# 设计文档

主体设计思路：

1. 将用户点击信息转化为” ^([1-8])([wasd])$”命令传入GameController中；
2. 逻辑部分，Controller为主体；GUI部分，JungleGUI为主体；
3. 逻辑部分与GUI部分的耦合通过两者都有的二维char数组left和right来沟通；GUI的作用为输入指令，读入逻辑部分的地图，进行显示；

一、 面向对象的设计

逻辑部分：

GameController用于接受GUI传来的信息，并进行转化，对自身进行刷新；

包括了两个char的二维数组left和right（通过getter来进行传递，通过updateLeftAndRight进行更新），和turn （有get和set方法） ， parse用于将GUI传入的坐标进行转化；还有就是undo，redo 和 restart方法分别用于悔棋，取消悔棋，和重新开始，可以被GUI直接调用；而update方法通过传入坐标对棋盘进行更新；

Board类 采用了单例模式，核心变量是board ； 附属关键变量是tiles和两个Animal数组；通过getInstance得到board；

Tile类是用于描述地形，拥有坐标，动物和地形属性；

Animal类，用于描述动物；它的子类有AnimalWithoutSkill ， AnimalCanJumpRiver ， AnimalCanSwim ， 而它们的子类就是各种动物了；

GameHistory用于保存历史信息；

GUI部分：

JungleGUI最核心的一个类，实现frontScene ， middleScene ，finalScene 和helpScene的切换；舞台是私有的myStage；

其中：

用于支持frontScene的部分：利用Front类的getInstance（）来得到我们所需要的StackPane；

用于支持middleScene的有：middleScene和middlePane；这个pane是由背景图片和上方的buttonArray组成的，然后呢，为了使button更炫一点，设置了两个处理器 分别使MouseEnter和MouseExit，如果鼠标悬浮在button上方，button的背景色会变化；

用于支持finalScene的有：两个二维char数组left和right，根据这两者来制造中心位置的animalPane，制造方法为animal Pane（）；中央的animalpane制成后再通过方法animalPaneWithMargin（） 来制造一个有边框和上方菜单的pane ，而 topOfGame（）是制造上方的菜单和玩家的方法；在点击的过程中，通过from和to两个一维int数组传递位置信息，而someOneIsMoving用于判断是否是第一次点击，当然还有一个fromOrToHandler用于处理点击事件；每次点击完后，通过updateFinalPane（）来刷新界面；SmallSquare则用于制作所有的动物小方块，它包含了声音等信息；

用于支持helpScene的有：主要是helpPane（）方法，传入的参数用于判断应该返回最后一个界面还是中间那个界面。

最后基础功能已经全部实现。

二、 图形化界面

图形化界面的主体功能由类JungleGUI实现；

JungleGUI主体则是四个Scene

1. frontScene：作用是用于引入游戏；由背景图片和时大时小的label组成；嗯，挺有逼格的，参考傲日与黑暗森林游戏；点击后进入下一个界面；
2. MiddleScene：作用是进入游戏和对游戏进行设置的窗口；大的主体是一个StackPane，这个主体是背景和塞着四个button的GridPane组成的；四个button的作用分别为开始新游戏，继续原来的游戏，帮助与设置（设置还没有做）和退出游戏；
3. FinalPane：也就是最终的游戏界面了；它首先是一个BoarderPane，由上面的topOfGame和左边的透明图片和中心的动物棋盘组成；首先介绍topOfGame ， 它是一个StackPane：下面是设置的label 显示左方还是右边行动，它的上面是两个player和一个MenuButton（菜单里面就有悔棋，取消悔棋，重新开始，帮助与设置，建立存档，还有返回）；然后，中心的AnimalPane 通过点击进行传递信息，每次都刷新一下；
4. HelpScene：主体是一个BoarderPane，由左边的button和右边的label组成；

三、 拓展功能

1. 图片还可以；
2. 音效还可以；
   1. 编程中遇到的问题和解决的策略
3. 如何得到用户点击的数据，

后来学了setOnAction;

1. 用户点击的数据如何转换为我们所需要的指令，

首先要分清楚是第一步还是第二步，于是有了someoneIsMoving；

而后出了bug，因为有的地方没有动物，也会有someoneIsMoving，然后就加了一个判断有没有动物的判断；

而后又出了bug，因为有时候是左边下有时候是右边下，然后又加了一个是左边还是右边的判断；

而后，，，当然又出了bug，只不过是其他方面的啦~

1. 用户点击的东西应该设置成GUI还是设置成动物类（将逻辑部分和GUI合并）；

动手写了一下，但是觉得耦合程度太高了，万一出了bug，，，九死一生，可能会牵一发而动全身；所以还是保留了原来的逻辑部分，仅对逻辑部分进行修改。

1. 如何实现场景的切换？

本来想弄一个KeyFrame，没多少秒对更新过了的数据进行图形界面化；后来，勇敢地尝试了一下，发现只要setScene整块都会变掉，豁然开朗；

1. 没有一次是一次成功的；

那就多试几次咯。