

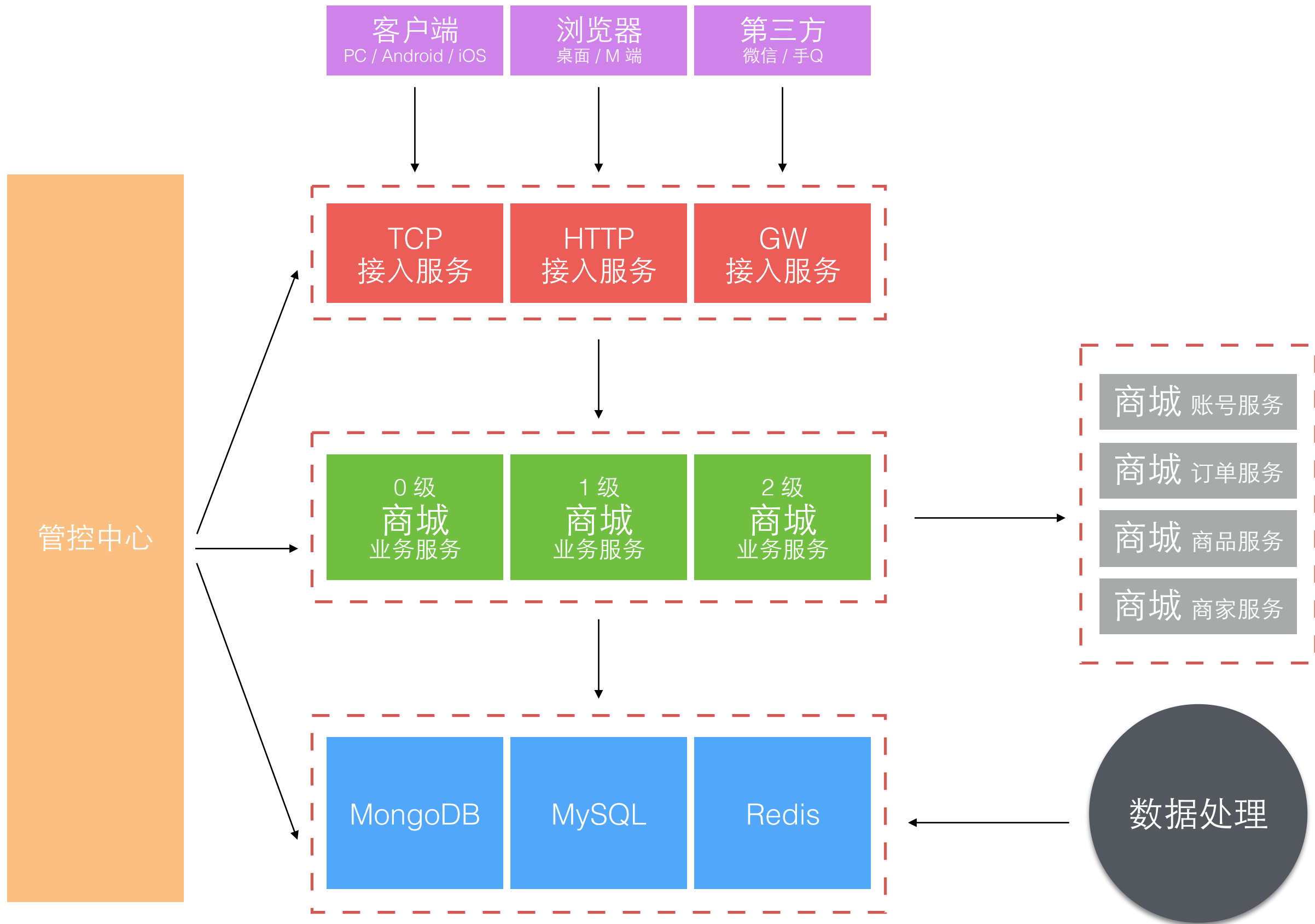
# 京东咚咚平台化架构

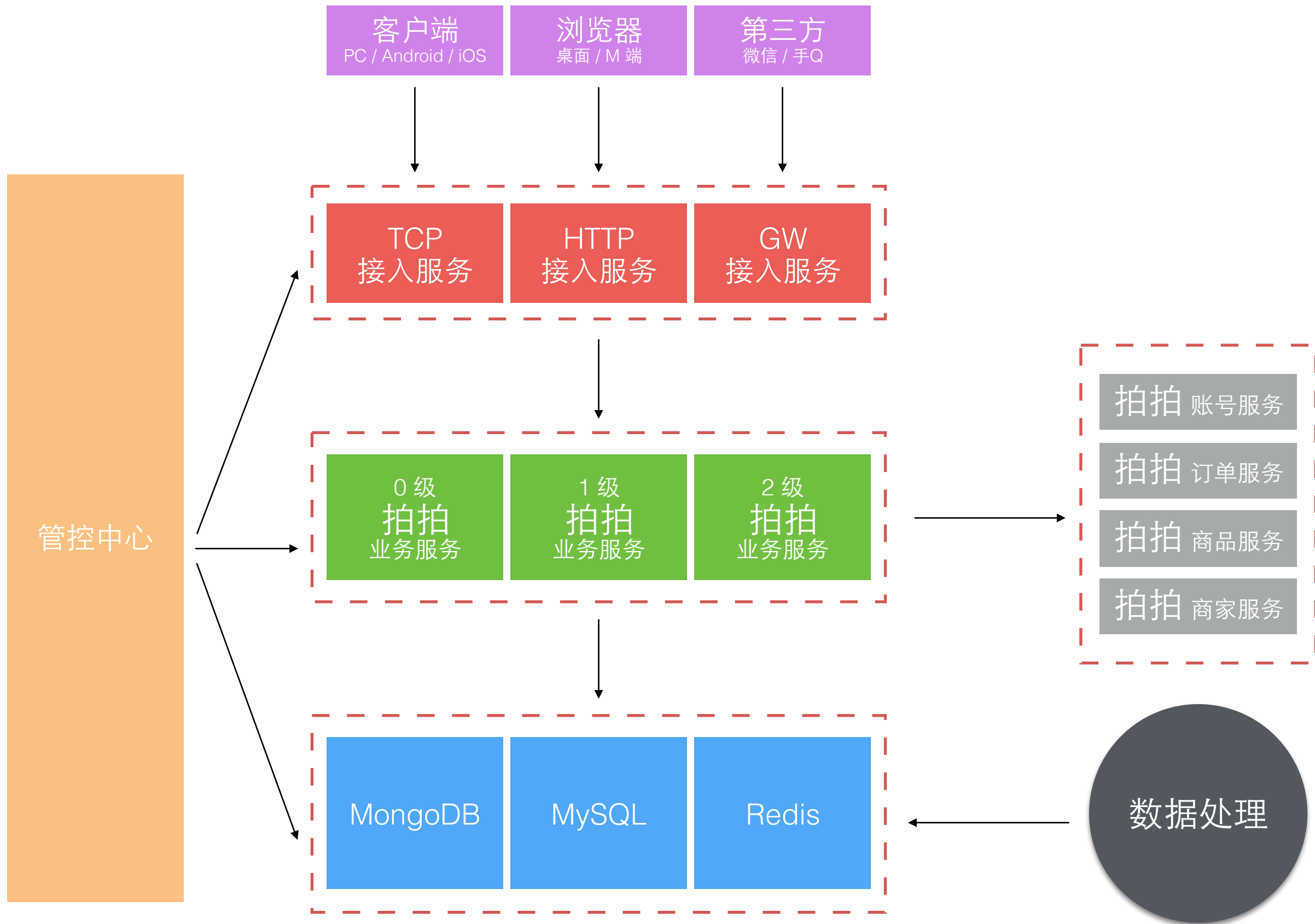
胡 峰

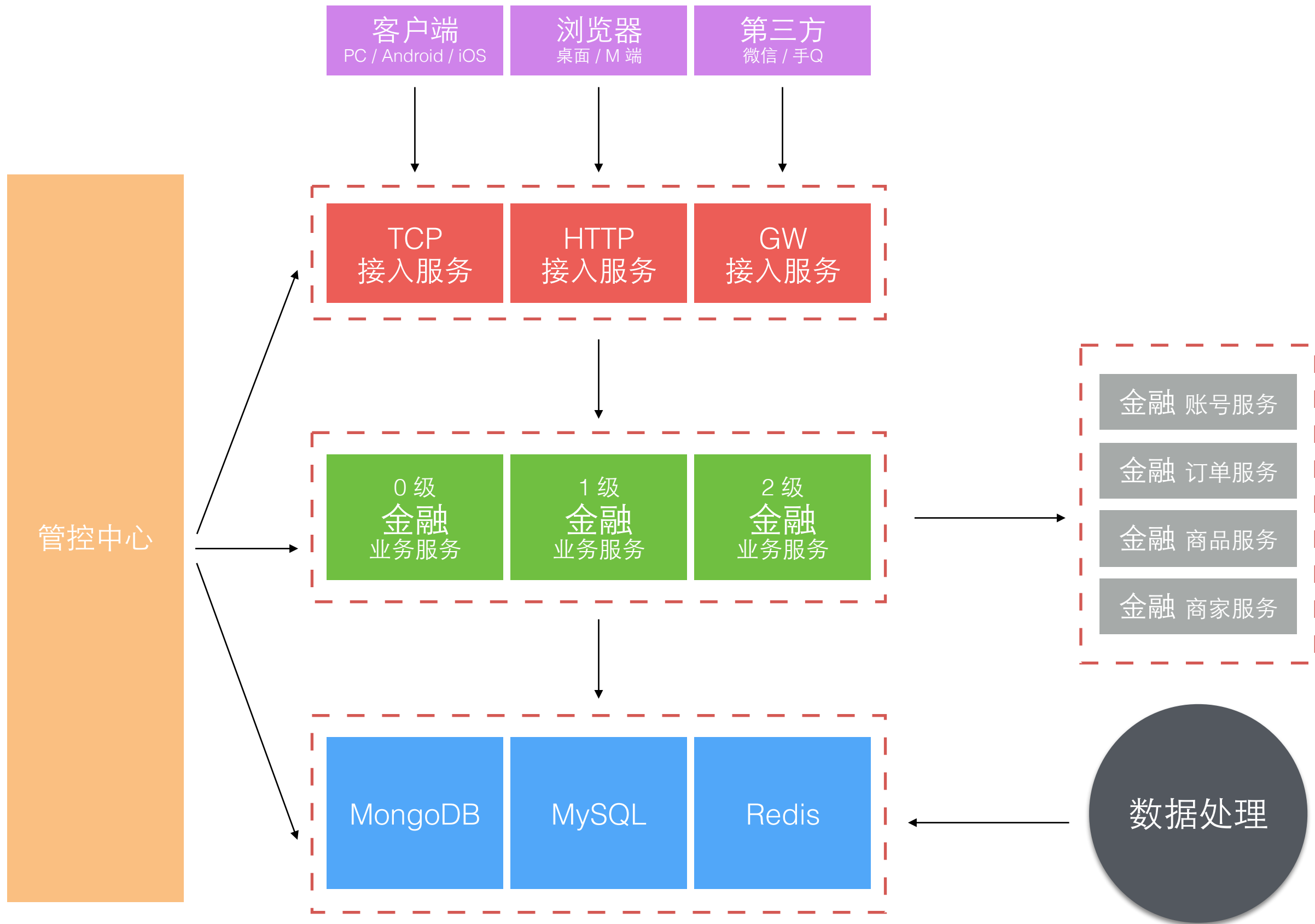
# 目录

- 面向业务架构
- 面向平台架构
- 一些技术细节
- 技术架构的核心维度

# 面向业务架构







# 问题

- 新业务接入，扩展维护成本高
  - 复制工程
  - 根据业务差异定制开发
  - 独立部署，每套部署含双机房主备和灰度环境，浪费资源

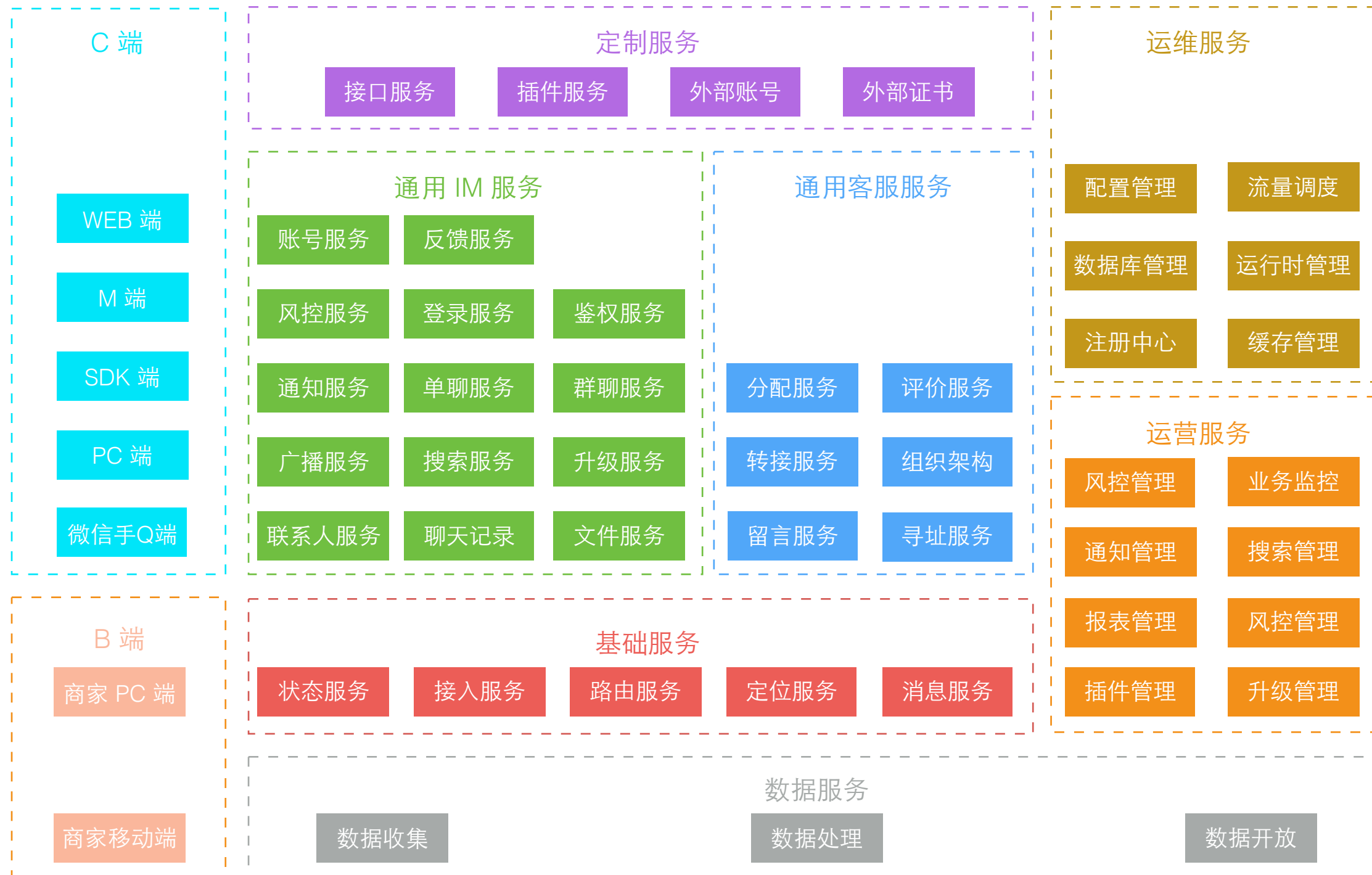
# 面向平台架构



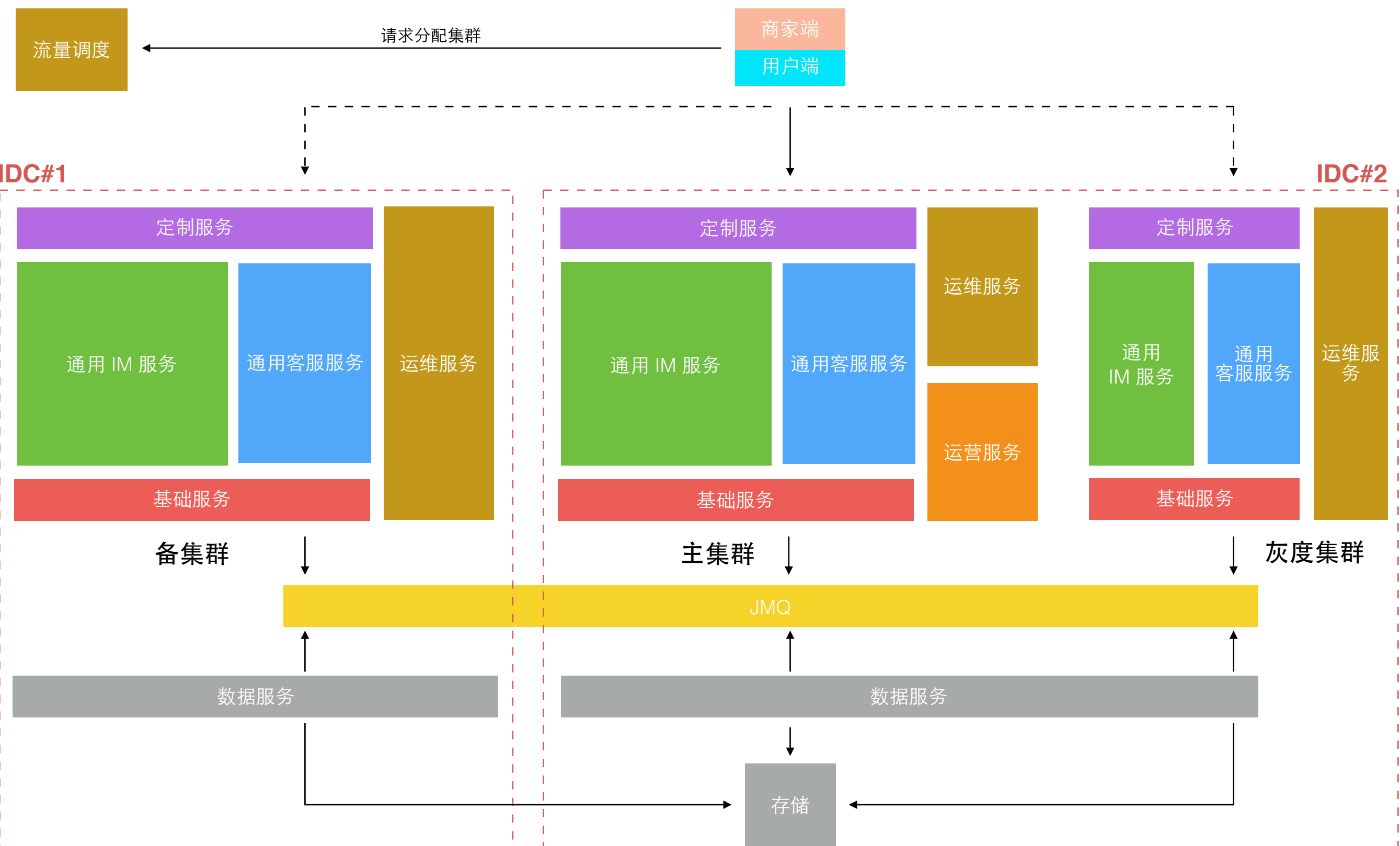
# 思考

- 越来越多的业务接入方，不能无限复制下去
  - 国际站、众客服、京东到家
- 定制业务差异很少，大部分通用业务逻辑可复用
- 服务微粒化，并行开发，局部更新，范围隔离

# 服务矩阵



# 服务部署



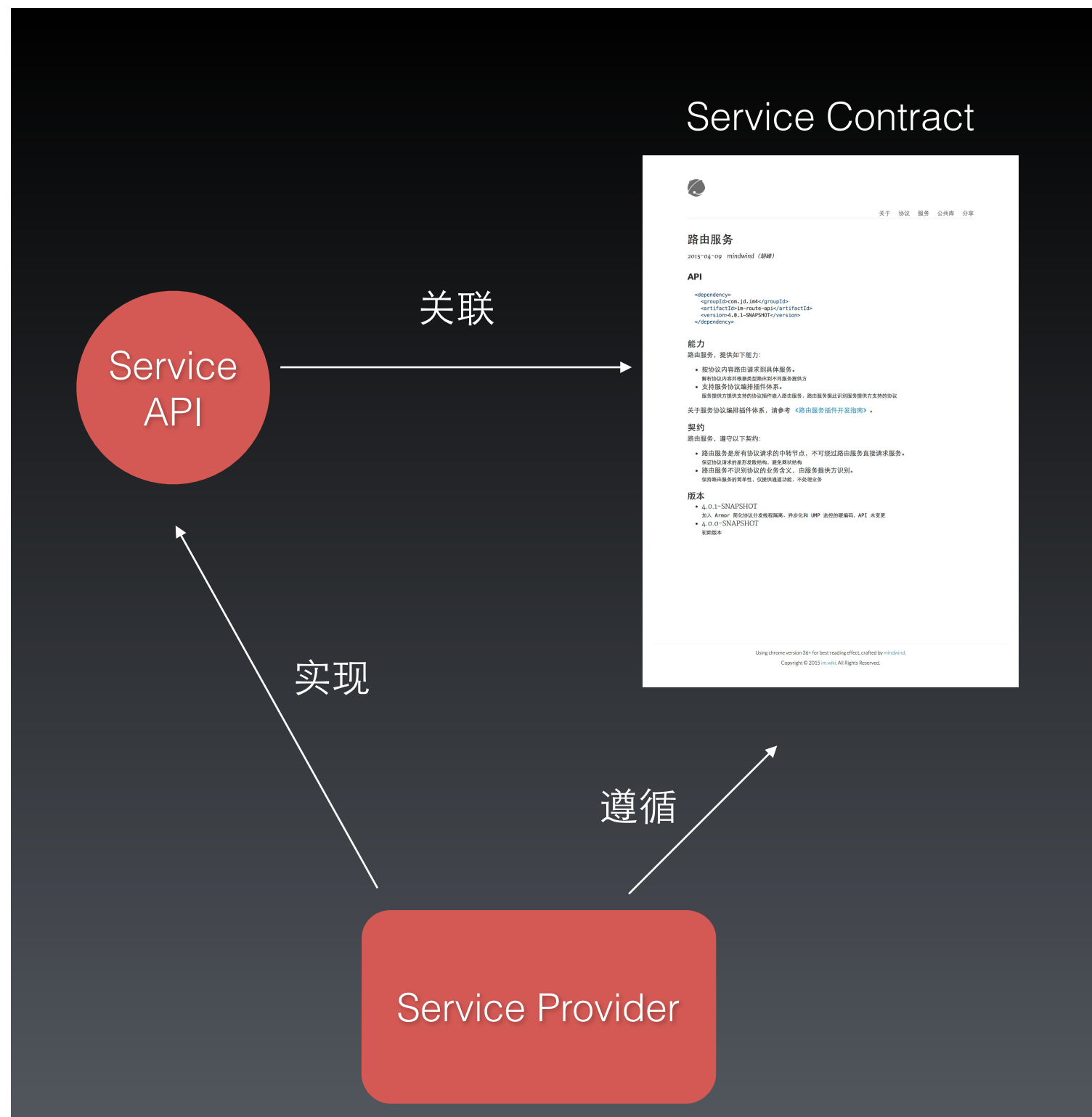
# 开发约束

- 服务调用协议化
  - 异步 HTTP、异步消息，避免同步长调用链，循环调用链
- 服务调用协议化
  - 内部协议，接口稳定性更高
- 服务调用透明化
  - 业务协议与服务处理链关联，跨进程调用方法监控

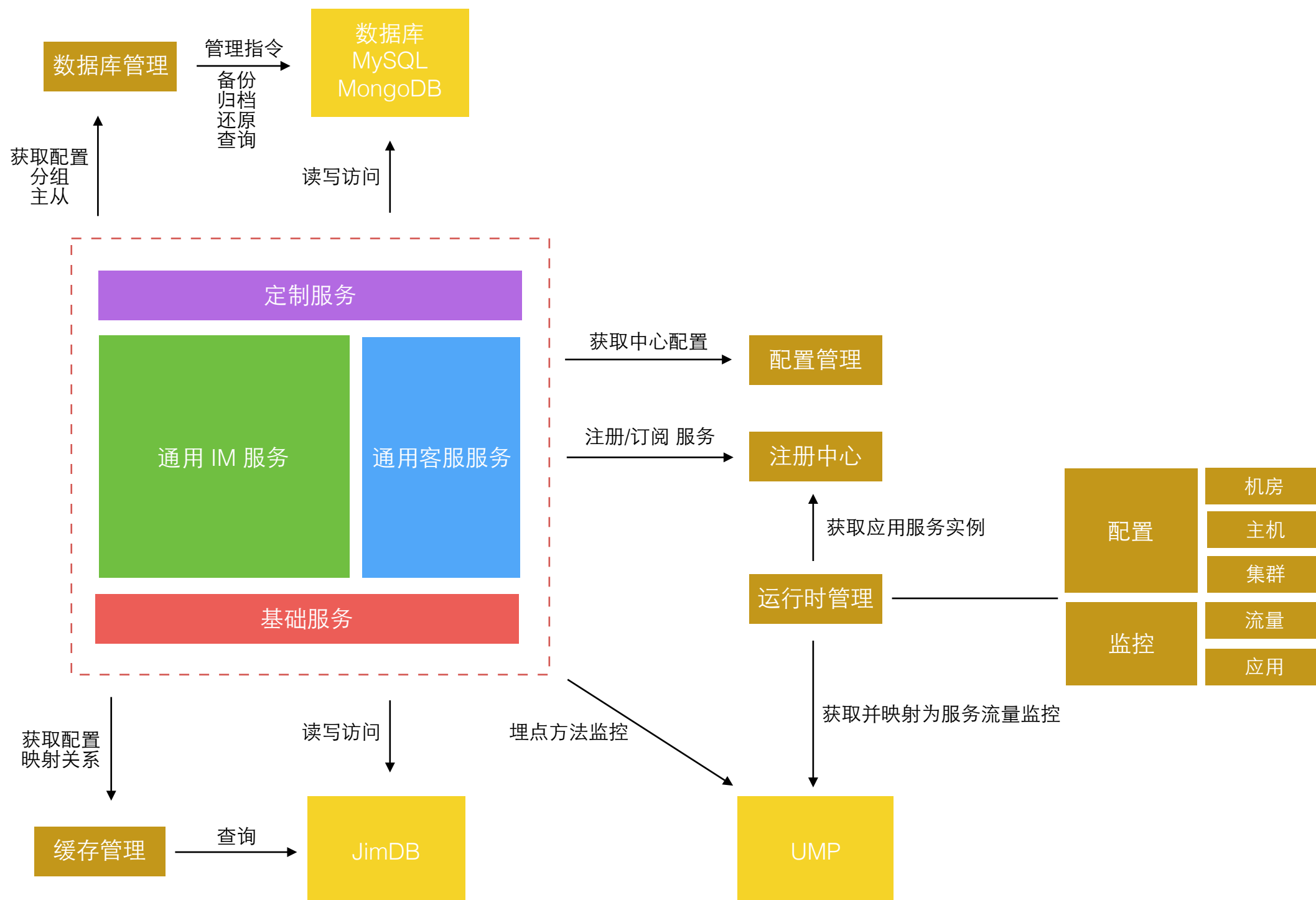
# 契约化开发

- Service Contract

- API
- 能力
- 契约
- 版本

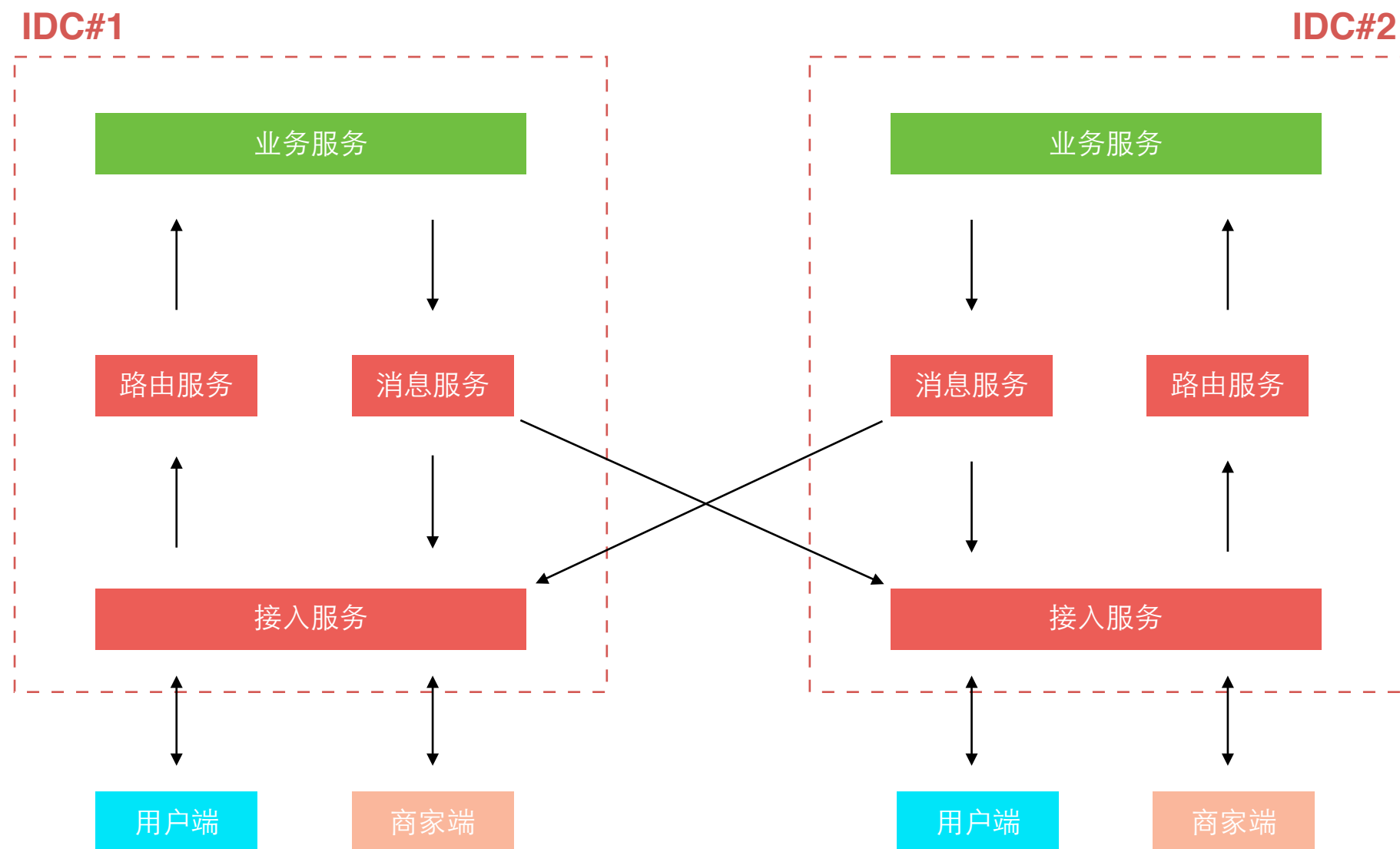


# 运维管理



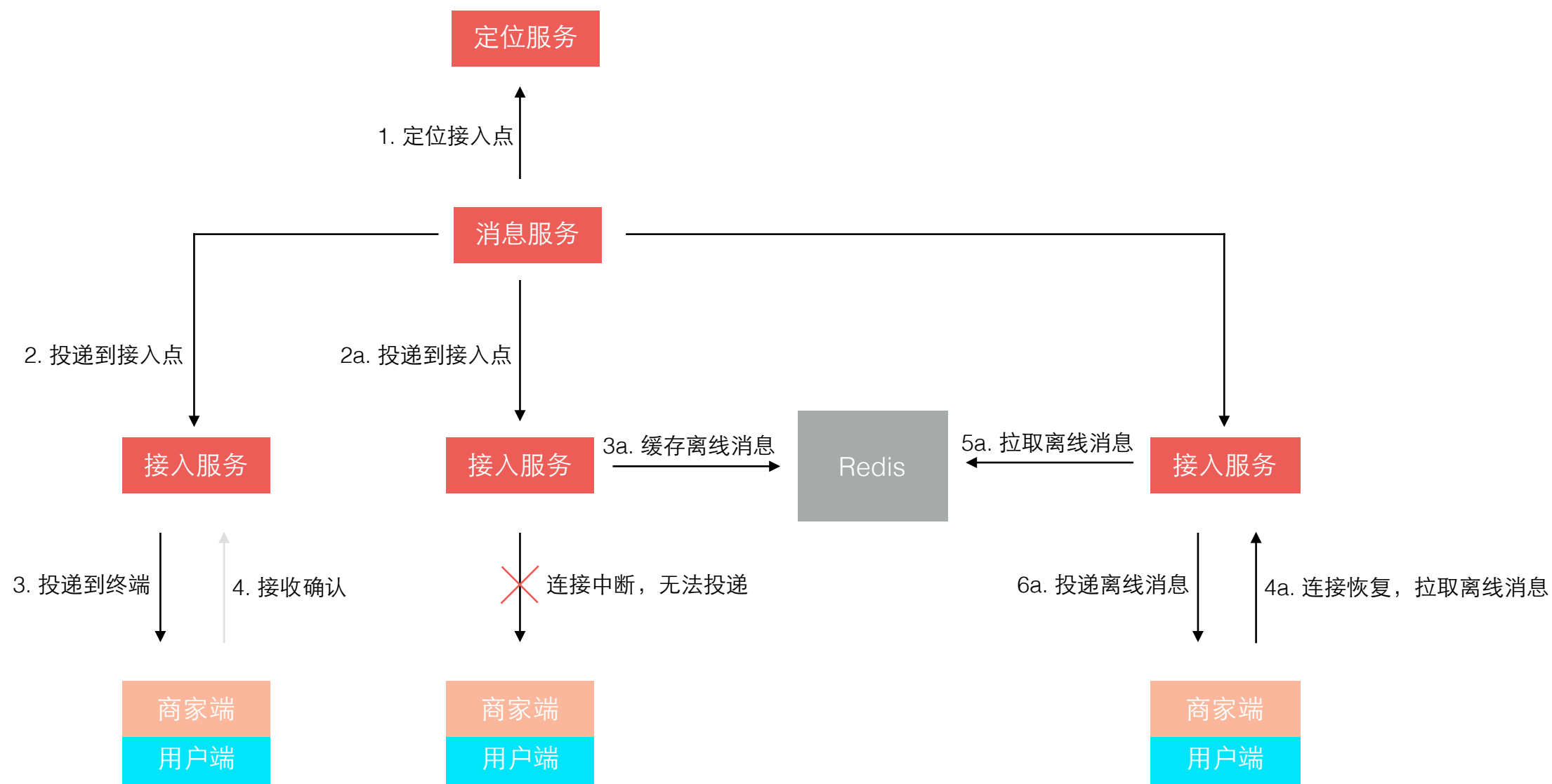
一些技术细节

# 消息流转

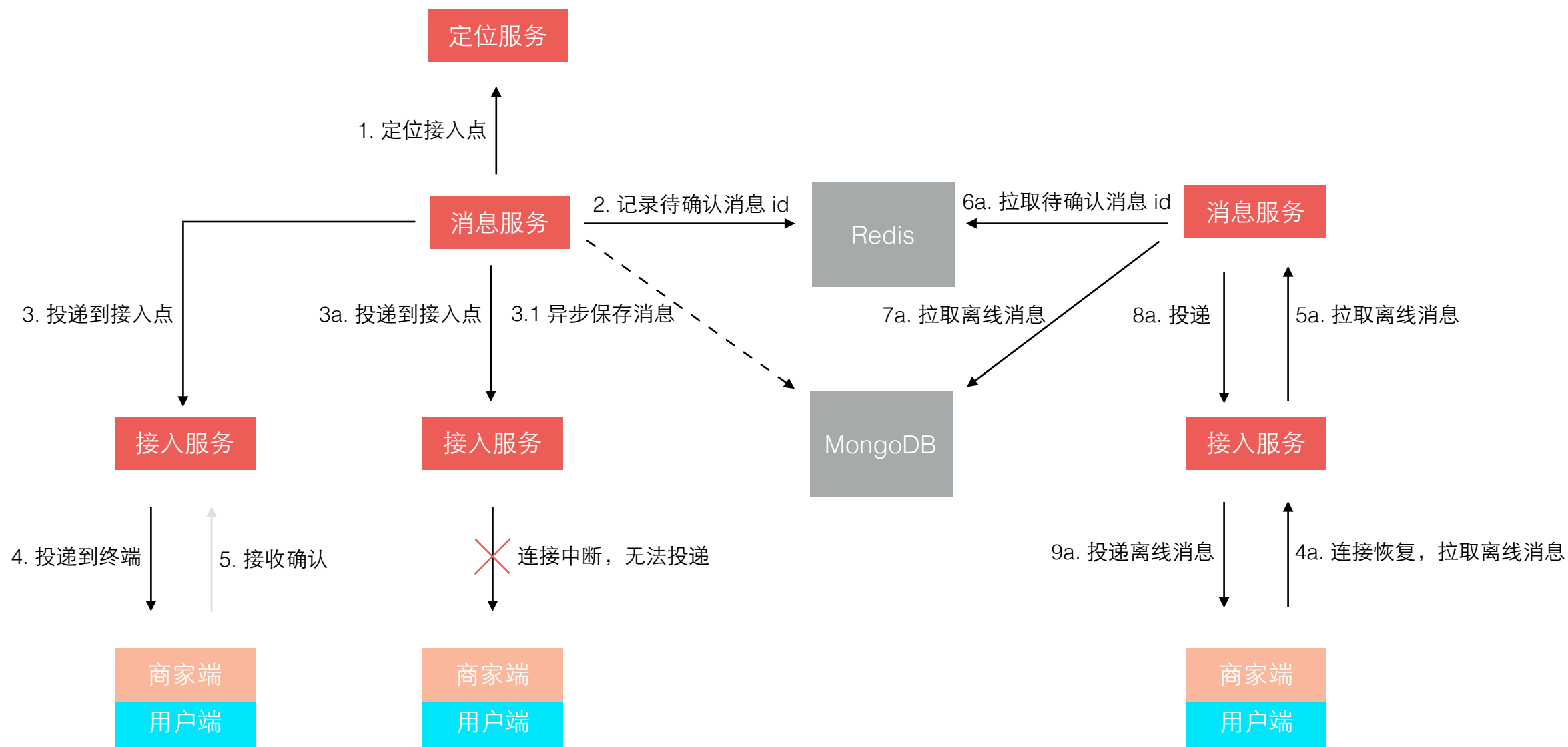




# 消息投递 - 早期模型

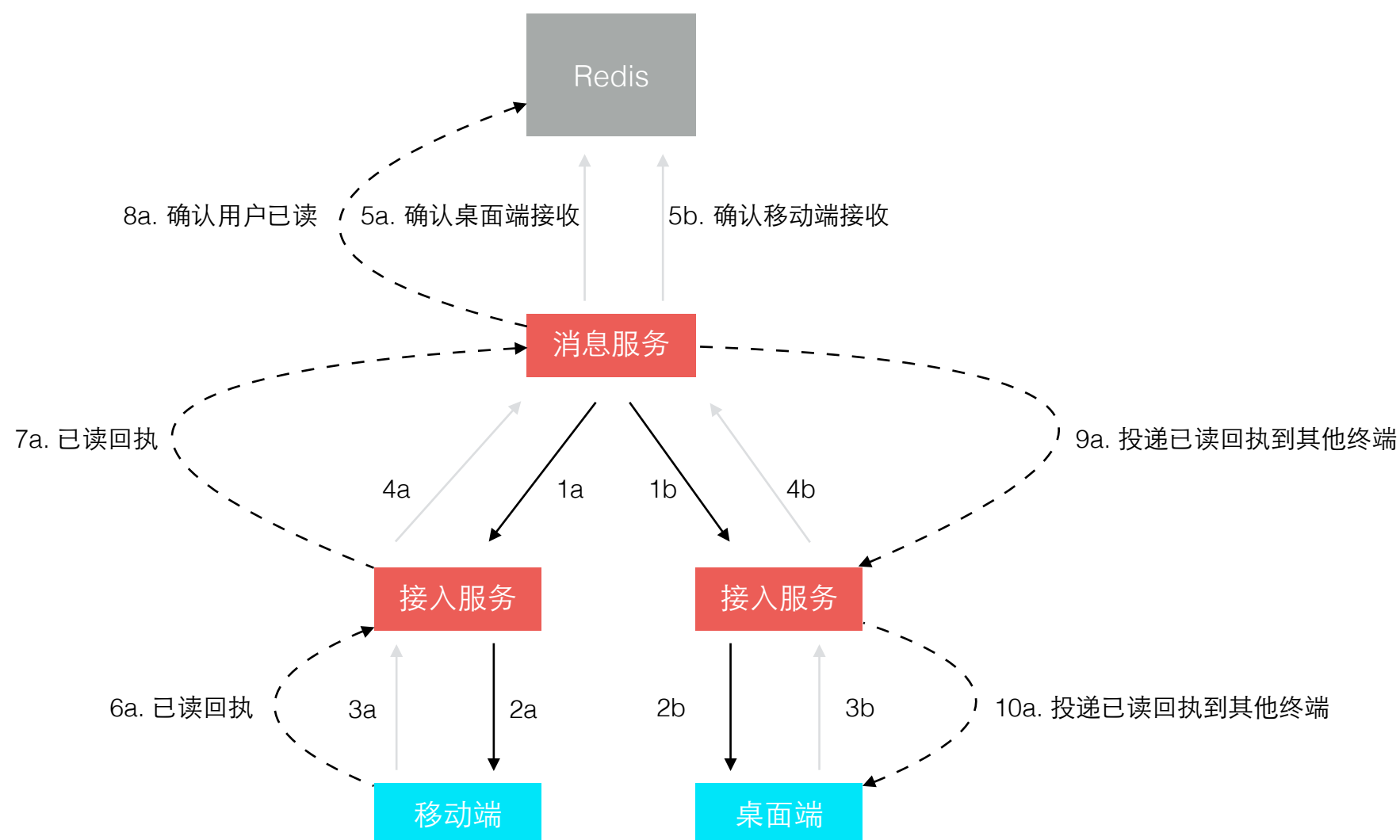


# 消息投递 - 优化模型



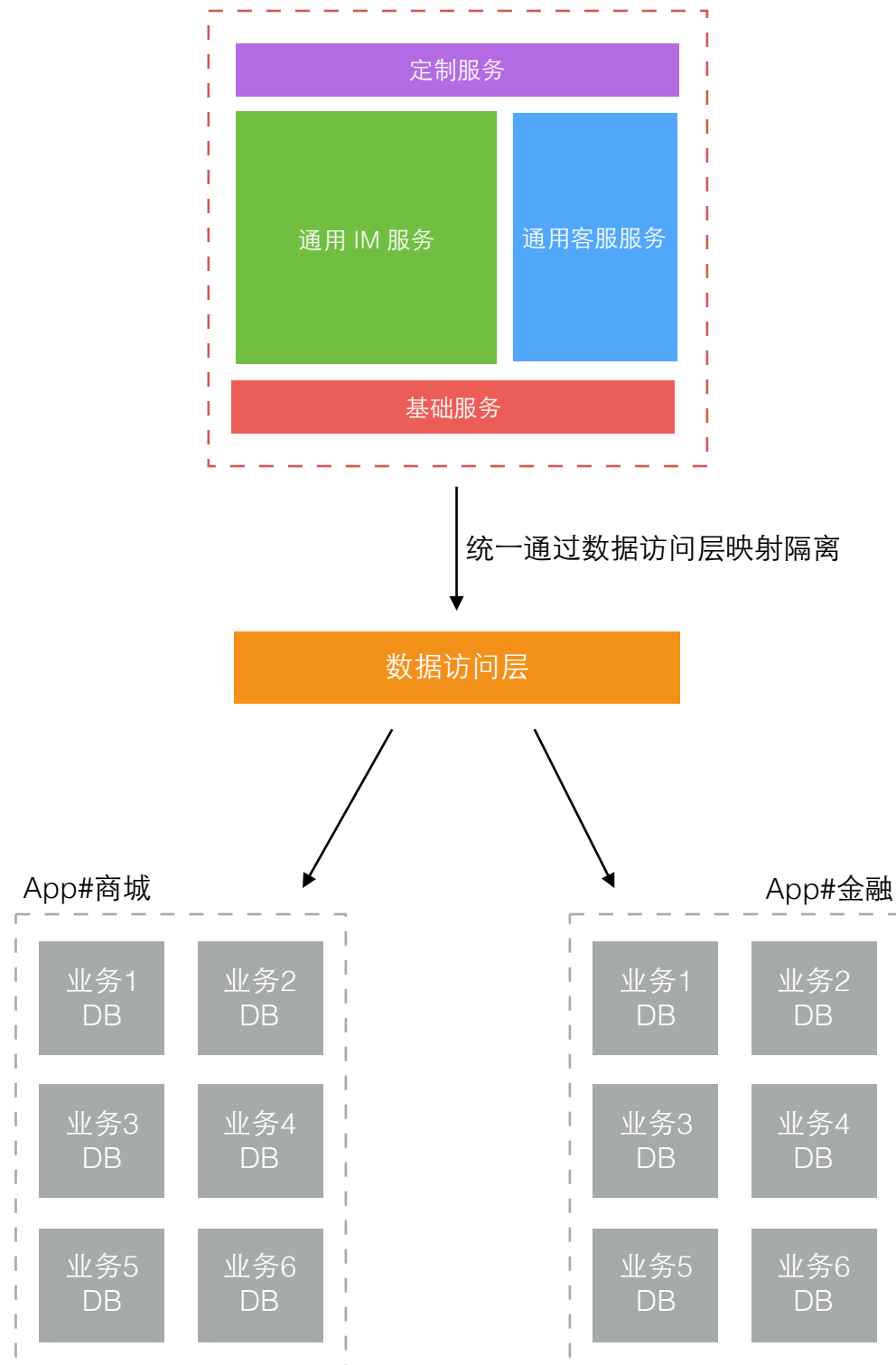
# 消息同步

- 按终端 - 消息确认
- 按用户 - 已读回执



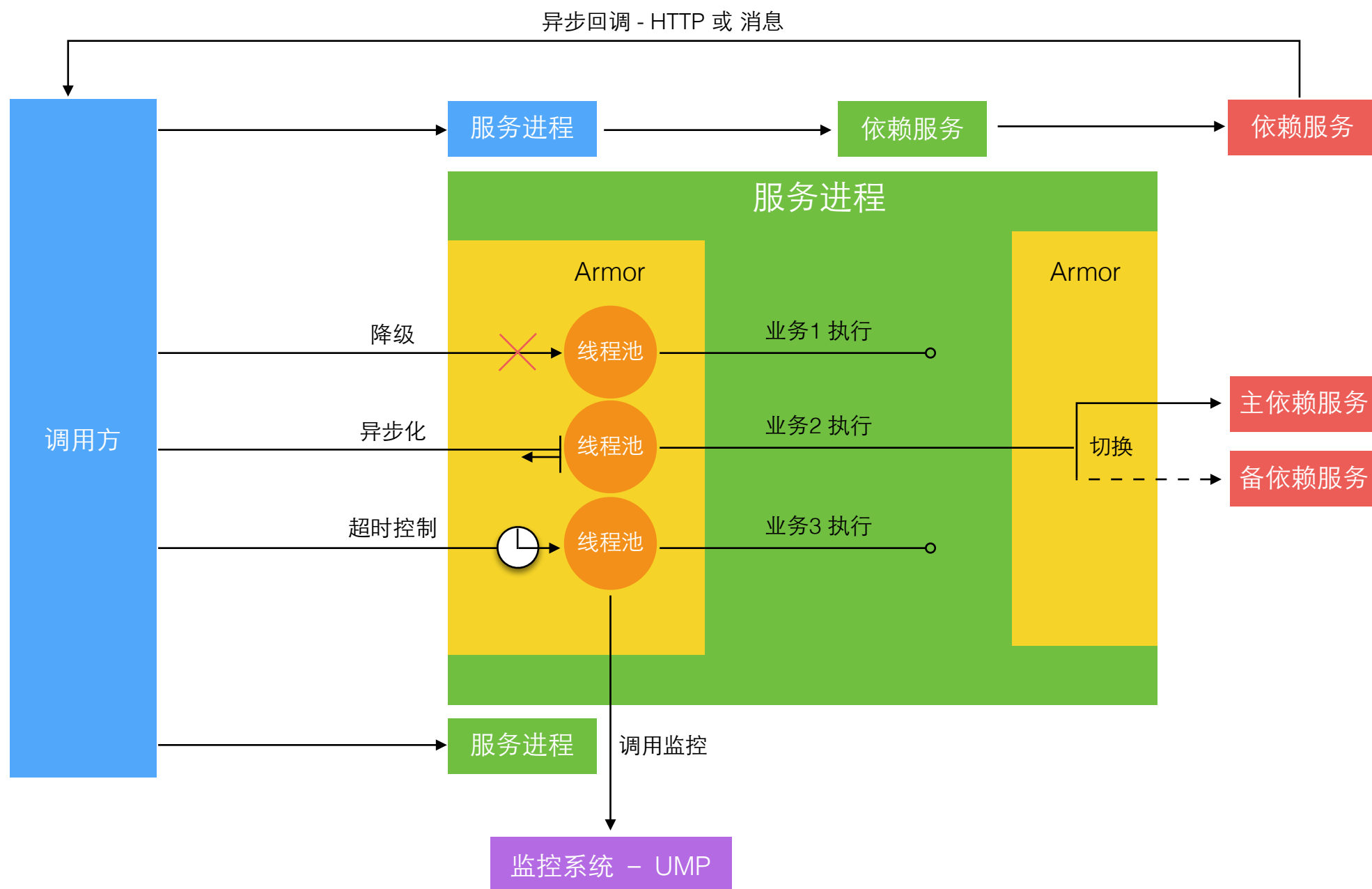
# 数据隔离

- 按业务划分
- 应用间隔离

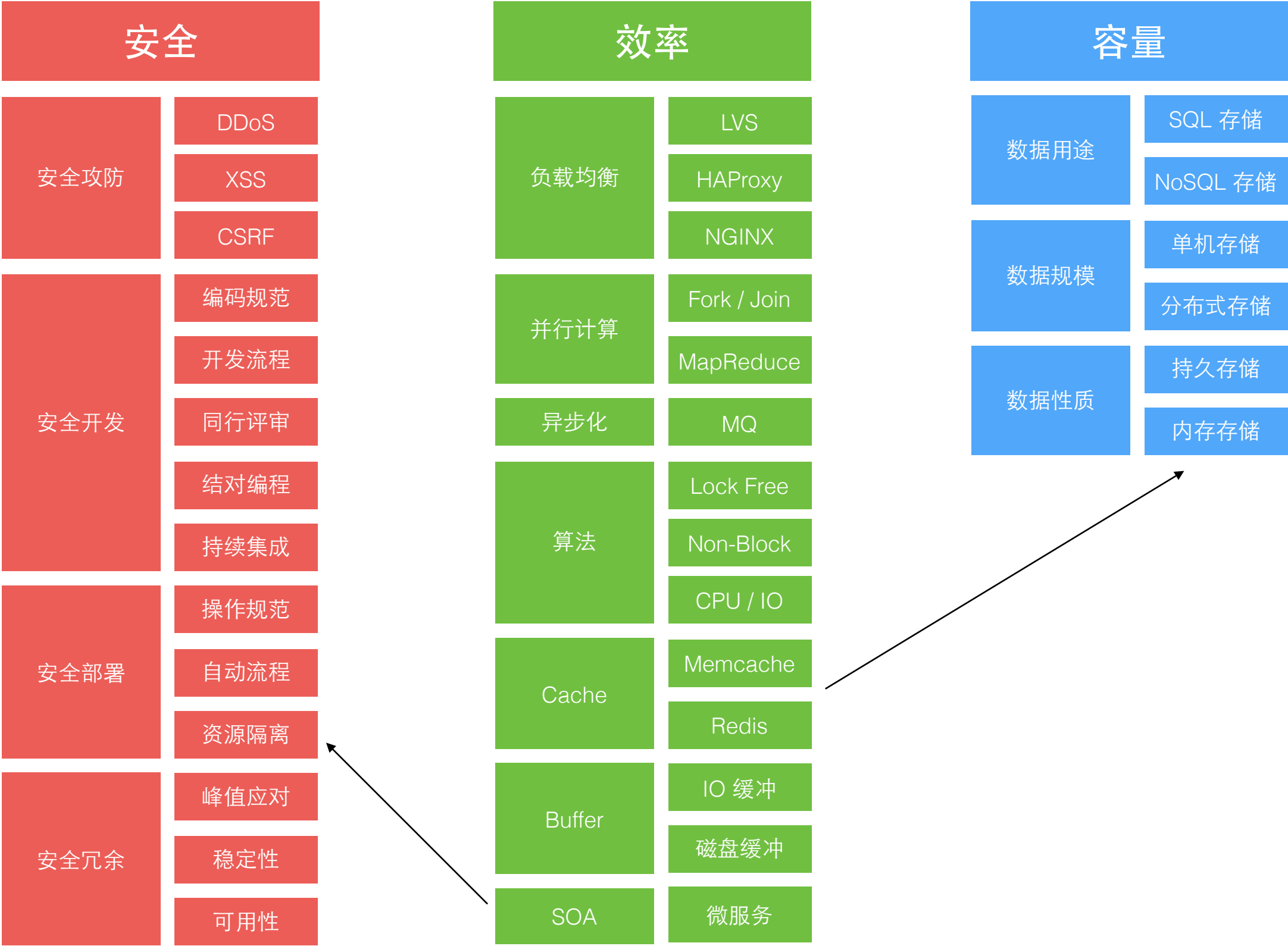


# 运行隔离

- 进程隔离
- 线程隔离
- 依赖隔离



# 技术架构的核心维度



# 谢谢

