



四川大學

计算机网络和分布式系统项目使用说明

题 目 基于socket编程的联机坦克大战

学 院 建筑与环境学院

专 业 力学-软件工程交叉实验班

学生姓名 高云飞(组长)

学 号 2019141410020 年级 2020

学生姓名 冉勤飞

学 号 2020141470100 年级 2020

指导教师 宋万忠

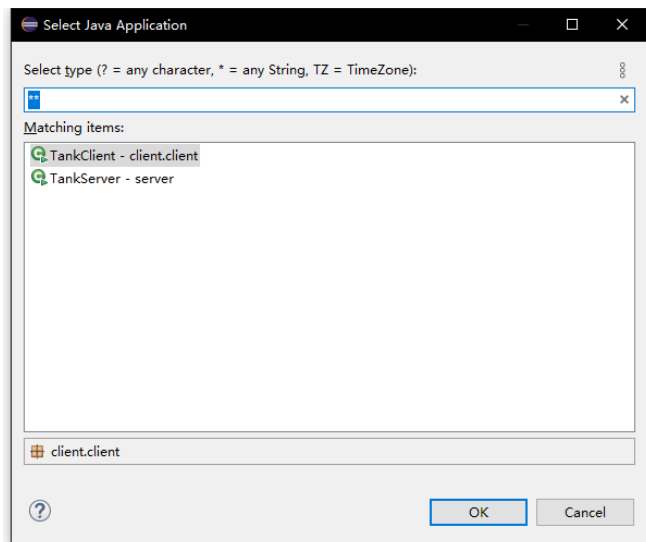
2022年06月10日

目录

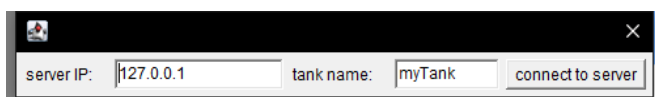
- 一、 项目准备
- 二、 功能说明
- 三、 系统设计
- 四、 实现通信功能
- 五、 后续完善和总结

一、 项目准备

环境：系统：win10,编译软件：eclipse。操作方法：将项目文件导入到eclipse中，对项目点击鼠标右键，选择run as，项目有两部分可执行程序，先在一台电脑上运行server部分，由于没有专属的服务器IP，只能用一台电脑来充当server，IP地址会变，但server端的图形界面的第一行会显示出自己目前的IP地址，后面的行会显示出所有连接到游戏上的客户端的IP地址，之后即可在同一局域网下的另一电脑或者就在原电脑上运行client部分。



输入正确的server IP，设置自己的玩家名字，即可进入游戏。





Eclipse 导入Java项目见百度经验：

<https://jingyan.baidu.com/article/456c463b5931774b583144a4.html>

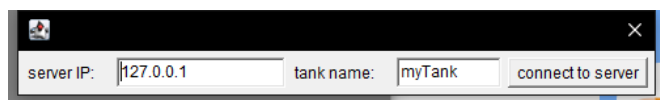
注意设置eclipse字符集为UTF-8以便查看注释。

设置字符集操作：https://blog.csdn.net/weixin_39915081/article/details/114430913

二、功能说明

1.登录

登录只需运行client端，并输入正确的server IP即可。如果是错误的或者不存在的server IP，会产生异常，游戏直接退出。



2.基本功能

坦克：每一辆坦克除了颜色以外没有其他区别，在默认模式下坦克分成两个阵营，分别是绿方和黄方。坦克可以按WASD自由移动，按空格键发射炮弹，拥有共100血量，每次中弹-20血。

移动：地图内坦克由不同玩家操作，可以根据键盘指令(WASD)上下左右移动，这样的方式来控制坦克的朝向，坦克不会停下来，会自动向着朝向向前走。

开火：玩家按下空格即可发射一枚炮弹，可以连射，每发炮弹在遇到非己方坦克后会造成20伤害，没有开启友伤。

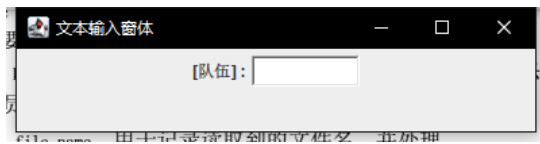
爆炸：炮弹在遇到坦克后会有一系列的爆炸画面，形成一个动画。

地图：图形化地图，有不可跨越的边界。

3.扩展功能

联机：server端负责接转消息，坦克和子弹的位置、方向和坦克的血量、类型和聊天内容等都由server接收并转发给client端。Client端则会把消息实时地画在屏幕上，这样各个client端的信息就相互统一了。

聊天：这里我尝试借鉴《星际争霸》中对于信息的发送功能：按键显示一个对话框，发送消息之后焦点转移回游戏上。这里，玩家按下回车键即可在对战窗体的下方唤起消息窗口，在窗口内输入需要发送消息，输入消息之后按enter键发送消息。可以通过按ctrl键来改变消息的类型，即：消息是友军可见还是全局消息。



结束：我方坦克血量为零时，直接弹出游戏结束信息，关闭游戏结束消息之后会退出程序。

三、系统设计

1.BattleMap

功能：画出地图。

2.Dir

定义方向的枚举类型，上、下、左、右。

3.Explode

播放爆炸动画，炮弹遇到非友方坦克时，会造成伤害并在相遇地点播放爆炸动画。当爆炸动画播放完时，会自动停止。

4.Message

消息类。

5.Missile

炮弹类，存储了炮弹的速度、方向、位置，在地图上画出了炮弹的位置并不断移动，判定碰撞，当遇到的坦克不是己方坦克时，会造成伤害。

6.Resource

资源类，读取了图片文件。

7.Tank

坦克类，在地图中添加坦克。设定了坦克的移动速度、类型、方向、位置和是否发射炮弹等信息，使用键盘事件监听器实现了坦克图像和位置在地图上根据键盘输入进行自由移动、开火、掉血等功能。

8.NetClient

网络方法接口。实现了Client端和server端进行建立连接并通信的功能。创建坦克、分配坦克阵营、监听坦克行为（如监听坦克是否死亡）、向服务器发送或接收游戏数据。

9.TankClient

坦克大战客户端，存储了游戏中的子弹集合、爆炸集合、坦克集合和消息集合。创建了客户端的窗体和聊天窗口。实现了图形绘制、开始游戏、结束游戏、发射子弹、坦克移动、键盘监听、注销窗口以及和服务器通信等功能。

10.TankHitEvent

击中坦克的事件。

11.TankHitListener

坦克被击中事件监听者(由坦克实现)。

12.各Msg

这里我们采用了Java的多态来实现多种报文，实现了自己编写的应用层协议，每一个报文里都有一个“报文类型”字段，来帮助客户端实现对应的报文解析。报文的类型如下图：

```
TANK_NEW_MSG = 1;  
TANK_MOVE_MSG = 2;  
MISSILE_NEW_MSG = 3;  
TANK_DEAD_MSG = 4;  
MISSILE_DEAD_MSG = 5;  
TANK_ALREADY_EXIST_MSG = 6;  
TANK_REDUCE_BLOOD_MSG = 7;  
MESSAGE_MSG = 8;
```

具体实现：首先定义了一个接口类Message，代表了不同信息类型的Message必须实现这个接口声明的两个函数：parse()和send ()。多态实现了多种协议报文，并完成对应的解读。

发送消息的方法：把坦克的相关信息编写成不同类型的完整的消息，根据IP和UDP端口把消息发送出去。通过writeInt(),writeBoolean这样的写进去。

接受信息的方法：先通过readInt()读入消息的第一个数，在我定义的所有协议中，也就是消息类型。消息类型读入后，由客户端来判断应该调用哪种报文的parse()方法来解读消息。消息的解读也是通过readInt()、readBoolean()、readUTF()这样的方法来实现的。

13.NormalFireAction、Fire、FireAction

确定子弹位置，发出子弹，并向服务器发送消息。

14.IPGetter

获得网络接口、服务器的IP地址以便显示在屏幕上。

15.TankServer

服务器端，存储着TCP端口号、各个UDP端口号、服务器画布。用窗口的形式显示出来，显示client端的IP地址，接发消息。建立客户端，保证客户端唯一。接受客户端连接进来的线程，将IP、UDP端口记录。

四、实现通信功能

现在来介绍下这个程序如何达成的通信：

我们实现了自己的应用层协议，一旦监听器监听到相关的事件（比如用户按下空格，代表坦克开炮了），将会调用自己tankclient的发送方法，发送一个对应类型的报文。但首先，我们得和服务器建立链接。服务器的图形界面上显示出了自己的IP地址，我们输入它，点击connect to server，现在会建立一个TCP链接，服务器端会提取出TCP报文的关于IP地址，UDP端口，设定的坦克名称等等信息，存到“坦克服务器的”客户列表中，存放了客户之后才能进行后续的广播。

我们的应用层协议用UDP进行传输。服务器接到一个UDP报文之后怎么办呢？服务器将会转发这些报文，转发到每一个服务器记录的客户列表的客户。客户端接到报文之后会进行parse()方法的调用，读出报文的类型、内容，反映到自己的画面上。具体每一个应用层的协议，在此就不再赘述了。

五、后续改进与总结

我们认为，目前的坦克大战项目仍然存在不足。能够改进的地方有：

- 1.游戏逻辑方面，坦克在和地图元素进行交互的时候应该，比如说在碰到砖墙处会停住，发射炮弹应该能击碎砖墙等等，总之就是和地图不能很好的交互。

- 2.由于开发的时间有限，游戏性不强，虽然它实现了联机对战，并可以实时通信，但只有一种游戏模式，后续增加可以加入比如合作模式等新模式。

总结：

通过计算机网络课程的学习，我们学会了计算机之间是如何实现的连接网络，而这次项目更是提高了我们对于网络协议栈的理解。通过运用、实践套接字编程，我们实现了建立TCP链接、发送UDP报文（里面装的信息是自己定义的应用层协议），这样实现了一个C/S架构的小游戏。游戏是一个载体，重要的是这个架构的实现过程。整个项目代码量大概有2100行（包括服务器和客户端），还是蛮大的