# 《破·房》 游戏技术文档

## 概要：游戏集成了单机与联机两种模式，基本满足需求文档中全部功能，包括主角的移动、攻击、伤害计算及得分结算，怪物的伤害和死亡以及幽灵的巡逻等特性，灯光的计时亮灭与手动操作以及陷阱和宝箱等场景道具的功能。在联机状态下可支持多人同局进行游戏，支持断线重连、排行榜、商城等。在开发过程中，对数值针对实际表现情况进行部分调整，以达到较优游戏体验。

## 亮点：

1. 将单机与联网模式集于一体，通过标志位区分不同模式，复用一套代码。
2. 采用状态同步，血量等重要数据通过服务器进行计算，交互、攻击等操作均经服务器模糊校验。
3. 有房间系统，可实现组队探索，支持断线检测和断线重连。

## 框架：

本组采用前后端分离的开发模式进行开发，核心沟通桥梁为protobuf协议。同步方式选用状态同步。但鉴于服务器不能取得客户端全部地图数据，因此部分判定仍依赖于客户端信息。

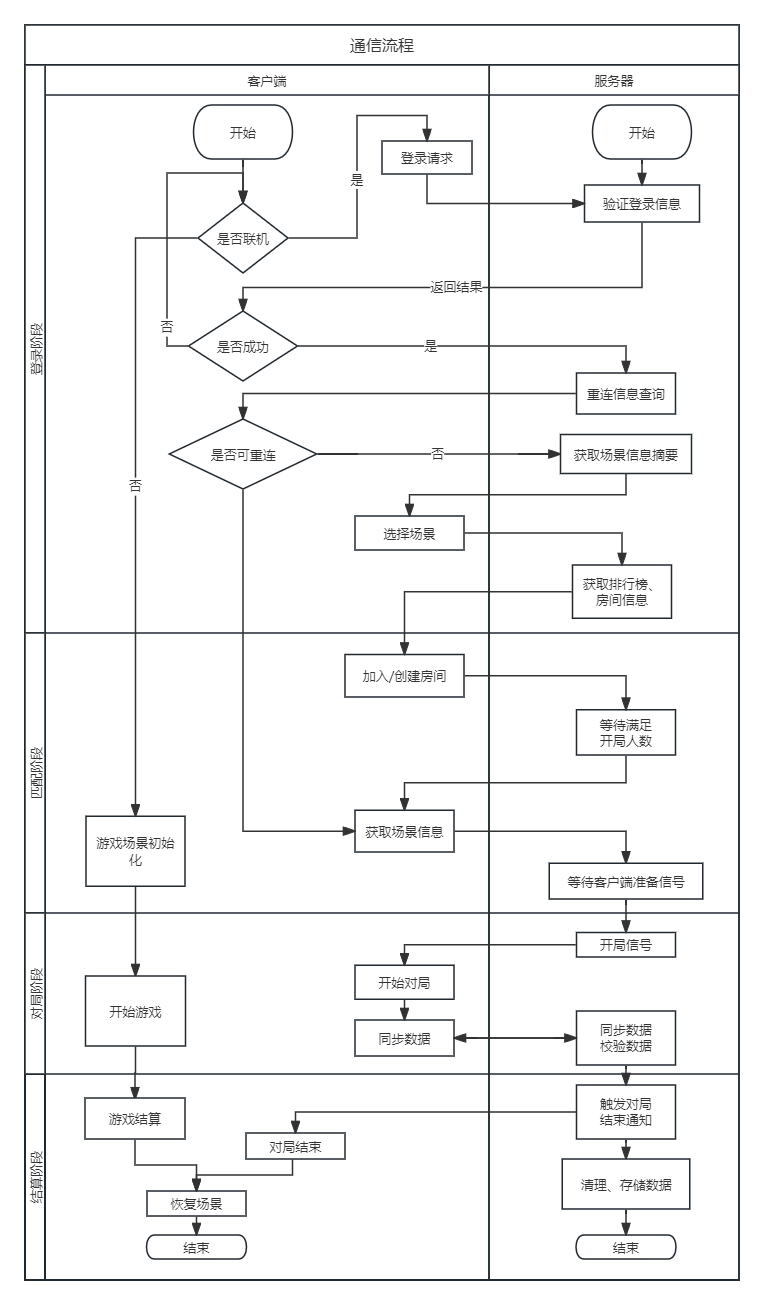


图 1 通信流程图

### 数据部分

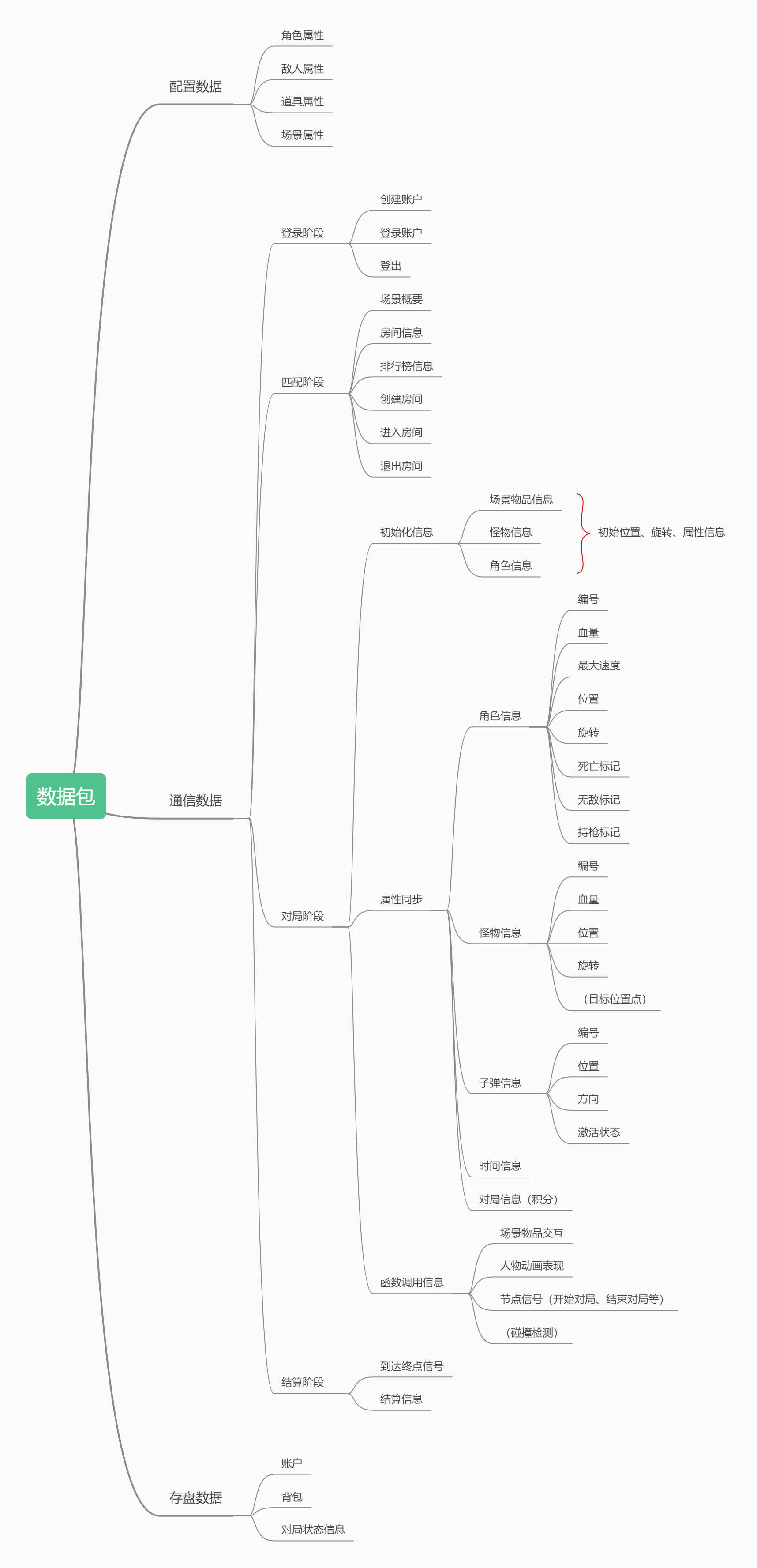


图 2 数据总览图

1. **配置数据**

角色属性（移动速度、初始血量、攻击力、子弹速度）、敌人属性（血量、攻击力、类型）以及场景属性（名称、怪物配置、出生点、终点）等配置属性存在服务器本地，采用json进行配置。

1. **存盘数据**

账户数据、背包数据等需持久化的数据在更新时进行存盘。

1. **通信数据**

登录阶段主要包含注册、登入、登出三条命令及对应的信息

在匹配（准备）阶段，登录结束后发送场景概要信息请求。在玩家选择场景后请求对应的房间和积分榜信息进行返回。随后是创建房间、进入房间、退出房间三种命令及其对应的包。

对局阶段中，首先由客户端拉取场景信息（怪物、角色、交互物），初始化后告知服务器已做好准备，等待服务器发送开始通知。正式对局中信息主要分为两类。高频的信息作为属性同步，在检测到数据存在修改时分组一起进行同步，主要包括角色、怪物、子弹的位置、旋转等信息。另一类低频的信息主要作为远程函数调用的信息，主要包括场景的物品交互、角色部分状态的动画等。此外，为了满足服务器的校验，会由客户端将检测到的碰撞信息进行上传，由服务端进行模糊校验裁决后进行缓存，用于伤害判定及交互请求的判定。

1. **结算阶段：**

主要为结算信息。

### 客户端

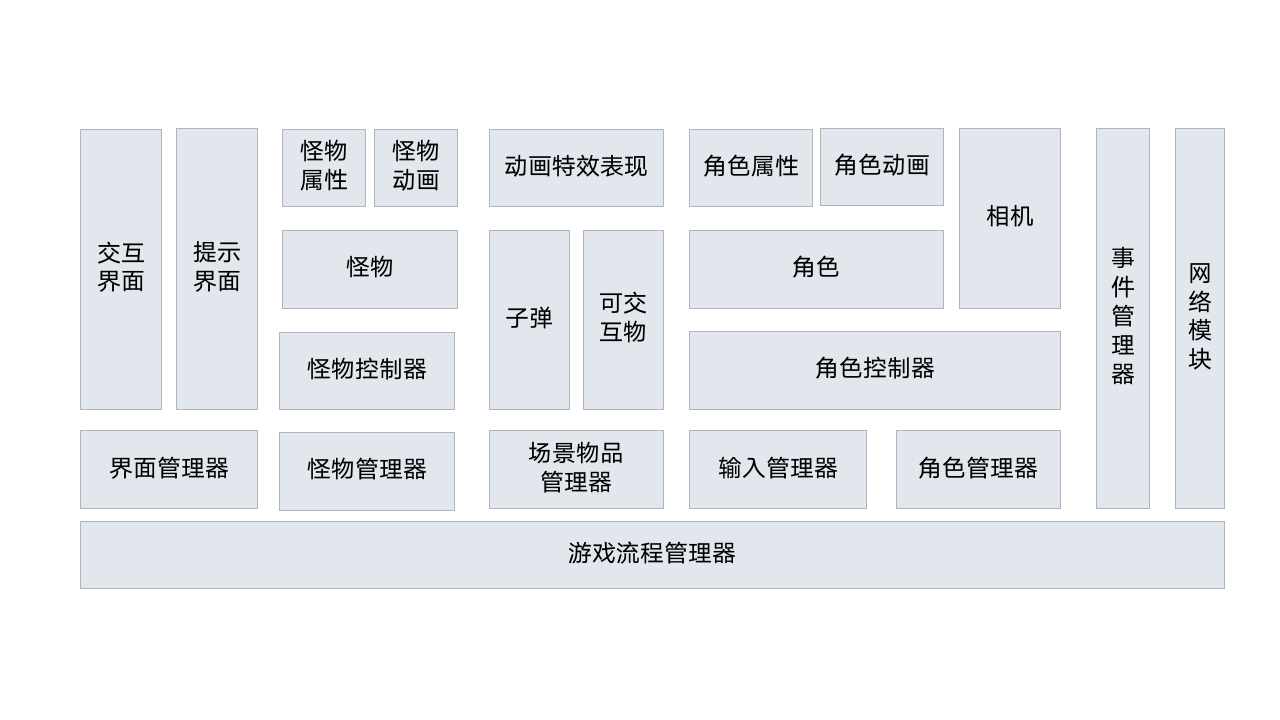


图 3 客户端架构图

**1. 游戏流程管理器：**

负责记录游戏信息，推进游戏流程，包括对局前通知各个系统进行初始化，等各个系统准备完毕后发出开始信号。在结束对局时，告知各系统清理现场。在登陆后进行断线重连相关逻辑的实现，出现网络中断等异常及时暂停游戏，等待一定时长可自动执行退出逻辑。记录游戏阶段、对局模式及积分等全局信息。

1. **事件管理器：**

负责接收和分发来自各个系统的信息，在架构中担任信使的工作。

1. **网络模块：**

负责与服务器通信。为了不阻塞主线程且保证通信，共分配三个线程进行。与服务器失去连接后会进行重连尝试。重连到达指定最大次数则告知网络异常，放弃重试。网络模块与其他模块的通信均通过事件管理器进行转发。

1. **角色管理器**

仅在联机模式下发挥作用，负责同步远端角色的信息，上传本地角色位置旋转等状态数据。在对局结束后清理所有角色。

1. **输入管理器**

将物理输入抽象为逻辑输入，作为角色控制器的输入

1. **角色控制器**

接收输入管理器或角色管理器的信息，进行角色的行为模拟。

1. **怪物管理器：**

负责管理怪物配置及初始位置。在单机模式下负责初始化，在联机模式下还需负责怪物信息的同步和上传。由于服务器无法获取场景信息，所有怪物的寻路信息仍需在客户端获取。因此采用监听模式，在同步时服务器会指定一个id，其对应的客户端无需同步位置旋转信息。

当对局结束时，负责所有怪物的回收。

1. **怪物控制器：**

当为联机模式时，通过同步信息判断是否进行模拟。若无需模拟，则根据怪物管理器下发位置信息进行插值运算，并同步其他属性。若需模拟，则与单机模式一样，进行寻路等行为树的执行。

1. **场景物品管理器：**

负责可交互物品的管理。在单机模式中负责初始化，在联机模式下负责收指令并进行相应响应。在对局结束时需进行现场的还原。

1. **子弹池：**

负责同步子弹信息，单机状态下进行子弹的回收复用。

1. **界面管理器：**

统一管理所有界面，实现界面的复用。

### 服务端

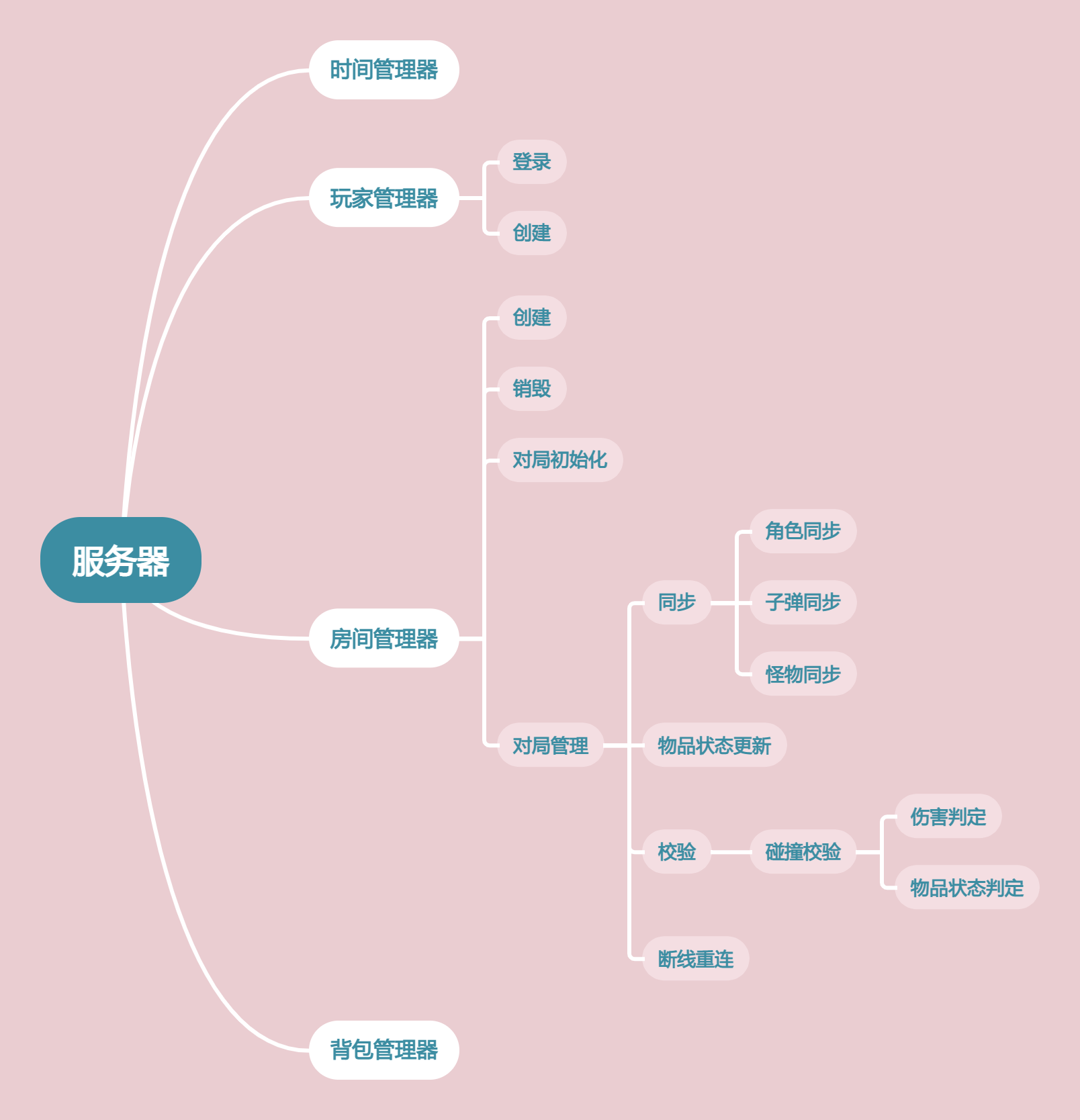


图 4 服务端架构图

**1.玩家管理器：**  
 负责玩家的登录及创建，记录在线的玩家信息。

**2.房间管理器：**

负责记录房间信息，控制房间的创建及销毁。在对局开始后，根据客户端发送的信息进行场景信息的维护与模拟，主要分为同步，校验与断线重连。  
**3.同步**

同步信息大致分为角色同步，子弹同步和怪物同步。服务器会按房间ID维护每个房间的信息及同步信息。同步信息将在服务器的每一帧更新，并把改变的信息广播给房间内所有玩家。

**3.1角色同步：**

服务器在接收到客户端角色同步时会更新该角色的信息，在每一帧广播给当前房间内玩家。

**3.2子弹同步：**

服务器在接收到客户端攻击请求后校验并生成一颗子弹，在每一帧更新子弹状态并广播给房间内玩家。

**3.3怪物同步：**

由于服务器不能取得客户端全部地图数据，所以怪物的位置信息由服务器选定的房间内一人做更新，再同步给其他玩家。

**4.校验**

主要为碰撞校验，在接收到客户端碰撞请求后将请求加入队列，在服务器每一帧处理碰撞队列中的请求，根据不同的碰撞类型制定了不同的模糊范围，如果碰撞类型为伤害碰撞则立即处理；否则判断碰撞物体间的距离，若距离小于最小模糊范围则立即处理，距离处在最小模糊范围与最大模糊范围之间则认为两者未来有可能发生碰撞，但在此时不做处理，将其继续保留在队列中，若距离大于最大模糊范围则认为碰撞没有发生，将其从队列中移除。

**5.断线重连**

在检测到玩家断线后记录玩家ID，停止对该玩家发送同步数据，在该玩家重连后恢复同步。