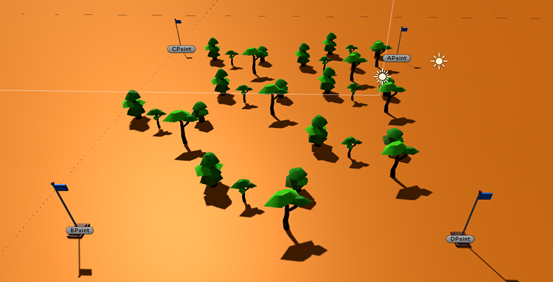
# **Unity完全自制游戏纸箱战争项目记录（20180705）**

作者：[Pluto](http://gad.qq.com/user/index?id=2267380)

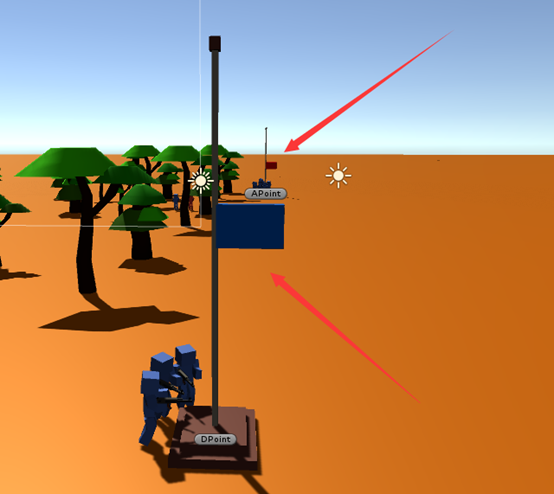
链接：<http://gad.qq.com/article/detail/190460>

今天完成了初代可试玩版本的发布，历经将近一周的时间的首个测试版本。



初代地图并不掺杂地形，只不过是把外建工程中的AI脚本适配进了原项目工程中，对细节部分做了一些优化和改动。

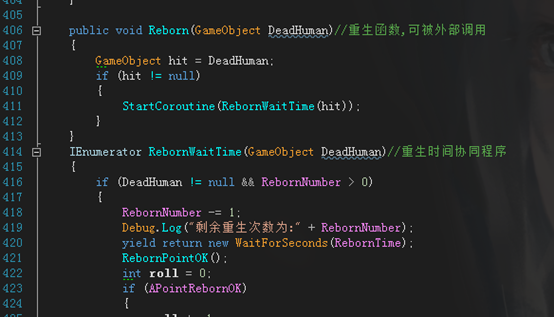
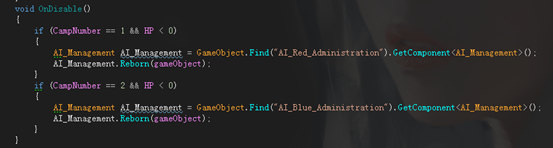
在占领点上增加的旗帜，方便更直观的看到旗帜的变化从而知晓占领点的占领进度，当然了，这在之后的UI上也会有所体现。



如图所示，不同占领点在AI接近之后发生了占领数值的变化，从而改变了旗帜的高低位置。

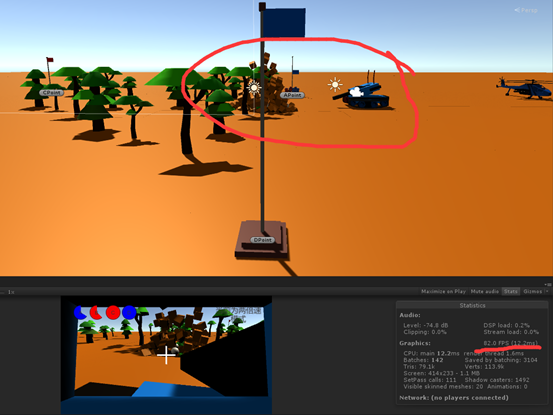
在旗帜的代码编写阶段还出现了一点意外情况，感觉到利用动画实现旗帜的升降会更加困难，所以便采用了代码检测目标点的数值动态调整旗帜高地的设定。

在最早的代码中，因为错误的把全局坐标当成了局部坐标，创建向量给position赋值的时候坐标位置总是不再预想中的位置上，期间还以为是因为模型的缩放值影响到了坐标位置的变化，最后检查过发现是在赋值的时候发生的错误，创建出的向量为全局坐标向量，但错误的赋值给了局部坐标，从而导致了错误，更正之后即修正了错误。

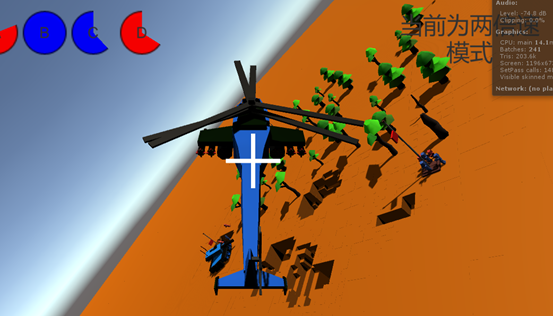


上图代码中是实现的NPC在死亡后所调用的函数，在NPC生命值小于0的同时关闭物体的显示，同时在OnDisable中调用管理AI中的重生方法，传导启用协同程序，在等待一段时间后根据随机值在可重生点复活。

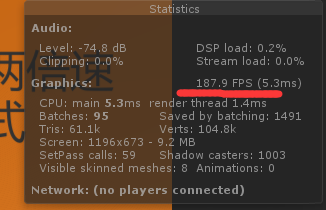
接下来的就是着重与游戏的优化，因为地形破坏的原因要消耗大量的系统资源，所以在其余能节省的地方要尽量节省，首先我考虑到了管理AI的代码深度过大，每次执行都要消耗扥多资源。

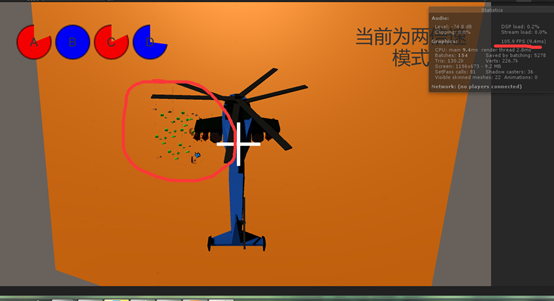
而在战争进行的同时，目标点肯定不会是瞬息万变的，所以统一调用时间设置成了五秒，每五秒钟更新一次目标点，饶是这样还感觉有点时间短了，在后续添加进地形之后可能会把该数值放大到10秒或是20。

如图产生爆炸效果的同时，工程页面的FPS值为82，因为我用到了两个屏幕的原因，消耗的性能可能会增加，如果是在玩家的电脑上运行帧率应该会进一步提高。



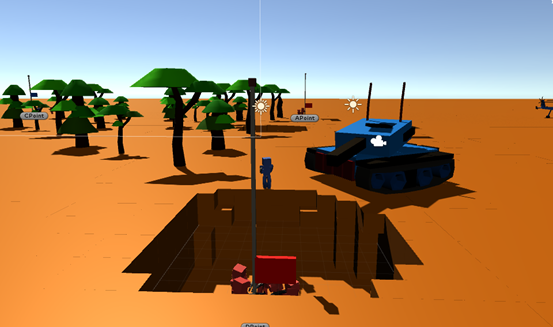
驾驶飞机纵观全局

驾驶飞机时的瞬时FPS达到了187，多数还是能维持在120；



飞机拉升到纵览全局的高度，地形全部加载，预先设定好的四个目标点区域在地图中只占到了很小的一部分，现在的FPS为105。

最后增加了算是一点恶趣味的东西，通常在这类游戏中，玩家进攻的目标点一般都是不可摧毁的，也不能改变的，但我给目标点旗帜增加了刚体，当地形破坏之后，旗帜就掉进了地面下的孔洞中，当然了使用坦克也可以推动旗帜。



此外，今天在编写代码和工程页面调试阶段Unity发生了多次崩溃，未能发现崩溃的具体原因，把工程发布成产品之后在别的电脑上均能流畅运行，并无报错闪退的情况，目前还不清楚是因为什么导致了这样的结果，难道是因为在编写代码的时候改变了系统的某些环境变量的缘故？