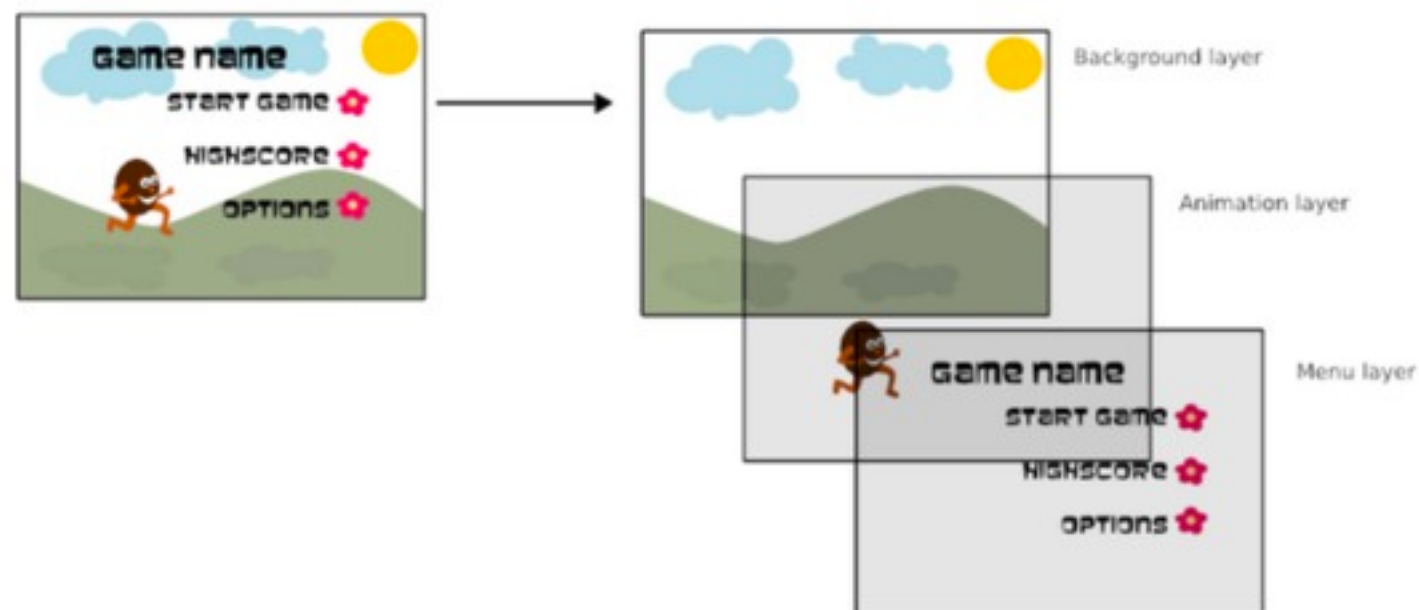
Возможная иерархия сцен в игре:



Слои (CCLayer)

- Занимают весь экран
- Принимают события касаний и акселерометра
- Могут содержать в себе другие слои, спрайты и т.д.

Сцена, состоящая из трех слоев:
фона, слоя анимации и слоя меню:



Спрайты (CCSprite)



Стандартные суффиксы для Retina и iPad графики:
-hd, -ipad, -ipadhd

Размеры текстуры всегда равны степени двойки, например: текстура для картинки 257 x 100 будет размером 512 x 128. Поэтому выгодно использовать атласы текстур. (об этом немножко дальше).

По умолчанию включен antialiasing.

CCDirector

- Управляет переходами между сценами
- Приостанавливает и возобновляет сцены
- Инициализирует и настраивает OpenGL ES

CCNode

- Похож на UIView
- Базовый класс для большинства объектов cocos2d: CCLayer, CCSprite, CCScene и т.д.
- Сам ничего отображать не может
- Имеет много свойств: размеры, прозрачность, OpenGL X position, увеличение и т.д.

Аналогии с UIKit

CCDirector	UINavigationController
pushScene: popScene: replaceScene:	pushViewController:, popViewController:
CCNode	UIView
addChild:	addSubview
CCLayer (базовый класс: CCNode)	UIViewController
onEnter:, onExit: ccTouchesBegan, ccTouchesMoved, ...	viewWillAppear:, viewWillDisappear: touchesBegan, touchesMoved, ...
CCSprite (базовый класс: CCNode)	UIImageView

Действия (Actions)

Пример

Перемещаем спрайт на 50 пикселей вправо и на 10 пикселей вверх за 2 секунды:

```
[sprite runAction:  
[CCMoveBy actionWithDuration:2 position: ccp(50,10)]];
```

Остановка / Возобновление действий

```
# Остановка действий  
[sprite1 pauseAllActions] ;  
  
# Возобновление действий  
[sprite1 resumeAllActions] ;
```

Список базовых действий

- ПОЗИЦИЯ
 - [CCMoveBy](#)
 - [CCMoveTo](#)
 - [CCJumpBy](#)
 - [CCJumpTo](#)
 - [CCBezierBy](#)
 - [CCBezierTo](#)
 - [CCPlace](#)
- увеличение
 - [CCScaleBy](#)
 - [CCScaleTo](#)
- поворот
 - [CCRotateBy](#)
 - [CCRotateTo](#)
- ВИДИМОСТЬ
 - [CCShow](#)
 - [CCHide](#)
 - [CCBlink](#)
 - [CCToggleVisibility](#)
- прозрачность
 - [CCFadeIn](#)
 - [CCFadeOut](#)
 - [CCFadeTo](#)
- ЦВЕТ
 - [CCTintBy](#)
 - [CCTintTo](#)

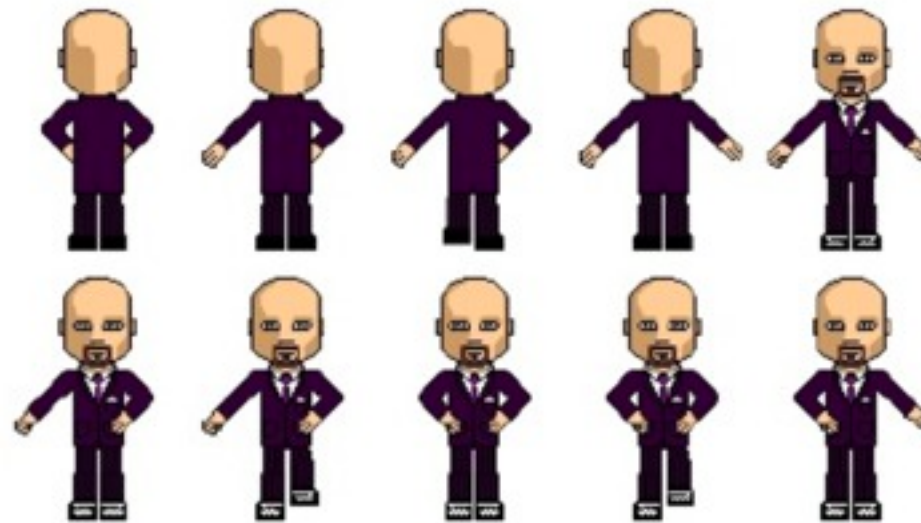
By - изменить значение
на заданную величину
(относительно)

To - установить значение
в заданную величину
(абсолютно)

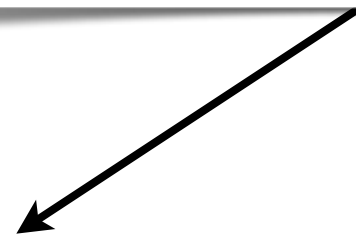
Атлас текстур (Sprite Sheet)



Zwoptex



+ .plist



```
[[CCSpriteFrameCache sharedSpriteFrameCache] addSpriteFramesWithFile:@"spritesheet.plist"];
```

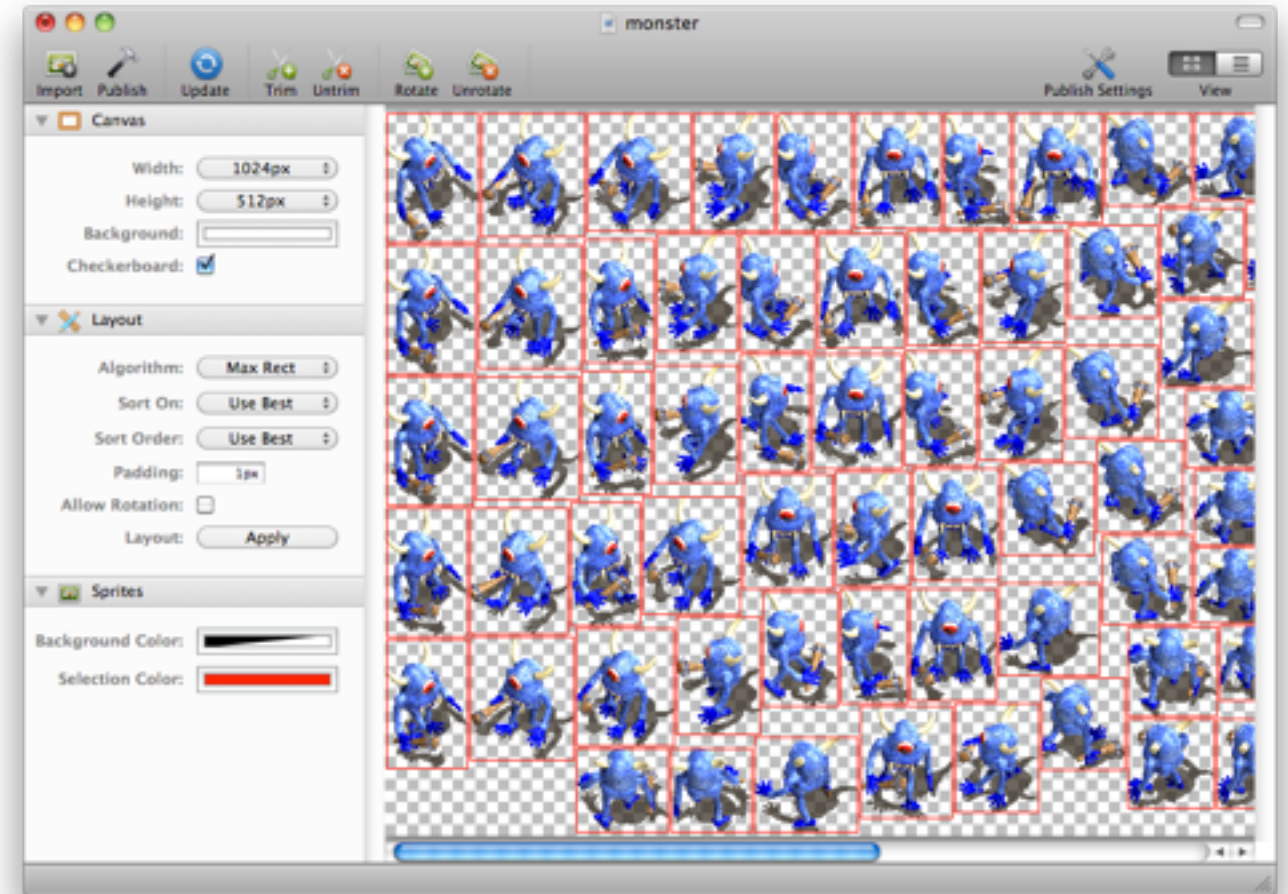


```
CCSprite *sprite1 = [CCSprite spriteWithSpriteFrameName:@"sprite1.png"];
```

Zwoptex

На входе - много картинок

На выходе - одна большая картинка с прилагающимся .plist файлом.



При сохранении не забываем добавить retina суффиксы и масштаб, например: -ipadhd 100%, -ipad 50%.

<http://www.zwopple.com/zwoptex/>

Преимущества Sprite Sheet

- Размеры текстуры всегда равны степени двойки, например: текстура для картинки 257 x 100 будет размером 512 x 128. Итог: используется много лишней памяти. Поэтому выгодно использовать атласы текстур, комбинируя текстуры разных размеров и пропорций в одну большую, как на предыдущем слайде.
- Одна большая картинка также занимает меньше места на диска чем много маленьких.
- В комбинации с CCSpriteBatchNode дают весомый прирост в производительности.

CCSpriteBatchNode

Используется для увеличения производительности при отрисовке большого количества объектов.

Инициализируется с файлом .plist, который идет в комплекте к sprite sheet, после создания этого sprite sheet через zwortex.

Содержит объекты CCSprite, инициализированные вот так:

```
CCSprite *sprite1 = [CCSprite spriteWithSpriteFrameName:@"sprite1.png"];
```

Производительность увеличивается за счет того, что все спрайты на CCSpriteBatchNode отрисовываются за один OpenGL вызов, в отличие от того, если бы спрайты были помещены НЕ на CCSpriteBatchNode.

Недостаток: может содержать спрайты только с той текстуры, к которой привязан .plist файл (с которым объект был инициализирован).

Система частиц

(смотрите ParticleTest в [cocos2d-ios.xcodeproj](#), идущем в комплекте с cocos2d)

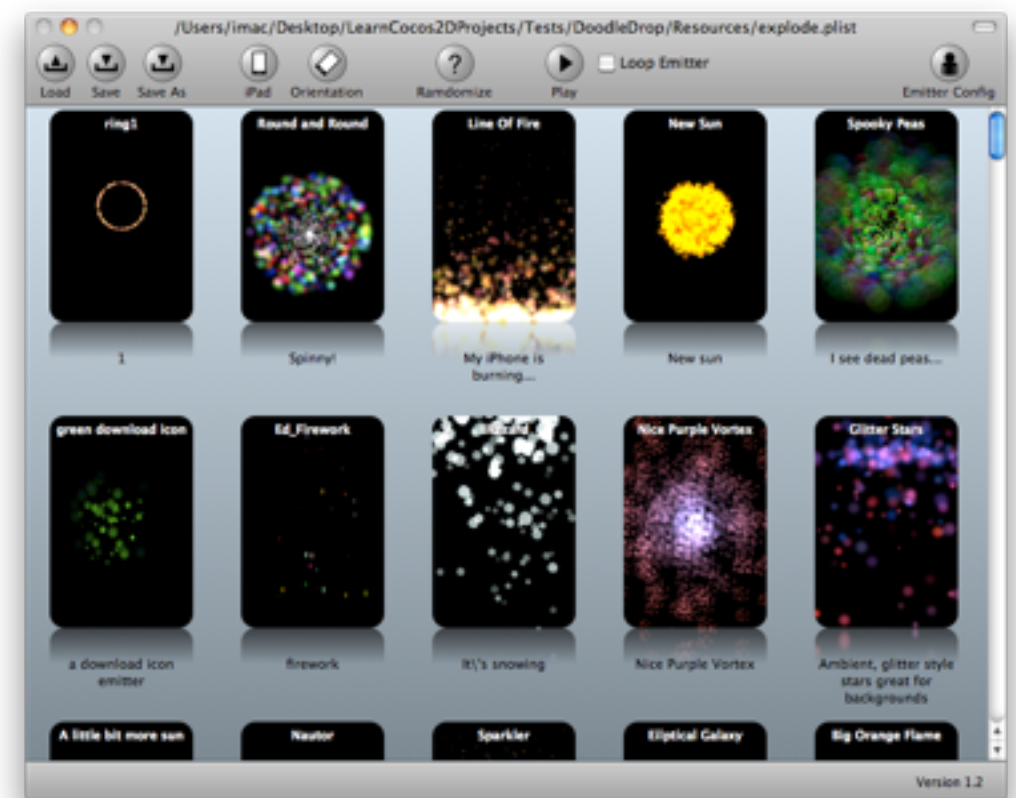
```
CCParticleSystemQuad * particleSystem = [[CCParticleSystemQuad alloc]  
initWithTotalParticles:numberOfParticles];
```

Некоторые свойства

- `startSize` Начальный размер частиц
- `startSizeVar` Вариация начального размера частиц
- `endSize`
- `endSizeVar`
- `startColor` (a `ccColor4F`) Например, `ccc4f(1.0f, 0.0f, 0.0f, 1.0f)` – красный
- `startColorVar` (a `ccColor4F`)
- `endColor` (a `ccColor4F`)
- `endColorVar` (a `ccColor4F`)
- `life`: Время жизни частиц
- `lifeVar`:
- `angle`: (a float). Начальный угол испускания частиц в градусах
- `angleVar`
- `position`: (CGPoint)
- `posVar`
- `centerOfGravity` (CGPoint)

Particle Designer.

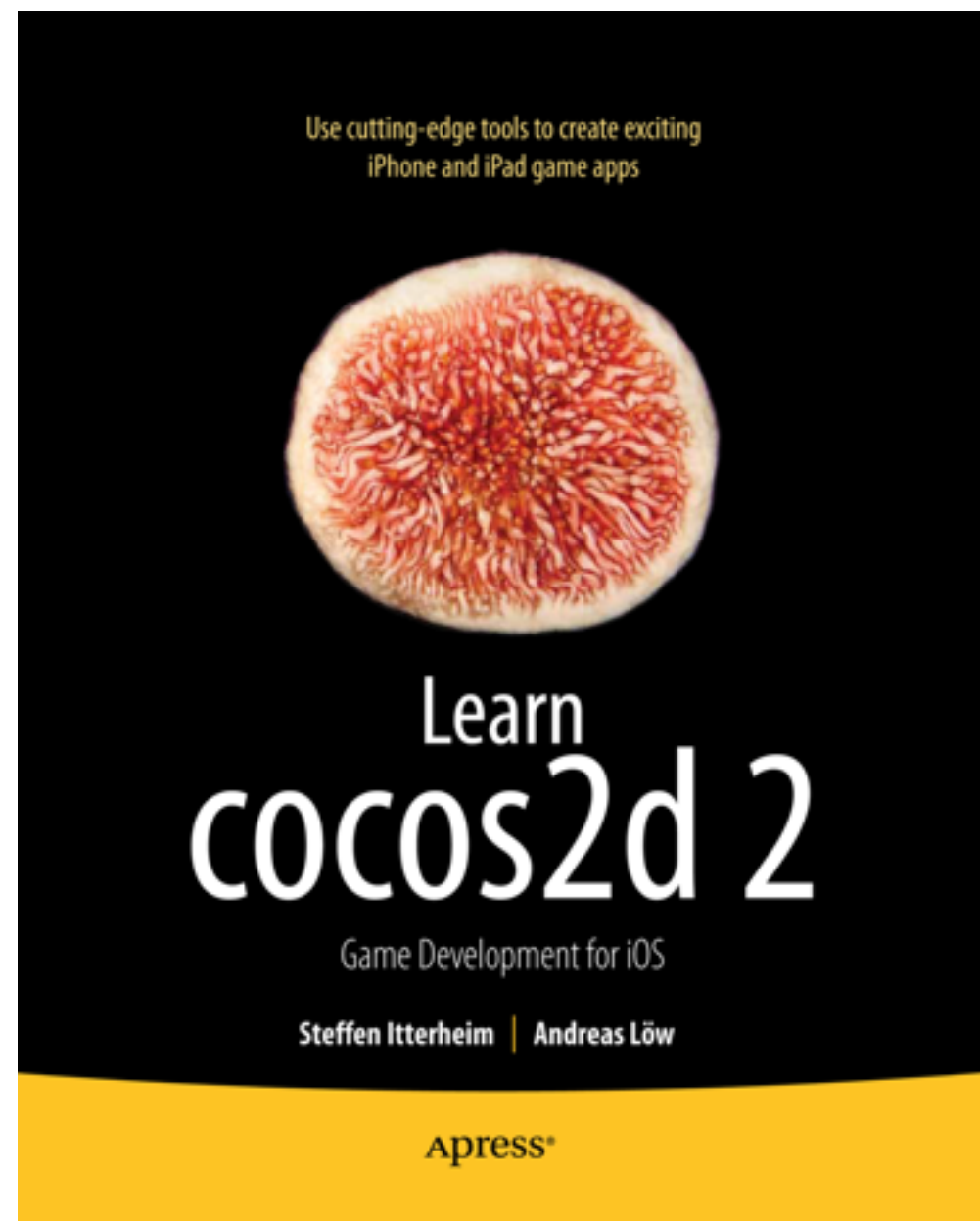
Must have.



<http://www.7lsquared.com/particle designer>

Литература

Learn cocos2d 2 Game Development for iOS, Steffen Itterheim



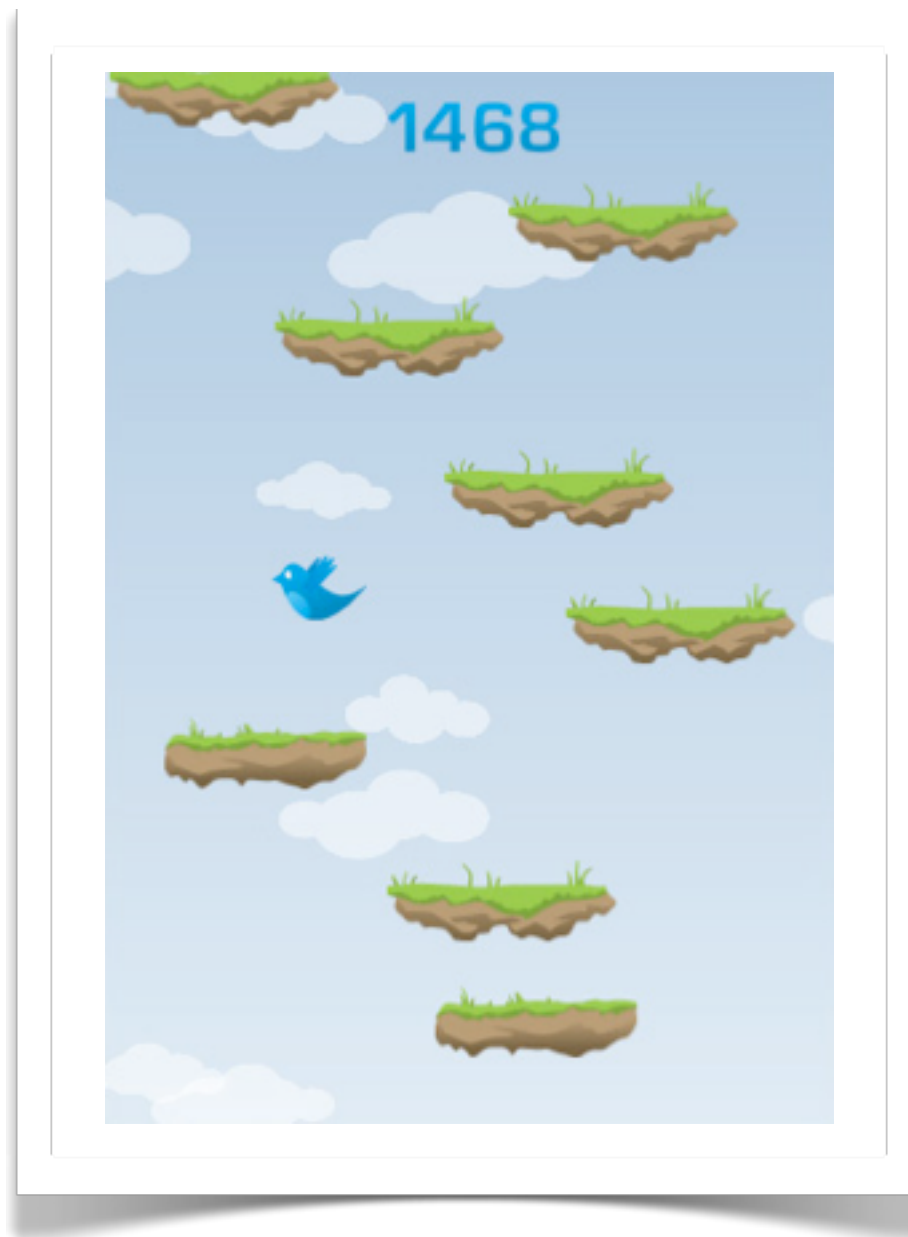
Ссылки

www.cocos2d-iphone.org

Особо рекомендуется к прочтению раздел “Documentation”, большая часть материалов взята именно с него.

Tweejump.

Простая игра на cocos2d.



<https://github.com/haqu/tweejump>