# **从零点五开始做半个不能玩的小游戏（三）**

作者：[枸杞忧天](http://gad.qq.com/user/index?id=1075523)

链接：<http://gad.qq.com/article/detail/48288>

**好久不见。**

这是本人认真写下的第15篇与游戏开发有关的文章。

在上一回《从零点五开始做半个不能玩的小游戏（二）》中，我们完成了下列事情：

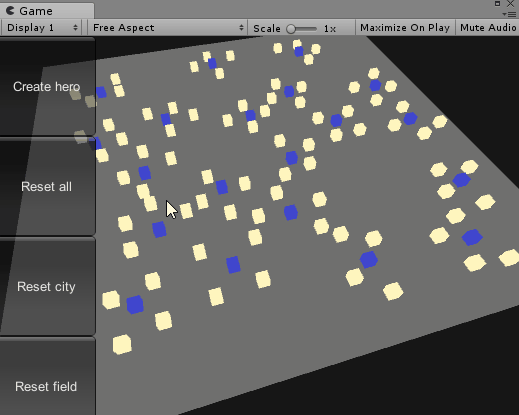
1、重构野外、城市及英雄代码。

2、为英雄添加了简单的状态机，使其可以在世界中玩耍。

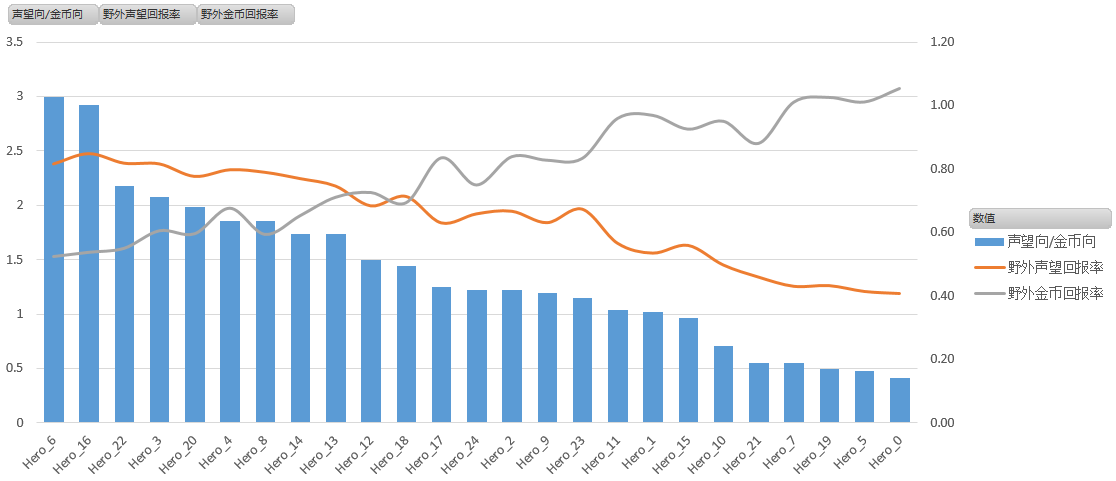
3、搭建了简单的数值体系，使英雄在玩耍时可以提升各种数值。

4、收集了英雄玩耍记录，分析数值结果与预期的偏差。

好像大概是这个样子：



愉快玩耍的英雄们



却不知在身后有一双眼睛在默默地注视着他们

**| 本回目标**

1、修改系统，使更多英雄可以在世界中同时玩耍。

2、添加一个简单(简陋)的UI界面，实时了解世界信息。

**| 修改系统**

我希望可以同时有1000名甚至更多英雄在世界中探索，当然我们无需(也不能)同时观看所有英雄的移动、探索过程，因为这样计算及渲染的压力会很大。

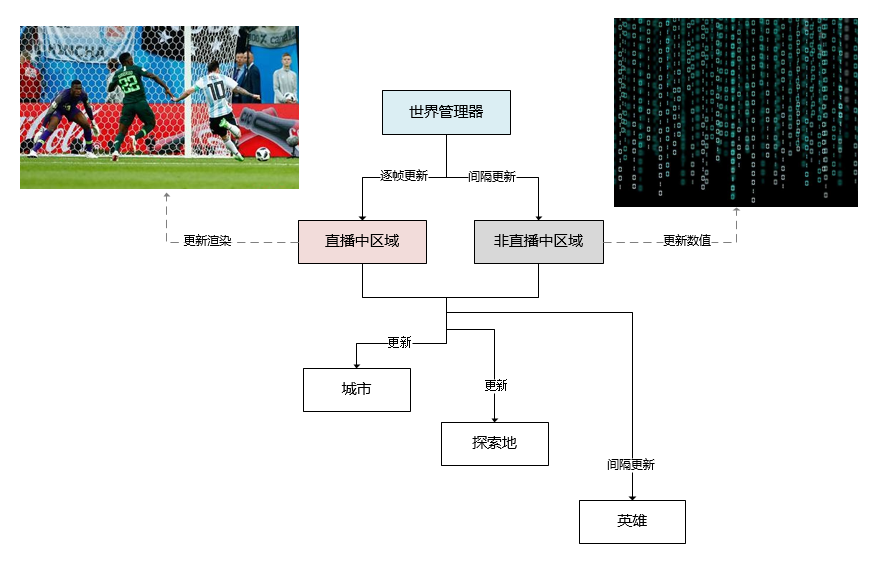
因此，我们要对系统进行一些简单的调整：

1、将世界分为若干区域(Area)，每个区域下有若干数量的城市(City)、探索地(Zone)和英雄(Hero)。

2、世界管理器管理所有区域，并在每帧更新一定数量的区域。

3、区域管理其拥有的城市、探索地和英雄，并在每次更新时更新城市、探索地和一定数量的英雄。

4、我们控制的镜头只能观察一个区域的实时情况，只有某个区域被观察(直播)时，才进行渲染。

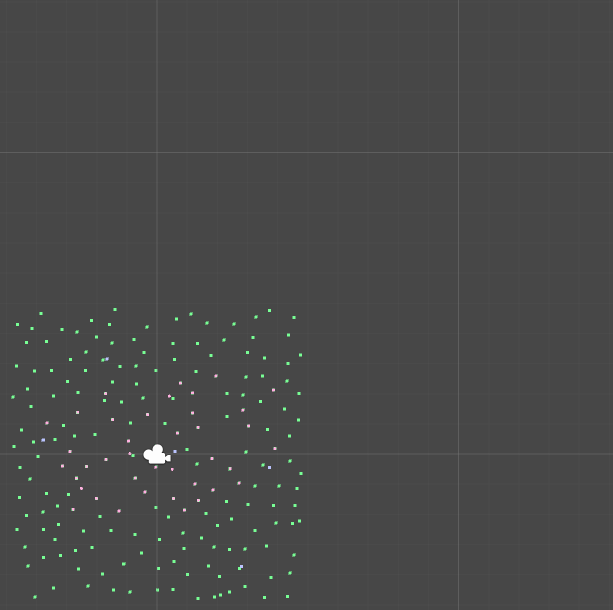


重新调整后的更新结构

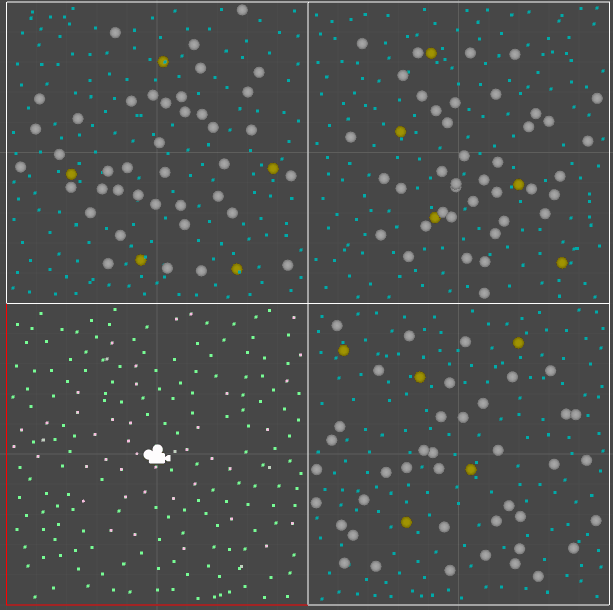
重点是：

**数据与渲染分开，上述1~3为对数据进行更新，4为将数据与渲染进行关联并呈现。**

最终的结果像这样：

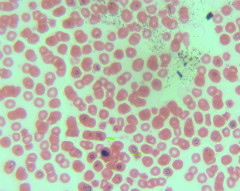


2x2世界里，只对镜头停留的区域进行逐帧更新及渲染



更直观的展示：镜头停留区域逐帧更新、绘制，非停留区域间隔更新及Gizmos绘制

怎样，是不是很像中学时用显微镜观察的细胞装片？



显微镜下红细胞装片

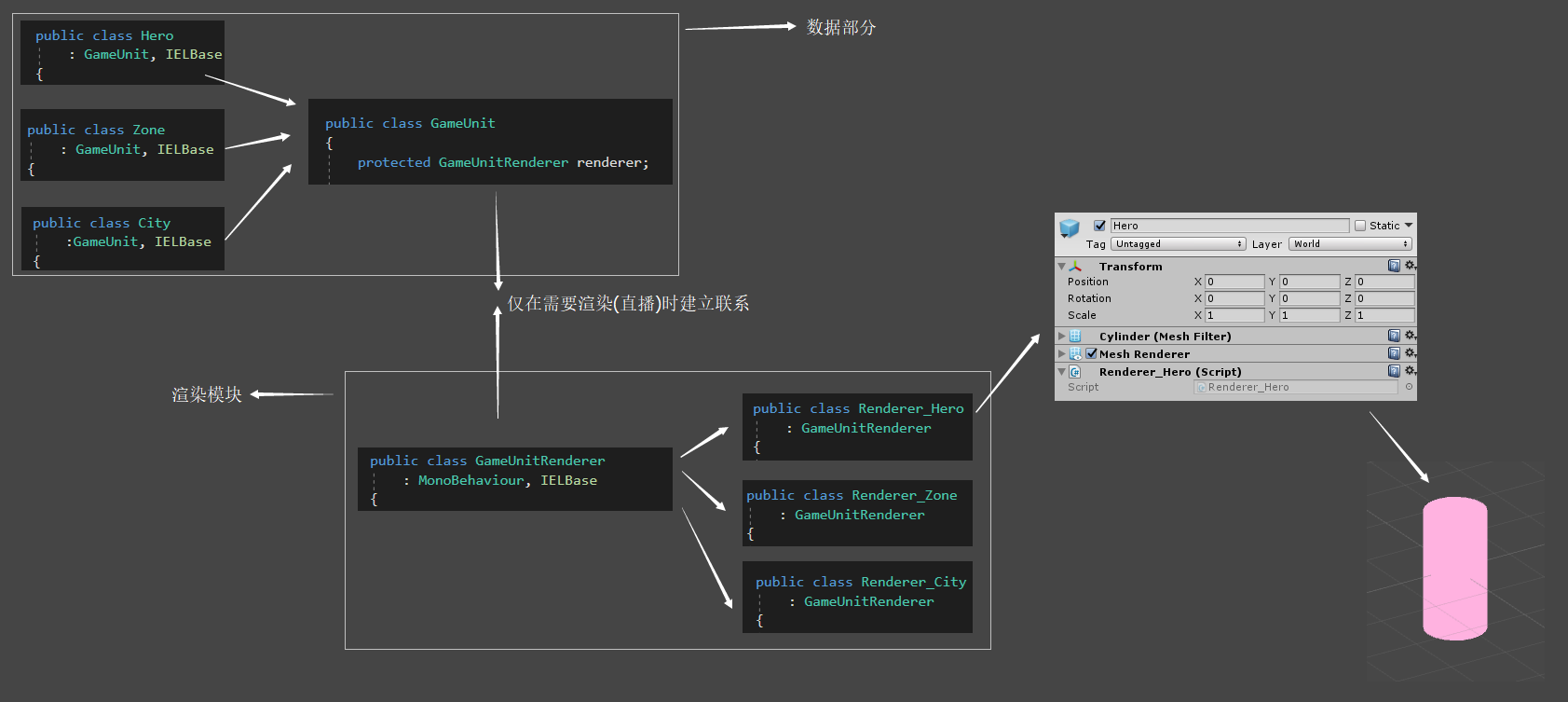
所以，知识都是相通的。具体表现为：上学时你若虚度光阴、不好好学习，并不会对你现在的成就有什么直接影响，它只会让你在后面的人生中大概率上过得比别人惨而已。



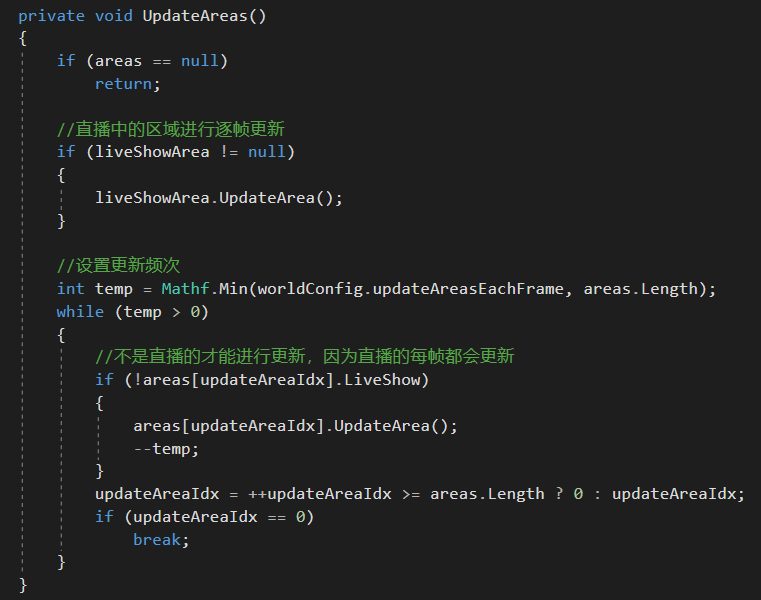
哼~那又如何

**| 关键步骤**

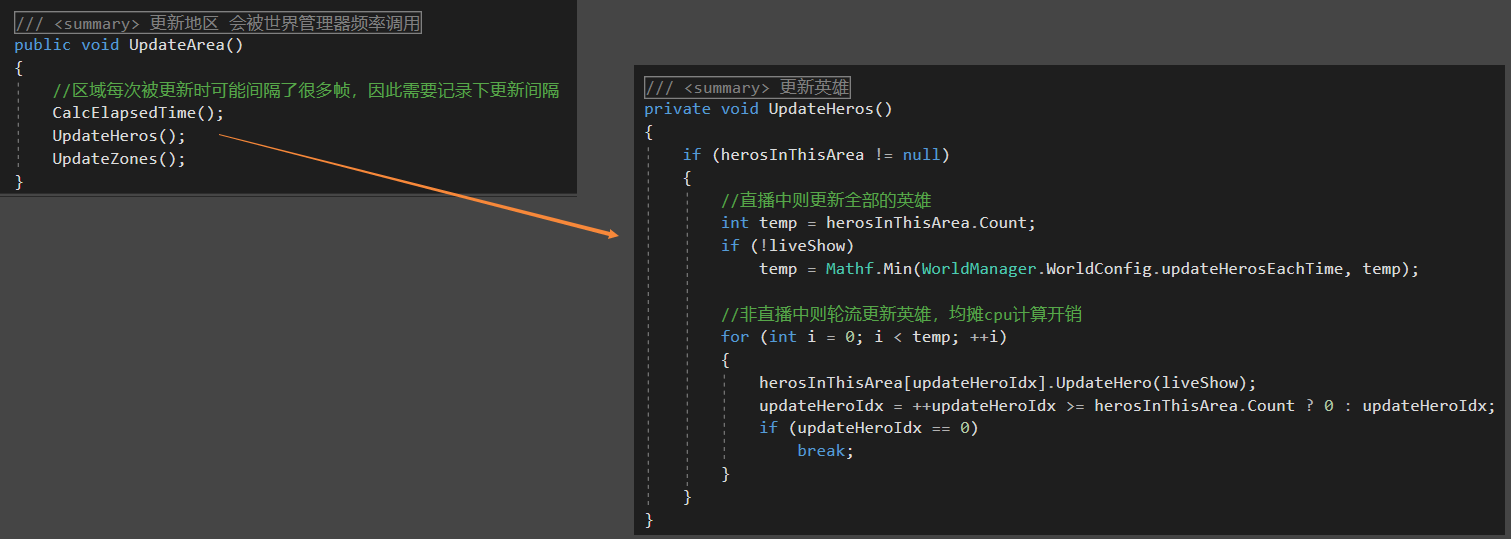
时间有限，只捡些重要的事情讲吧：



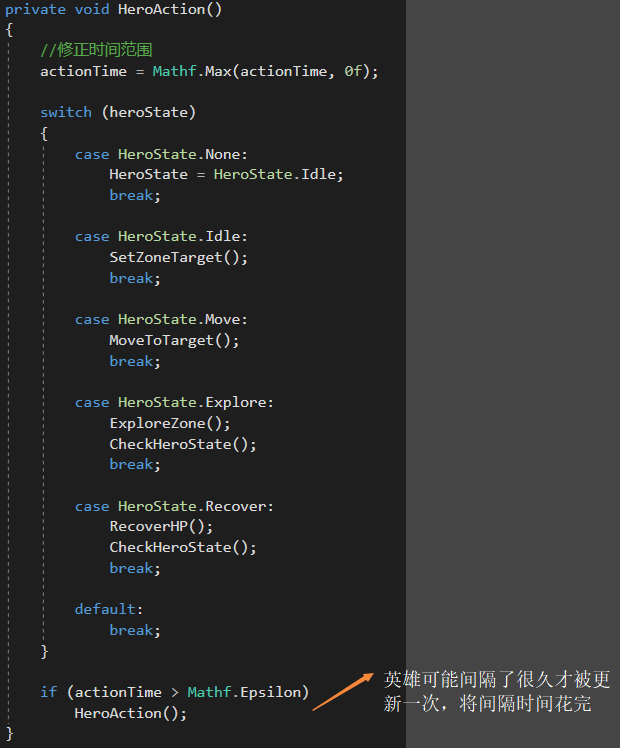
数据与渲染分离



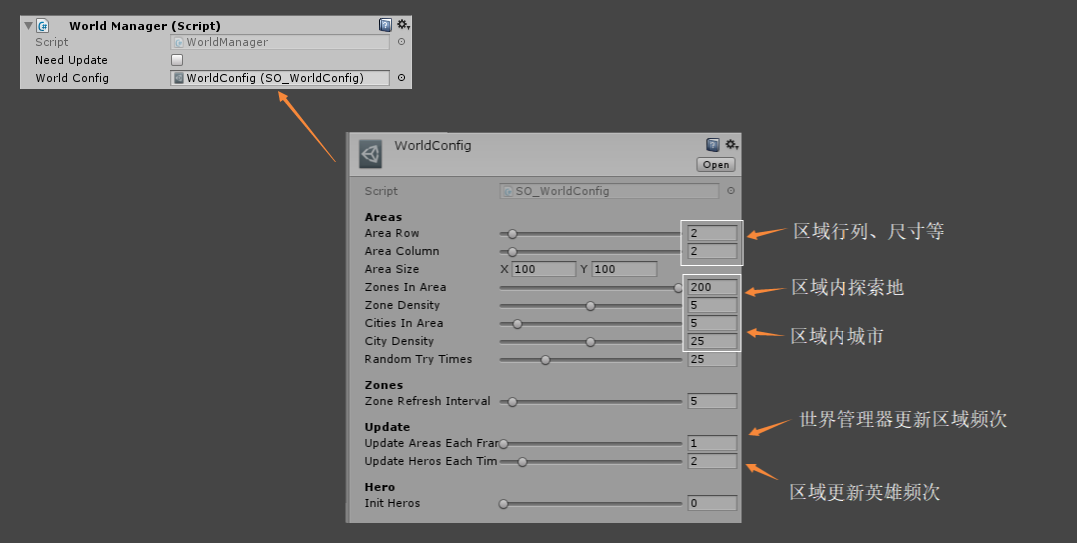
世界管理器更新区域



区域更新英雄、探索地



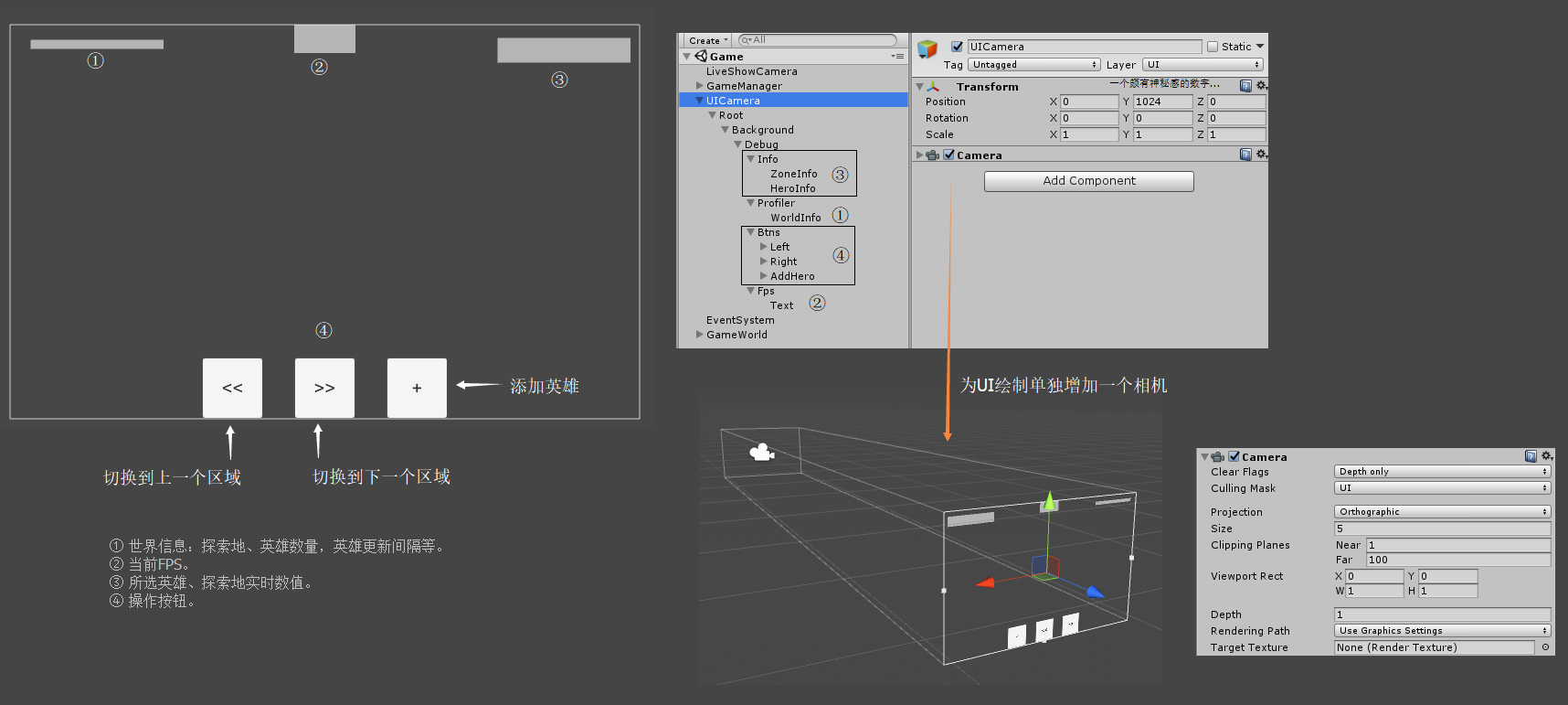
新的英雄状态机



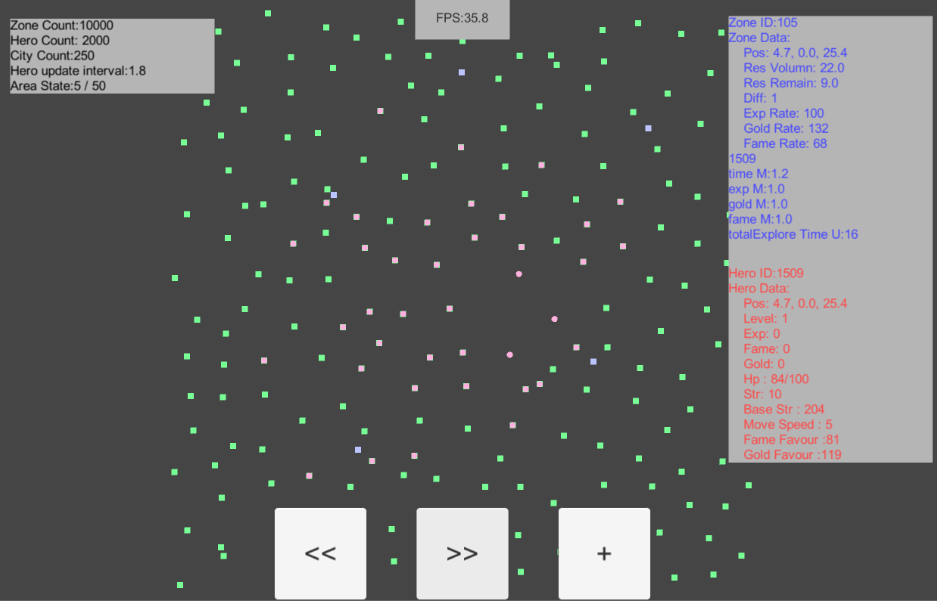
世界配置文件及使用

因为具体代码与上次并没有太大差异，因此不再赘述。

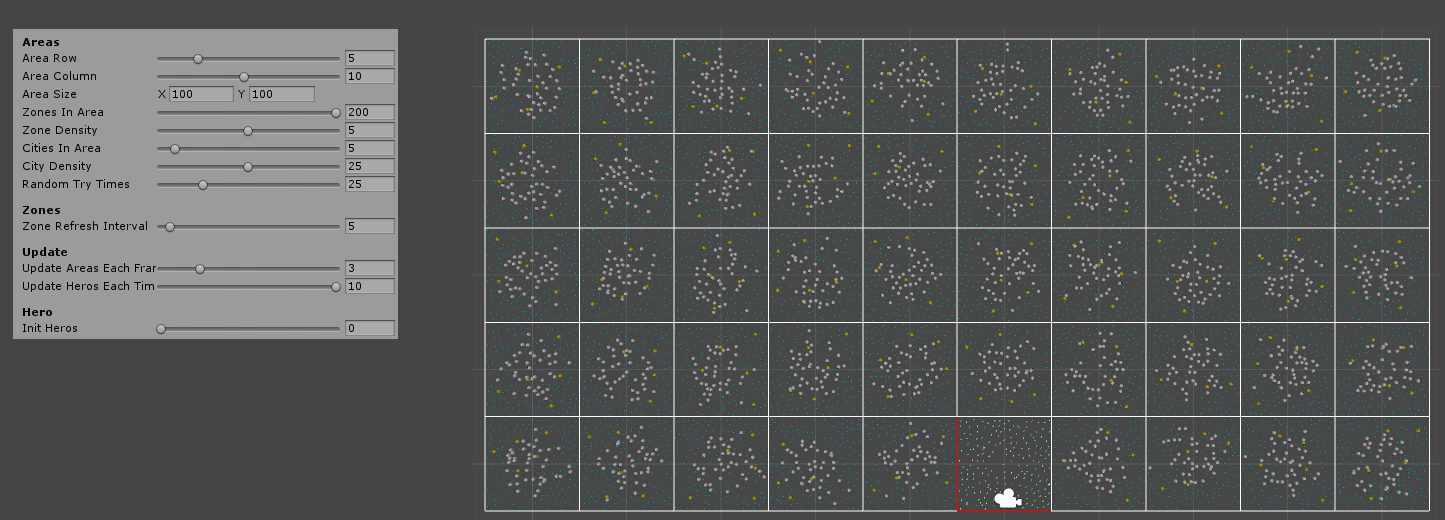
添加一个简单(简陋)的UI界面



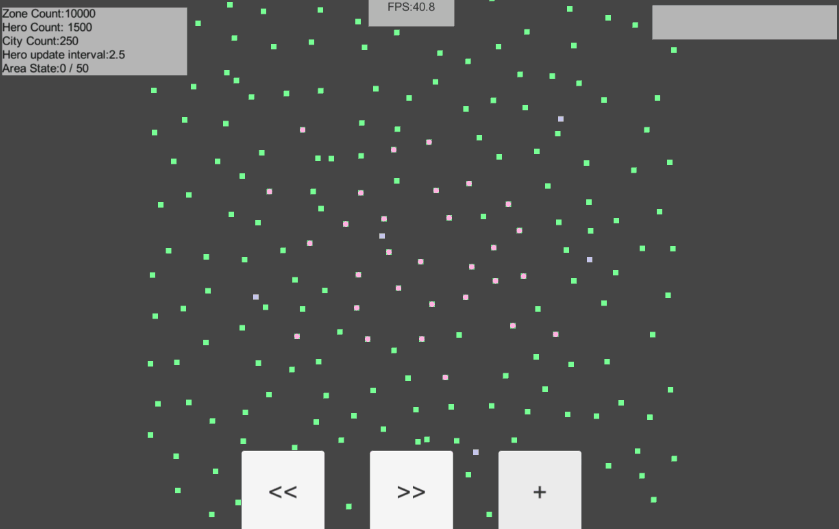
使用一个新的相机来渲染UI



运行起来像这样



5x10世界下的鸟瞰图



实际运行起来的细胞装片

**| 总结**

至此，《从零点五开始做半个不能玩的小游戏》系列将暂告一段落。

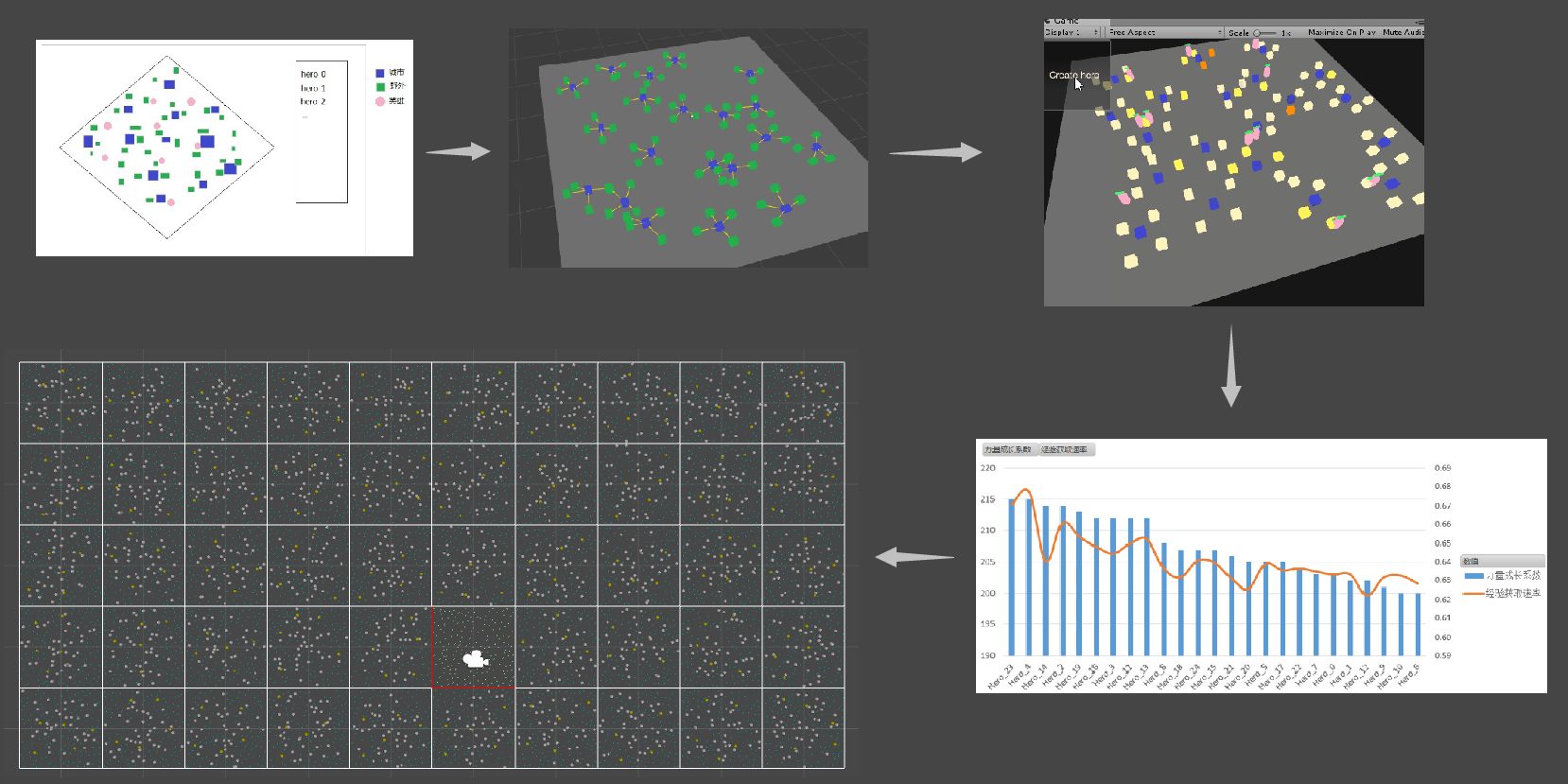
又到了该抒情的时候了(我想这就是我与传统程序员之间最大的差异：点了“擅长将一件小事搞得很有仪式感来感动自己”的天赋)。

为它，从落笔那天起，至今花费了近3个月时间。想也正常：首先，与很多人一样，工作繁忙，只能利用闲暇、且偶尔战胜懒惰君的机会下，一点点的丰富内容。其次，我并不是一个随便的人(当然随便起来也与一般人没有两样)，每句话、每个截图总是斟酌好久，生怕会有什么不妥，闹出什么笑话，因此更降低了不少效率。只是很可惜，能力有限，并没有写出什么干货。

不过我更坚定了一个道理：

*有些时候，就是要先行动，而后思考。*

从仅有一点点小想法开始，不断的写、改，再写、再改，别着急，慢慢来，你会发现改变也在一点点的发生！比如我现在就一点想法都没有了。



看着这个小游戏一点点的发生变化，一步步的完善功能，到最后变成了芝麻烙饼一样的东西，不由的感叹道：

我TM这到底是把时间花在了啥子东西上啊......



就到这了，下次见。

**请关注微信公众号：偶尔学学Unity**