



课堂教学辅助系统

BY 五律协同观乙方

陈俊达，苗沐之，廖均达，钱宇辰，靳炳淑，李冰，梁楠

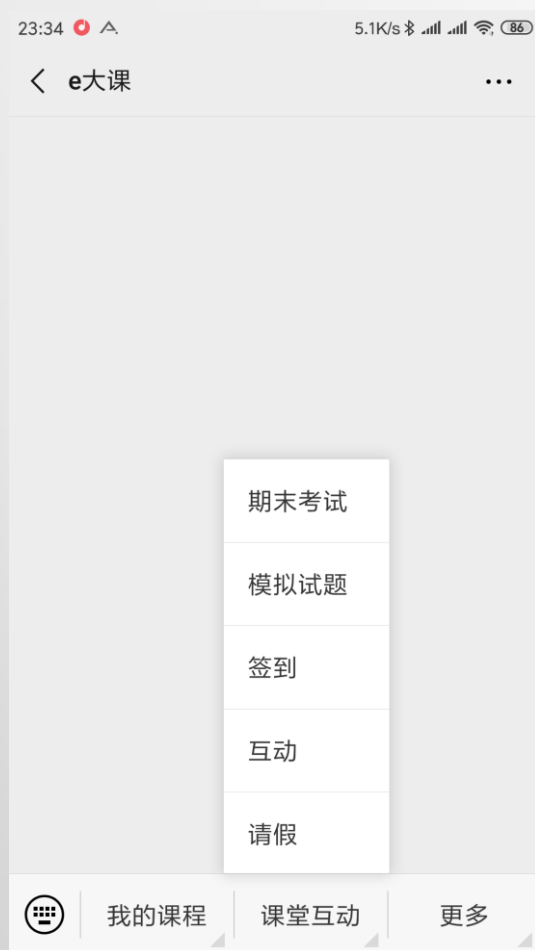
2019年3月28日

目录

- 项目背景和功能需求
- 关键质量属性和场景
- 针对质量属性的设计
- 最终架构设计和比较
- 架构Demo视频展示

项目背景

- 课程“五律协同观”所使用的系统



各种功能



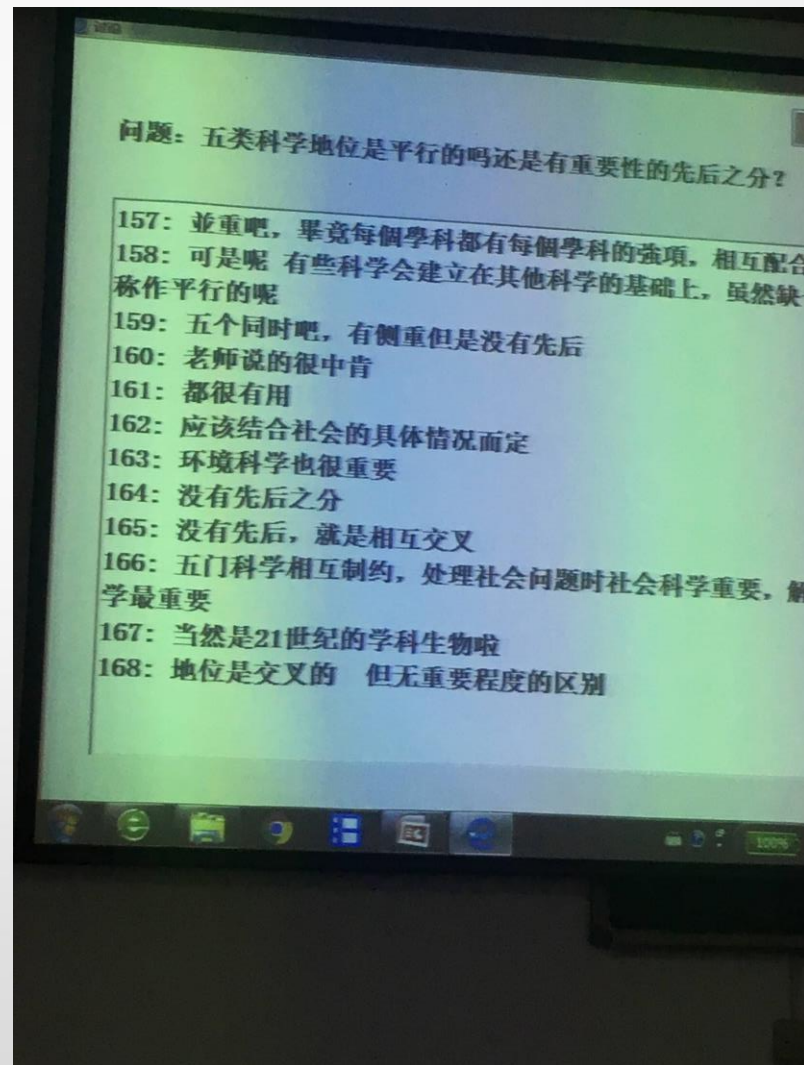
签到记录

项目背景

学生学期讨论记录

课堂名称	提问内容	讨论内容
多样观	您认为当前的教育制度应该如何改变呢?	
多样观	一个人, 一个组织, 一种文化, 为了获得多样性, 必要素质有哪些?	
多样观	宗教在现代社会的作用是什么?	
多样观	文化冲突的根本在哪里呢? 通过互相尊重能够真正解决吗?	
多样观	有些价值观确实很落后, 比如封建社会的君主集权专制制度、三从四德、官僚主义, 此时还要坚持多样观吗?	
多样观	请问您认为当前教育制度是否应该改为12年乃至15年义务教育?	
多样观	您对开设科学世界观这门课的期待是什么? 您希望开设这门课达到什么样的教学效果?	

课上讨论



功能需求

- 课程签到
- 发言举手
- 参与讨论
- 共享文件
- 参加考试
- 信息推送

质量属性和场景

- 性能

- 场景1：大量用户短时间内进行数据量交换较小的操作
- 场景2：系统快速响应用户操作
- 场景4：推送信息需要快速有效达到用户

- 可用性

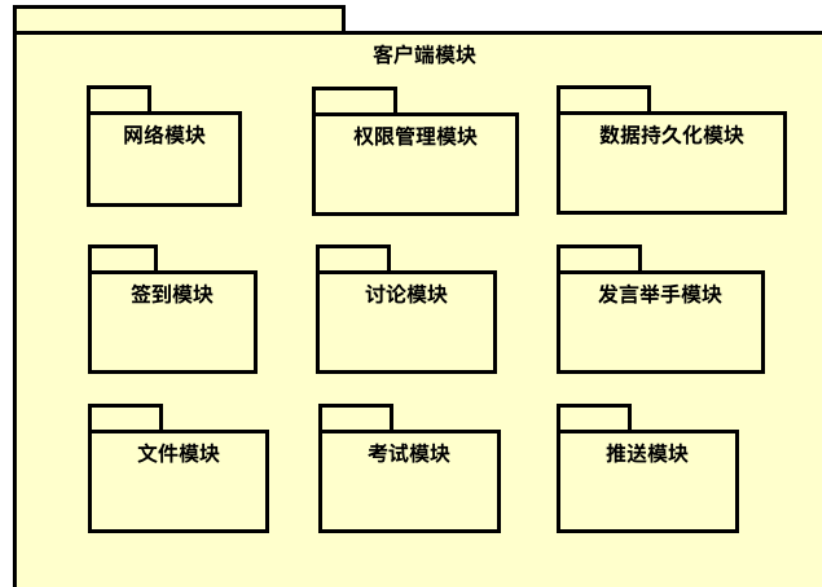
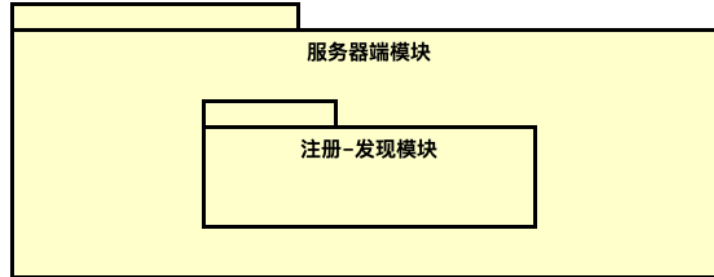
- 场景3：由于各种原因导致的系统出错
- 场景6：断线重连

- 安全性

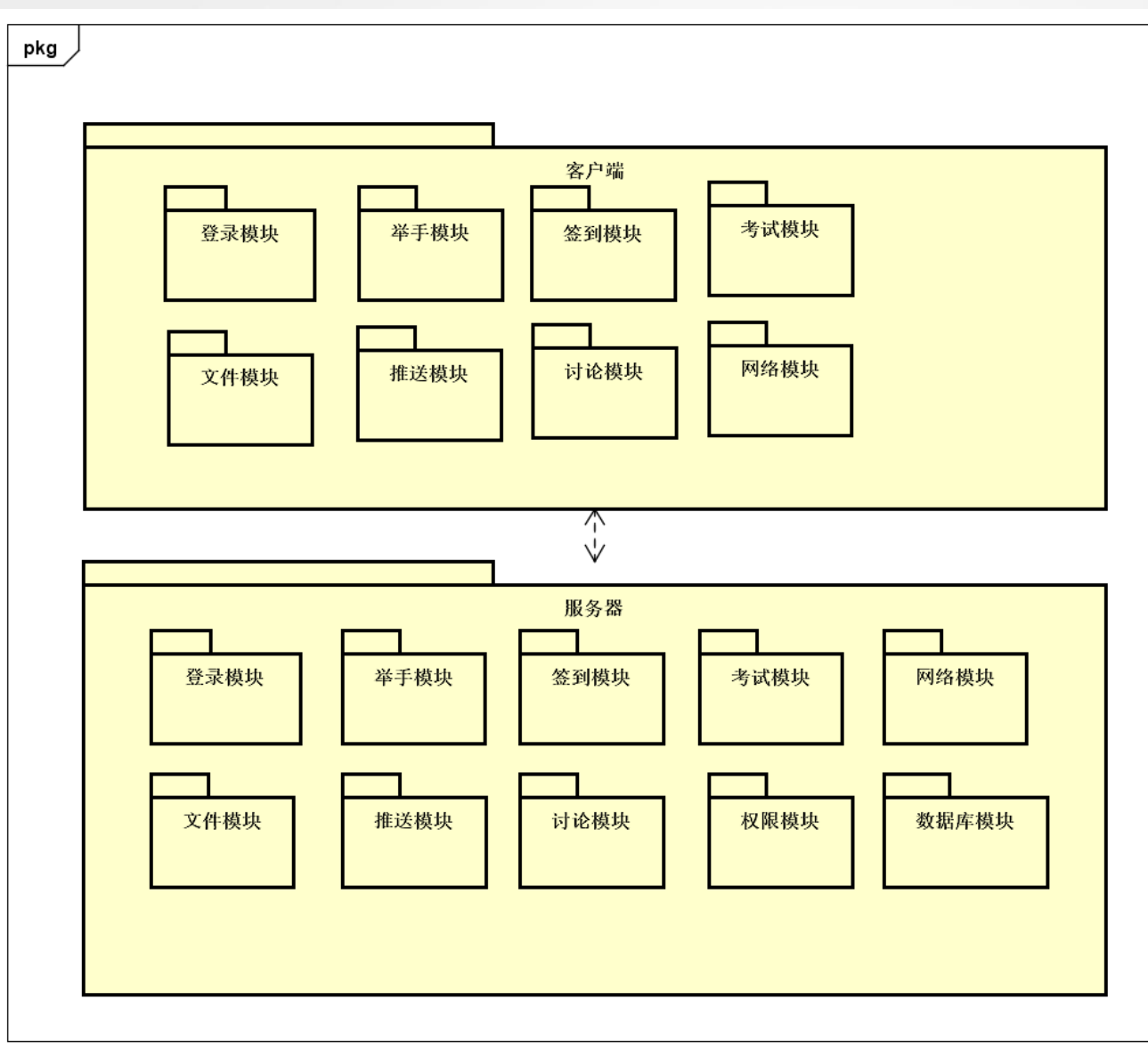
- 约束1：用户的数据不得被泄露和非法修改
- 场景7：学生数据防止被攻击和篡改
- 场景8：用户执行未授权的操作

P2P初步设计

pkg



C/S初步设计



迭代

P2P (12次)

- 分解系统
- 分解节点
- 注册-发现模块
- 网络模块
- 数据持久化模块
- 签到模块
- 考试模块
- 举手模块
- 讨论模块
- 文件模块
- 权限管理模块
- 推送模块

C/S (11次)

- 分解系统
- 考试模块
- 文件模块
- 登录模块
- 推送模块
- 权限管理模块
- 举手模块
- 讨论模块
- 网络模块
- 签到模块
- 数据库模块

针对性能的设计

P2P

- 第6次迭代（签到模块）
- 第7次迭代（考试模块）
- 第8次迭代（举手模块）
- 第12次迭代（推送模块）
-

C/S

- 第2次迭代（考试模块）
- 第3次迭代（文件模块）
- 第6次迭代（权限管理模块）
- 第10次迭代（签到模块）
-

P2P第6次迭代（签到模块）

3.6.2 确定架构驱动因素

#	架构驱动因素	重要性	难度
1	场景1：大量用户短时间内进行数据量交换较小的操作	高	高

3.6.3.1 设计关注点

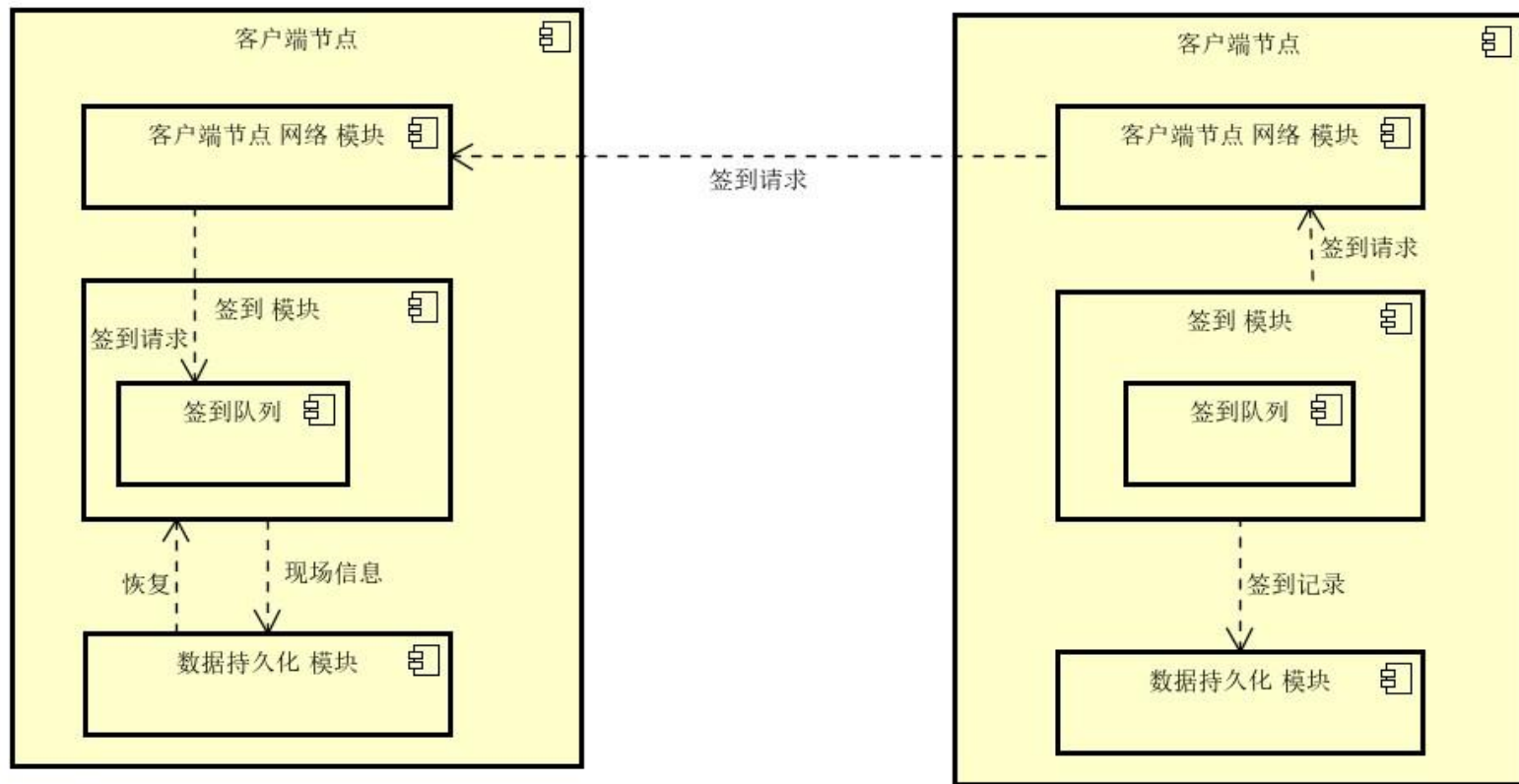
质量属性	设计关注点	子关注点
性能	高并发	密集网络请求

P2P第6次迭代（签到模块）

密集网络请求

#	模式	实现难度	并发性能
1	同步请求	低	<60（取决于硬件和容器）
2	集群+负载均衡	高	>10000（取决于硬件和容器）
3	消息队列	中	>150（取决于硬件和容器）

P2P第6次迭代（签到模块）



签到模块C&C图

C/S第10 次迭代（签到模块）

4.10.2 确定架构驱动因素

#	架构驱动因素	重要性	难度
2	场景1：大量用户短时间内进行数据量交换较小的操作	高	高

4.10.3.1 设计关注点

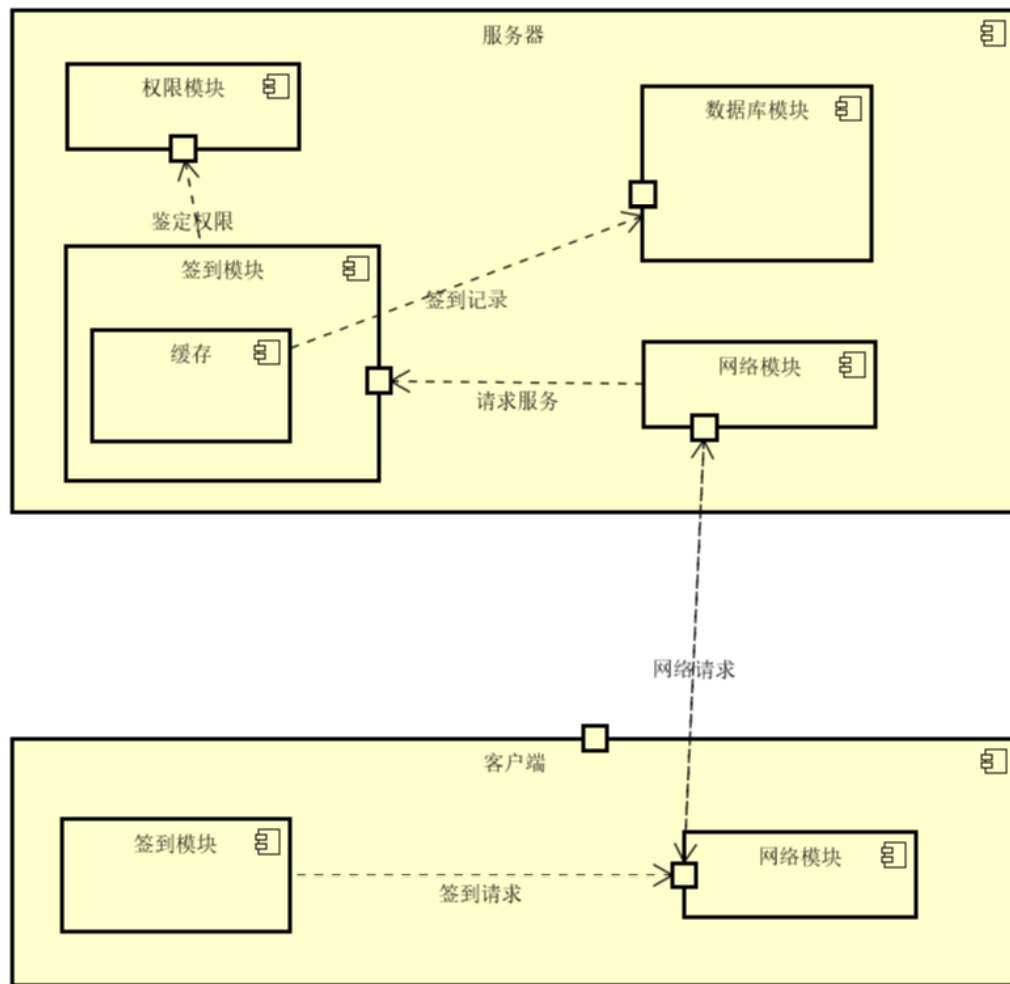
质量属性	设计关注点	子关注点
性能	高并发	快速的处理时间

C/S第10次迭代（签到模块）

快速的处理时间

#	模式	速度提升效果	硬件要求	成本
1	引入并发	低	中	中
2	增加服务器数量	高	高	高
3	对数据进行缓存	中	中	中
4	提高硬件性能	高	高	高

C/S第10次迭代（签到模块）



签到模块C&C图

针对可用性的设计

P2P

- 第3次迭代（注册-发现模块）
- 第4次迭代（网络模块）
- 第5次迭代（数据持久化）
- 第10次迭代（文件模块）
-

C/S

- 第2次迭代（考试模块）
- 第3次迭代（文件模块）
- 第9次迭代（网络模块）
- 第10次迭代（签到模块）
-

C/S第9次迭代（网络模块）

4.9.2 确定架构驱动因素

#	架构驱动因素	重要性	难度
2	场景3：各种原因导致系统出错	高	中
3	场景6：断线重连	高	中

4.9.3.1 设计关注点

质量属性	设计关注点	子关注点
可用性	故障恢复	对客户端透明 健康检测

C/S第9次迭代（网络模块）

对客户端透明

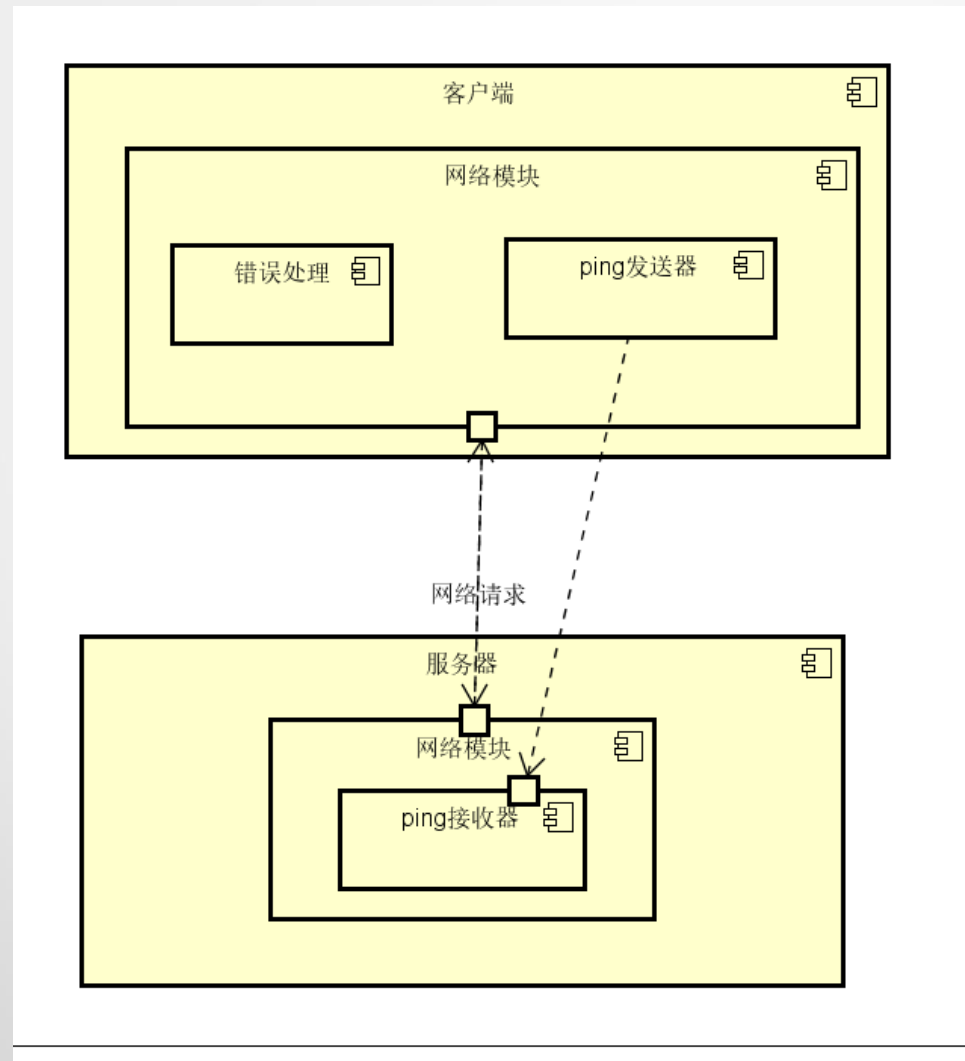
#	模式名称	通信方式	超时位置
1	客户端处理故障	单播	客户端
2	代理处理故障	单播	代理
3	基础设施处理故障	广播	基础设施内部

C/S第9次迭代（网络模块）

健康检测

#	模式名称	成本	检测效果	支持HTTP
1	Ping/Echo	中	好	是
2	心跳	中	好	否

C/S第9次迭代（网络模块）



网络模块C&C图

针对安全性的设计

P2P

- 第5次迭代（数据持久化模块）
- 第6次迭代（签到模块）
- 第10次迭代（文件模块）
- 第11次迭代（权限管理模块）
-

C/S

- 第2次迭代（考试模块）
- 第3次迭代（文件模块）
- 第6次迭代（权限管理模块）
- 第11次迭代（数据库模块）

C/S第3次迭代（文件模块）

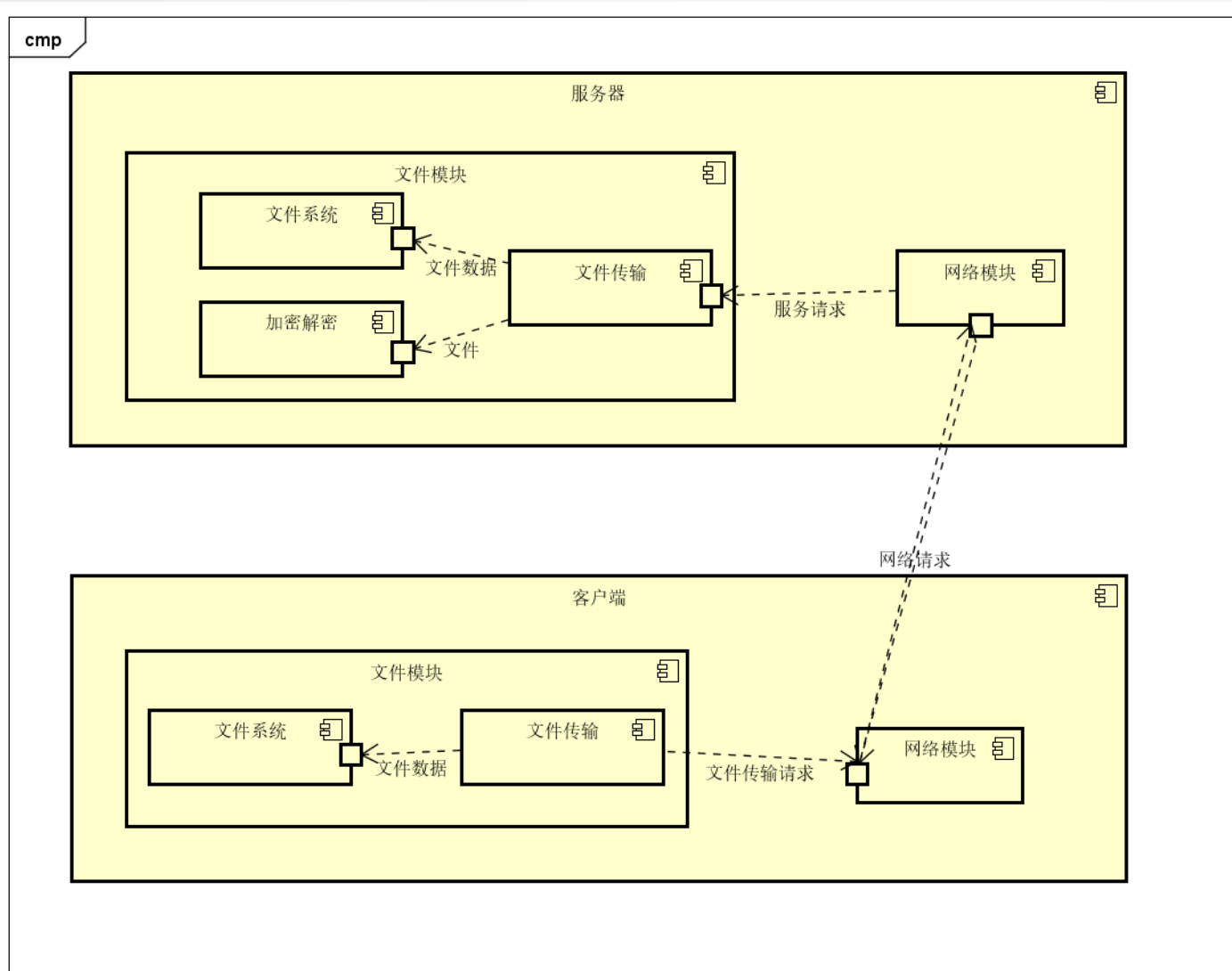
4.3.3.1 设计关注点

质量属性	设计关注点	子关注点
安全性	数据不能泄露	防止数据被非法访问

防止数据非法访问

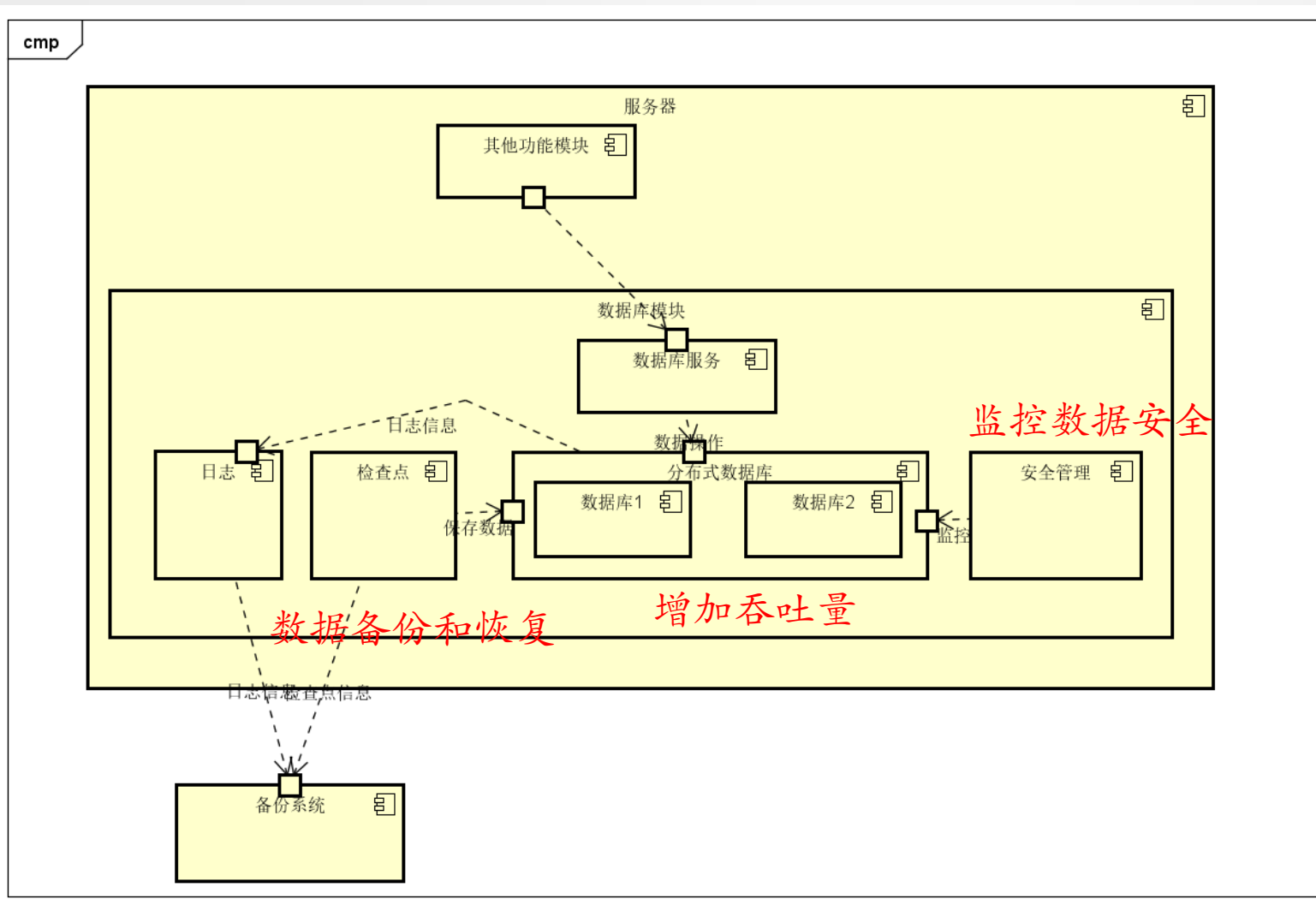
#	模式	对性能的影响	数据泄露影响	开发难度
1	加密文件	高	低	低
2	定时更改文件接口的地址	中	高	高

C/S第3次迭代（文件模块）



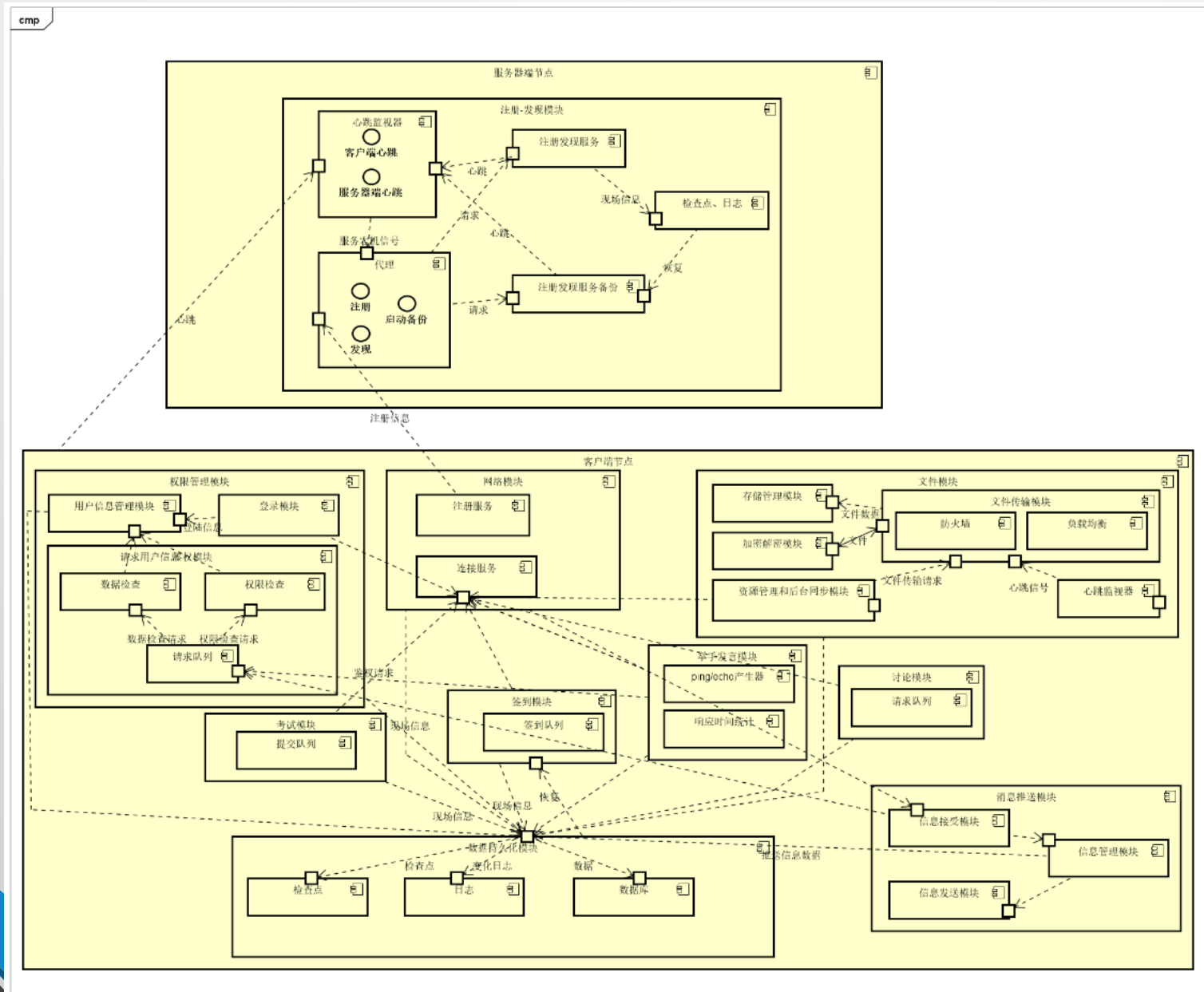
文件模块C&C图

C/S第11次迭代（数据库模块）

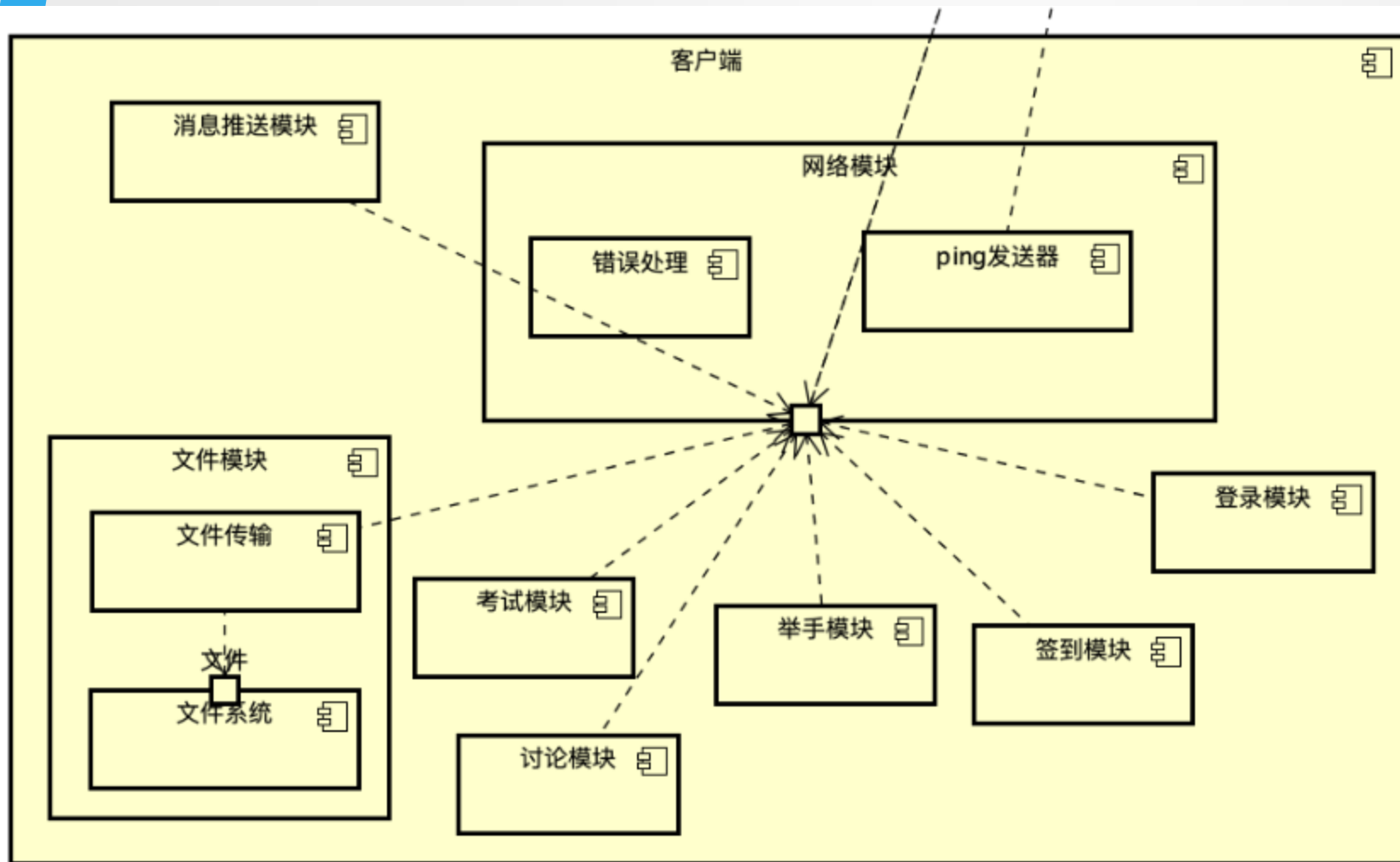


数据库模块C&C图

P2P架构设计

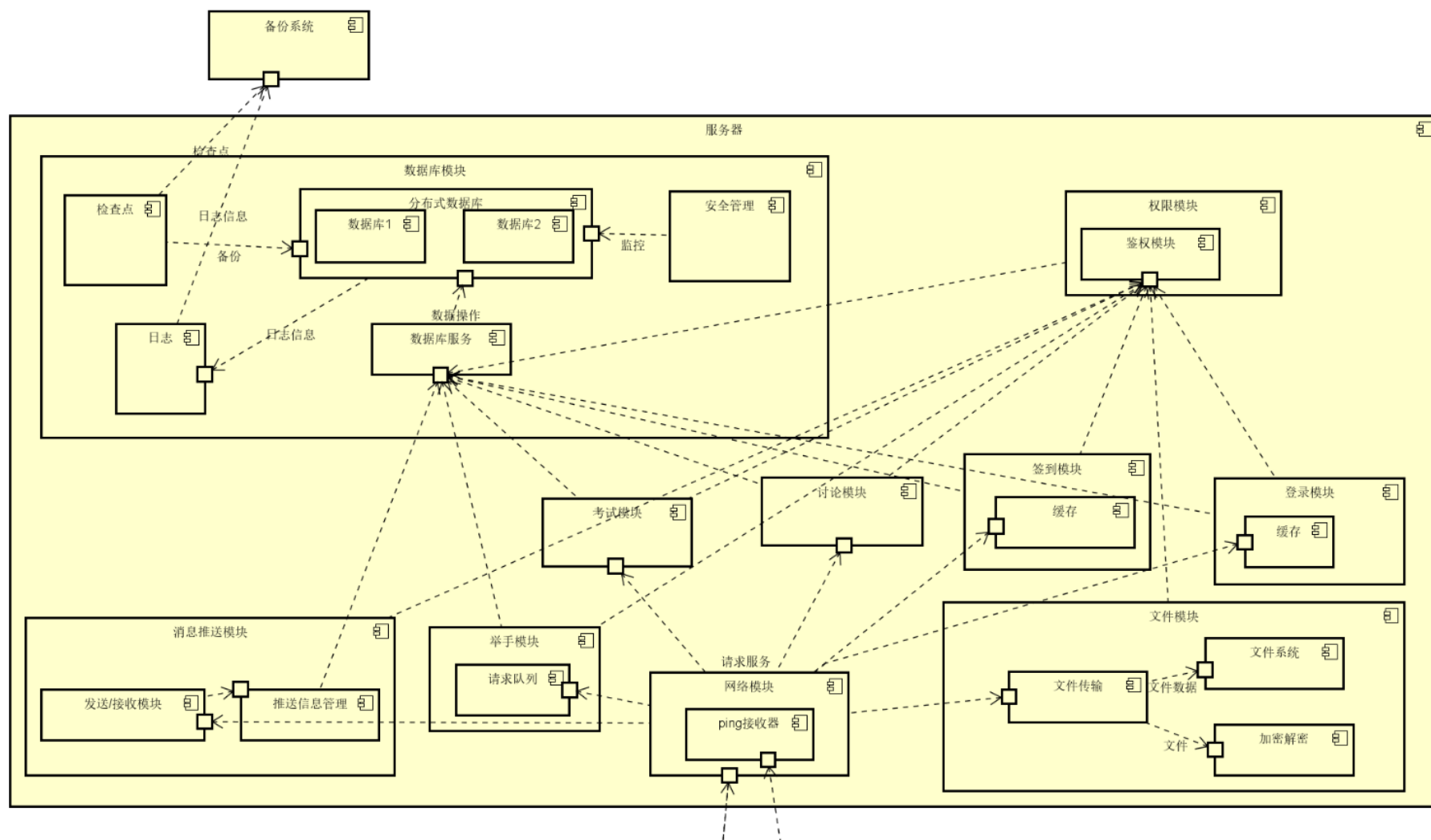


C/S架构设计-客户端



C/S架构设计-服务器端

cmp



架构比较

P2P

- 低成本
 - 不依赖中心服务器
 - 或者中心服务器功能简单
 - 低维护和硬件成本
- 高可用性
 - 能够支持大量用户
 - 一个节点挂了不影响其他节点

C/S

- 实现容易
 - 架构传统且简单
- 用户体验好
 - 复杂计算、存储、安全均在服务器
 - 客户端硬件资源消耗低
- 功能实现
 - 云备份
 - 全时可用的文件共享

Demo

- 严格按照C&C的接口和模块设计
- Kotlin + JavaFX + RMI
- 实现了网络模块Ping-Echo机制
- GitHub: [viccrubs/ClassroomAssistant](https://github.com/viccrubs/ClassroomAssistant)

