# 掉落的苹果——b2Body刚体

上一节中，我们认识了Box2D世界，学习了它的创建方法。今天我们一起来扮演牛顿，认识一下Box2D世界中掉落的苹果。

在开始之前，我们想先说明两个内容：

1. Box2D中的计量单位是米  
   在Box2D中的计量单位是米m，而不是Flash中的像素px，在布置坐标时，要进行一个转换，1米=30个像素。所以Box2D中(a,b)点对于Flash中的(a\*30,b\*30)的位置，或者说Flash中的(c,d)位置对应Box2D中的(c/30,d/30)位置。
2. Box2D用b2DebugDraw进行模拟调试  
   Box2D是一个物理引擎，不会向Flash显示列表中添加任何显示对象。不过Box2D中有一个b2DebugDraw类，可以绑定一个显示对象，进行模拟调试。至于如何在Box2D中添加显示对象，在以后的学习中，我们再深入讨论。

b2DebugDraw的用法：

* 创建一个空的Sprite对象debugSprite，并添加到舞台中
* 创建一个b2DebugDraw对象debugDraw，并设置它的m\_sprite属性值为debugSprite.
* 用world的SetDebugDraw方法，绑定debugDraw

1. World的更新：

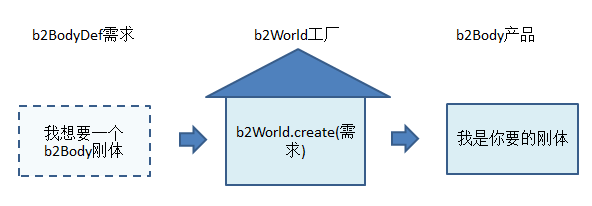
Box2D的和Flash一样，需要实时更新，Flash中有ENTER\_FRAME，b2world有step，其参数一表示模拟的时长，二表示限制碰撞后检测程式运行的次数，防止死循环。

刚体与DisplayObject的区别

掉落的苹果实际上是Box2D里的一个刚体b2Body，当然可以把他理解成Flash中的sprite，不过：

1. b2Body
   1. 它是Box2D中的一个物理模拟对象，不是可显示对象，但是可以通过b2DebugDraw来进行调试模拟
   2. 它的计量单位是米m
   3. 它通过b2World.createBody()来创建
2. Sprite
   1. 它是显示对象，通过addChild添加即可显示
   2. 它的计量单位是像素px
   3. 通过new来创建，然后addChild添加后直接显示在舞台上

刚体创建的过程，与AS中new一个对象，addChild到舞台中显示比起来要复制很多，刚体的创建可以看做是一个工厂模式，根据你的需求(b2BodyDef)生产指定的刚体



刚体的创建过程可以分为两个步骤：

1. 创建刚体需求b2BodyDef。

var bodyRequest:b2BodyDef = new b2BodyDef();

在这个需求中，我们可以包括下面的内容：

* 1. position：刚体的坐标位置，单位是米m
  2. angle：刚体的角度
  3. 其他，在以后的学习中我们会陆续介绍

1. b2World工厂用createBody方法创建刚体产品，并自动将其添加到Box2D世界中，然后返回该刚体。

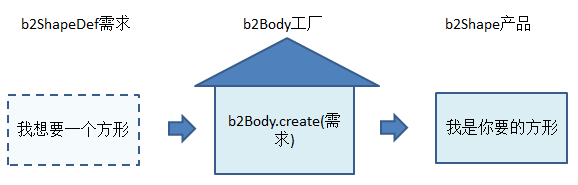
body=world.CreateBody(bodyRequest);

定义刚体形状

到目前位置，我们已经成功添加了一个刚体。

讲个题外话，今天我同时的老婆生了，听到这个消息，你会问的第一个问题是什么？”男孩，女孩？”答对了！

把刚体当成我们的孩子吧。创建一个刚体后，你应该问什么？”方的，圆的？”(这孩子好挫啊！)非常正确。刚体的形状不是与生俱来的，需要用一个b2Shape的子类对象指定。刚体形状的创建过程同样也是一个工厂模式。



这个创建过程与b2Body的创建过程是一样的。我们的形状需求b2ShapeDef可以报考下面的内容：

1. density：质量
2. friction：表面摩擦力
3. restitution：表面张力，这个值越大，刚体越硬
4. SetAsBox：设置刚体为矩形
5. 其他，在以后的学习中我们会陆续介绍

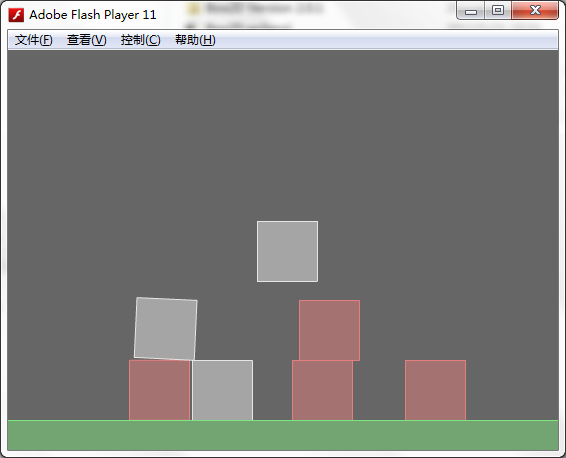
好了创建b2Body刚体的完整过程就讲完了。我们在总结一下：

1. 创建刚体需求b2BodyDef
2. Box2D世界工厂根据需求用createBody方法创建刚体
3. 创建刚体形状需求b2ShapeDef的子类对象
4. B2Body工厂根据需求用createShape方法创建形状

实例练习

掌握了上面的内容，我相信完成下面的这个练习就不是什么难事了。

点击舞台的任意位置创建一个方形刚体。



代码

package

{

import Box2D.Collision.b2AABB;

import Box2D.Collision.Shapes.b2PolygonDef;

import Box2D.Common.Math.b2Vec2;

import Box2D.Dynamics.b2Body;

import Box2D.Dynamics.b2BodyDef;

import Box2D.Dynamics.b2DebugDraw;

import Box2D.Dynamics.b2World;

import flash.display.Sprite;

import flash.events.Event;

import flash.events.MouseEvent;

/\*\*

\* ...

\* @author ladeng6666

\*/

public class Main extends Sprite

{

private var world:b2World;

private var body:b2Body;

public function Main()

{

createWorld();

createDebug();

createBody(stage.stageWidth/2,0);

createGround();

addEventListener(Event.ENTER\_FRAME, loop);

stage.addEventListener(MouseEvent.MOUSE\_DOWN, onStageMouseDown);

}

private function onStageMouseDown(e:MouseEvent):void

{

createBody(mouseX,mouseY);

}

private function loop(e:Event):void

{

world.Step(1/30, 10);

}

private function createWorld():void

{

//1.创建一个环境

var environment:b2AABB = new b2AABB();

environment.lowerBound = new b2Vec2( -100, -100);

environment.upperBound = new b2Vec2(100, 100);

//2.声明重力

var gravity:b2Vec2 = new b2Vec2(0, 10);

//3.睡着的对象是否模拟

var doSleep:Boolean = true;

//4.创建b2World世界

world = new b2World(environment, gravity, doSleep);

}

private function createDebug():void

{

var debugSprite:Sprite = new Sprite();

addChild(debugSprite);

var debugDraw:b2DebugDraw = new b2DebugDraw();

debugDraw.m\_sprite = debugSprite;

debugDraw.m\_drawScale = 30.0;

debugDraw.m\_fillAlpha = 0.5;

debugDraw.m\_lineThickness = 1.0;

debugDraw.m\_drawFlags = b2DebugDraw.e\_shapeBit | b2DebugDraw.e\_jointBit;

world.SetDebugDraw(debugDraw);

}

private function createBody(posX:Number,posY:Number):void

{

//1.创建刚体需求b2BodyDef

var bodyRequest:b2BodyDef = new b2BodyDef();

bodyRequest.position.Set(posX/ 30, posY/30);//记得米和像素的转换关系

//2.Box2D世界工厂更具需求创建createBody()生产刚体

body=world.CreateBody(bodyRequest);

//3.创建敢提形状需求b2ShapeDef的子类

var shapeRequest:b2PolygonDef = new b2PolygonDef();

//详细说明我们的需求

shapeRequest.density = 3;

shapeRequest.friction = 0.3;

shapeRequest.restitution = 0.2;

shapeRequest.SetAsBox(1, 1);

//4.b2Body刚体工厂根据需求createShape生产形状

body.CreateShape(shapeRequest);

body.SetMassFromShapes();

}

private function createGround():void

{

//1.创建刚体需求b2BodyDef

var bodyRequest:b2BodyDef = new b2BodyDef();

bodyRequest.position.Set(stage.stageWidth/2 / 30, stage.stageHeight/30);//记得米和像素的转换关系

//2.Box2D世界工厂更具需求创建createBody()生产刚体

body=world.CreateBody(bodyRequest);

//3.创建敢提形状需求b2ShapeDef的子类

var shapeRequest:b2PolygonDef = new b2PolygonDef();

//详细说明我们的需求

shapeRequest.density = 0;

shapeRequest.friction = 0.3;

shapeRequest.restitution = 0.2;

shapeRequest.SetAsBox(stage.stageWidth/30, 1);

//4.b2Body刚体工厂根据需求createShape生产形状

body.CreateShape(shapeRequest);

body.SetMassFromShapes();

}

}

}