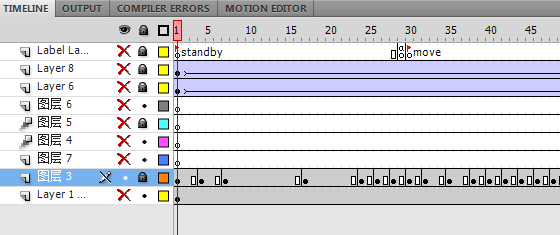
# 序列帧压缩器使用指南

以YongJiang.fla为例来说明

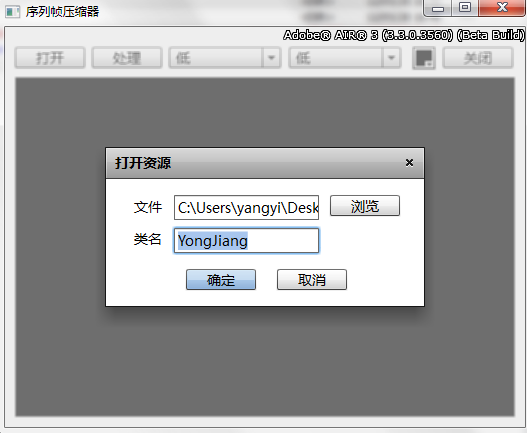
### 序列帧压缩

1. 首先使用FlashProfessional打开动画资源文件，编辑包含序列帧资源的MovieClip
2. 将序列帧层之外的层隐藏，然后发布

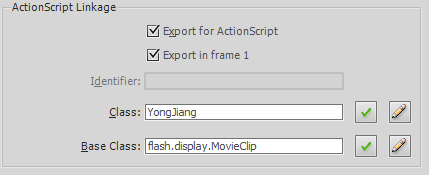


注意发布设置中要取消“包括隐藏图层”选项（Include hidden Layers）

1. 打开序列帧压缩器，找到刚才发布的swf，并指定正确的类名



类名就是元件属性中定义的类名



1. 在预览窗口中拖动动画，并调整压缩器程序窗口大小，确保预览窗口范围能完整的看到动画的各帧。压缩器将使用预览窗口的大小来捕捉图形，但窗口也不要太大， 否则会影响压缩器的速度，窗口过小，可能会导致部分帧被裁剪
2. 可以修改预览窗口的背景色
3. 选择透明的质量。层级1只有透明或不透明；层级4表示从不透明到透明有4个级别，可以表示出半透明效果
4. 点击处理按钮开始压缩处理，信息窗口中会显示出压缩信息，有三种信息：

【286】帧大小：4494　　Alpha通道：912

第286帧，压缩后大小是4494字节，其中透明通道大小是912字节

【291】引用〖286〗

第291帧与286帧内容相同，直接引用

【313】null

空白帧

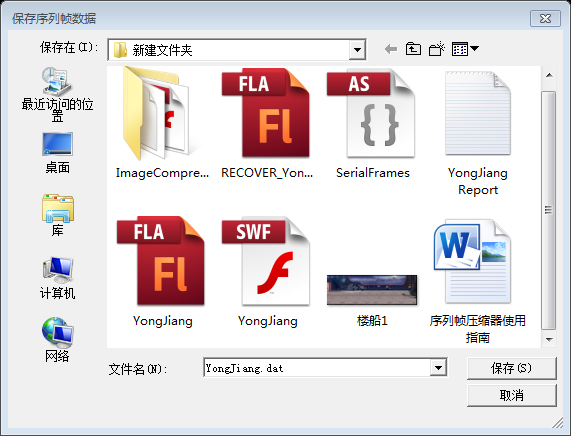
1. 处理完成后，可以点击预览按钮查看压缩后的效果。



和主窗口一样，也可以通过拖动动画来调整位置，也可以修改背景色，还可以替换背景来查阅显示效果

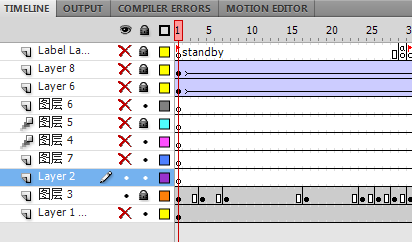


1. 信息窗口中点击保存按钮，可以将压缩的数据保存为文件，扩展名任意，不过建议使用dat这类常见名称

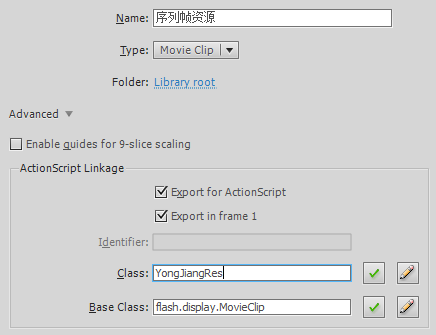


### 数据文件的使用

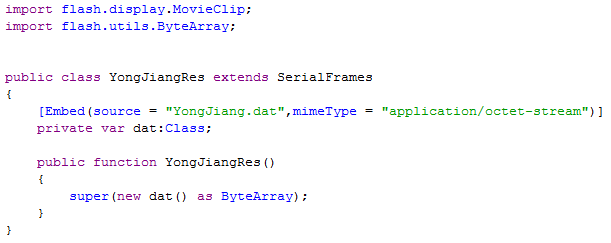
1. 回到FlashProfessional中，在刚才的序列帧图层之上新建一个空白层



1. 再创建一个空白MovieClip元件，取名为“序列帧资源”（名称任意），类名为YongJiangRes，注意类名要唯一，否则多个资源的载入会冲突

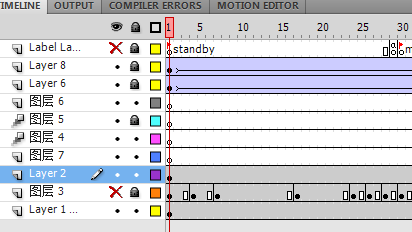


1. 点击编辑按钮来创建这个类，我们使用Embed标签来嵌入刚才导出的YongJiang.dat



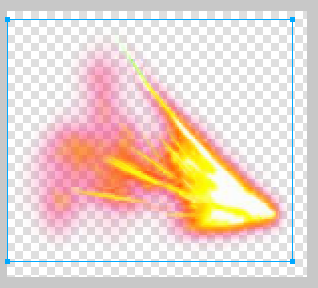
这里扩展了包中提供的SerialFrames类，它用于解析保存的数据文件，并将其正确播放出来，编辑完成后，保存

1. 回到导出序列帧的元件中，将刚才创建的新元件拖放到添加的空白层中，注意坐标设置为（0，0）
2. 现在隐藏掉原来的序列帧层，就可以看到效果了

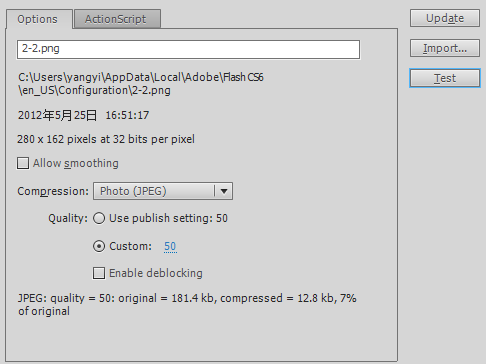


# 附录：美术处理原则

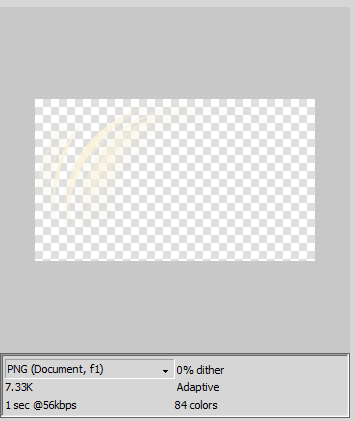
* 压缩器主要针对序列帧动画资源，因此，非序列帧动画（如时间线动画）、矢量图就不要用压缩器来处理了
* 图片坐标尽可能使用整数，有小数的坐标可能会导致图片的显示质量下降
* 给压缩器处理的图片之外的其它资源图片，需要手工处理，确保资源裁剪到正确尺寸，冗余的区域会造成图片大小的浪费



* 没有动画，就不要设置MotionTween，画面没有发生变化时，就不应设置关键帧
* 注意使用FlashProfessional中的“生成尺寸报告”项，观察位图的大小，有没有浪费的图源。如资源库中的“2-2.png”，按质量50输出时大小为12.8K



将图片导入到Fireworks中编辑优化，使用PNG8，Alpha透明时，得到的大小是7.33K，那么这张图就应该重新处理并再导入。



最终得到的优化大小是6.8K：

2-2.png 6869 45360 Lossless