

# ET 框架拖拉机项目试改装

deepwaterooo

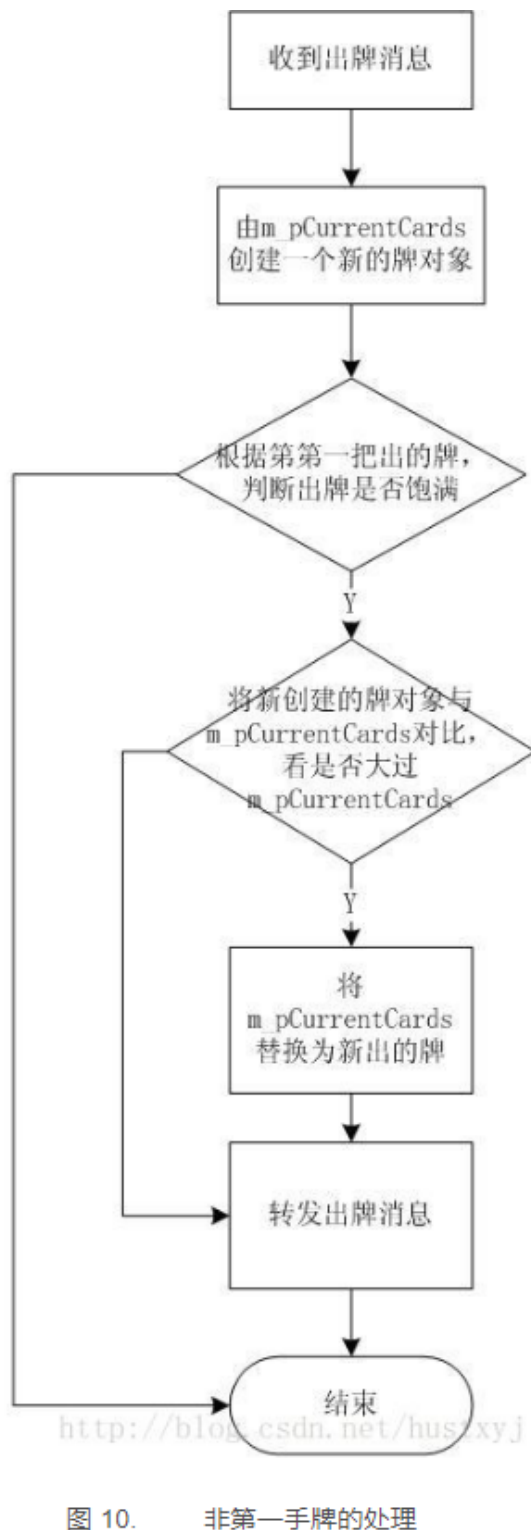
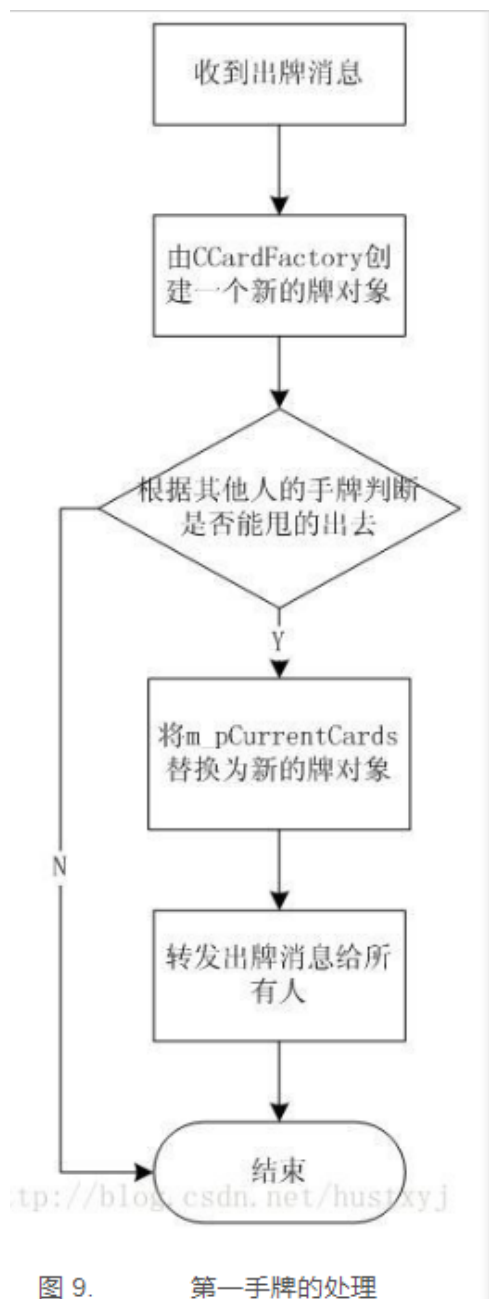
May 19, 2023

## Contents

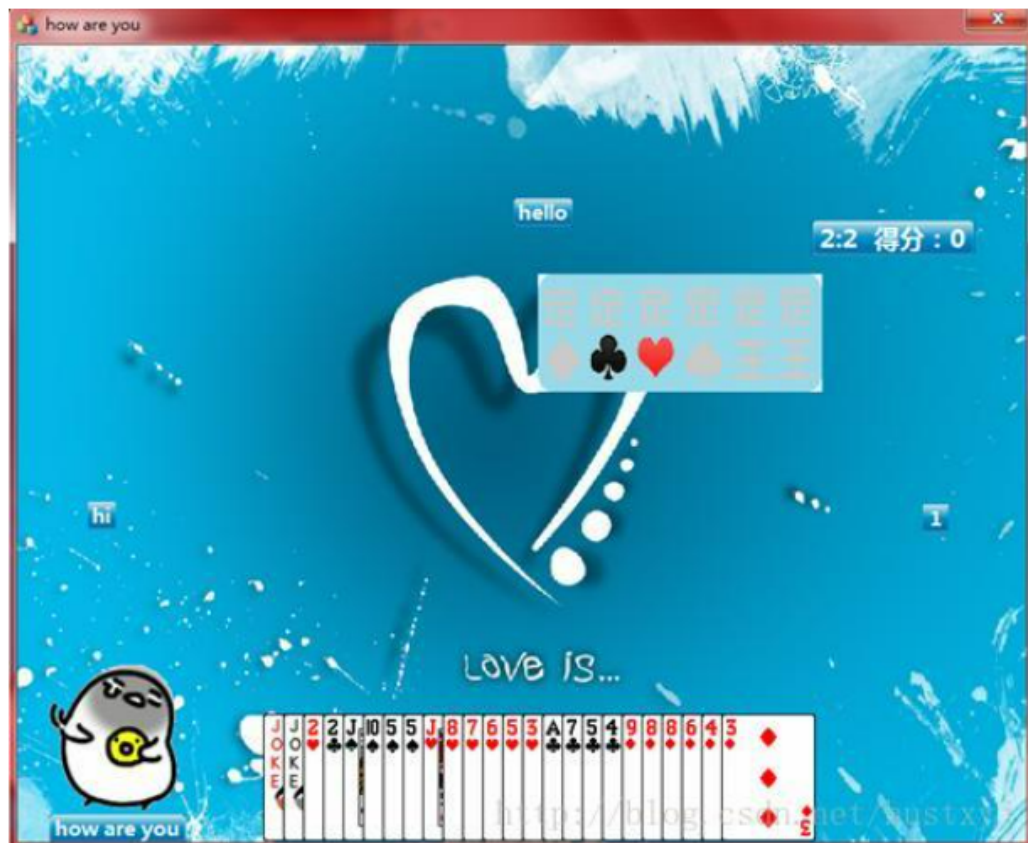
1 双副牌双升 108 张卡牌游戏	1
2 主要参考项目:	6
3 项目主要进展	6
4 残余的几个小 bug	7

## 1 双副牌双升 108 张卡牌游戏

- **【游戏可试玩程序】**: 放在 Release/Tractor.exe. Windows 用户大家可以下载试玩儿体验一下。
- 昨天晚上找见了别人几年前就开发出来的卡五星麻将, 所以写麻将游戏的想法就被恶杀在摇篮中。现在再写什么好呢? 就只能写**【双升拖拉机】**了, 就是两副牌 108 张来打的拖拉机。现在已经 ios iPhone 上有的双升游戏, 可能搜索一下设计, 写安卓版的双升了, 看下能否套用 ET 框架, 写成四人网络**【客户端与服务器双热更新的】**网络游戏
- 现在先搜索必要的框架设计, 出版规则比大小算法之类的。
- **【服务器与客户端的同步】**: 尤其是在分四人牌后, 亮主拖底的时候, 谁先亮, 亮什么主, 顺序重要, 结果重要。**【ET 框架有专用的游戏服, 由游戏服来状态同步】**在本程序中, 采用的是服务器保存所有的状态, 处理所有的逻辑。比如, 客户端在点击亮主后, 做的事情就是发一个消息给服务器, 不做任何显示操作, 等待服务器传来亮主的消息后再显示
  - **【发牌, 公正性】**: 随机分牌。第一步就是要发牌。需要做到一个完全随机的发牌, 就要保证每张牌发到每个玩家手里的概率都是一样的, 而且牌的顺序是等概率随机打乱的。程序中采用的是如下的发牌算法 (感谢 Dr.Light 提供): 假如有两幅牌, 编号从 1 到 108, 首先随机选出一个, 并且将牌发给玩家, 然后将这个编号的牌与 108 号牌交换编号, 那么剩下的牌就是从 1 到 107 号。于是再从中选出一个, 重复以上的过程, 这样一来, 算法的复杂度就是  $O(n)$ 。
- **【牌的逻辑 OOD/OOP】设计**: 三个类, 对应单张, 拖拉机 (对子是长度为 1 的拖拉机), 和混合单张与拖拉机
- 第一手牌与非第一手牌的处理:



• 参考一个很 Q 的界面：



- 亮着牌打的不好玩，一定会把其它三副牌藏起来的
- <http://www.homygame.com/ngscom/help/shengji.htm>
- 简易版设计原理：模拟拖拉机（升级）玩法；

- 1. 创建两副牌的集合：HashMap
  - 2. 创建纸牌：四个花色共 108 张
  - 3. 创建 poker 的 ArrayList 操作集合
  - 4. 创建亮主牌的操作
  - 5. 将所有牌放入牌盒中
  - 6. 创建四个玩家与底牌的集合：HashSet wj1,wj2,wj3,wj4,dipai
  - 7. 洗牌
  - 8. 发牌操作
  - 9. 创建看牌方法
  - 10. 调用方法看牌
- 安卓上的游戏现在是这样的：还要再写一个吗？【活宝妹就是一定要嫁给亲爱的表哥!!!】还是说更为完善或是好玩儿的游戏逻辑？或是 UI 视图画面，或是性能表现？反正一定是套用 ET 框架写得最容易快速方便。【感觉现在这个截图的 UI 长得有点儿丑怪。。】不好看不经典，看了就不想玩儿了。。
- 主要想要改进的地方：前辈十年前开发出的游戏，游戏整体【差强人意】现官宣的现项目基础上，主要游戏逻辑与改进方向【除了重构适配 ET 框架以实现客户端服务器双端热更新，besides 之外】如下。
- **【界面设计：】**八十年前没有 ET 框架，不知道原作者是如何设计这个游戏的。感觉游戏的整体走桌面游戏风，要把菜单设计等改成【手游风格】。
- **【逻辑设计，用户意愿：】**不【尊重用户游戏配置选择】的游戏，永远是固执不受欢迎的。【游戏逻辑、玩法】需要能够给用户留足选择配置空间，如下：
- 现有的游戏规则：再找了解一下：现找到原项目中提到这些。但作为程序员，不把它所有相关逻辑读懂，都感觉不明白它说的很多是什么意思



配置用户选项	选项的功能说明
<b>2 为常主</b>	是常主，就常主会比较多，尤其打王的时候；否则常主少，尤其打王的时候
<b>单 J 勾到 6, 双 JJ 勾到 2</b>	开历史倒车，增加游戏的无穷乐趣：惊险刺激：逢对家对 J, 谁不想把对方逢自家打到 J, 会想要被对方废到 6 或是 2?
<b>5 10 K 必打</b>	因为比较难打
<b>逢 J 必打</b>	因为上面不能言说的【惊险刺激】，不给任何双方逃跑的好玩机会
<b>J Q A 必打</b>	因为他们所添加的各种乐趣。若是 J 与 Q 合并为 JJ, 则 Q 可以跳过
<b>小光升 1 级，大光连升 3 级</b>	不满 40 算小光，0 分为大光头。。。连升三级
<b>游戏亮牌与反牌逻辑完善</b>	添加闲家反牌时急速填扩张抠牌底分的机会
<b>捡分方扣底，底分翻倍</b>	单扣乘 2, 双扣乘 4, 甩牌或是拖拉机扣，则牌的张数，乘以 2. 如此，才能让捡分方快速超越，以风马牛不能及之势火箭升级。。。

• **【关于 J Q】**，游戏设置：

- 增强的规则：在一些地方流行一 J 到底、Q 到半的玩法。
- 庄家在打 J 时，如果下台，并且最后一把被 J 抠底，那么此庄家再上台时将从 2 开始打。
- 庄家在打 Q 时，如果下台，并且最后一把被 Q 抠底，那么此庄家再上台时将从 6 开始打。

• **【亮主规则】**：可以选择是否允许自反，加固和亮无主

• **【亮牌规则】**：（注：以打 8 为例）**【这里有狠大的游戏逻辑改进，和游戏用户体验提升空间】**

- 在发牌过程中，第一次亮出的 8 的花色作为主牌花色。
- 有以下几种情况可改变或加强主牌花色：
  - \* 自保
  - \* 反主
- 以上后三条以先出现者为准。
- 若发牌结束仍无人亮牌，则以底牌第一张的花色作为主牌花色。

• 以上三点应该是自己着重想要改进的地方

• **【轮庄规则】**：为创造出好玩儿的玩法，这里是可以优化改进的。对家的本意是，两人合作，快速升级，所以需要两者配合。不需要，或可以配置不规定严格的顺序，给予他们无数无限合作可能，给予对方继续反副反主的机会，增加游戏趣味。

- 开局中，双方争庄，先亮者为庄家。
- 庄家升级时，下一副牌由其对家当庄家。
- 闲家上台时，下一副牌由此副牌的庄家的下家当庄家。

• 其它这里没有列出来的，主要是我现在还不曾了解那些是在说什么，比如下面网络上提到过的：提供六种配置选项：**【允许自反】**，**允许对家保**，**允许反无将**，**A 必打**（是为什么呢，K 易跑光，不好捡分？）等

• **【点击触屏、用户交互的性能优化】**：需要优化。玩家就算玩得不久，一直点鼠标，也是痛苦的事。需要 AI 辅助，智能帮助用户出牌，让鼠标点击、选牌灵敏、反应快。

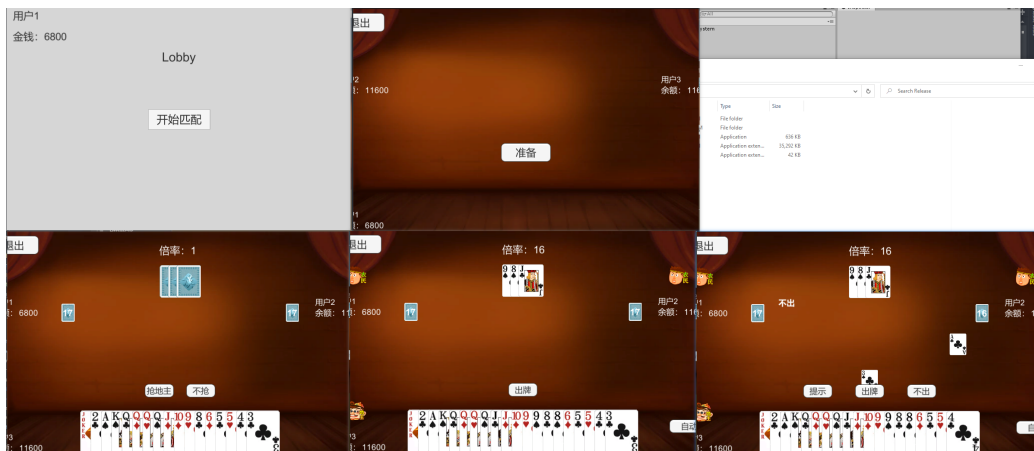
- 原游戏应该是桌面游戏，所以会有快捷键设置。但手游，就需要自己将触屏设置优化出来
- **【去想】**：怎么才能既有良好的用户触屏点击的反应灵敏性，又不失传统方式紧密摆牌？因为不同于桌面游戏，鼠标可以精准点击到位，手游上手手指触屏的射线检测，视图上摆放过于紧密可能不利用射线检测成功，点击灵敏度降低。需要好好考虑一下手游上如何实现优化。这处牌又比麻将窄了很多，确实不好检测。



- **【逻辑设计，用户意愿】**：逻辑上，为能实现以上种种好玩玩法，游戏逻辑需要 **规定，约束严格的反牌规则：从高到低为【王黑红梅方】**，就是别人叫方块的主，其它都可以反，但若是已经反到黑桃，接下来就只能反王或说是常主。允许捡分方按照以上规则反牌，这样才给给予捡分方底牌放 80 分，拖拉机扣底，火箭升级的机会。规则明确，公正。现游戏中一个 **【“流局”】** 界面，抹杀了这一切好玩儿的过程与结果，太不好玩了。。游戏界面，也需要必要的文字提示等，帮助玩家理解游戏中的这些好玩儿规则，让玩家上瘾。。

## 2 主要参考项目：

- **【ET- 斗地主游戏】** 年代久远，ET 框架的版本古旧。手游风格的界面设计
- **【ET- 卡五星游戏】** 年代久远，ET 框架的版本古旧。主要看牌类相关的房间内的逻辑
- 分支 7.2 与主分支的区别：都有最开始必须取消主相机。区别是进入地图时，7.2 如果再激活相机，可以看见地图，小人也能跟随鼠标右键动。主分支可能还需要我再看一下，为什么主分支进入不了地图。两个分支对比一下，主要是让自己放心，现所使用的主分支应该没有 blocking **【BUG:】** 存在。去看下主分支，可以显示地图吗
- 消息，没再区分热更新的消息，分成了内网与外网消息
- 截几个斗地主游戏里的流程图供自己写时参考：



## 3 项目主要进展

- 因为两个参考项目的版本都很古旧，现用最新版本的 ET **【我觉得这里，最好我还是先去试 ET7.2 官方版本可能更好】**就以自己能有的相对快速，按照现有的理解，把能重组的重组出来。不理解的，再去看下两个项目分别的实现逻辑。
- UI Lobby 的三个命令按钮很好创建，但因为接下来的逻辑没能创建连通，程序仍然是运行不通，出报错。接下来应该是一段程序都极可能报错。但会试着修改每个界面上的错，把这个游戏的重构任务完成。
- 现在的游戏主要是参考斗地主游戏，并根据最新 ET7 框架的重构来，来重构自己的游戏，会在现各种文件报错的基础上一一修改掉所有的错，直到自己的游戏重构结束。因为对自己有绝对的自信，相信自己，就不怕它会出现的各种报错。真正改起来，也是 a-piece-of-cake 一口咬下。。拿下。。**【爱表哥，爱生活!!! 活宝妹就是一定要嫁给亲爱的表哥!!!】**

- 程序本身感觉不到困难，会试着解决过程中遇到的所有问题。只是今天状态不够，晚上会给妈打电话，会试着建几个 Unity 上会要用到的预设，再不行，今天晚上就早点儿休息。**【任何时候，活宝妹就是一定要嫁给亲爱的表哥!!!】**
- 感觉程序的逻辑现在已经比较简单了：但是仍然需要花点儿时间，来熟悉这个重构过的 ET 7 框架。今天下午会熟悉一部分，剩余未完成的明天上午来完成。今天熟悉划重点，与自己游戏的重构相关的重点。比如 **先前的多服务器系统变成为现在的服务器的路由系统**，，不同层级的路由？网络调用下的各种 Helper 类，比如 LoginHelper, 其它 Helper 等

## 4 残余的几个小 bug

- 当进入 UILobby 的时候，要把前面的 UILogin 删除。应该是有逻辑，只是因为我现在的 **【BUG:】**，后面就座地执行的逻辑中断了