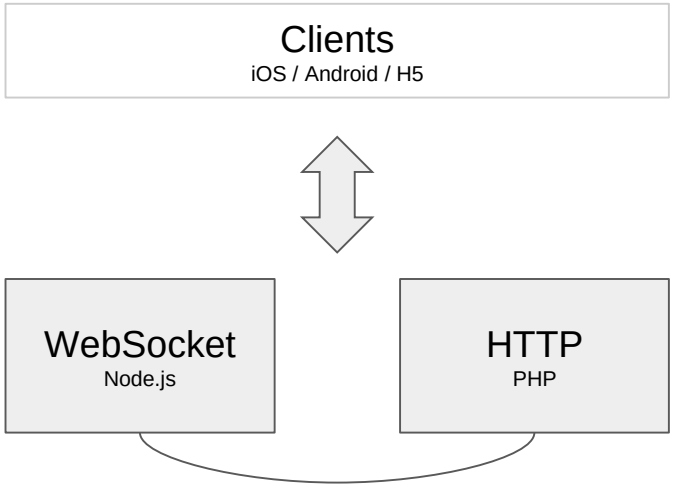
直播平台 IM 系统实战

张皓聪

1.0 需求

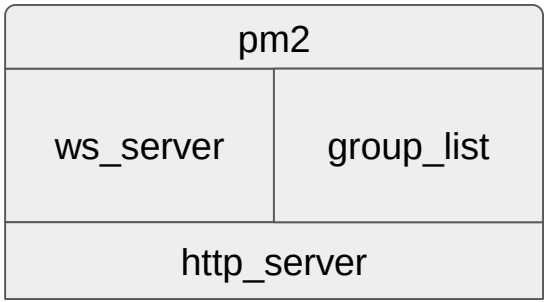
- 一周出 Demo
- 还要差不多能用
- 给老板演示的时候不要突然崩溃
- 万一崩溃了最好不要被发现
- 性能无所谓
- 总之就是要快点做出来

1.0 架构

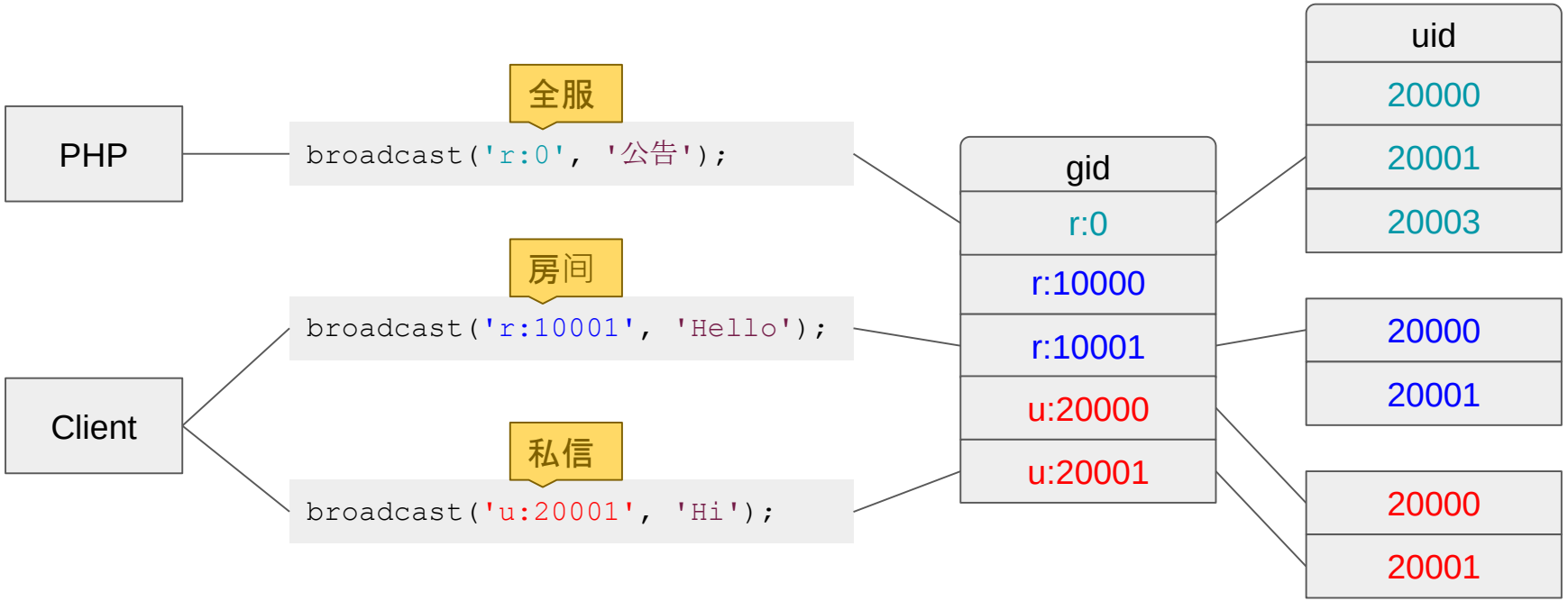


前端服务器

- 也叫接入服务器
- 使用 WebSocket 长连
- 使用 `pm2` 守护进程
- 使用组的形式管理连接
- 负责业务
    - 用户信息和登录
    - 房间在线列表和聊天信息推送
    - PHP 推送



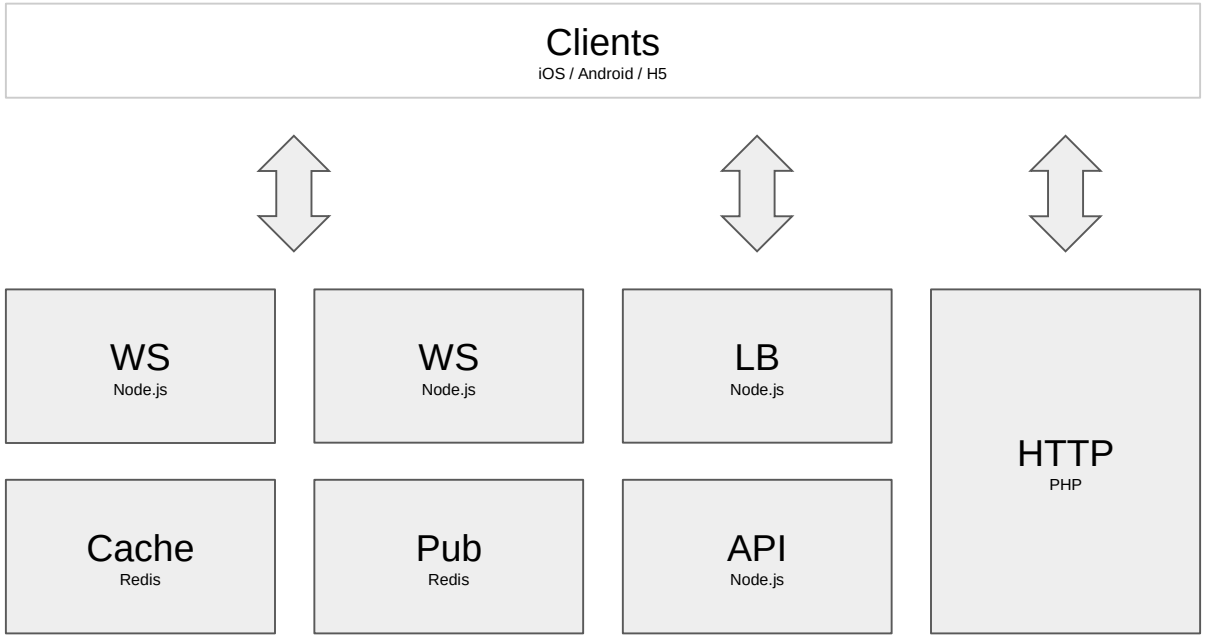
向组推送消息


PHP
Client
全服
broadcast('r:0', '公告');
房间
broadcast('r:10001', 'Hello');
私信
broadcast('u:20001', 'Hi');
gid
r:0
r:10000
r:10001
u:20000
u:20001
uid
20000
20001
20003
20000
20001
20000
20001

2.0 需求

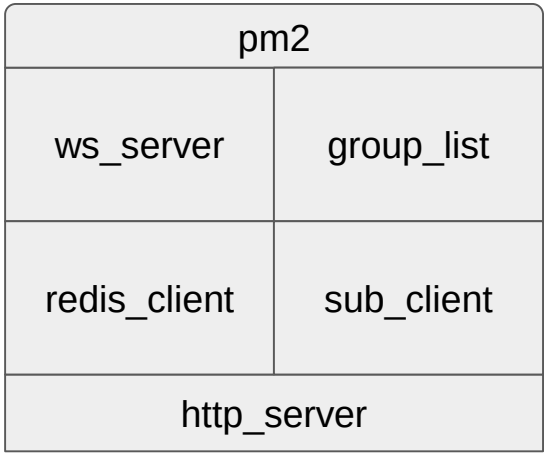
- 这是一个内测版本
- 支持扩容
- 需要更高的可用性
- 礼物信息要比聊天优先级高
- 特定消息要发送回执

2.0 架构

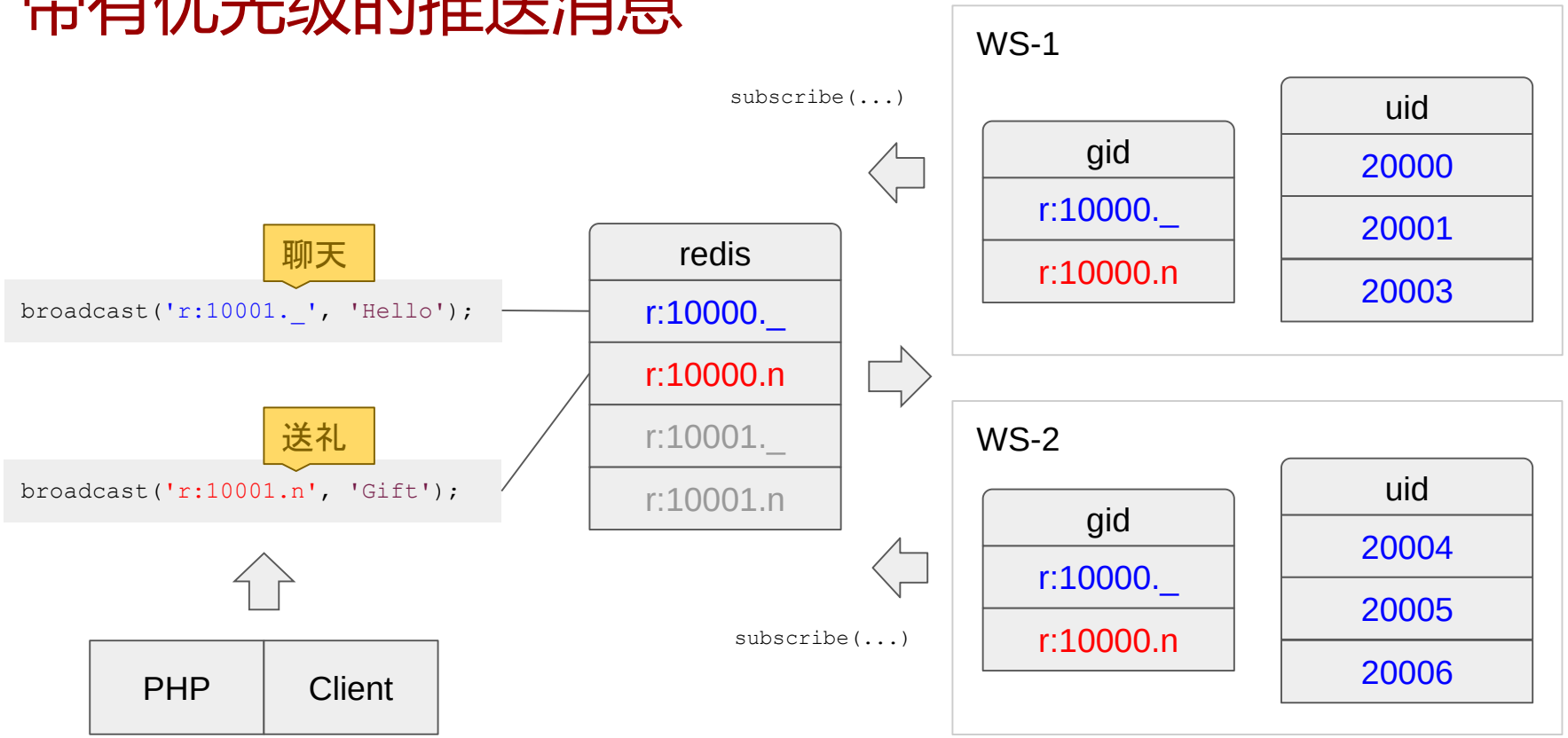

Clients
iOS / Android / H5
WS
Node.js
WS
Node.js
LB
Node.js
HTTP
PHP
Cache
Redis
Pub
Redis
API
Node.js

前端服务器

- 使用 Redis 保存在线列表
  以便跨进程 / 主机共享
- 增加了广播订阅器，用于接收推送
- 增加了广播优先级，确保礼物信息优先到达



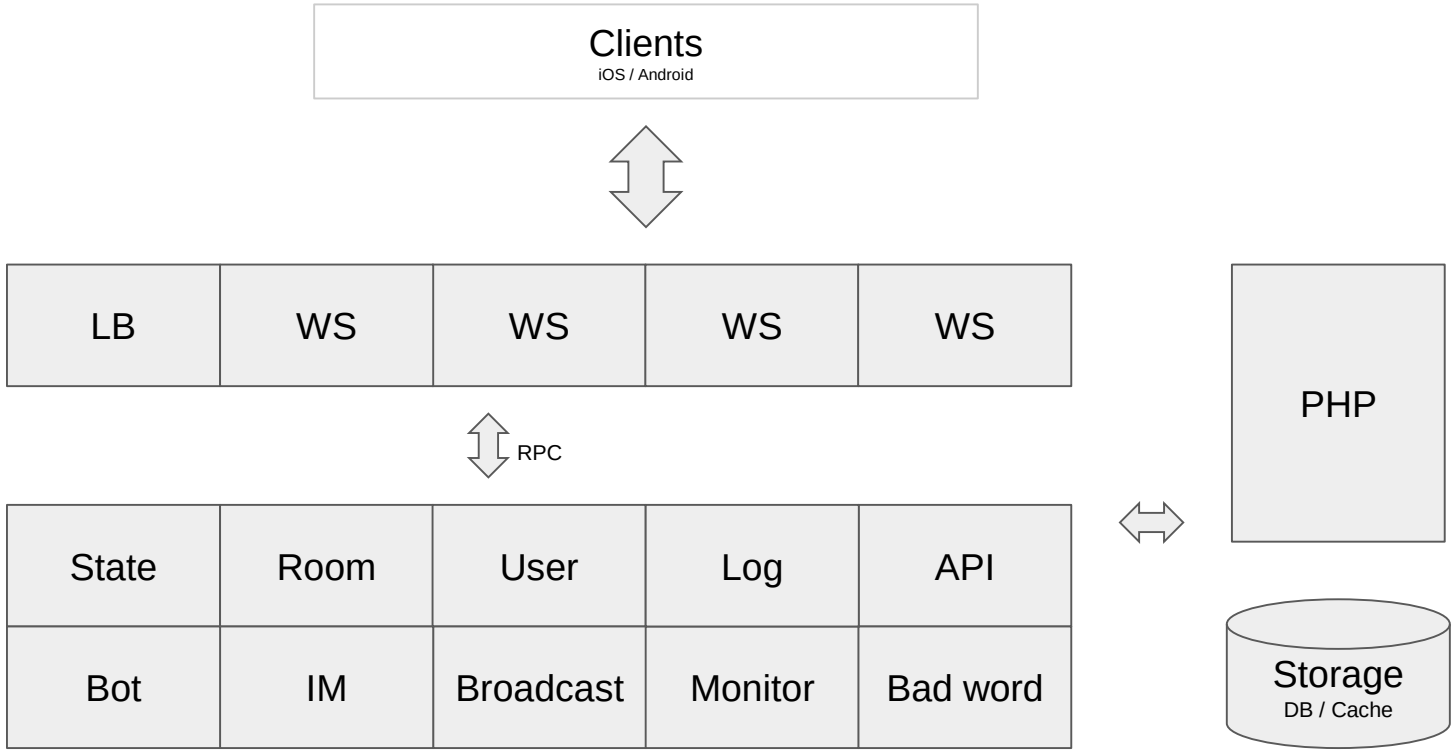
带有优先级的推送消息


WS-1
subscribe(...)
gid
r:10000._
r:10000.n
uid
20000
20001
20003
聊天
broadcast('r:10001._', 'Hello');
redis
r:10000._
r:10000.n
r:10001._
r:10001.n
送礼
broadcast('r:10001.n', 'Gift');
WS-2
gid
r:10000._
r:10000.n
uid
20004
20005
20006
subscribe(...)
PHP
Client

3.0 需求

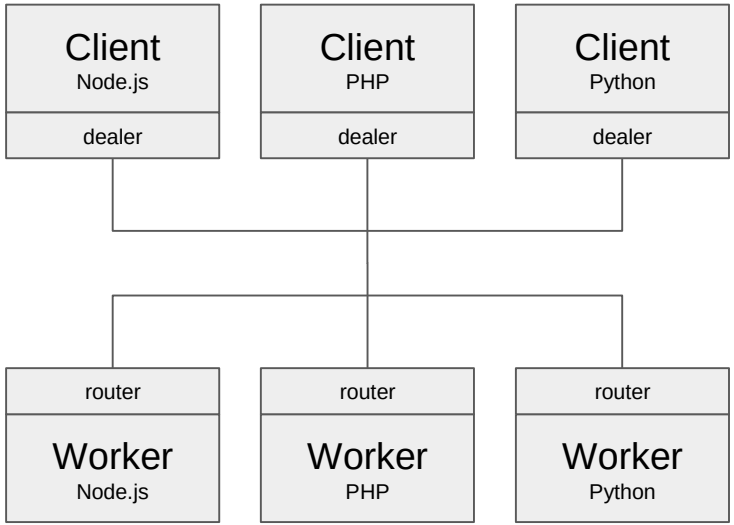
- 业务逻辑越来越多，需要拆分
- 需要支持热更
- 更好的广播性能
- 优化与其他服务通信
- 日志系统
- 部署脚本

3.0 版本架构


Clients
iOS / Android
LB
WS
WS
WS
WS
RPC
State
Room
User
Log
API
Bot
IM
Broadcast
Monitor
Bad word
PHP
Storage
DB / Cache

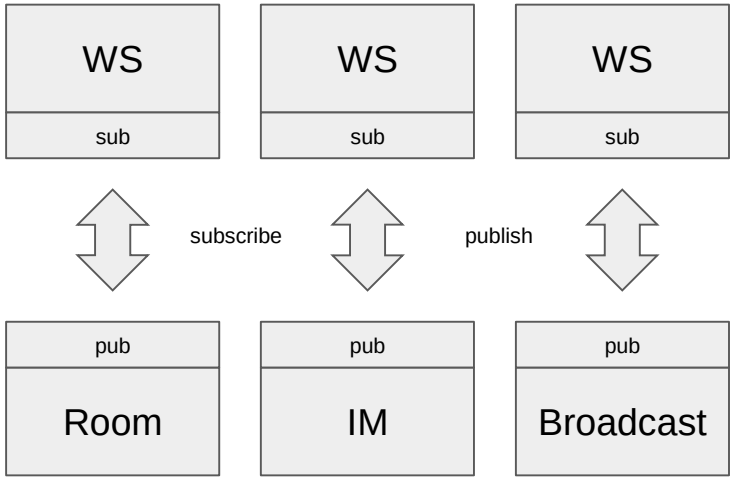
RPC 服务框架

- 使用 ZeroMQ 的 Dealer 和 Router
- 数据交换格式使是 JSON
- 负载均衡
- 支持热更
- 方便调试
- 很多语言绑定

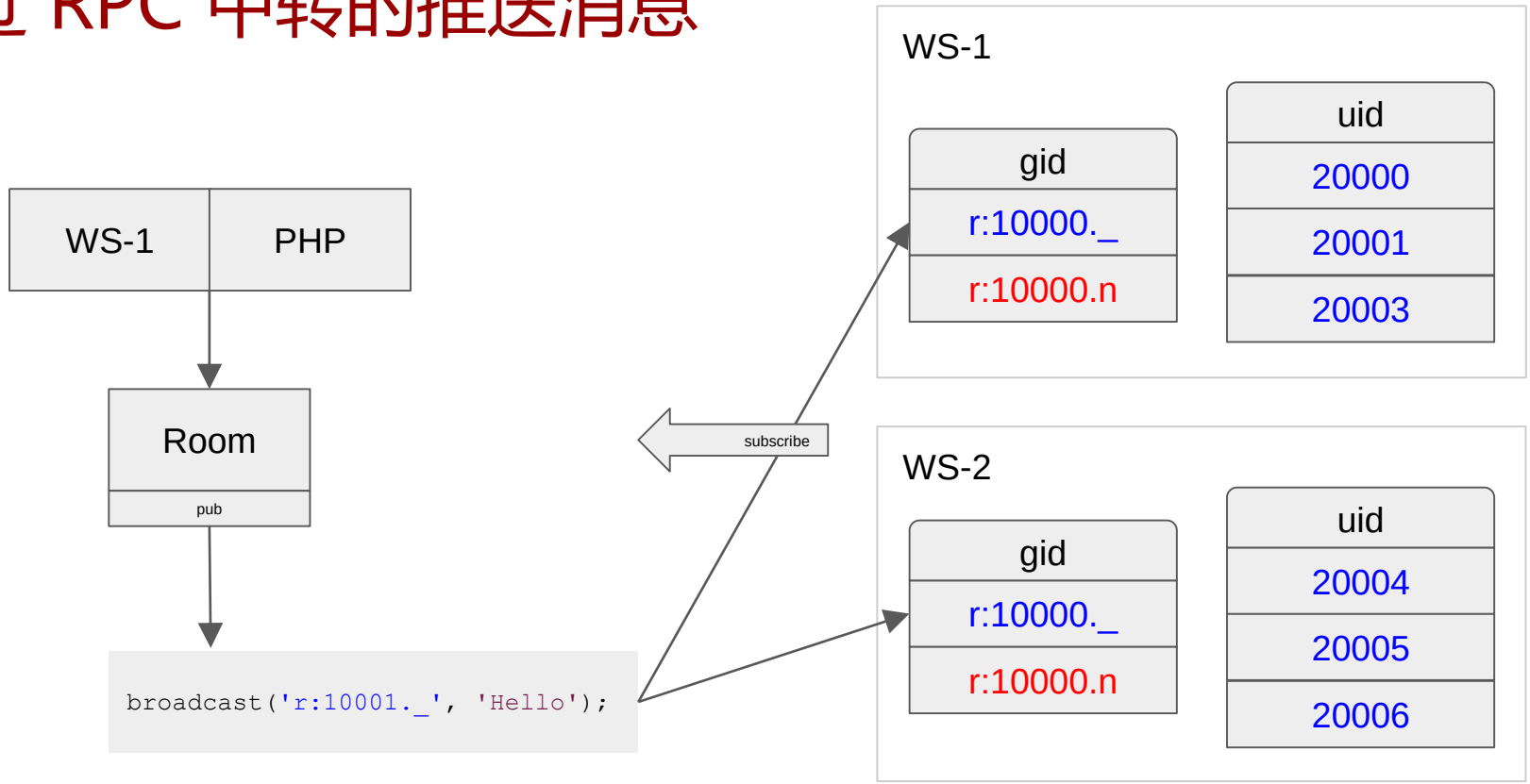


广播服务

- 使用 ZeroMQ 的 Pub 和 Sub
- 广播逻辑和业务是绑定的
  不需要广播的业务就不用起 Pub
- 定期向监控服务器 ping



通过 RPC 中转的推送消息



ZeroMQ 性能

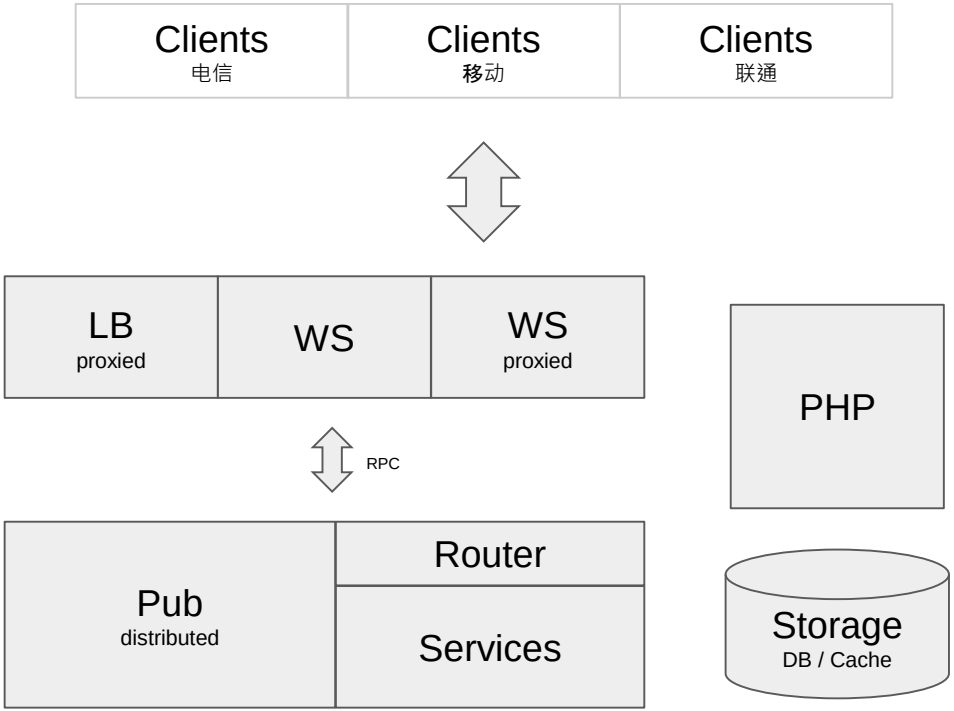
- E5-2630，8G 内存，千兆网卡的测试结果

```
# local_thr tcp://*:5555 100 10000000
message size: 100 [B]
message count: 10000000
mean throughput: 1097051 [msg/s]
mean throughput: 877.641 [Mb/s]
```

4.0 需求

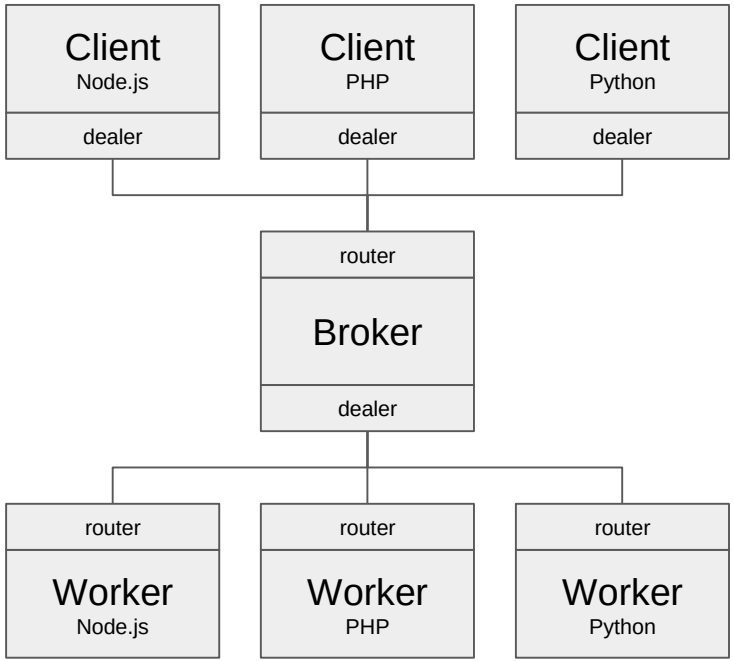
- 广播和业务分离
- 优化 RPC 协议
- 针对个别地区网络优化

4.0 架构


Clients
电信
Clients
移动
Clients
联通
LB
proxied
WS
WS
proxied
RPC
PHP
Pub
distributed
Router
Services
Storage
DB / Cache

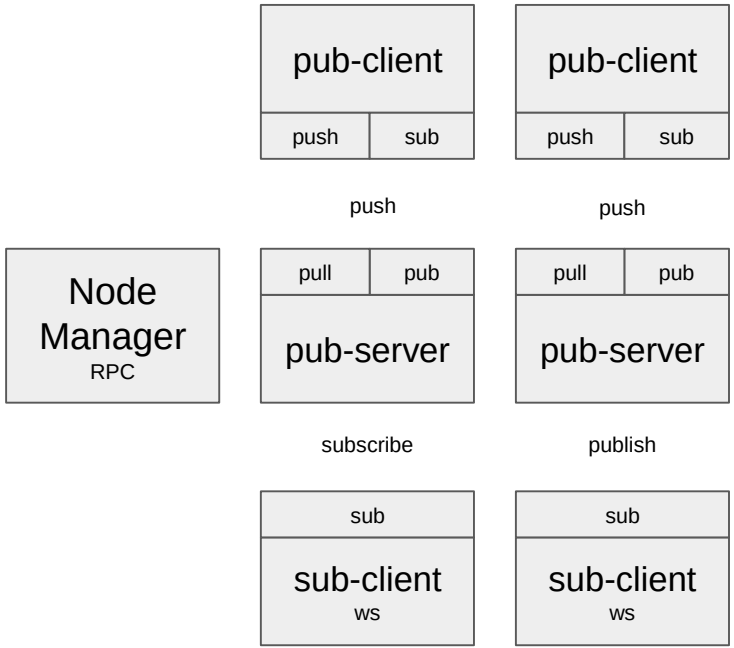
RPC 服务路由

- 客户端不必知道每个 Worker 的地址
- 由 Broker 负责统一转发请求
- Worker 的种类，数量也可以随意增减

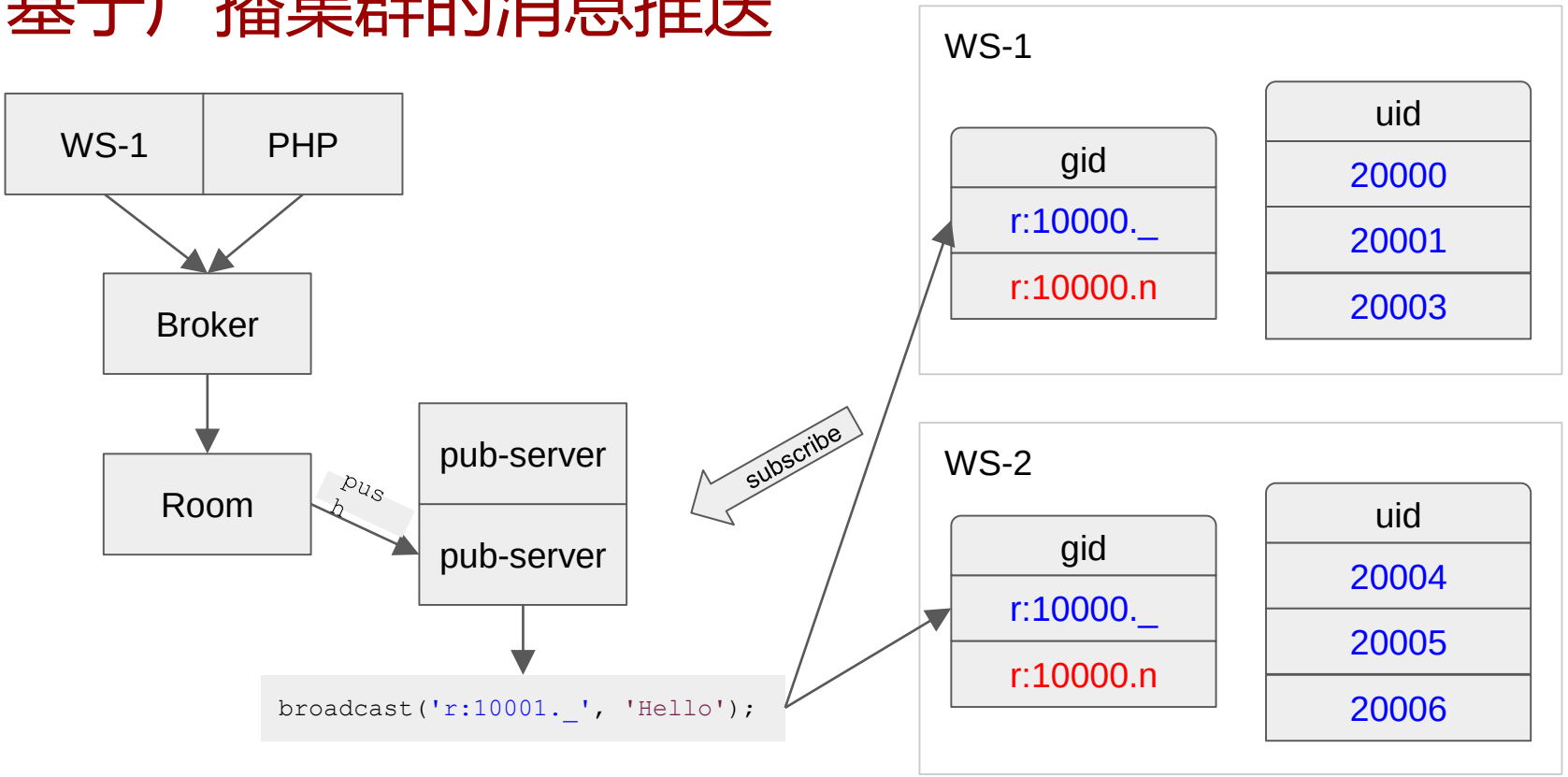


广播服务集群

- 基于 ZeroMQ
- 分别用到了 push, pull, pub, sub 等模式
- 支持动态扩容
- 新的 Node Manager



基于广播集群的消息推送



Q & A