

Developing a Java Game from Scratch

孙文戈¹

1. 南京大学计算机科学与技术系, 南京 210023

E-mail: 191220102@smail.nju.edu.cn

收稿日期: ; 接受日期:

国家自然科学基金 (批准号: 00000000, 00000000, 00000000)

摘要 本文有关 Java 高级程序设计课程第八次作业, 作为葫芦娃游戏大作业的报告。本次游戏主要基于曹春老师教授的面向对象思想, 使用 JavaFX 设计图形界面, 并使用到 java.io 与 java.nio 完成存档与网络对战的设计。下文将介绍游戏有关的设计理念与问题, 并对于本学期所学课程进行总结。

关键词 Java, 葫芦娃, JavaFx, IO, NIO

1 开发目标

本次大作业主要有三个部分, 分别是基于 jw04 的迷宫任务改造为一个 ruguelike 的葫芦娃与妖精两方对战游戏, 同时实现多线程; 实现游戏保存功能, 使用 maven 管理项目以及编写 junit 单元测试; 实现网络对战。为实现以上要求, 我主要基于 jw04 的作业, 运用面向对象的知识, 学习 JavaFX 用法代替之前的 ASCII Panel 实现 GUI 页面, 使用 IO 完成游戏存档, NIO 实现网络对战。

下面是关于我的游戏具体介绍与其灵感来源。

1.1 游戏介绍

本次作业我写的游戏名为 The story of the hulus, 即讲述葫芦娃们的故事。主要剧情为蛇精诱骗和葫芦娃们重新划分山界 (占格子游戏, 即人物移动会将走过的格子染色, 人物只能走自己的颜色的格子与未被走过的白色格子), 并在结束之后用剑攻击所有的葫芦娃, 使得全地图仍为蛇精所有 (全部恢复为蛇精的灰色)。同时为游戏增添的多人模式中, 将占格子游戏改为最高可以支持九人 (贴图只找了九个, 理论上限为地图大小) 同时在线联机游玩。

1.2 灵感来源

不知道从哪里看到的一句话——“游戏的意义在于讲一个好的故事”, 所以此次我想通过作业来写一个有剧情的游戏, 来讲述一个蛇精要阴谋诡计打倒葫芦娃们的故事。

其实最开始我的目标并不是这个。在作业提及继续完成游戏时，因为 jw04 获奖的经历，我想继续在游戏中加入能让人耳目一新的元素。我很自然地联想到拳皇这种经典格斗对战游戏，里面攻击攒能量释放绝招的画面引起我的注意。于是我最先的设想是，设计一个双人对战，在他们移动攻击的过程中会积攒能量条，能量条满释放绝招。释放绝招时我想使整个游戏窗口播放技能释放动画，如拳皇里放绝招的背景，圣斗士里天马流星拳的招式展示。当时就感觉这种想法设计出来展示时候会比较 amazing 吧。

但是由于迟迟没有行动，新的作业要求“多人对战”发布，让我不禁联想到一些以前我玩过的多人在线游戏，比如 Pummel Party，一个集合各种趣味小游戏的派对型游戏。它里面集成了各种类型的小游戏，玩家们每回合会进行其中一项游戏进行竞技，根据每轮游戏得分最终得到的总分决出名次。由于许久没玩，比较容易改编成课程作业易于实现的我那时候只想到了占格子游戏（前文有介绍），当时觉得占格子实现起来应该比较简单，便初步把作业形式定为 pummel party 这种的集成式游戏，可以自由选择游戏模式（格斗与占格子）。

又是迟迟没有开始，也因为个人觉得单纯的将两个不相关游戏塞到一起供选择比较生硬，为此要取舍苦恼了一段时间。不过有天早上刷牙时我突然想到了可以以故事形式，情节场景转游戏场景实现不同游戏的切换来消除两个游戏间切换的生硬。于是本次作业所设计的游戏模式雏形诞生了！

2 设计理念

2.1 总体设计

本次作业依托于作业四的迷宫游戏继续完成，故总体对象设计依旧以作业四的内容为框架，即依靠 world 类完成地图生成与生物构造，screen 类完成图像展示与总体调用生成。为了实现更好的 GUI 效果，本次作业放弃了作业四中的 AsciiPanel 方案，转而使用 JavaFx 来实现图形化，同时新增 scene 类来负责图形化界面的生成与界面跳转。在 world 中，分别由 Creature 完成对生物属性以及移动的定义，Tile 定义地图每个方格的基本属性，World 完成场景搭建。同时简化了 screen，仅使用 Playscreen 来实现整体类初始化调用与场景变化。

同时由于剧情式的游戏设计，较为复杂的场景变换让我决定在完整的游戏流程中加入存档机制，而多人对战在占格子游戏场景中单独实现。下面分别介绍这两个方面的设计。

2.1.1 正常模式

在正常模式游戏设计上，为实现多场景设计，分别设立了 StartScene, PlayScene, PlotScene 来完成开始场景、游戏场景、情节场景的构造。

在开始场景中，会选择是否加载存档。当加载存档时，便 io 加载保存的文件跳转到对应的场景。同时在游戏窗口关闭时会将对应该场景信息如地图信息、生物位置等保存在存档文件中，以便下一次开启游戏时加载。

2.1.2 多人对战

在多人对战模式中，将源代码进行功能分割并加上 nio，生成服务端和客户端两类。

整体思路为在建立连接后,由服务端向客户端发送地图与生物信息,客户端接收后负责展示,并接受键盘输入传给服务端运行。大致可以理解为服务端由原来的 screen 和 world 类构成,而客户端即为原来的 scene 类。同时为进一步简化,将 scene 类仅保存 PlayScene,当判断游戏结束时跳出弹窗来宣告结束。

2.2 设计优点

由于依托于作业四的框架,而作业四的框架由老师提供,这给我提供了最稳定的面向对象保障,即使在完成作业过程中有许多设计不合理的地方,也能保持最基本的面向对象设计结构。多个游戏场景均有对应的 scene 的支持,面向对象设计提高效率,降低维护修改成本。

其次,使用 JavaFx 的框架,相比简单的 AsciiPanel 能够实现更加多样且自定义的图像支持,优化了游戏外观图像设计。

在多人游戏的设计中,由于采用拆分的结构,所有计算与生物移动地图变化等都由服务端完成,可以保证多个客户端能够同时接收到相同的数据,即保证了数据一致性。并且服务端只需要保存基本的如地图布局和生物状态数据,以及简单的多线程操作与接收键盘操作等,本身负载小,且减少与客户端通信的数据规模,能够提高数据传输效率,降低网络延迟的影响。还有 selector 的使用可以减少线程开销带来的性能问题,节省计算资源。

3 技术问题

3.1 通信效率

由于本次作业多人对战的要求需要实现网络通信,自然需要考虑网络速度限制的问题,如当传输数据过大时会出现卡顿现象等。因此将较大的资源文件都放在客户端本地,由客户端加载。服务器只需实时提供各种素材的位置,客户端即可通过位置数据加载生成动态页面。而这样的处理方式仍然存在优化空间,即每次服务端都是将全部地图位置与生物位置数据传输,而每次传输时其实只有少数地图贴图会发生变化,可以进行优化使得服务端只需传输发生变化的位置信息。

3.2 并发控制

为避免两个生物走到同一个格子上,需要在生物移动时判断下一个格子上是否有其他生物。但是单纯的如此判断仍然不能解决线程冲突的问题,如两个生物同时想走进同一个没人的格子,此时便会发生线程冲突两者到同一个格子上。为避免此类线程冲突问题,需要给线程上锁。即在生物需要查询前先上锁,查询结束后再释放锁,这样既可避免多线程同时检测同一个格子为空。

3.2.1 输入输出

为完成游戏存档功能,本次作业涉及到文件 io,即游戏关闭时将游戏状态信息(场景号,地图信息,生物信息等)保存到文件中,待加载游戏时读取文件。在起初尝试中会遇到 io 报错,原因为自建的类没有实现 serializable,即 io 操作中无法将设计到的类序列化。本以为根据报错提示给自建的类添加 serializable 接口即可,结果还是遇到了报错。查询可知 javafx 的 color 类无法序列化,经

询问老师及自己思考后,发现并无必要直接保存 world 与 creature。采用给不同地形标号以及记录生物位置的方式即可通过少量信息获取全部存档,相比之下原本想将类全部保存的行为反而会使数据冗余,影响加载速度。同时有了这里的经验,在网络通信中同样只需传输少量位置信息即可,极大地提升了 io 效率。

3.2.2 面向对象

维护简单:面向对象设计的一个特征就是模块化。实体诸如方格、生物、场景可以被表示为类以及同一名字空间中具有相同功能的类,可以在名字空间中添加一个类而不影响该名字空间的其他成员。这种特征为程序的维护提供了便捷性。

可扩充性:如果有一个具有某一种功能的类,就可以扩充这个类,创建一个具有扩充功能的类。

代码重用:功能是被封装在类中的,类是作为一个独立实体而存在的,因此可以很简单的提供类库,使代码得以重复使用。

4 工程问题

不过还是由于依托老师构建的作业四的优势,作业四中采用了工厂方法模式,在本次作业中也顺延继承了下来,使用工厂模式来生成各类对象。使用工厂模式的优点主要为以下两点:

解耦:通过工厂模式可以把对象的创建和使用过程分割开来。比如说在 PalyScreen 中想要生成葫芦娃,那么我们无需关心葫芦娃是如何创建的,直接去工厂获取就行。

减少代码量,易于维护:通过工厂模式,我们把对象创建的具体逻辑给隐藏起来了,交给工厂统一管理,这样不仅减少了代码量,以后如果需要修改代码,只需要改一处即可,方便日常维护。

但很可惜的一点是,由于本次作业开始较晚,遇到各项课程作业堆积以及临近期末考试的问题,在多人对战的模式设计中,没有采用老师推荐的 reactor 设计模式,对此感到非常遗憾。

5 课程感言(对课程形式、内容等方面提出具体的意见和建议)

不知不觉一学期的 Java 课程已经要结束了,本篇作业报告也水到了尾声。之前以为写满 5 页最好凑字数的地方就是课程感言这里,但轻轻敲着跟前老师送的高斯 ALT-83D 键盘,有很多话想说,又不知道该说什么。课程形式与内容方面的意见和建议,相信其他同学都已经说的很好了,也不需再赘言。这里想到一些这门课程对我自己的改变,胡乱写下来也算是对自己一学期课程学习生活的记录。

在本学期接触此门课程与 iOS 智能设计程序之前,我对于所学课程的态度可能大多是这门课好难赶紧水过去吧和这门课还好水过去比较容易,因此糟糕的成绩导致早早地被保研拒之门外。虽然心里一直有着对读研的向往,由于畏难情绪和对考研失败可能性的抗拒,以及学习过程的枯燥,会拿着“对科研没兴趣”之类的口号来搪塞自己,准备着未来的就业。但这学期的课程,让我有了不一样的感觉。

在最早的可视化排序,在宿舍比较大家 256 规模的排序时间要多久,还用舍友的电脑挂着一个多小时没跑完,然后发现影响时间的不止是算法本身的快慢,还有每次排序过程的输出会占用大量

时间,因此使用交换最少的选择排序效率极高。虽然这只是很小的一件事,也与作业本身要求并不直接相关,但却非常触动我。不知如何形容,通俗一点说,大概是这里很能发现学习的快乐。

会比较喜欢过去发生的一些偶然事件。比如跟舍友聊天问他作业四如何设计的时候,想到加入双人竞技或许不错;比如起床刷牙时想到以情节的形式将想到的不同游戏模式结合;比如同学没来上课导致键盘落到了我手上(狗头保命)。

非常喜欢一学期以来的老师和同学们。从一开始水群看这个老师哈哈真有趣,到学期末“嘿,我的春春”。“老师对于学生的影响真的是很大的”,在我说要考研时,女朋友如此评论。那当然了,不过我肯定不是贪图老曹的五千助研费和mbp。大家都会很在乎别人的评价,作业四要求将运行录屏发到群里,每个人的作业老师都给了一个单词的赞赏。”splendid”,我的是这个词。不止是老师的鼓励,很多作业录屏在发到群里时,会收到许多群友的夸赞。完成作业的自豪感,在此时极度“膨胀”。我很爱这个班的氛围。

回望这一学期以来的经历,遇上喜欢的老师,好像学习也可以是一个有趣的过程。“对科研没兴趣”貌似也不再足以作为逃避的借口,虽然课上还有很多东西没听懂,Java学的也不如课上的很多同学,最后考试也比较拉胯,不过厚脸皮还是可以具备一下的,希望能继续当曹老师的学生,不是这学期,不止下学期。

致谢 感谢曹老师一学期在学习生活上的指导。感谢舍友们在我遇到问题时的帮助。感谢女朋友在我写作业时候的陪伴。