Flash移动开发优化技巧

杜增强

自我介绍

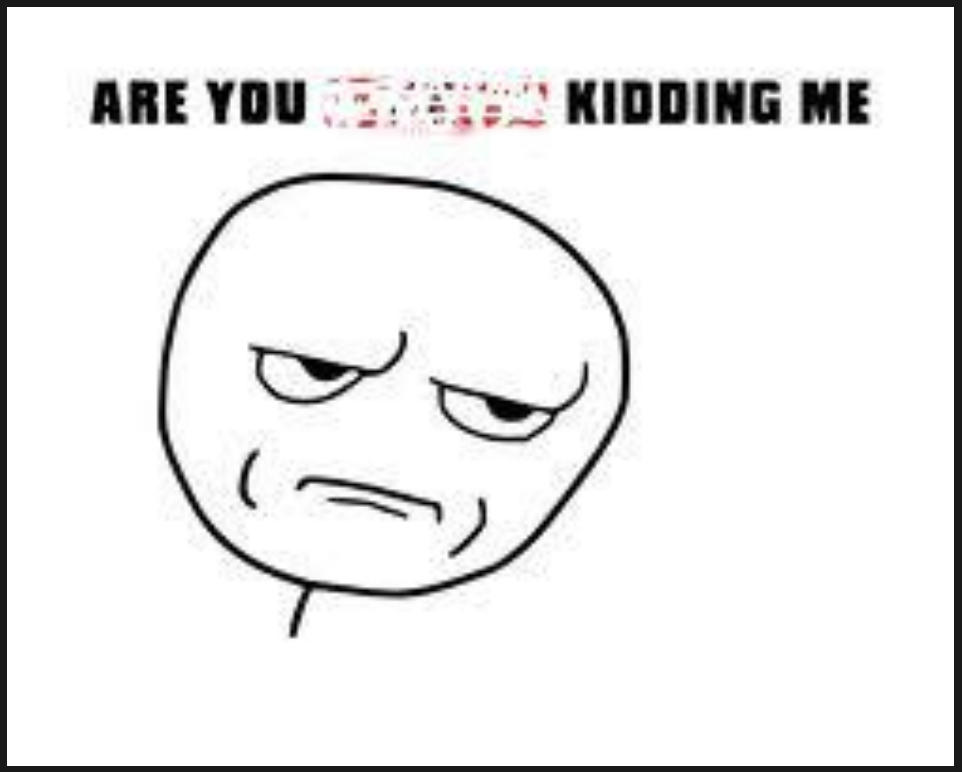
-  杜增强
- Adobe Flex Developer
- Adobe社区认证RIA专家
- 天地培训专家讲师
- www.duzengqiang.com
- Weibo.com/duzengqiang
- dzq@9ria.com

Are you “攻城师”?

“攻城师”的梦想

(注:本图片来自网络)

”攻城师”永远的痛



(注:本图片来自网络)

又是效率问题?!

理想和现实的差距

理想很丰满

现实很骨感

为什么会有”效率”问题?

- 组件使用(AS3基础)
- 算法效率(程序基础)
- 架构选择(设计模式)
- 引擎选择(新技术)

主题

- Object Pooling(对象池)
- Blitting(块传输)
- Stage3d Engine

对象池

- 原理
- 实现方法
- 使用示例
- 优点
- 缺点

原理

- 对象池的工作原理的核心有两点：使用和缓存，即对于那些被频繁使用的对象，在使用完后，不立即将它们释放，而是将它们缓存起来，以供后续的应用程序重复使用，从而减少创建对象和释放对象的次数，进而改善应用程序的性能。

实现方法

```
private static var pools:Dictionary = new Dictionary();

private static function getPool( type:Class ):Array

/**
public static function getObject( type:Class, ...parameters ):*
{
    var pool:Array = getPool( type );
    if( pool.length > 0 )
    {
        return pool.pop();
    }
    else
    {
        return construct( type, parameters );
    }
    return null;
}
/**
public static function returnObject( object:*, type:Class = null ):void
{
    if( !type )
    {
        var typeName:String = getQualifiedClassName( object );
        type = getDefinitionByName( typeName ) as Class;
    }
    var pool:Array = getPool( type );
    pool.push( object );
}
```

使用示例

从ObjectPool中获取一个SomeClass的实例：

```
var obj:SomeClass
= ObjectPool.getObject( SomeClass );
```

当你不需要一个对象，把它存进对象池中：

```
  ObjectPool.returnObject( obj );
```

优点

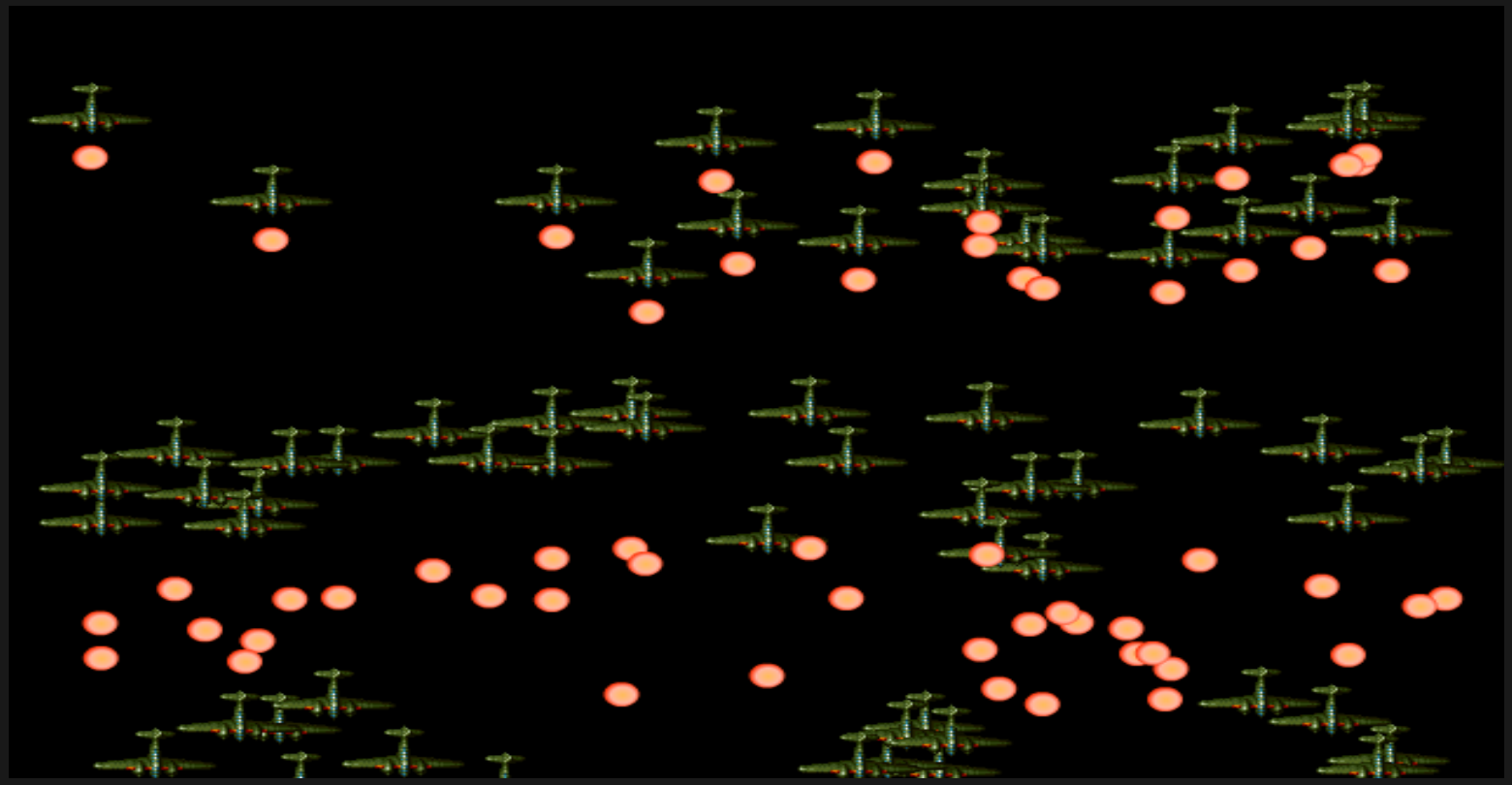
- 能快速取出对象节省了new对象所产生的cpu,时间的消耗。
- 能很好的控制内存的占用，使用时从对象池取出，使用完毕放回。中间不涉及到对象销毁创建，所以内存占用是定量的。

缺点

- 对象池同样消耗new同样对象所消耗的时间，对象池从创建到结束消耗定量的内存。
- 非提前初始化式内存池，缓存as值类型的对象如：Point反而产生更大的消耗。

适用情况

- 对象池只适合大量的对象需要被重复使用的情况。



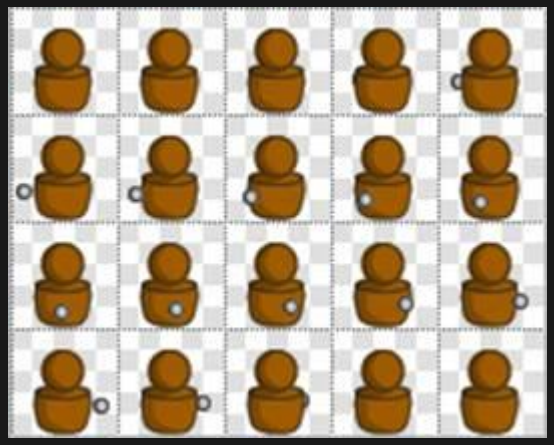
blitting(块传输)

- 原理
- Sprite Sheet
- 实现方法
- 工具

原理

- *Blitting*主张*bit-block (image) transfer(*图像基于位块传输)，是一种采用数张位图并合并成一张位图的技术。
- 在Flash Player中复制位图像素到一张渲染图中比分别地渲染每个DisplayObject更为快速。

sprite sheet



- 一个sprite sheet可以对不同大小的位图进行组合。将所有图形元素组装到一个大的图像文件中会减少加载时间（打开和读取一个包含100帧的较大文件比打开读取100个小文件更为快速）并提供压缩的好处。

实现方法

```
var bd:BitmapData;
var unitRect:Rectangle = new Rectangle(0, 0, _unitWidth, _unitHeight);
var i:int;
for(i = _startIndex;i<= _endIndex;i++)
{
    unitRect.x = i%columnCount*_unitWidth;
    unitRect.y = int(i/columnCount)*_unitHeight;
    bd = new BitmapData(_unitWidth, _unitHeight, _transparent, 0);
    bd.copyPixels(bigBD, unitRect, ORIGIN_POINT);
    bdv[i-_startIndex] = bd;
}
```

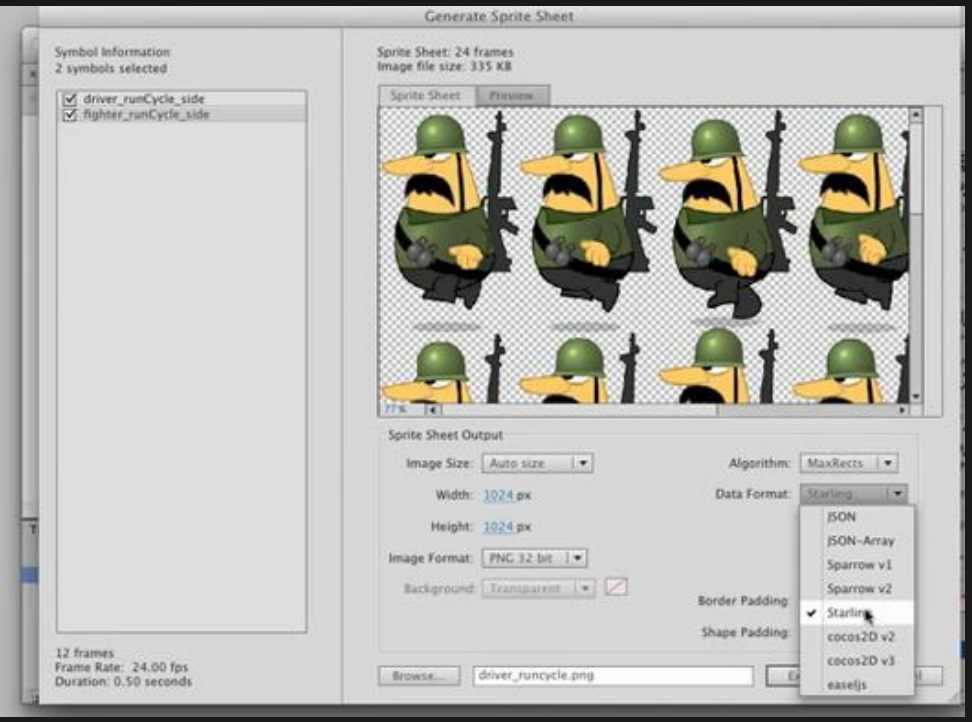
```
var movieTimer:Timer = new Timer(10, 0);
movieTimer.addEventListener(TimerEvent.TIMER, playOnTimer);
var bmp:Bitmap = new Bitmap();

private function playOnTimer(event:TimerEvent):void
{
    bmp.bitmapData  =  bdv[frame];
}
```

小结

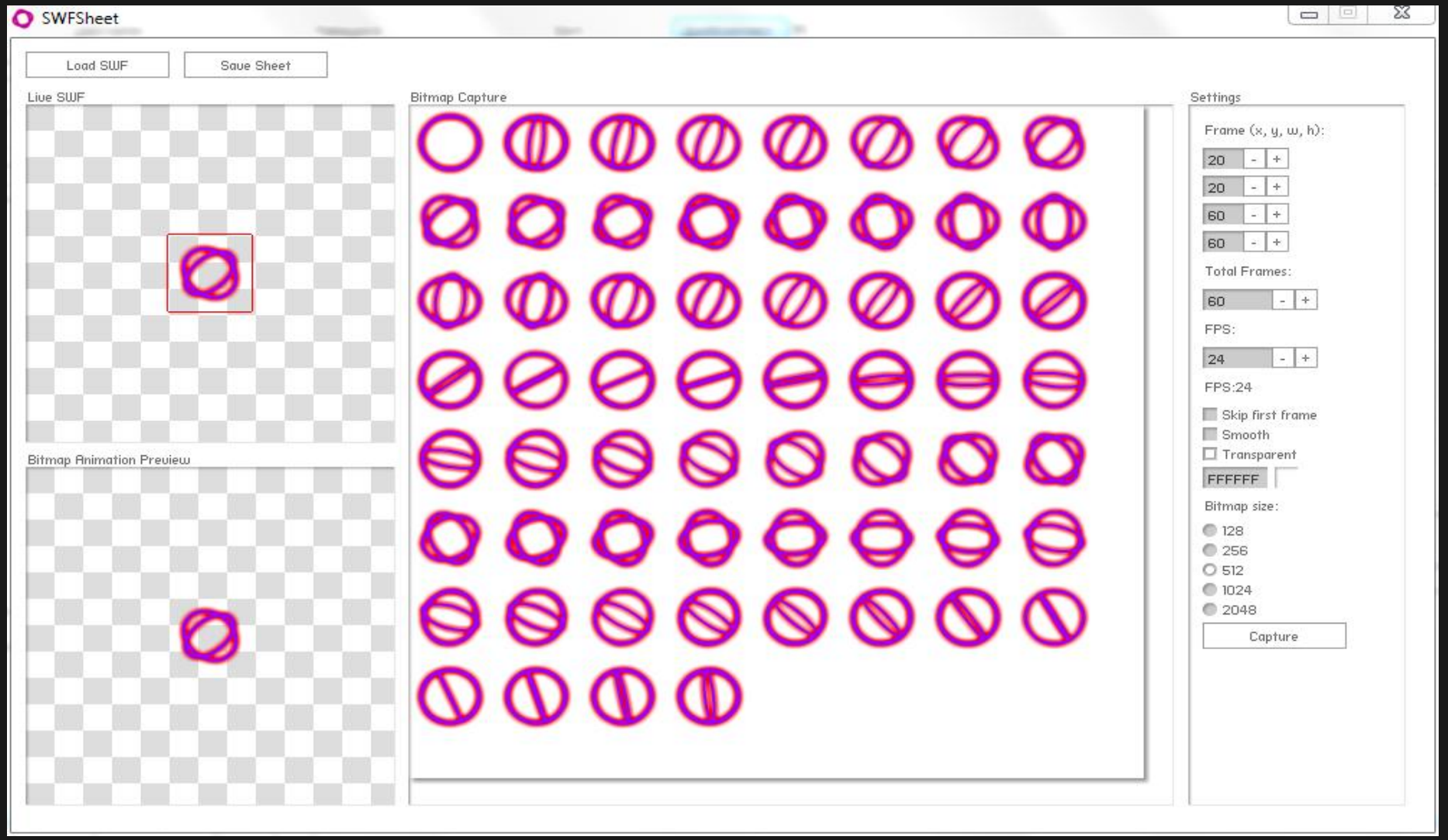
- 块传输虽不能解决所有性能问题，但它能使动画运行平滑，统一大多数机器上动画帧频。

Tools

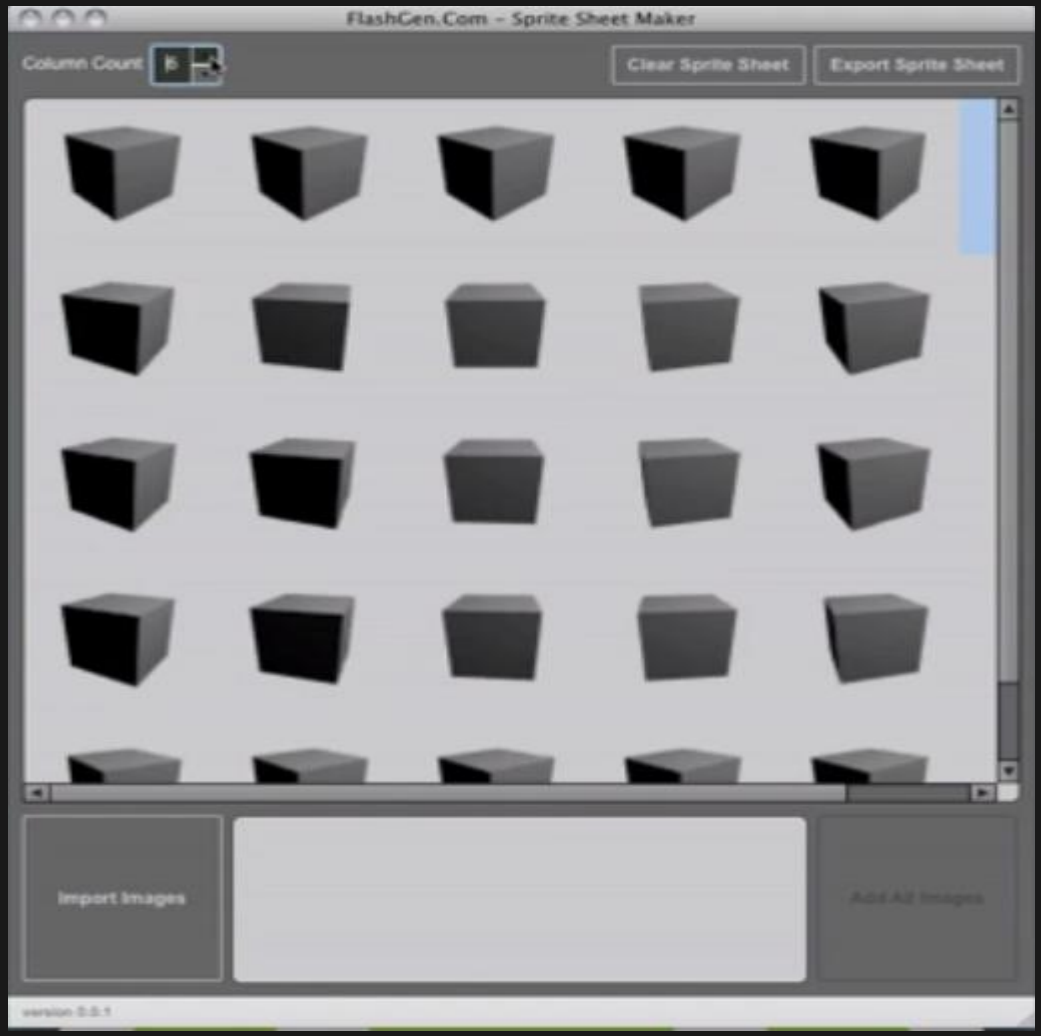


- #Adobe MAX 2011#上演示下一Flash Pro版本CS6中将出现的Sprite Sheet导出功能，你可以将一个或者多个MovieClip/Graphic 输出为位图序列合并图以及对应XML文件来标注每个位图元件大小及偏移量。

SWFSheet


SWFSheet
Load SWF
Save Sheet
Live SWF
Bitmap Animation Preview
Bitmap Capture
Settings
Frame (x, y, w, h):
20
20
60
60
Total Frames:
60
FPS:
24
FPS:24
Skip first frame
Smooth
Transparent
FFFFFF
Bitmap size:
128
256
512
1024
2048
Capture

Sprite Sheet Maker

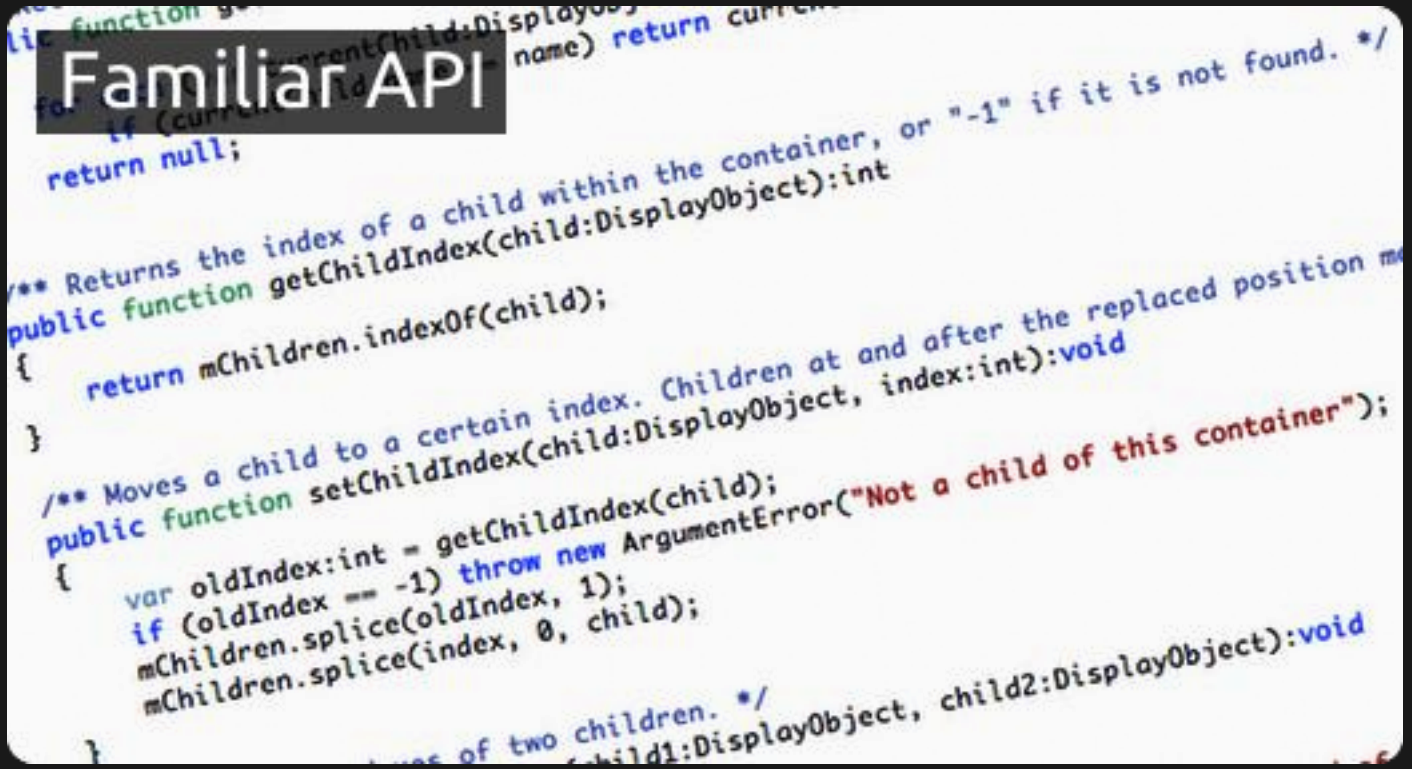


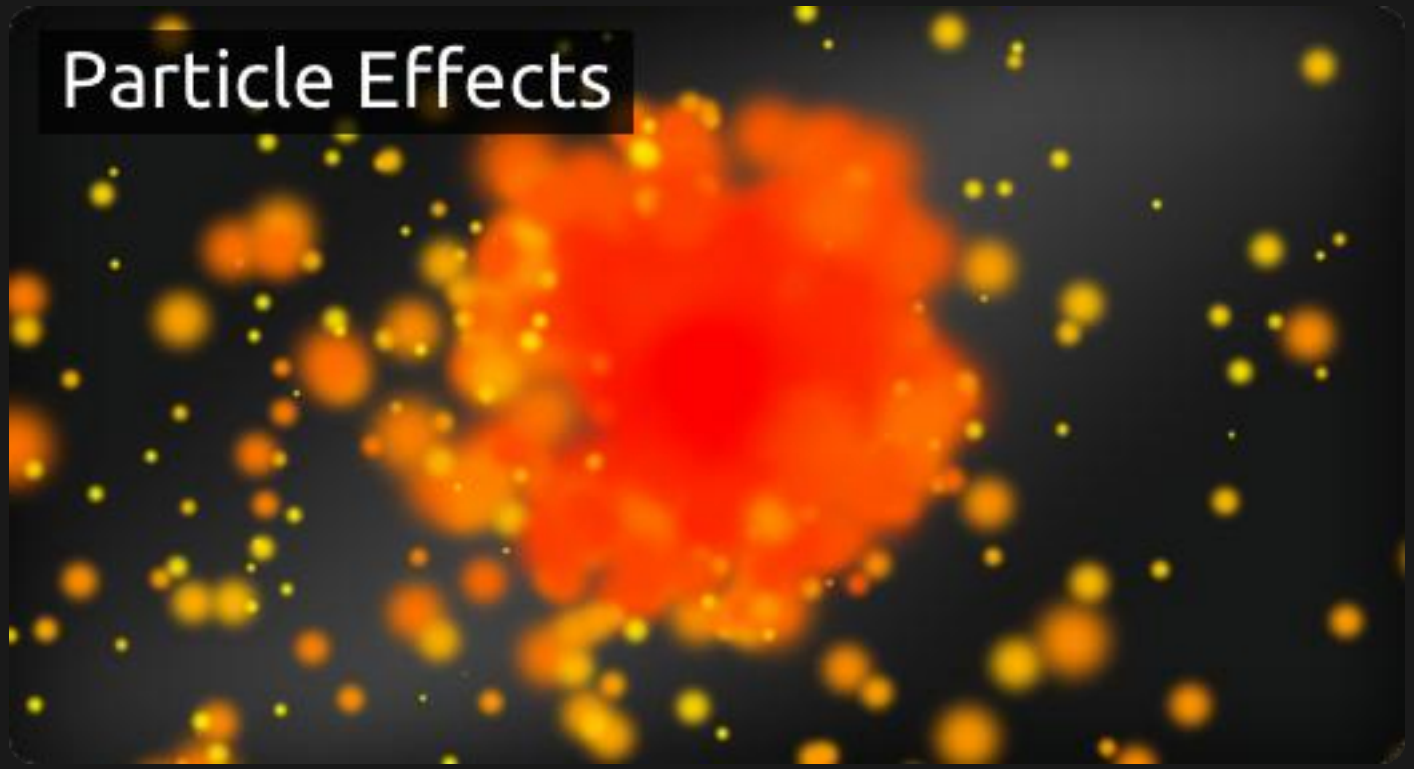
Stage3D 2D Engine

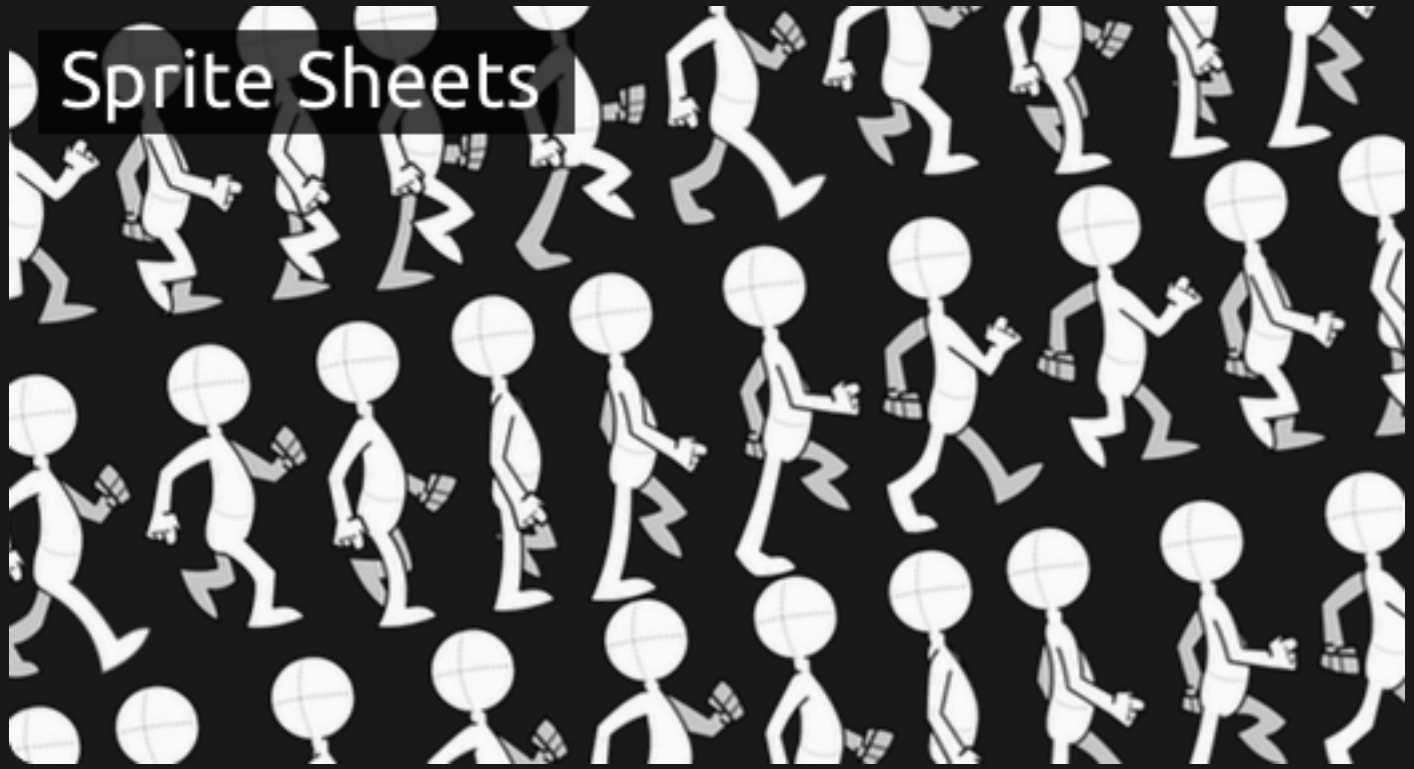
- StarlingFramework
- Citrus Engine
- YCanvas
- IsoHill Framework
- ND2D
- **Napoleon**

- http://www.starling-framework.org/

Powering **Angry Birds** on **Facebook**!


Familiar API
return null;
/** Returns the index of a child within the container, or "-1" if it is not found. */
public function getChildIndex(child:DisplayObject):int
{
return mChildren.indexOf(child);
}
/** Moves a child to a certain index. Children at and after the replaced position m
public function setChildIndex(child:DisplayObject, index:int):void
{
var oldIndex:int = getChildIndex(child);
if (oldIndex == -1) throw new ArgumentError("Not a child of this container");
mChildren.splice(oldIndex, 1);
mChildren.splice(index, 0, child);
}
of two children. */
child1:DisplayObject, child2:DisplayObject):void


Particle Effects

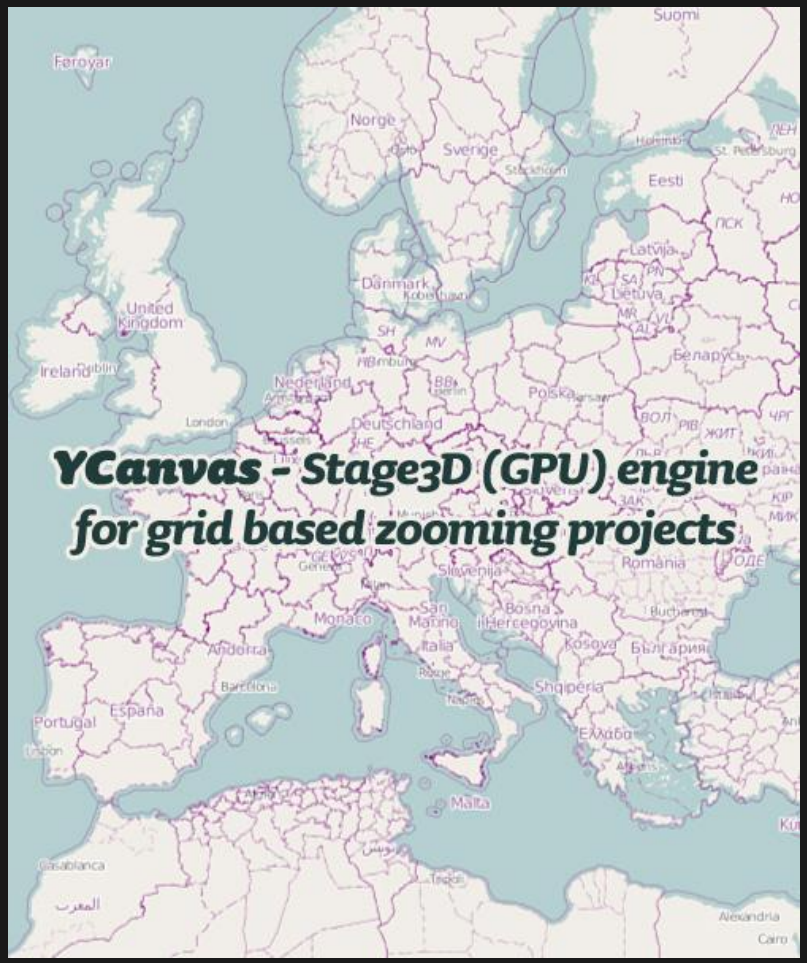

Sprite Sheets


Bitmap Fonts

Citrus Engine

- Box2D Physics Alchemy
- Signal libraries
- Sound Manager
- Input management
- Loader manager
- Level editors
- **Built on Starling**

YCanvas


YCanvas - Stage3D (GPU) engine
for grid based zooming projects

YCanvas is an open source ActionScript 3 library that provides stage3D (GPU) engine for perfectly overlapping grid based zooming projects like world maps or onBoard or simply a tile maps renderer.

Repository content:

- ApplicationDemo - sources for simpliest YCanvas implementation demonstrating implementation over Open Street Map available live on http://ycanvas.yoz.sk/demo
- ApplicationExplrorer - sources for YCanvas explorer available on http://ycanvas.yoz.sk/explorer
- ApplicationRemotair - sources for YCanvas with Remotair available on http://ycanvas.yoz.sk/remotair/ . This app lets you remote controll YCanvas application with Remotair. Uses multitouch (two-finger) gestures to move, scale, rotate
- bin - compiled binaries
- YCanvasLibrary - sources for YCanvas library

You can see Ycanvas explorer demo live on ycanvas.yoz.sk (+stats) to compare the speed of native implementation vs. YCanvas, or browse repository for sources to understand the simplicity of implementation of 3rd party systems:

- onBoard
- Wall Of Fame
- WebCanvas
- MapQuest
- ArcGIS
- Open Street Map
- Flickr - gallery experiment

The engine is optimized for rendering speed as well as for relasing unused system resources. While GPU api for flash player requires textures at width and height at power of two, you can reach the best performance when using 256x256 or 512x512 source images etc.

IsoHill Framework

《文明III》（Civilization III）

《暗黑破坏神》（Diablo）

《城市农场》（CityVille）

《咖啡世界》（Café World）

腾讯《QQ餐厅》

新浪微博《微城市》

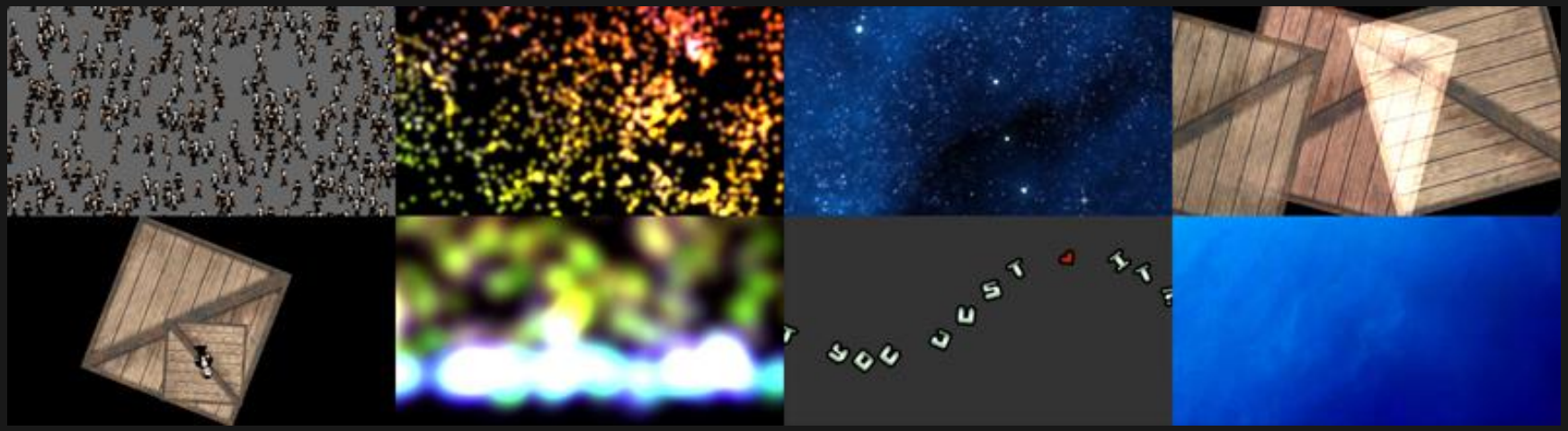
IsoHill Framework

IsoHill is a new isometric engine for Flash Player 11 [molehill] built on top of the Starling 2D framework.

Featuring:

+ Plugins - engine-wide modifications
+ Includes a TILED map parser
+ Components - logic on the IsoSprites
+ Virtually unlimited layers
+ Anti-aliasing and Mipmapping
+ Built for robust browser game

ND2D



Features

- Flash-displaylist-like hierarchy for 2D elements
- 2D sprites with tinting, blendmodes, colorTransforms, pivot points
- Support for spritesheets and texture atlases
- Scenes
- 2D camera

Features

- SpriteCloud / Batch for massive sprite rendering
- Powerful particlesystem that runs entirely on the GPU
- Full mouseevent support for 2D sprites
- Utils (Color mixing, random number generation, …)
- Fast bitmap fonts

Features

- Distortable 2D grid for wave / ripple effects
- Texturerenderer for post process effects
- Extendable material system, that makes it easy to write own effects and shaders with PixelBender3D or AGAL
- Device loss is handled by the framework automatically, you don't have to care about this.

Napoleon

Napoleon

Extending ND2D with physics and more...

This experiment adds the 2D phyiscs of Nape and some extra 2D object types to the 2D GPU accelerated ND2D framework.

To use it you will need to clone (or fork) this repository and my ND2D fork, which can be found here.

I use FlashDevelop, so that's the format of the example project file.

Try out the current build of the napoleon example/test project here. There's info in the console at the top, but you can close it and use 'n' to skip through the tests if you prefer. (ctrl+shift+enter to show/hide the console)

The project is still early in development, so things are likely to change and break along the way.

- StarlingFramework
  - Citrus Engine
  - Ycanvas
  - IsoHill Framework
- ND2D
  - **Napoleon**

参考

- http://leo398.blog.51cto.com/658992/328120
- http://lostinactionscript.com/2008/10/30/object-pooling-in-as3/
- http://stage3d.flashdaily.net/
- http://www.cnblogs.com/turingbooks/archive/2011/12/26/2301890.html

Q&A

- www.duzengqiang.com
- Weibo.com/duzengqiang
- dzq@9ria.com

君道科技 以信致远

旗下网站

9RIA.com 天地会
中国最大的RIA开发者社区，关注Flash/Flex/AIR/Silverlight
Adobe亚太区Community Champion
业务范围：人才推荐、广告宣传、开发者技术大会

RIATraining.net
专业RIA教育咨询平台
一流师资、前沿教学、实训课程、推荐就业
业务范围：职业教育、企业咨询、行业会议、高校课程合作

joyClouds.com
一流软件外包团队
PC、移动端跨平台游戏及应用开发
业务范围：网站开发、iOS开发、Android开发、RIA程序开发

联系电话： (+86)0571-87664537