参考：

《CryENGINE3 GameDevelopm  Beginner Guide》第六章：http://blog.csdn.net/u011707076/article/details/41729359

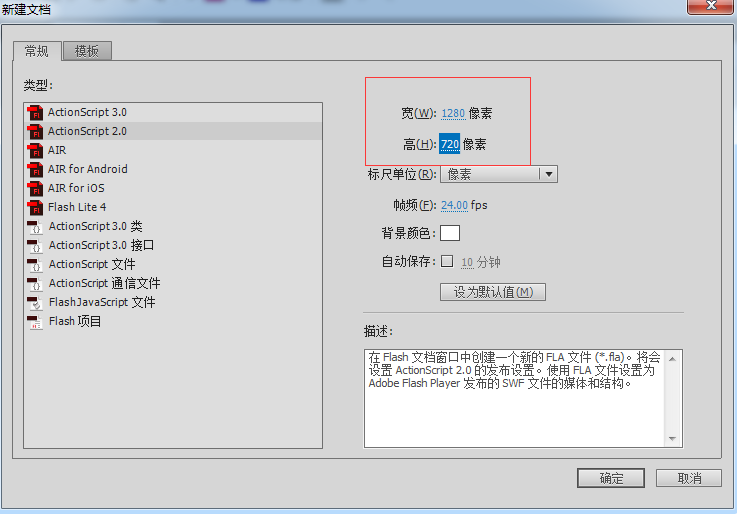
http://blog.csdn.net/kasteluo/article/details/79087363

UI

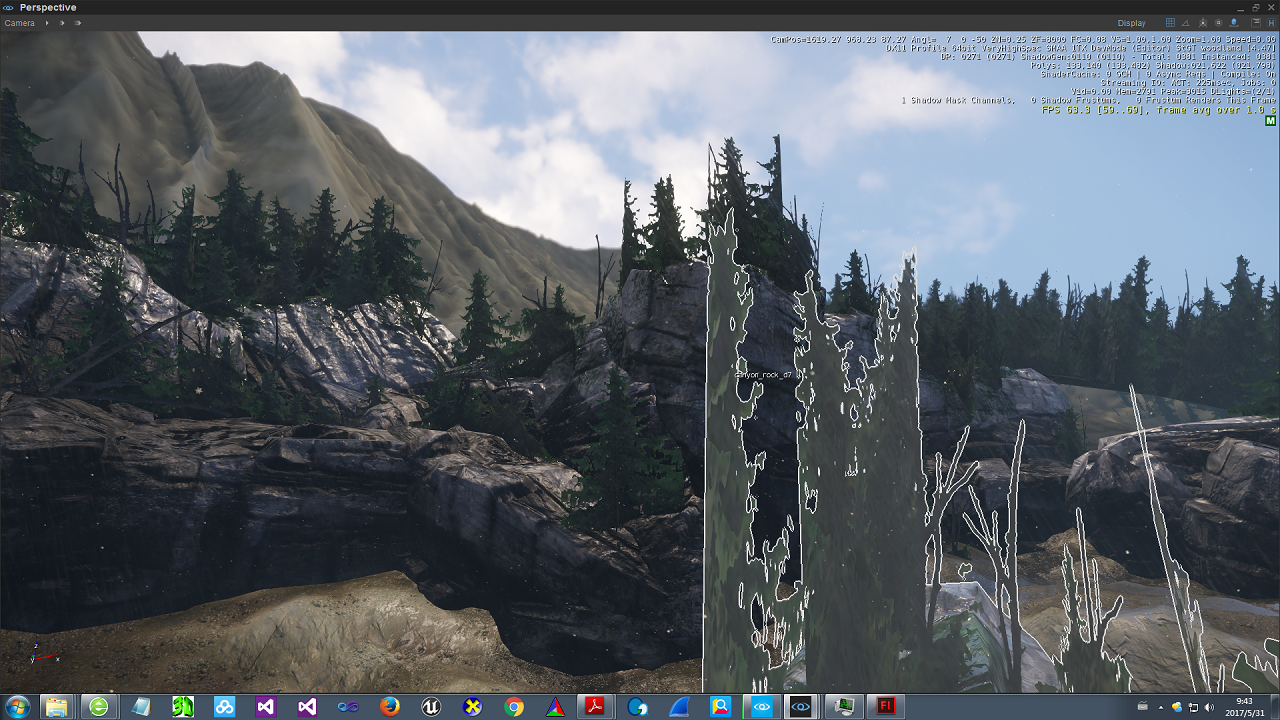
1、Flash UI制作，举例：生命值和弹药数目显示

1.1创建flash资源

1.1.1、创建flash actionscription2.0工程，设置分辨率为1280\*720,保存为：存储为NewHUD.fla。



1.1.2、准备资源从程序中抓取一张截图，保存为1280\*720的一张截图：screenshot.png：

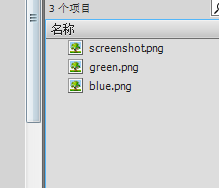


1.1.3、创建button资源：

使用photpshop制作两个半透明按钮图片 green.png(512\*128) 、blue.png(512\*128)

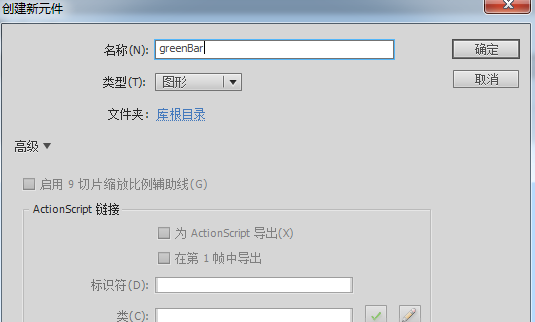


1.1.4导入资源到库中：

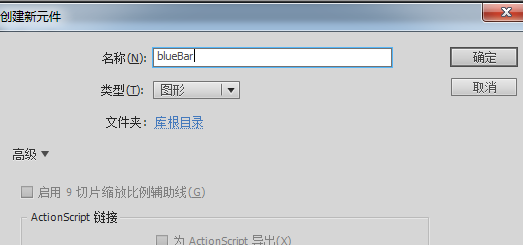


1.1.5、创建 图元

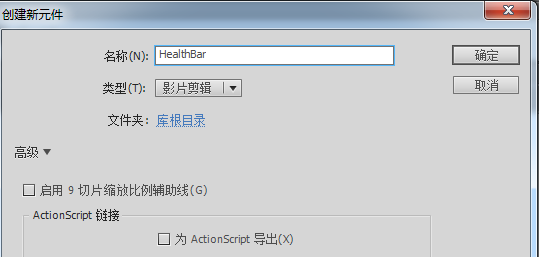
创建greenBar:



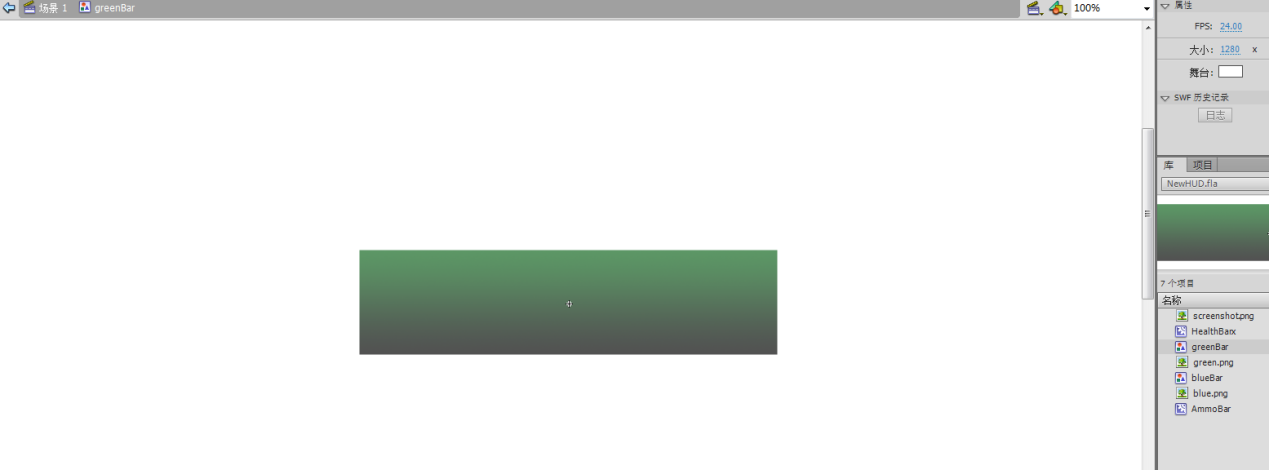
创建blueBar:



创建生命值元件：



将png资源放置到对应名称的图片元件上。



1.1.6 编辑生命值图元

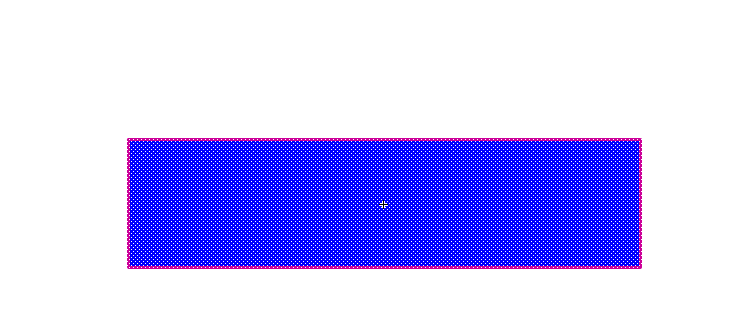
打开healthBar 图元；

将greenbar图元拖入healthbar中；

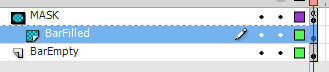
复制图层，分别取名为BarFilled、BarEmpty；

创建MASK空图层，放置与最上部；

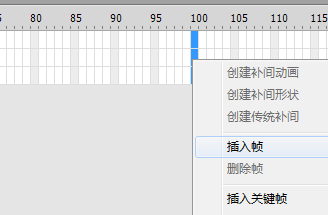
MASK属性设置为遮罩层，并绘制一个矩形，覆盖在原来按钮上，如下图：



将BarFilled为MASK的子层：



选择三个图层的第一百帧，插入帧：



在MASK层的100帧设置关键帧；

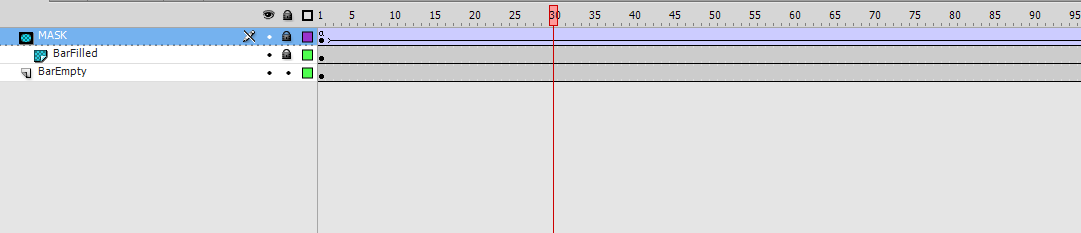
在MASK层中间位置右键，选择：创建传统补间；

将MASK的时间轴设置在第0帧；

移动蓝色矩形区域到原来绿色矩形的左侧：



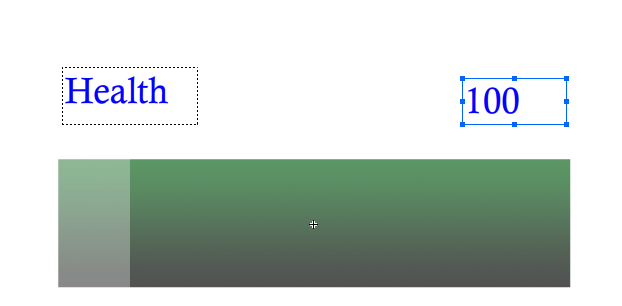
锁定MASK和BarFilled层，移动时间轴，出现下图效果：





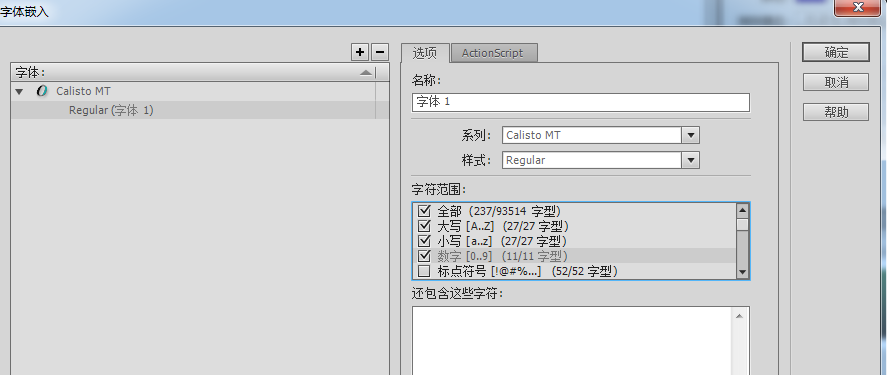
创建Text层，用于输入文字：

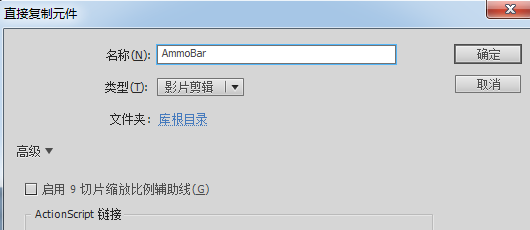
选中该层，添加文字如下：



并对生命值实例命名为：currHealth；

对文本进行字体嵌入设置，勾选字符、数字等设置：



创建AmmoBar，使用复制功能，从HealthBar复制：

修改Text中的文本为：

将Health修改为：Ammo；

添加文本，当前导弹名称为：123（实例名currWeaponName）

添加文本，当前导弹数目 12/12 (实例名：currAmmo）

添加文本，当前导弹 12 (实例名：currAmmoName)

切换到场景1，添加HealthBar和AmmoBar，并取名为：：ammoMC，和healthMC。

插入图层2，设置为引导层，选中该层，将screenshot.png放置在层上。

创建一个Button按钮测试

Button测试 :

按照教程创建的button不能动,做如下修改就可以,帧标签要使用字符类型:

stop();

onRollOver = function()

{

//gotoAndStop(2);

this.gotoAndStop("over");

}

onRollOut = function()

{

//gotoAndStop(1);

this.gotoAndStop("out");

}

onPress = function()

{

//gotoAndStop(3);

this.gotoAndStop("down");

fscommand("onPressed",btnText.text);

}

onRelease = function()

{

gotoAndStop(over);

}