CircleBattle--软件设计文档

	学号	姓名
小组成员	12330379	易圣舟
	12330404	张伟越
	12330405	张炜杰
	12330415	郑安恺（组长）
	12330426	钟声波

目录

技术选型

所选技术	理由
开发语言 Python	Python 具有学习使用简单、开发高效的特性，适合原型开发，对于我们小规模游戏项目的编写开发有很好的支持。
开发框架 Pygame	Pygame 封装自 SDL 引擎的游戏模块，包括图形图像处理、声卡驱动并且对硬件输入也提供了良好的封装，而提供的相对底层可定制性高的函数允许实现丰富的游戏元素。
开发工具 PyBox2D	PyBox2D 是开源的 Box2D 平面物理引擎的 Python 封装版，提供真实高效的平面物理仿真功能，丰富的物理特性和友好的学习文档是游戏开发爱好者很好的选择。

架构设计

[架构图见附录]

游戏系统核心包括网络层、控制输入层、游戏演算层、渲染层和作业逻辑层，资源层为整个系统提供全局静态资源。

网络层包含三个子模块，HTTP 模块处理 WEB 服务器数据上传、主机模式和客户机模式负责不同机器间的渲染信息和控制信息交换。网络层运作在作业层的网络通信工作线程上。

控制输入层负责收集本地硬件和来自网络主机的控制输入信息，供系统其他模块调用。

游戏运算层包括游戏对象实体集合、游戏演算模块、本地渲染队列，通过更新实体机和状态信息再进行游戏逻辑演算得到渲染指令序列存入本地渲染队列等待系统取出。

渲染层包含序列解析和图形渲染两个子模块，负责接收渲染指令序列并将解码序列与图形资源匹配，再调用 Pygame 渲染接口绘制屏幕。

作业逻辑层包含网络通信、游戏逻辑和渲染三大功能模块的工作线程以及线程间的共享数据队列。

模块划分

系统核心概念模型 [见附录]

CoSource.py

全局静态资源模块，负责通用数据的初始化并提供全局常量。

CoGame.py

游戏系统的核心模块， 负责管理所有功能型组件模块（CoController、CoPhyWorld、CoNetwork、CoRender），包括对所有组件模块的初始化、数据交互、线程分配任务处理以及资源回收，并负责提供直接的命令行菜单交互。

CoRender.py

游戏渲染模块，功能型组件之一，负责读入游戏对象数据并根据角色状态信息和图片映射 ID 获取并渲染图片帧至用户屏幕。

CoNetwork.py

网络通信主模块，功能型组件之一，负责与远程主机的信息交互以及云端服务器的数据上传，下辖网络主机模块、客户机模块以及云服务器 HTTP 协议通信模块。

CoNetworkClient.py

网络通信子模块，负责提供客户机模式下的通信规则。

CoNetworkServer.py

网络通信子模块，负责提供主机模式下的通信规则。

CoNetworkHTTP.py

网络通信子模块，负责提供 HTTP 网络访问协议。

CoController.py

系统输入控制信息模块，功能型组件之一，负责接收并缓存外部控制指令，包括通过 Pygame 接口函数持续从本地硬件输入设备提取输入指令（键盘和鼠标）以及从网络通信模块 CoNetwork 接收远程主机指令。模块对系统内部提供调用接口，获取控制数据。

CoDataCollector.py

数据采集模块，负责运行时修改类和对象的方法并注入检测代码，实时提取游戏运行数据并缓存。模块对系统提供调用接口，获取已收集的数据。

CoPhyWorld.py

游戏物理世界模块，实体数据存储组件和功能型组件之一，负责游戏实体数据的缓存以及根据物理演算和游戏逻辑对游戏实体数据进行更新，提供游戏数据的提取方法。

CoGameEntity.py

游戏实体模块，包含当前游戏所有可用角色类的声明和定义，实体类包含每个游戏角色的基本状态信息以及关联的图片 ID，提供游戏系统内角色数据与相应图片的关联信息（映射 ID），维护每一个角色对象数据以及资源。[模块中类关系图见附录]

CoGamePass.py

游戏关卡模块，负责识别并读取自定义的游戏关卡代码文件，引导游戏世界系统生成游戏角色对象进入预备状态。

GamePass.py

游戏关卡文件，关卡的实际配置代码。

软件技术

AOP

运用 Python 装饰器模式和 AOP 技术在程序运行时进行代码注入在程序关键代码块执行前后提取运行数据，达到无需修改源码的前提下进行游戏数据采集的目的。

设计模式

1）装饰器模式

模块 CoDataCollector

通过闭包对类的方法进行运行时注入，在需要检测的类方法执行前后安插消息记录函数捕获该方法的传入参数和返回值。

2）外观模式

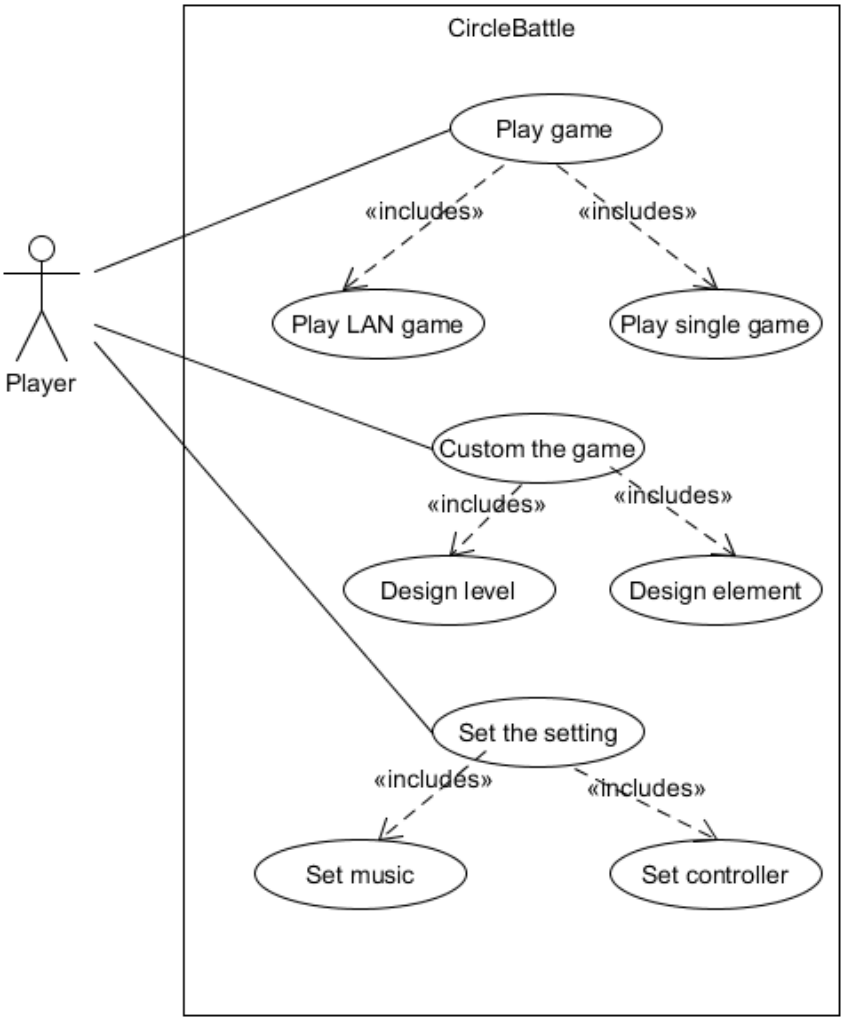
模块 CoGame

创建一个 CoGame 游戏系统实例之后，通过 Init()方法初始化系统，系统所持有的其他组件实例如 CoController、CoRender 能够一同进行初始化。

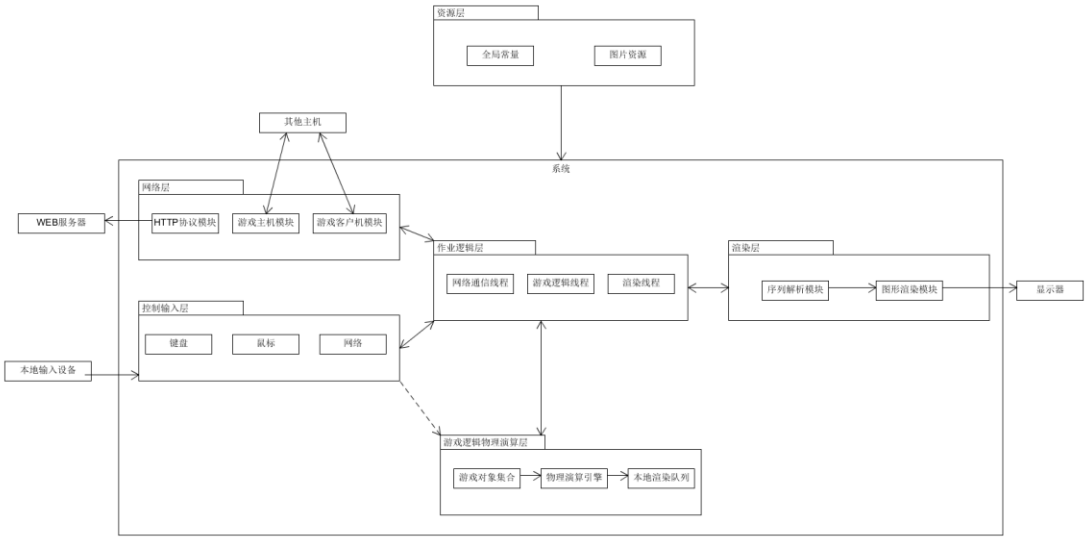
附录

模型

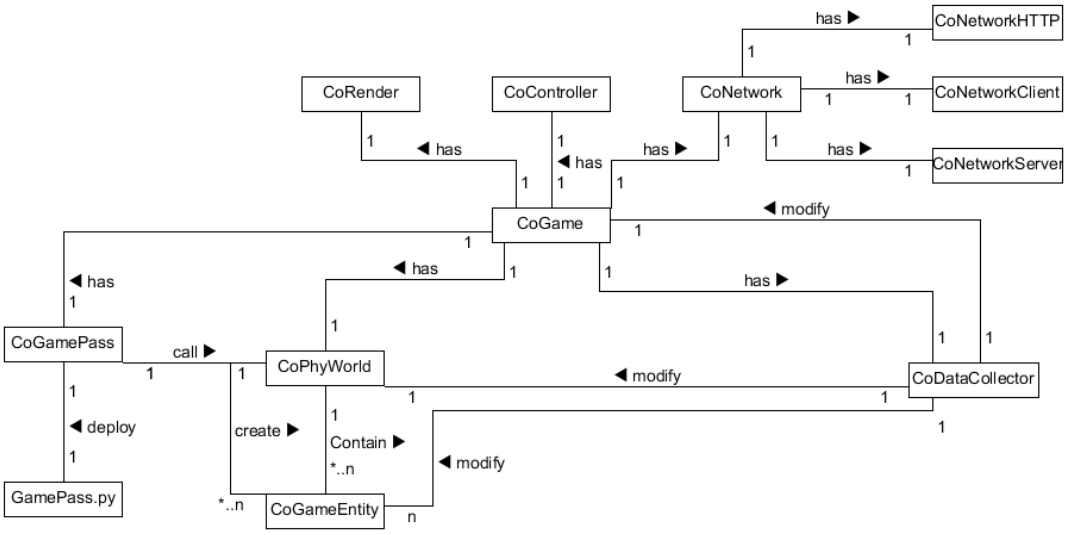
用例图



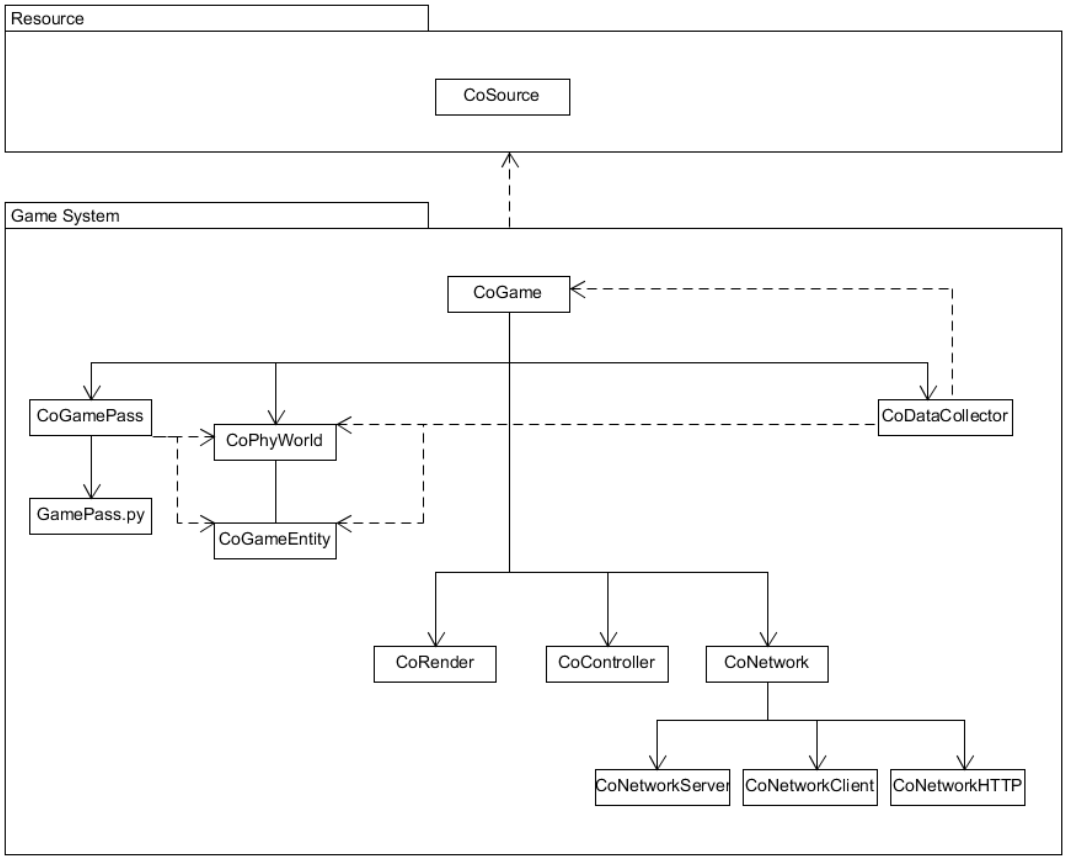
架构设计



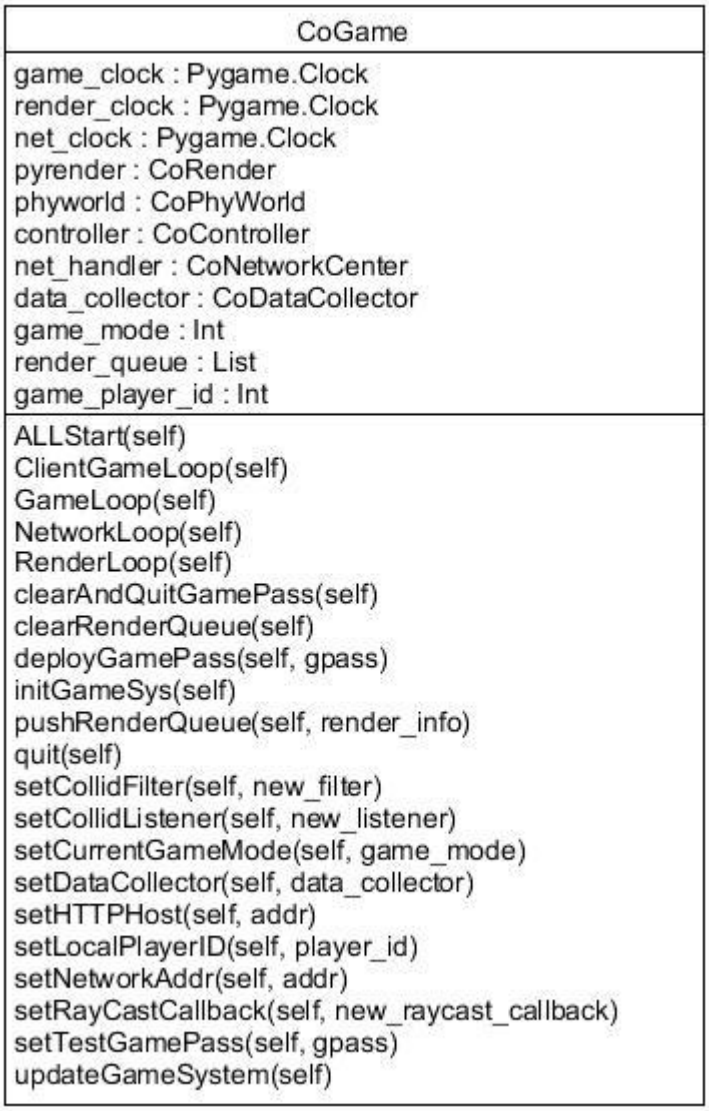
概念模型



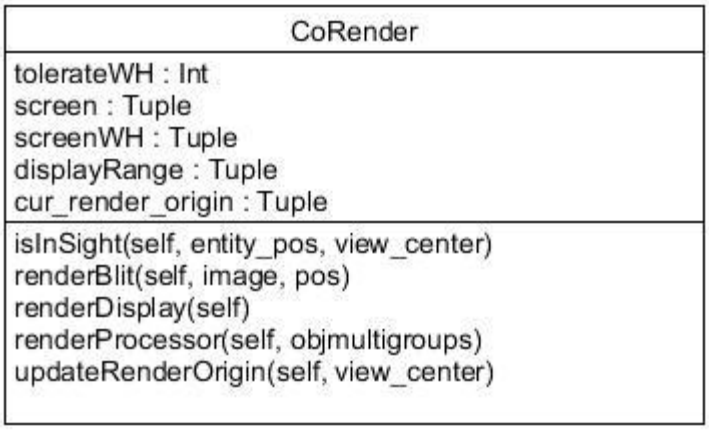
类结构关系，模块间的数据引用关系以 → 指示


Resource
CoSource
Game System
CoGame
CoGamePass
CoPhyWorld
CoDataCollector
GamePass.py
CoGameEntity
CoRender
CoController
CoNetwork
CoNetworkServer
CoNetworkClient
CoNetworkHTTP

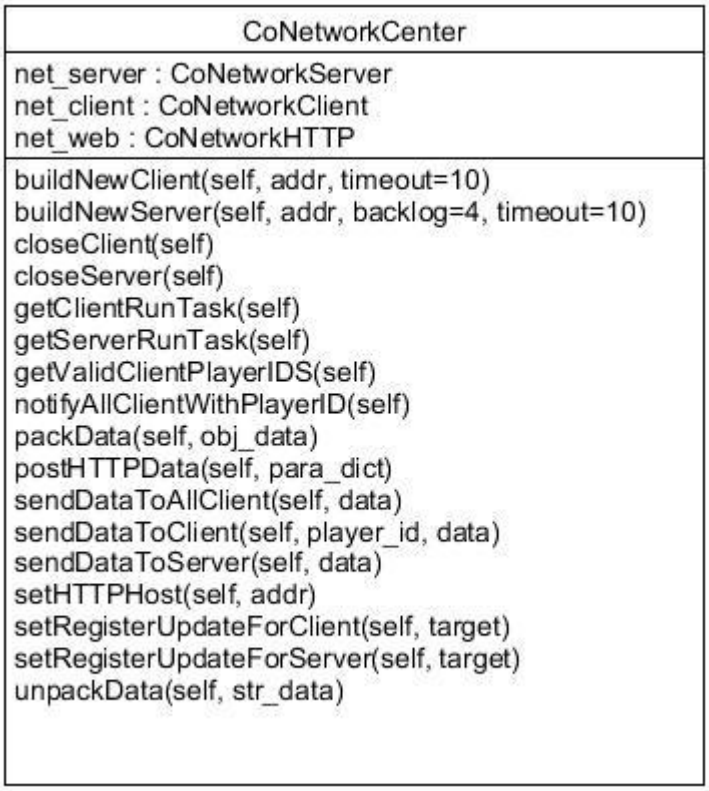
CoGame.py


CoGame
game_clock : Pygame.Clock
render_clock : Pygame.Clock
net_clock : Pygame.Clock
pyrender : CoRender
phyworld : CoPhyWorld
controller : CoController
net_handler : CoNetworkCenter
data_collector : CoDataCollector
game_mode : Int
render_queue : List
game_player_id : Int
ALLStart(self)
ClientGameLoop(self)
GameLoop(self)
NetworkLoop(self)
RenderLoop(self)
clearAndQuitGamePass(self)
clearRenderQueue(self)
deployGamePass(self, gpass)
initGameSys(self)
pushRenderQueue(self, render_info)
quit(self)
setCollidFilter(self, new_filter)
setCollidListener(self, new_listener)
setCurrentGameMode(self, game_mode)
setDataCollector(self, data_collector)
setHTTPHost(self, addr)
setLocalPlayerID(self, player_id)
setNetworkAddr(self, addr)
setRayCastCallback(self, new_raycast_callback)
setTestGamePass(self, gpass)
updateGameSystem(self)

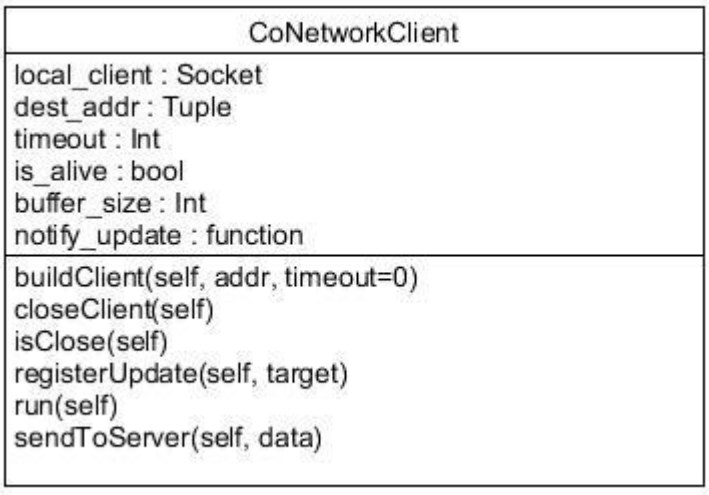
CoRender.py


CoRender
tolerateWH : Int
screen : Tuple
screenWH : Tuple
displayRange : Tuple
cur_render_origin : Tuple
isInSight(self, entity_pos, view_center)
renderBlit(self, image, pos)
renderDisplay(self)
renderProcessor(self, objmultigroups)
updateRenderOrigin(self, view_center)

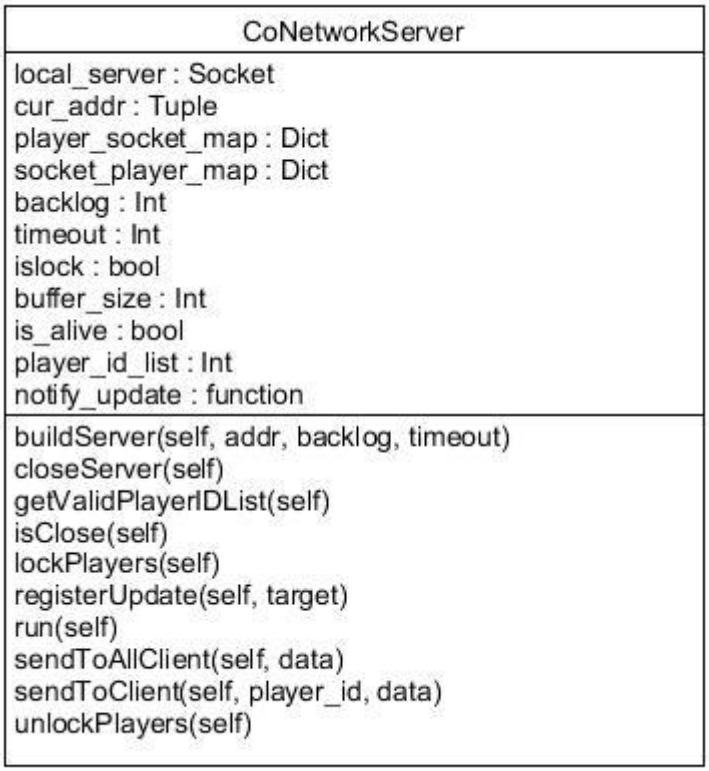
CoNetwork.py



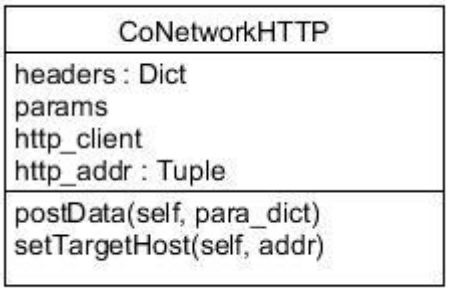
CoNetworkClient.py



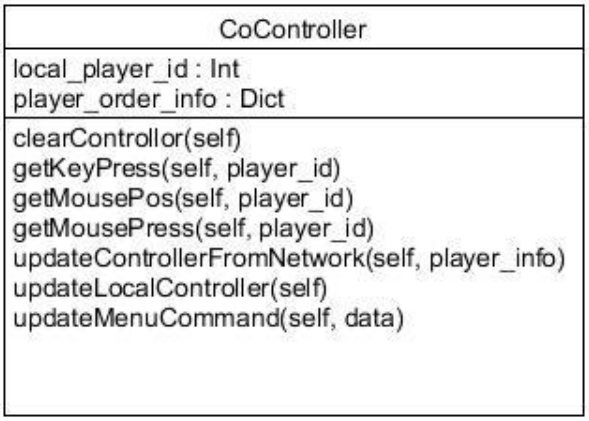
CoNetworkServer.py



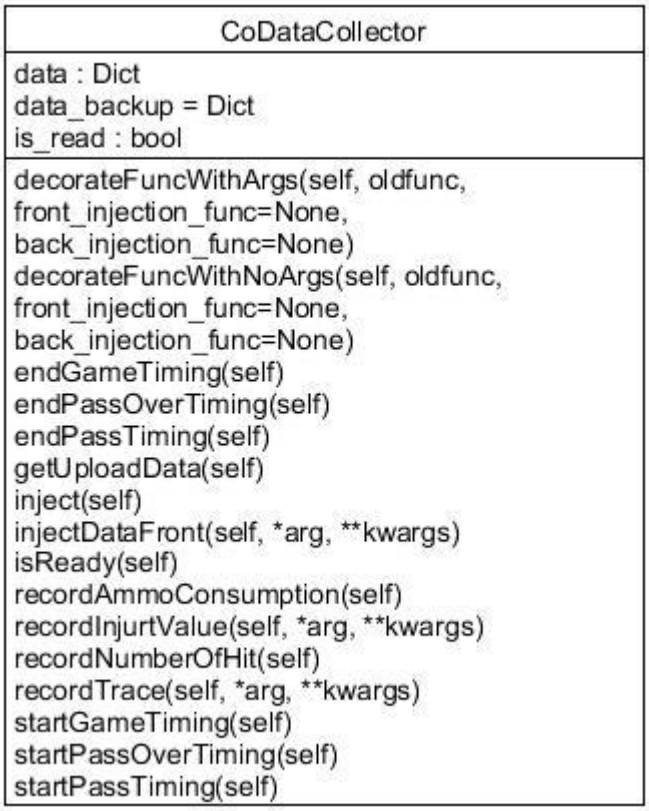
CoNetworkHTTP.py



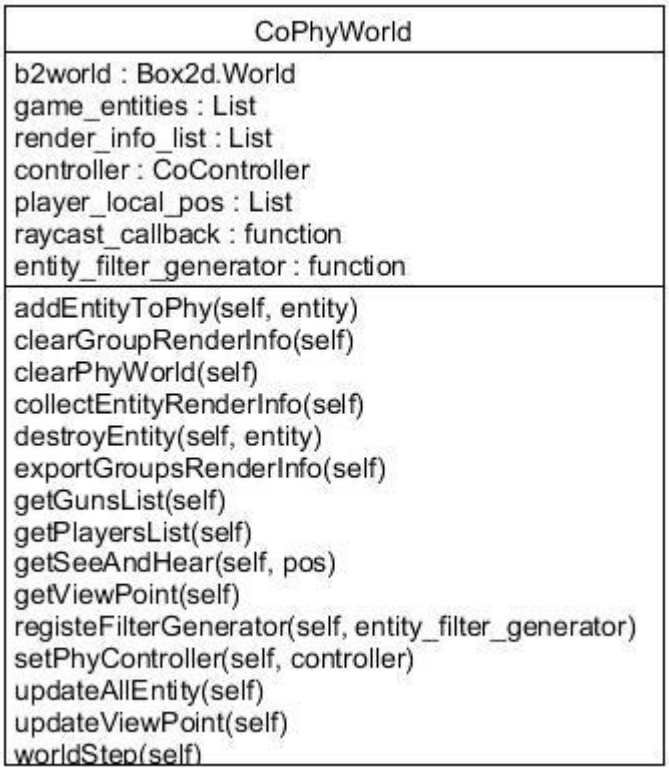
CoController.py



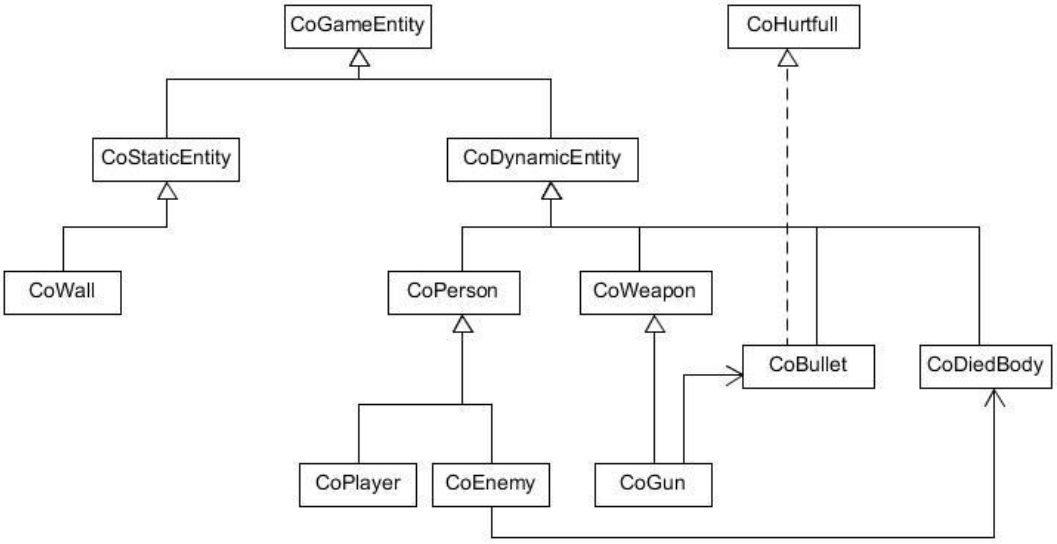
CoDataCollector.py



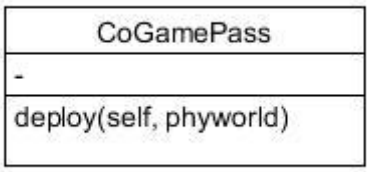
CoPhyWorld.py



CoGameEntity.py



CoGamePass.py



源码清单		**备注**
img	----------	资源文件
CoControllerM.py		
CoDataCollector.py		
CoEntityBase.py		
CoEntityBullet.py		
CoEntityDiedBody.py		
CoEntityEnemy.py		
CoEntityGun.py		
CoEntityPerson.py		
CoEntityPlayer.py		
CoEntityProperty.py		
CoEntityWall.py		
CoEntityWeapon.py		
CoGameEntity.py		
CoGameM.py		
CoGamePass.py		
CoMain.py		
CoNetwork.py		
CoNetworkClientM.py		

CoNetworkHTTPM.py
CoNetworkServerM.py
CoPhySystemProperty.py
CoPhyWorldM.py
CoRenderM.py
CoSource.py
CoStateMachine.py
GamePass.py
pygame2exe.py ---------- 第三方打包脚本