



千万级高性能长连接Go服务架构实践



彭宝江

百度公司
资深研发工程师



目 录

统一长连接服务背景

01

统一长连接服务介绍

02

统一长连接服务架构

03

统一长连接golang实践

04

总结和规划

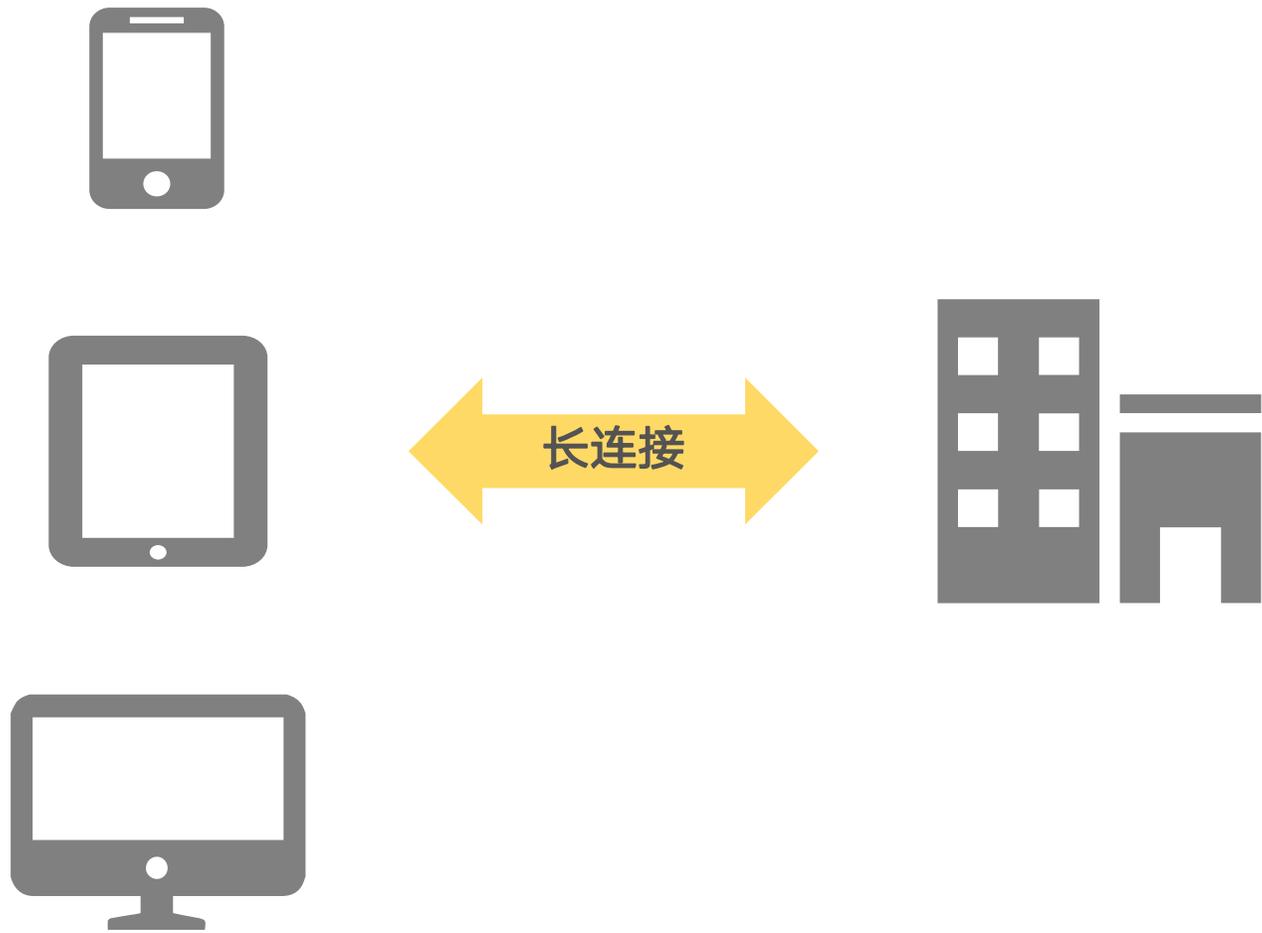
05

01

统一长连接服务背景



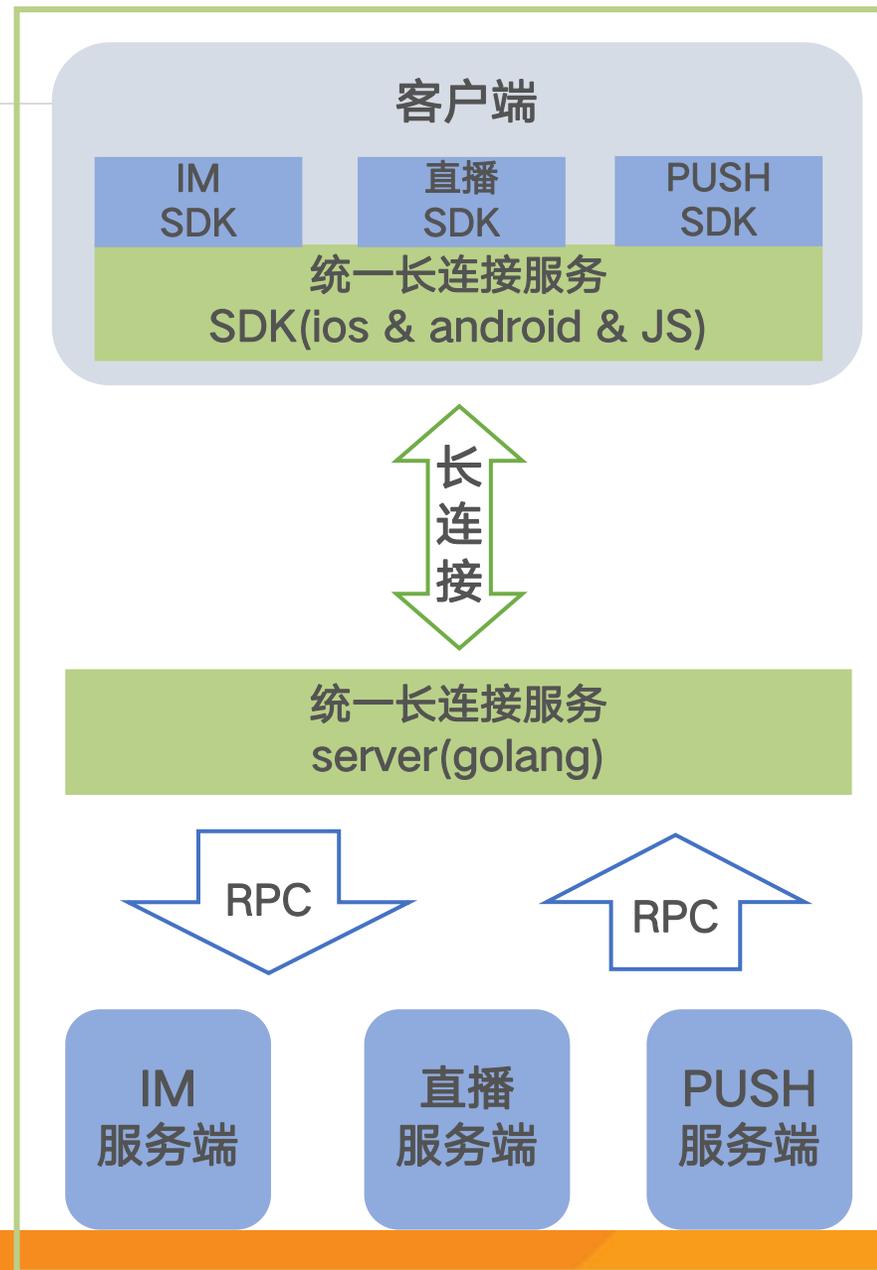
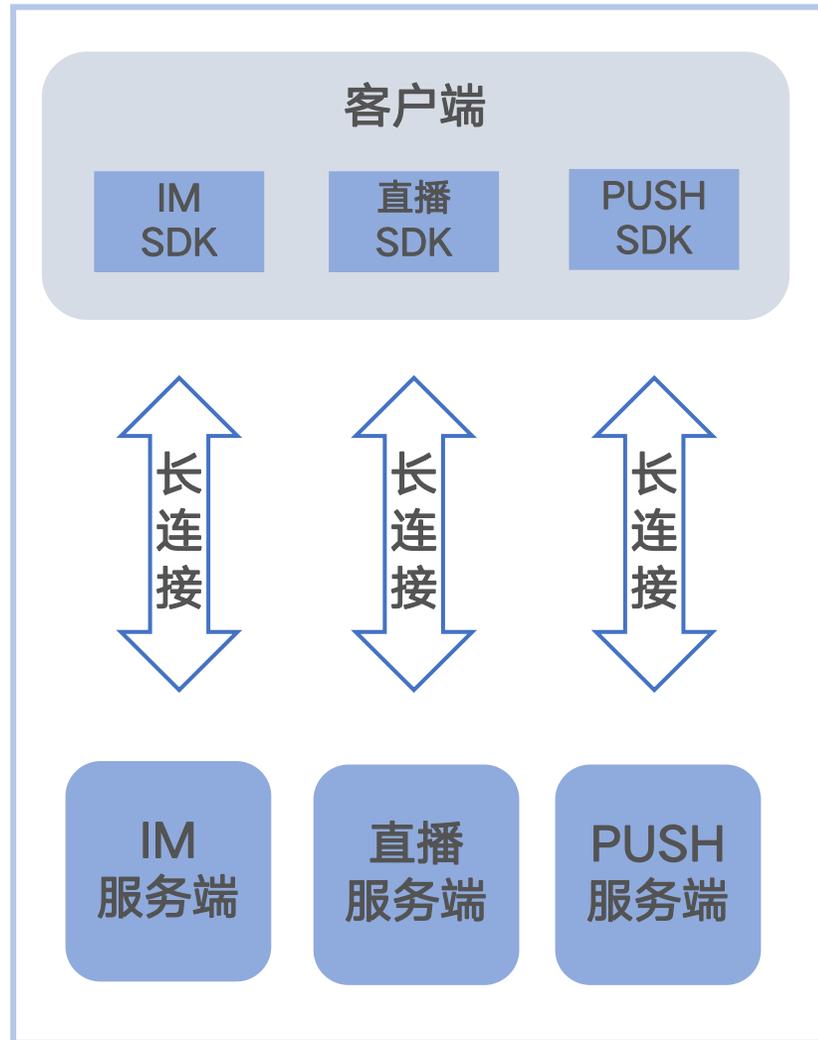
什么是长连接



长连接

- APP 生命周期常驻
- 支持全双工上下行
- 提升实时性、互动性
- 应用场景：消息&直播&PUSH

统一长连接服务背景



02

统一长连接服务介绍



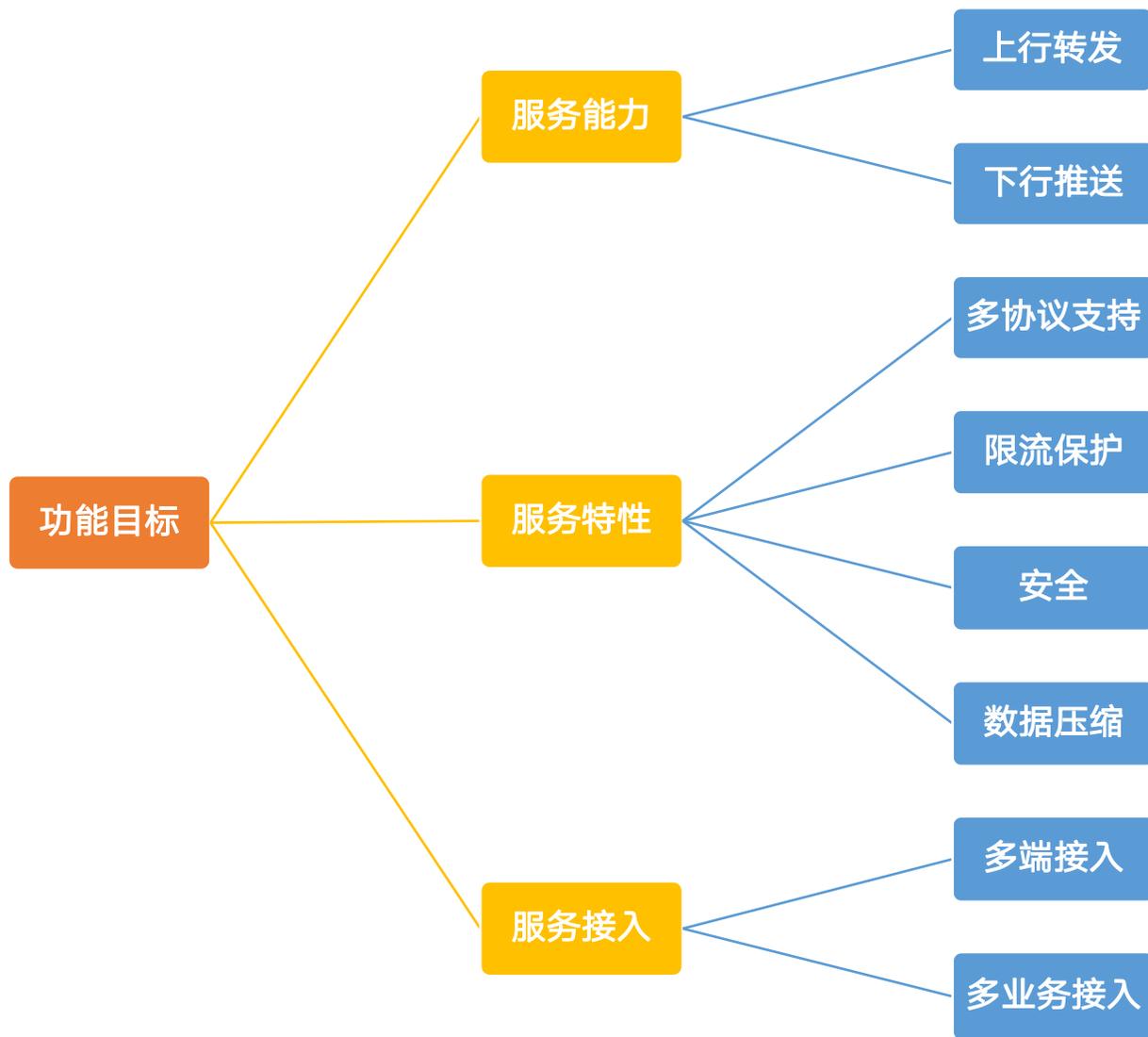
“

支持的业务场景



业务	支持能力	推送场景	推送预计UPS
消息	请求转发 下行推送	单播/批量单播	万级
直播	请求转发 下行推送	组播	千万级
云控	请求转发 下行推送	批量单播	百万级
PUSH	请求转发 下行推送	批量单播	百万级

统一长连接-功能目标



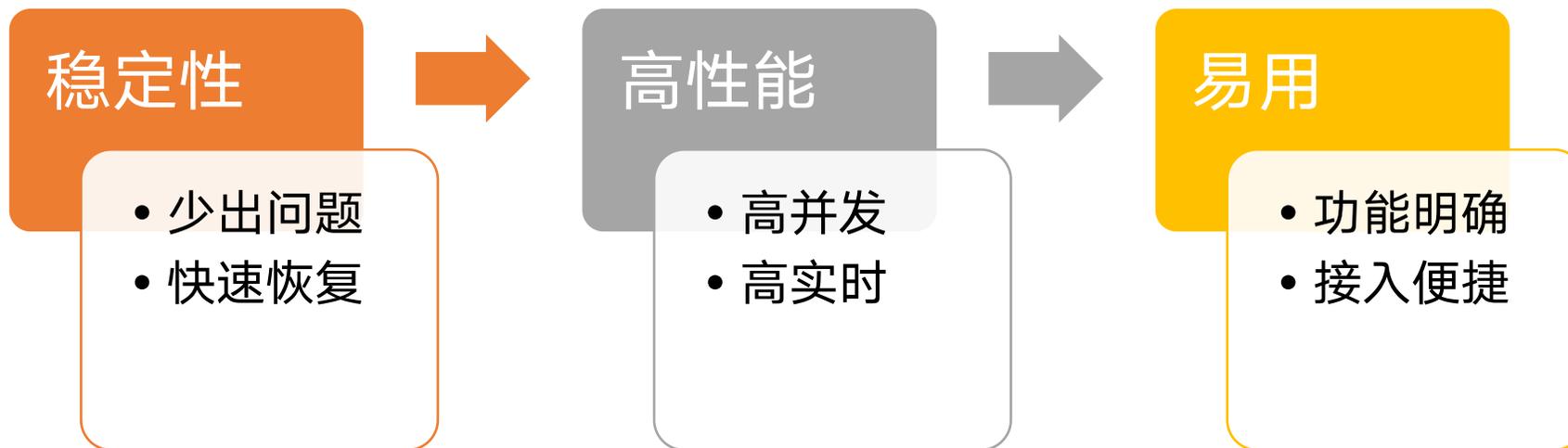
功能目标

- 服务能力
- 服务特性
- 服务接入

统一长连接-性能目标

性能项	性能支持	说明
并发连接数	千万级长连接	支持横向扩容
下行QPS	百万级批量单播推送 千万级组播推送	支持横向扩容
服务延时	毫秒级	-

统一长连接设计目标

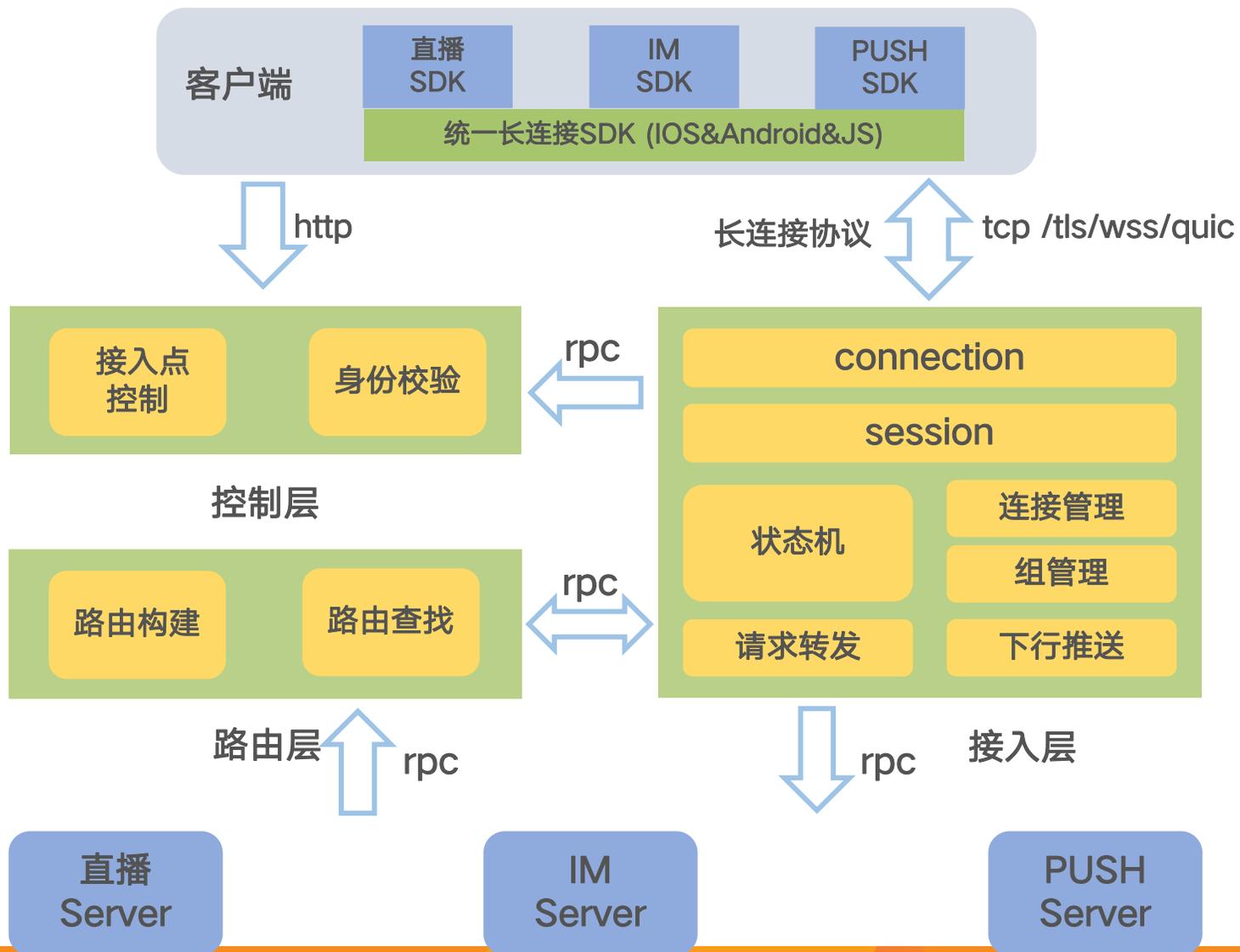


03

统一长连接服务架构



统一长连接服务架构



服务组成

- 长连接协议
- 统一长连接SDK
- 控制层
- 接入层
- 路由层

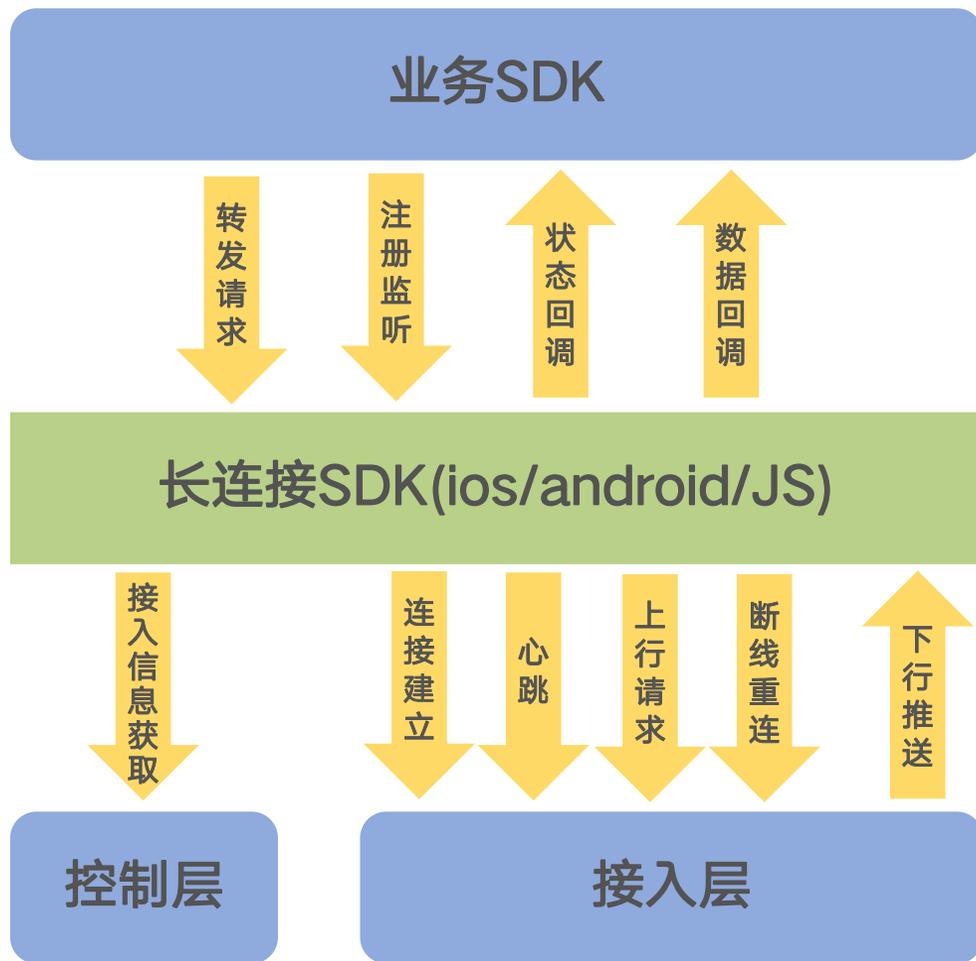
长连接消息协议



对端协议

- 协议组成：协议头，公参，业务数据
- 协议头：协议标识，协议版本
- 公参：设备标识、应用标识、业务标识、请求元数据
- 业务数据：业务自定义，长连接层透传

长连接-SDK



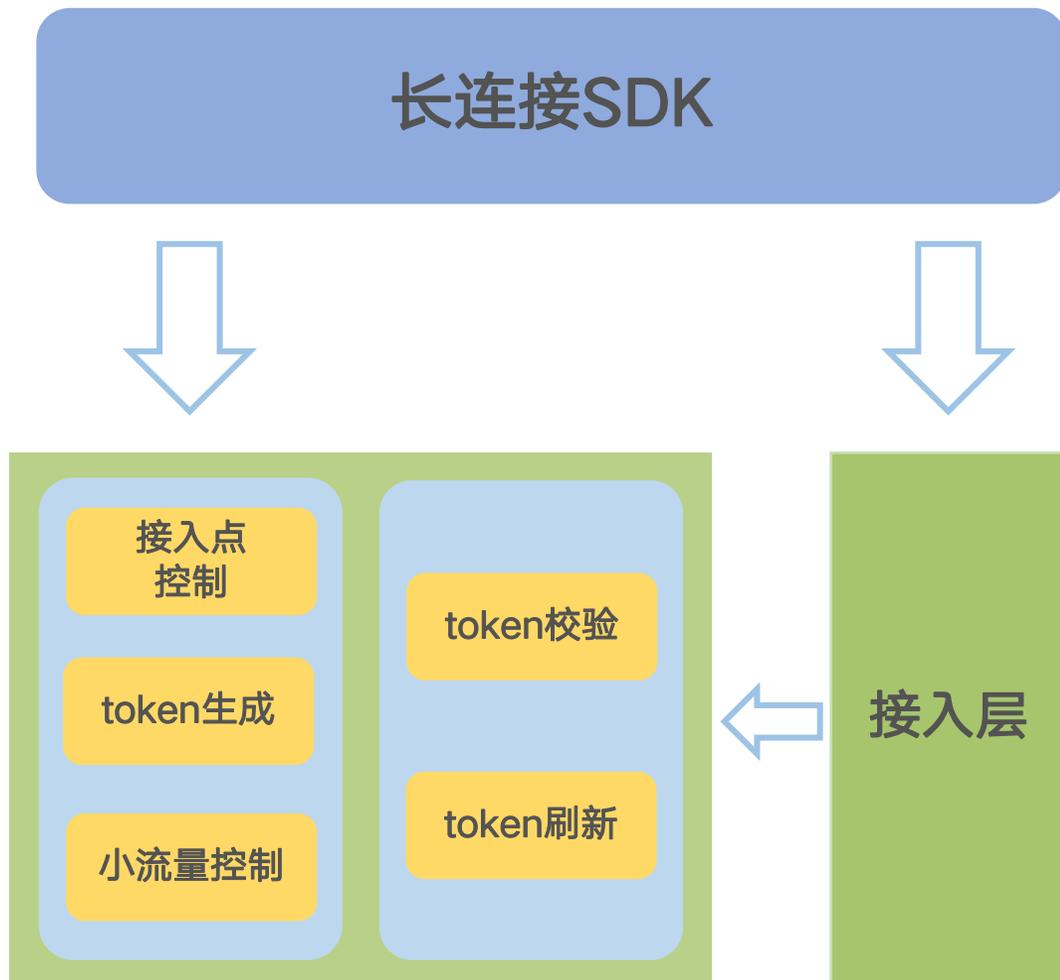
对业务SDK

- 转发请求
- 注册监听
- 连接状态回调
- 下行数据回调

对统一长连接Server

- 接入信息获取: 接入点&协议&token
- 连接建立: 通讯连接建立&长连接登录
- 发送心跳: 维持连接
- 上行请求
- 断线重连
- 下行推送数据收取

长连接-控制层



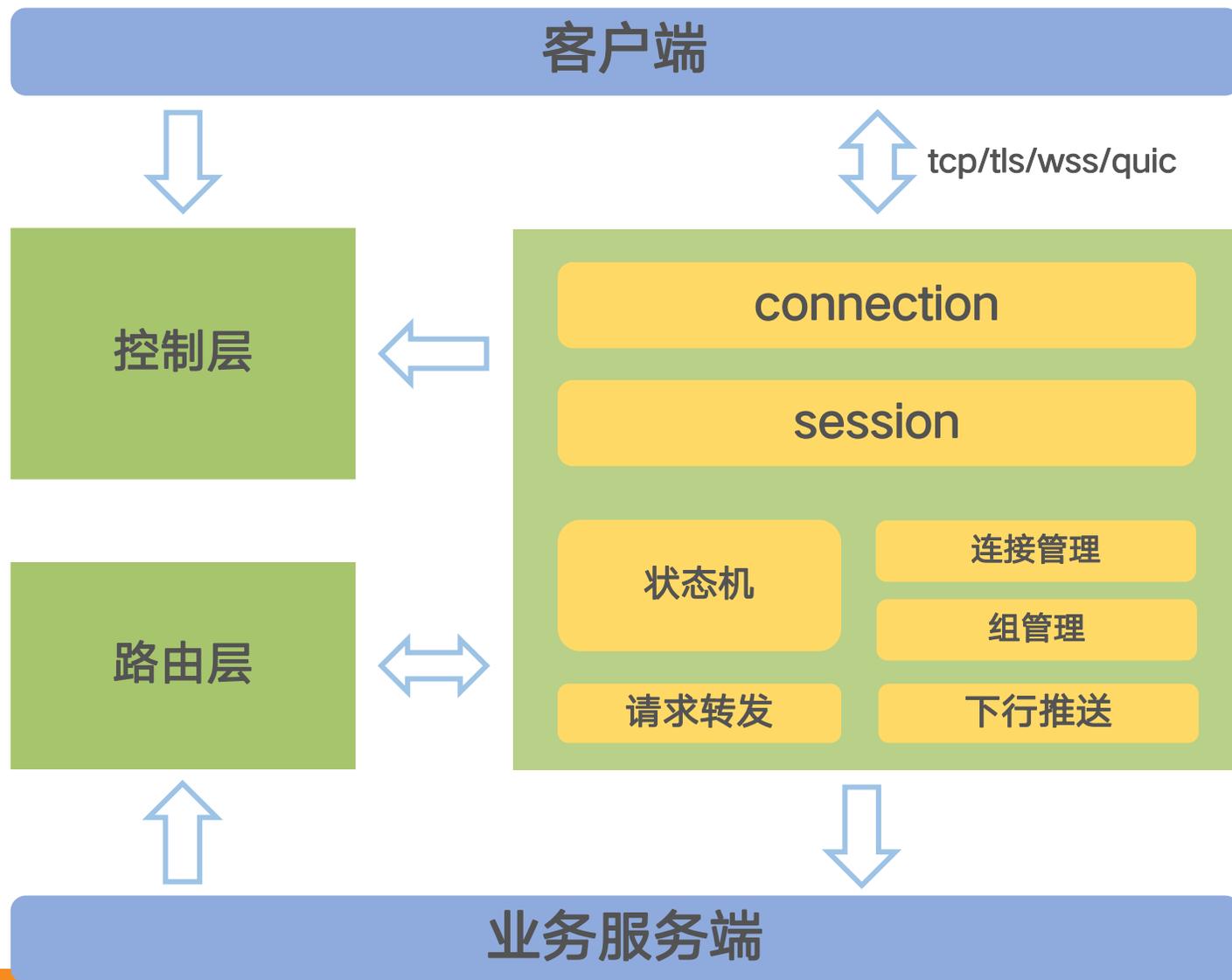
对长连接SDK

- HTTP接口
- 接入点&协议控制
- token生成
- 小流量控制

对接入层Server

- token校验
- token刷新

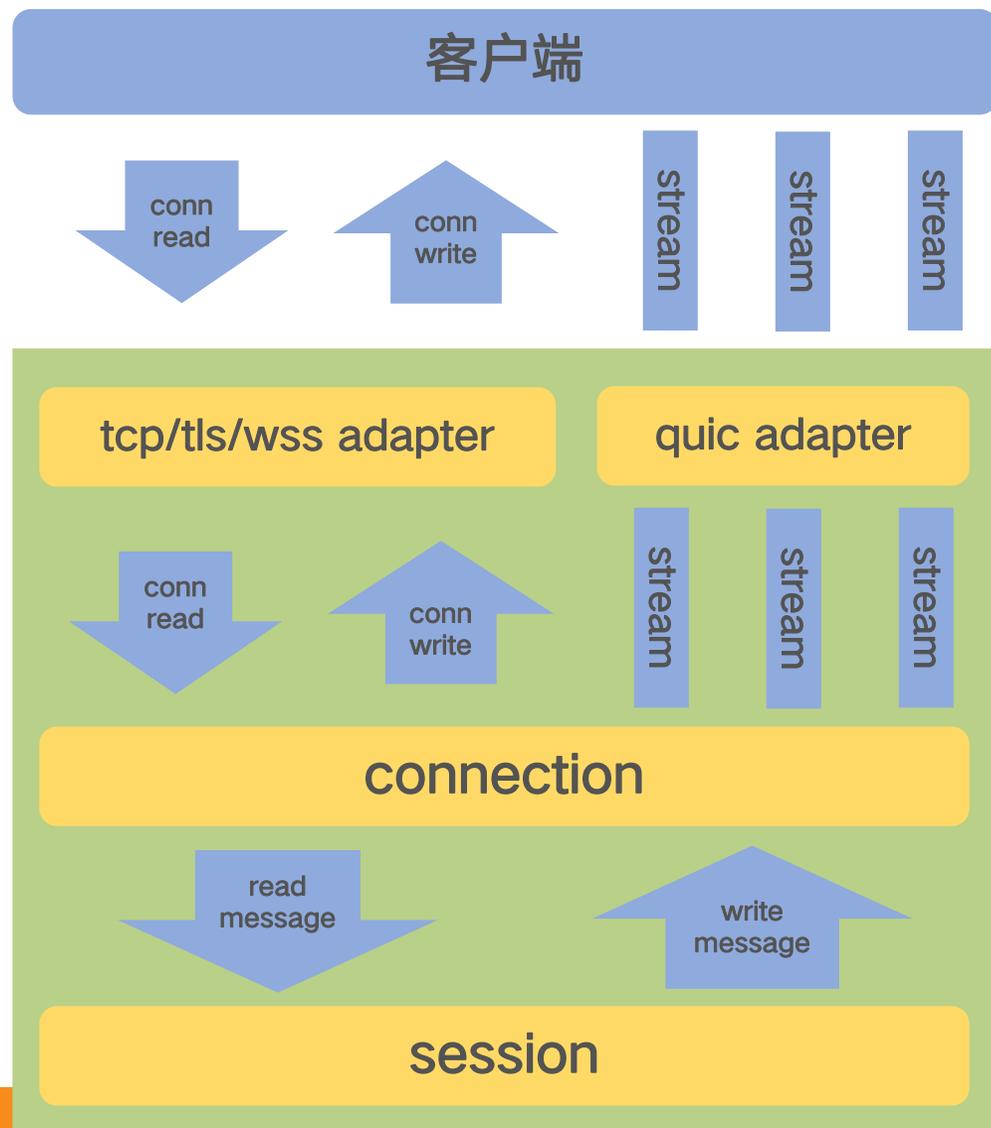
长连接-接入层



接入层功能

- 对端通讯
- 连接管理
- 组管理
- 请求转发
- 下行推送

多协议支持



支持多协议

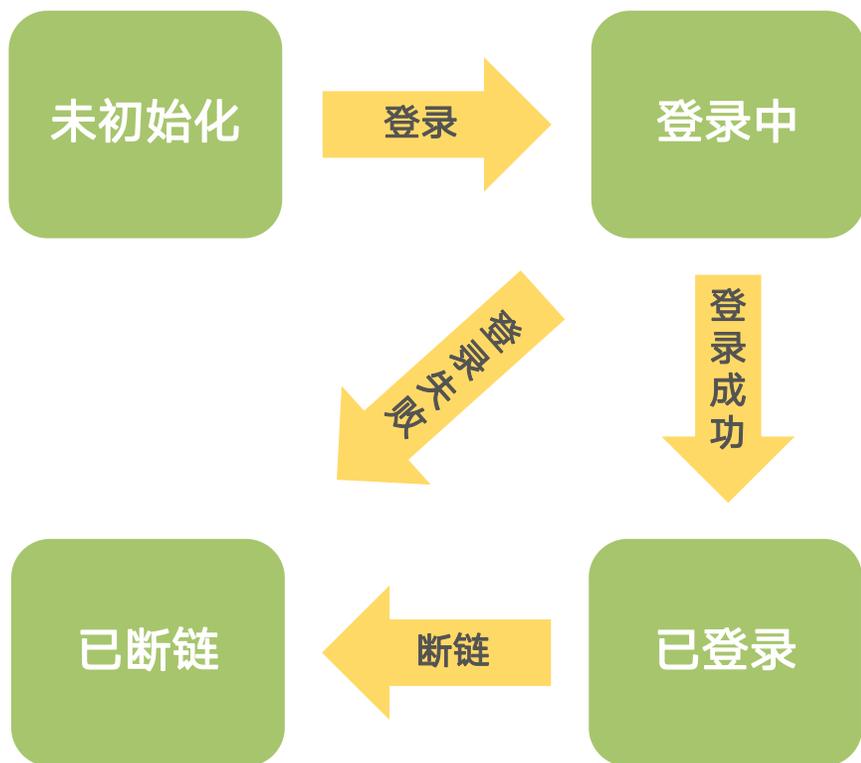
➤ connection层

封装通讯层逻辑，对session提供统一的读写消息的接口，支持多通讯协议

➤ session层

抽象的会话层，提供连接的统一接口

连接状态机



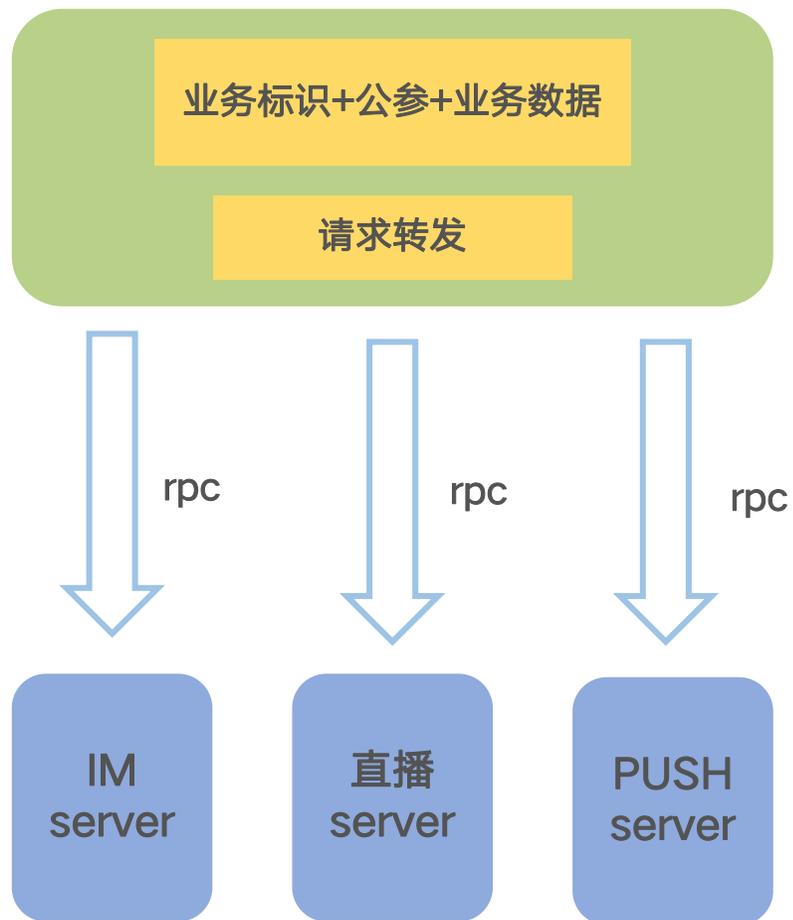
连接管理

连接有状态；存在异常；长连接服务
状态流转需要有明确预期

状态机

- 简化状态管理逻辑
- 处理所有异常事件

请求转发



转发规则：基于消息协议中的业务ID

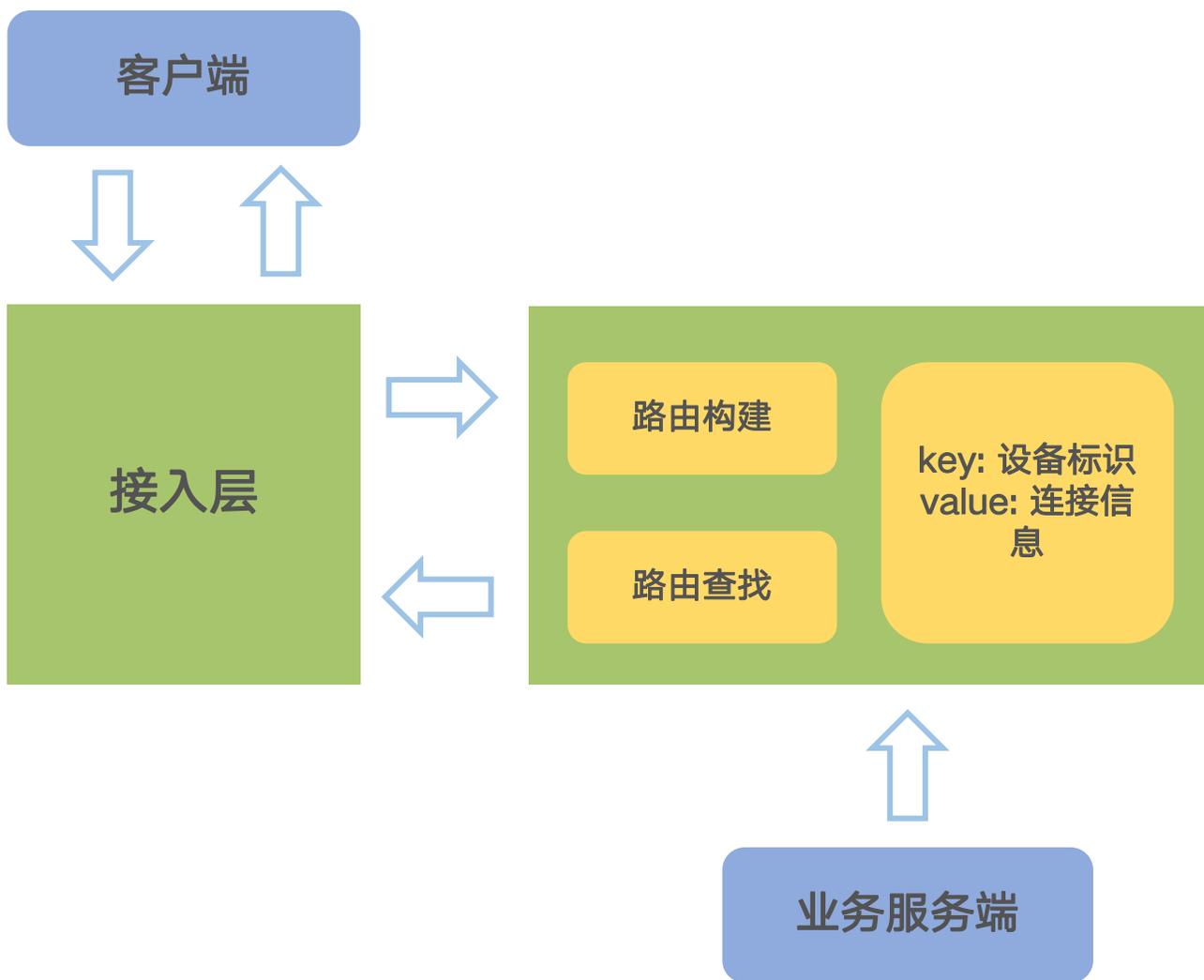
转发数据接收

- 回调数据：长连接公参 + 业务数据
- 业务回调方式：回调接口，基于RPC回调
- 状态回调：长连接断连回调

批量单播推送 VS 组播推送

推送方式	推送场景	用户包稳定性	推送频率	实时性要求	实际场景
批量单播推送	给一组用户分批推不同或者相同消息	用户包基本不变, 或则变化频率低	中	中(全量用户完全推完分钟级)	PUSH、IM
组播推送	给一组用户推相同的消息	用户包变化快	高	高(秒级)	直播

批量单播推送



批量单播推送

基于用户标识，给批量指定用户推送数据

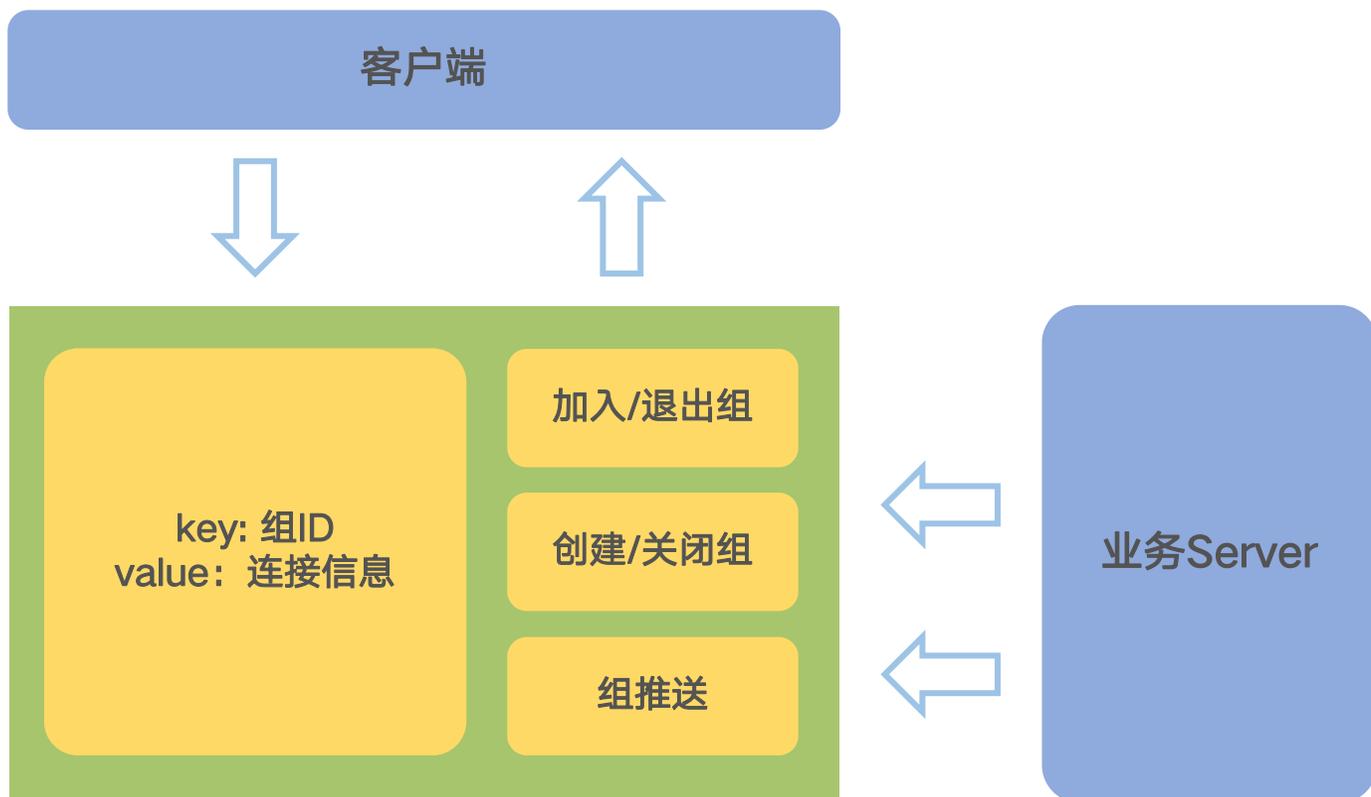
路由构建

- 接入层：连接标识->session信息
- 路由层：设备标识->连接标识+连接IP

路由查找

- 设备标识->连接信息
- 连接信息：连接标识 + 连接IP

组播推送



组播推送

连接加入组，指定组推送消息

组创建

- 使用方创建
- 组ID 全局唯一
- 接入层维护组关系

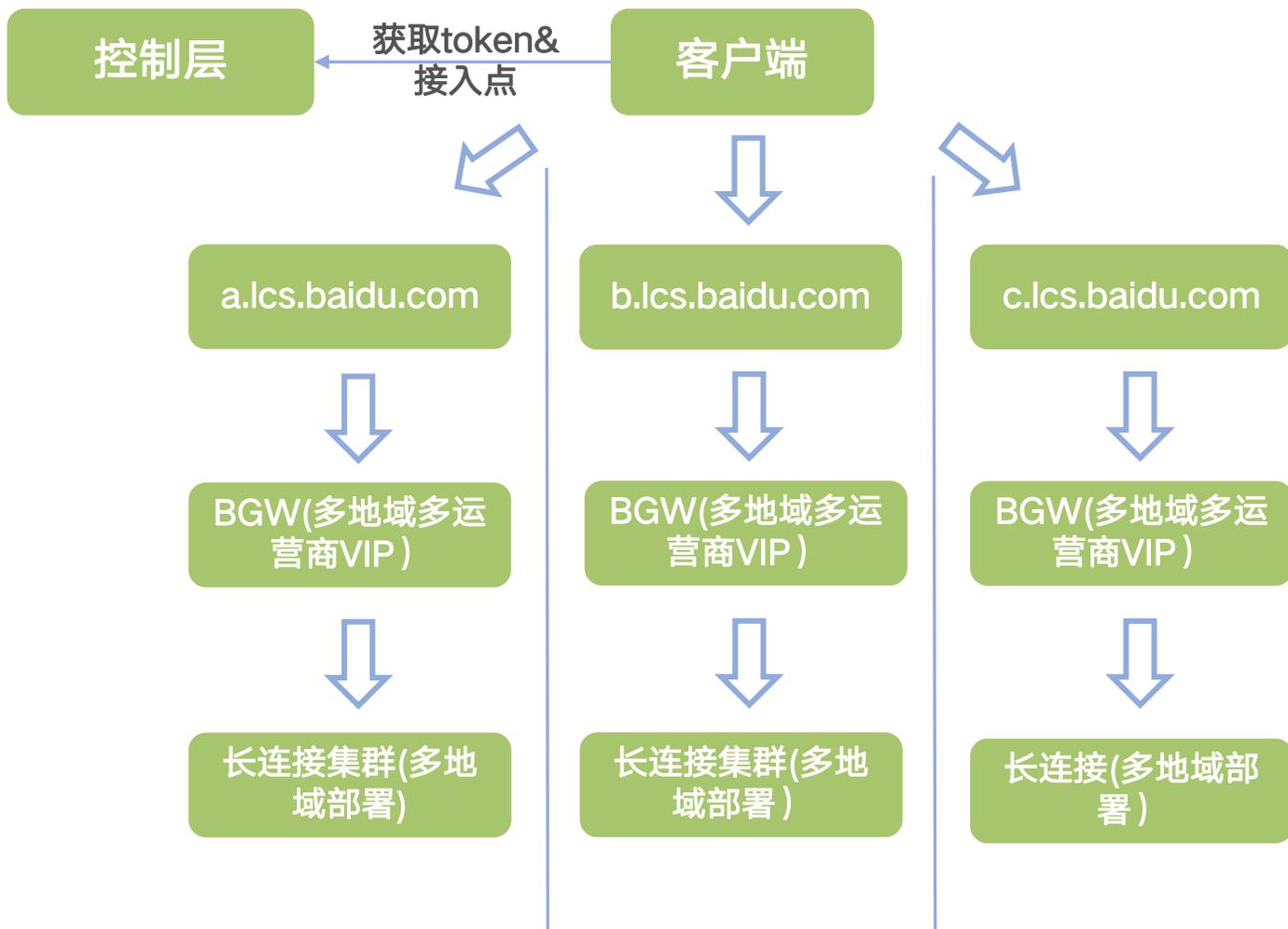
组加入

- 连接期间用户可以主动加入,退出
- 连接关闭即退出
- 组ID -> 一组连接信息

优势

接入层做组->连接路由拆分，支持高并发，频繁推送场景

服务部署



服务端部署

- 控制层：控制业务接入域名
- 域名：多地域多运营商VIP
- VIP：对应多机房集群
- 多集群：多机房容灾

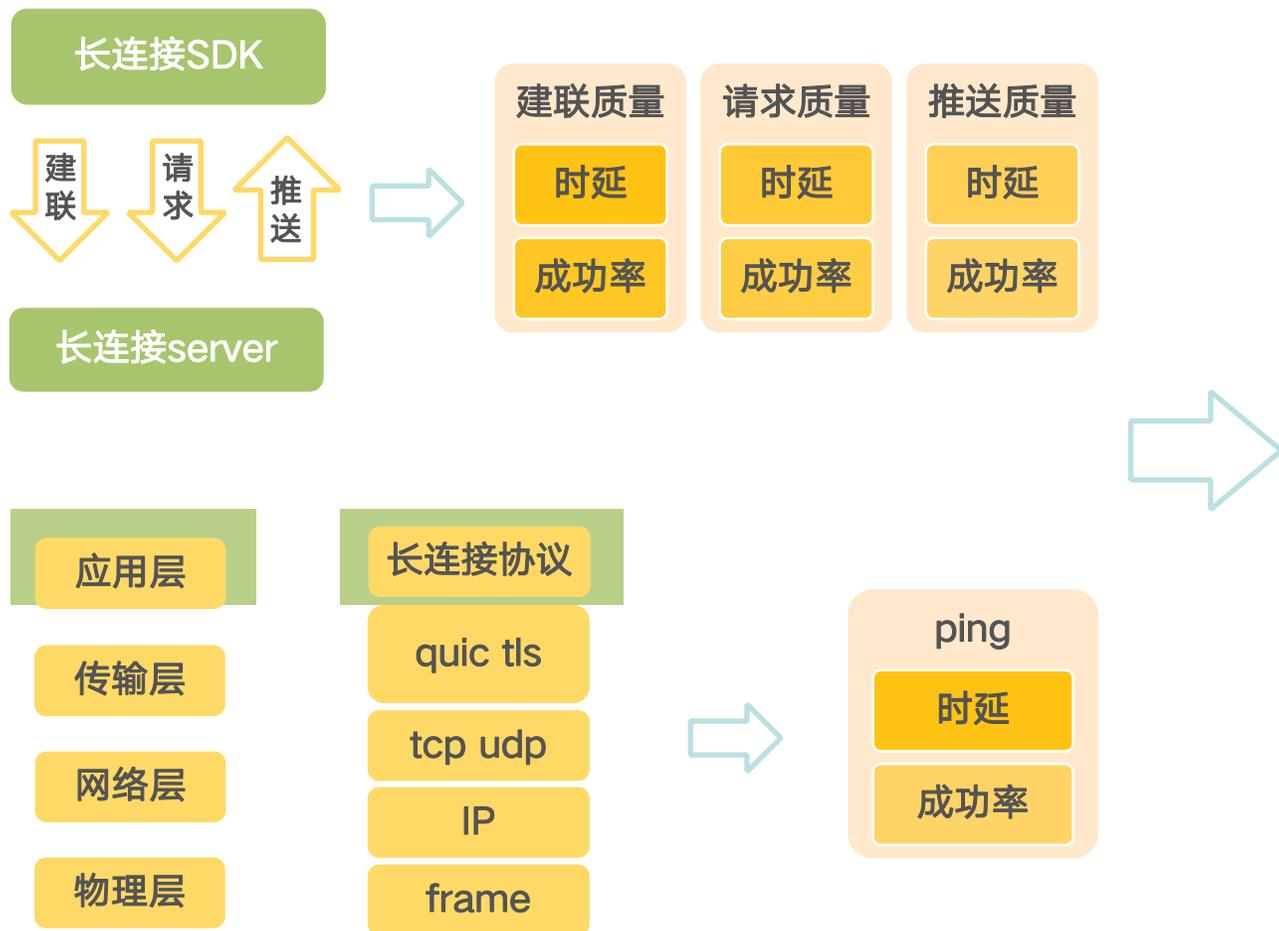
部署优势

业务隔离、多地域容灾

质量监控-服务端监控

监控项	监控指标	说明
在线用户数	在线session数	在线量级，和DAU相关
上行服务监控	上行请求 qps、延时、成功率	影响请求转发质量
推送服务监控	下行推送qps，延时、成功率	影响在线推送质量
资源监控	Cpu 内存 磁盘 网络带宽	集群资源压力情况

质量监控-端质量监控



	Ping成功率 时延A	Ping成功率 时延B
建联成功率	xxx	xxx
建联时延	xxx	xxx

统一长连接质量体系

- 网络质量由ping 成功率&延时来衡量，根据实际数据分布，分组划分弱网、中网、优网
- 对比不同网络质量下，不同协议的建联成功率&建联时延，来衡量不同协议在不同网络质量下表现优劣

04

统一长连接golang实践



golang开发优势

开发

- 开发效率
- 依赖工具&组件

测试

- 测试工具
- 压测

运维

- 稳定性
- 问题定位

开发

- 语法简单，易于上手
- 网络开发模式清晰，心智负担低
- 长连接开发依赖工具&组件完整
- 代码依赖清晰，利于问题追踪

测试

- 原生支持单测
- 自带profile工具，性能问题易于定位

运维

- 支持异常捕获，保障服务稳定
- 线上问题易于定位

协议序列化&反序列化

- **数据拼接**：避免内存拷贝、多次分配内存
- **内存管理**：利用内存池&多级内存池

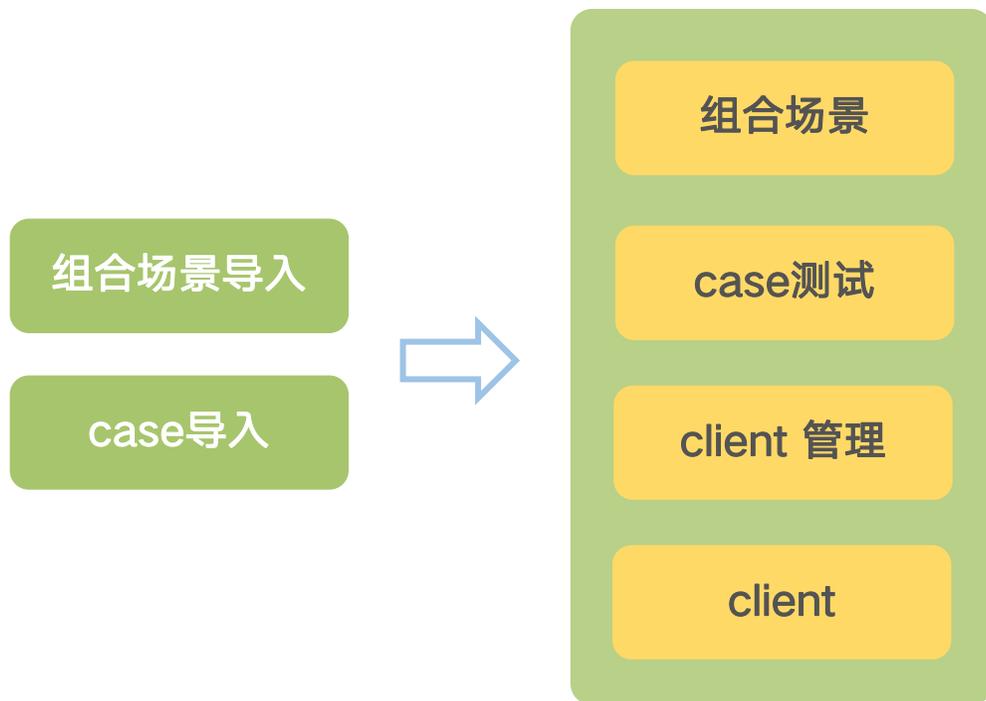
三方库

- **相关依赖库**
 - pb、JSON、数据压缩、RPC、日志库
- **三方库选择**
 - 功能-功能测试，理解三方库设计
 - 性能-业务场景性能压测

对象复用

- **sync.Pool**
 - 对象复用减少GC 压力
- **goroutine 复用**
 - 限制并发，减少瞬时压力冲击
 - 减少goroutine 频繁申请释放

测试工具开发



长连接测试：私有协议、有状态，场景测试

测试工具

- case测试
- 组合场景测试

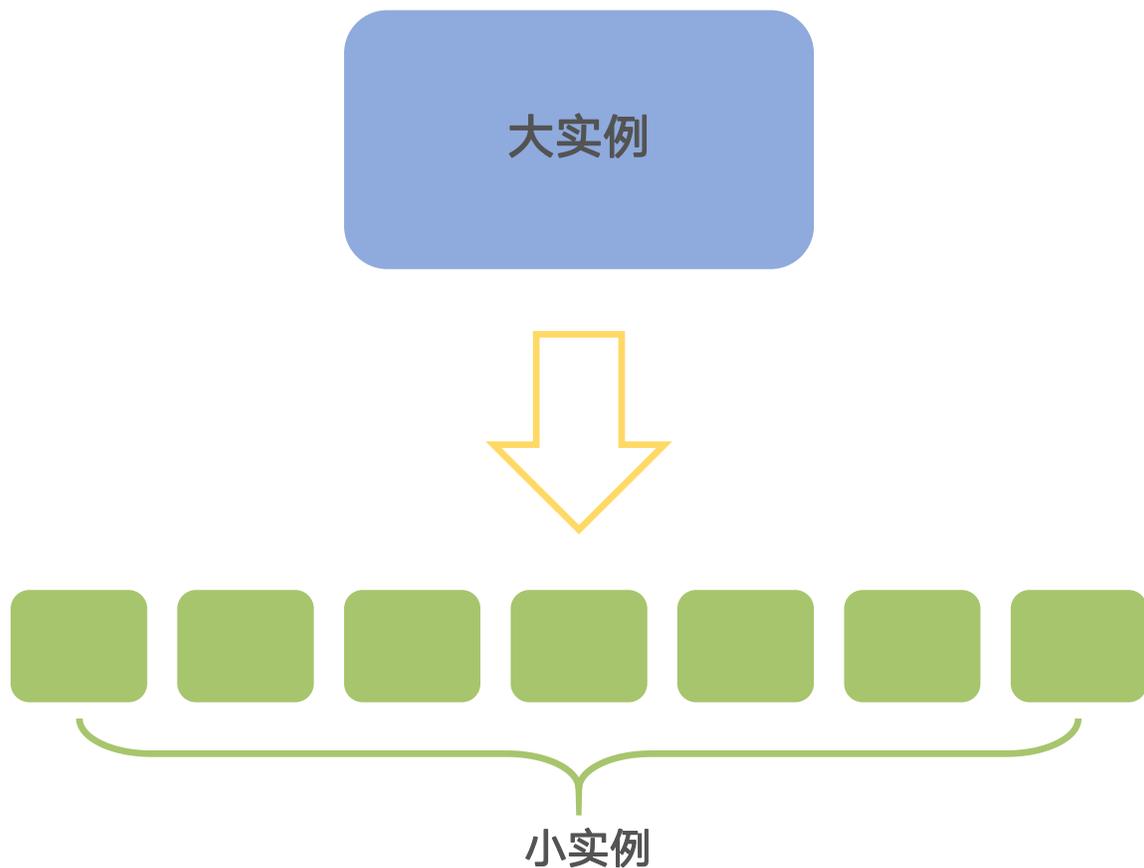
压测工具

- 压测场景：连接数压测、批量单播压测、组播压测
- 统计数据：触达、延时
- 压测工具：client 模拟用户行为，性能高、支持数据统计

golang测试工具

- 技术栈统一，基础组件复用，保障性能
- 引入cli 框架，快速构建
- 场景配置化，支持多场景用户行为模拟
- Prometheus 支持监控

运维



大实例问题

- 大实例占用内存多，GC压力大
- 实例重，对机器性能要求高
- 上线、迁移，运维成本高

结论

集群需要单独部署，串行上线，实例限制迁移

小实例集群

- 单实例占用内存小，GC压力小
- 单实例对资源要求不高
- 单实例上线，迁移，对整个集群影响有限

引入问题

集群实例增多，集群管理复杂度上升

05

总结与规划



总结

需求

- 明确边界，时刻记住什么该做，什么不该做

设计

- 方案选择在满足需求的情况下，尽量做到简单

运维

- 有时候大且强不一定比小且灵活更好用

规划

精细化

完善长连接全链路网络质量数据统计和分析能力

智能化

端上连接、接入点支持依赖实际网络环境切换

场景拓展

探索长连接支持更多的场景和业务



Thank You!