QCon 全球软件开发大会
INTERNATIONAL SOFTWARE DEVELOPMENT CONFERENCE

BEIJING 2017

Golang微服务

在腾讯游戏用户运营领域的探索及实践

刘家雄

InfoQ queue 促进软件开发领域知识与创新的传播

关注InfoQ官方信息
及时获取QCon软件开发者
大会演讲视频信息

扫码，获取限时优惠

ArchSummit
全球架构师峰会2017 [深圳站]
2017年7月7－8日 深圳·华侨城洲际酒店
咨询热线：010-89880682

QCon
全球软件开发大会 [上海站]
2017年10月19-21日
咨询热线：010-64738142

Web Architecture

Platform

关于我

刘家雄 <@楚吟风>

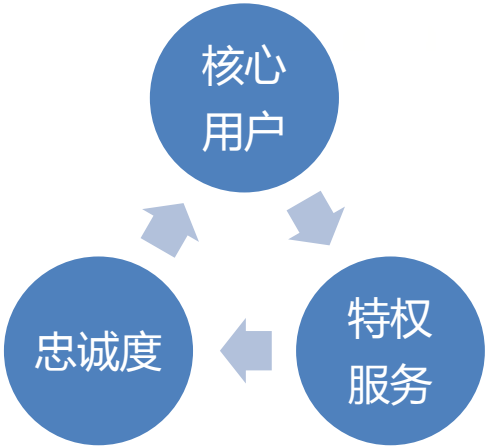
提纲

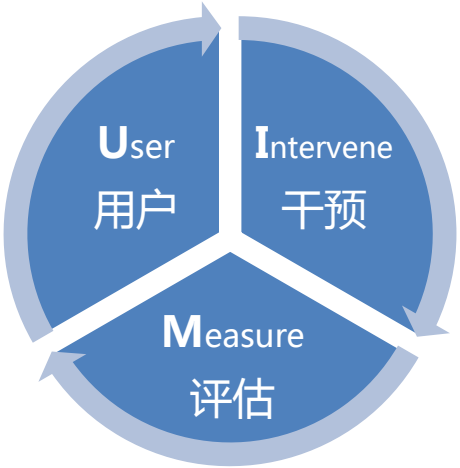
- 腾讯游戏用户运营介绍
- 服务化架构演进
- DSL加速敏捷
- 基准数据
- 总结及展望

腾讯游戏用户运营

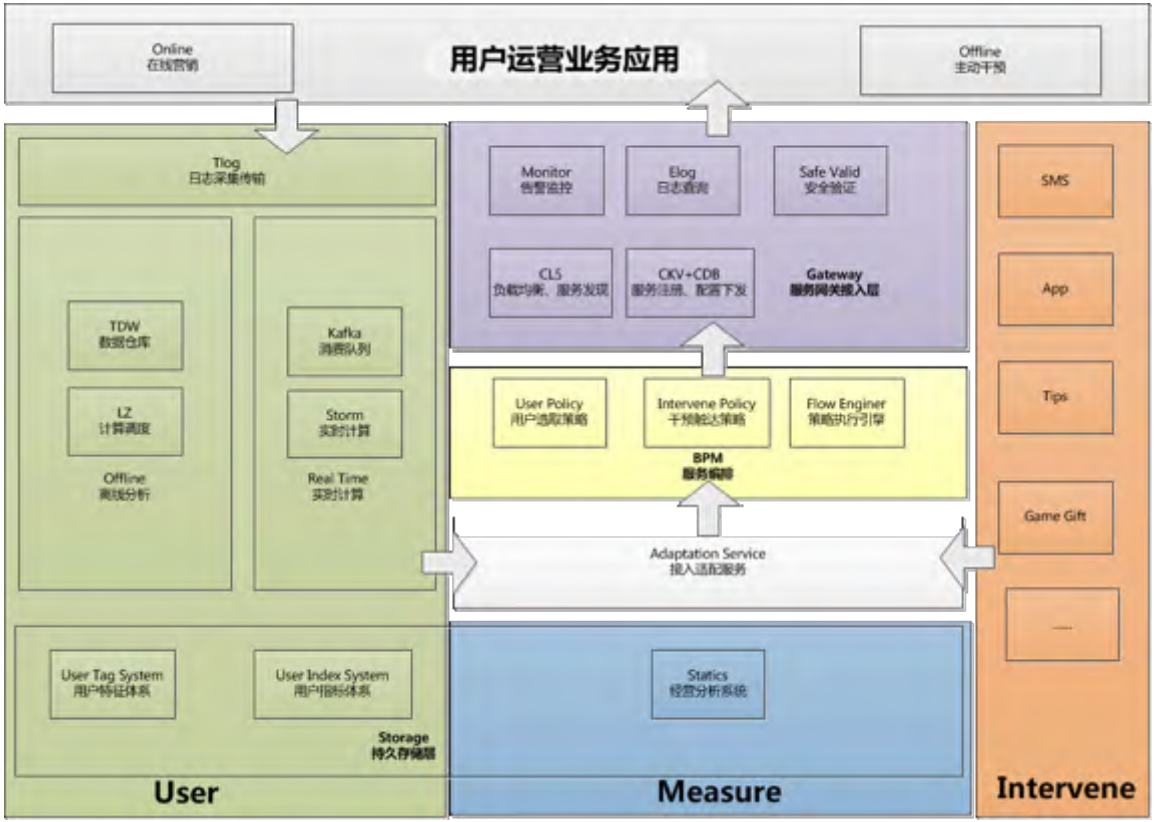
关于腾讯游戏用户运营

腾讯游戏 Tencent Joy Club
心悦俱乐部
心悦·诚服





技术体系


Online
在线营销
用户运营业务应用
Offline
主动干预
Tlog
日志采集传输
TDW
数据仓库
LZ
计算调度
Offline
离线分析
Kafka
消费队列
Storm
实时计算
Real Time
实时计算
Monitor
告警监控
Elog
日志查询
Safe Valid
安全验证
CLS
负载均衡、服务发现
CKV+CDB
服务注册、配置下发
Gateway
服务网关接入层
User Policy
用户选取策略
Intervene Policy
干预触达策略
Flow Enginer
策略执行引擎
BPM
服务编排
Adaptation Service
接入适配服务
SMS
App
Tips
Game Gift
User Tag System
用户特征体系
User Index System
用户指标体系
Storage
持久存储层
Statics
经营分析系统
User
Measure
Intervene

服务化架构演进

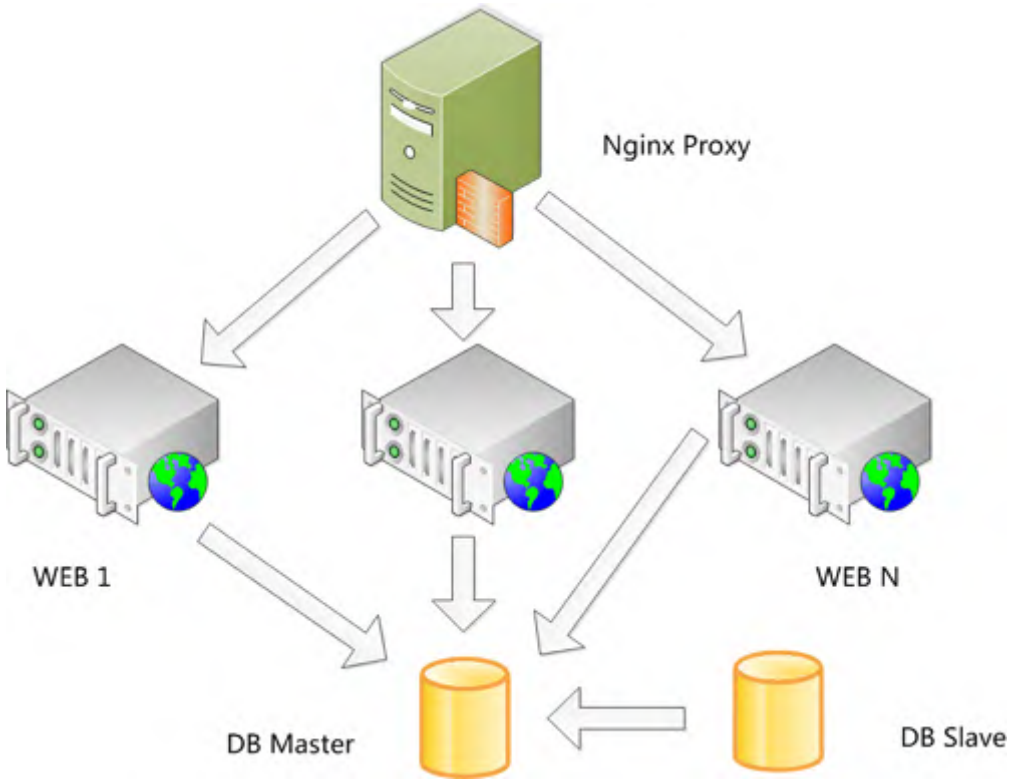
演进历程

经典LNMPA架构

先抗住再优化

业务混合部署

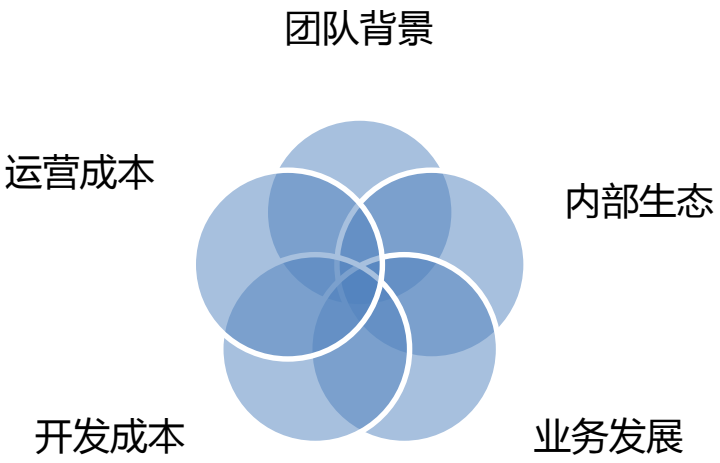
面向需求实现



效率质量之痛

- 流程
- 敏捷
- 隔离
- 性能
- 安全
- 监控

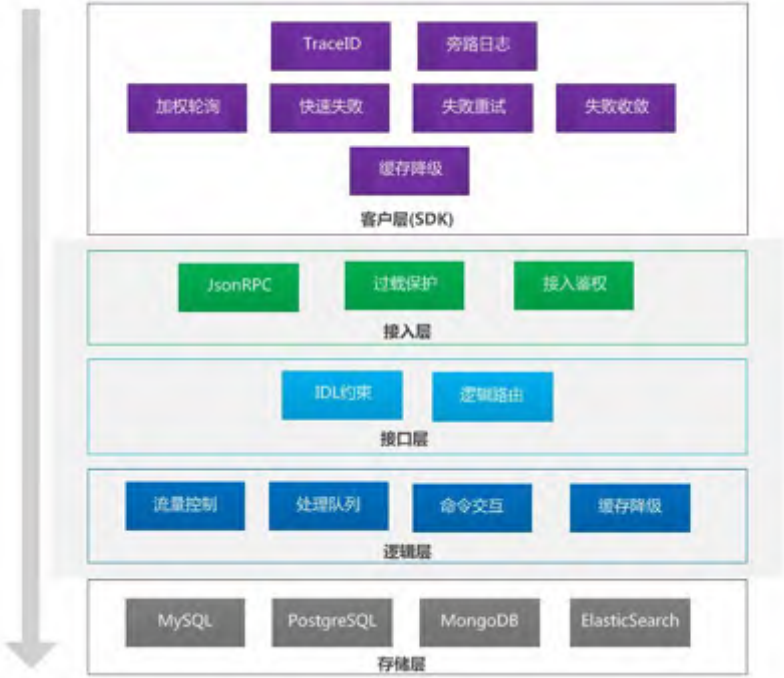
中间件技术选型


团队背景
运营成本
内部生态
开发成本
业务发展

Golang
CSP并发
多核友好
自动GC
语法简洁
开源库多
CGO集成
快速交付
内核稳定
生态趋势

数据中间件——指标服务

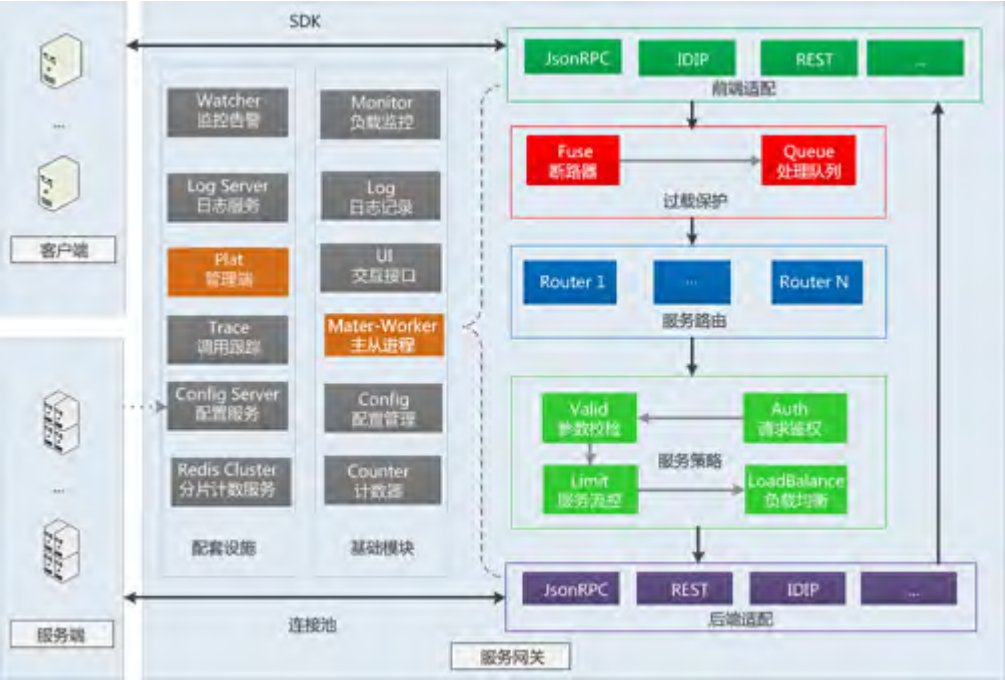
- Golang
- 高性能
- 高可用
- 扩展性
- 安全性
- 可运营性



业务发展驱动技术演进

服务网关

- 过载保护
- 流控降级
- SET部署
- 立体监控
- 敏捷集成

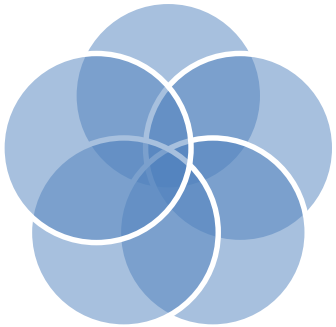


微服务探索

问题

跨部门

业务隔离



异构系统

托管接入

非标接口

手段

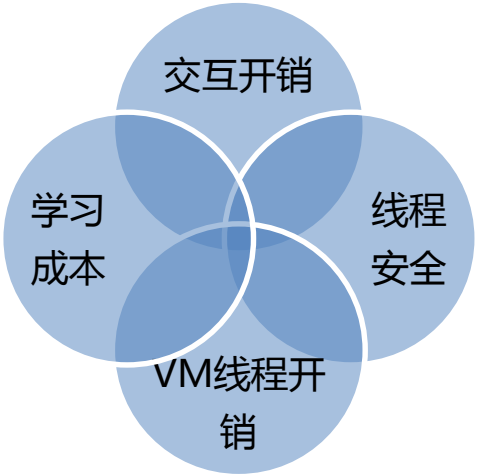
解析适配
IDL

- 标准接口
- 字段约定

Script
转码逻辑

- 非标接口
- 逻辑编码

嵌入脚本选型



Custom	• Expression、Learning cost
Javascript(V8)	• CGO overhead
Javascript (otto)	• Performance
Python/PHP	• CGO / Thread Safe
cLua/cLuaJIT	• CGO / Thread overhead
GopherLua	• Performance / API friendly

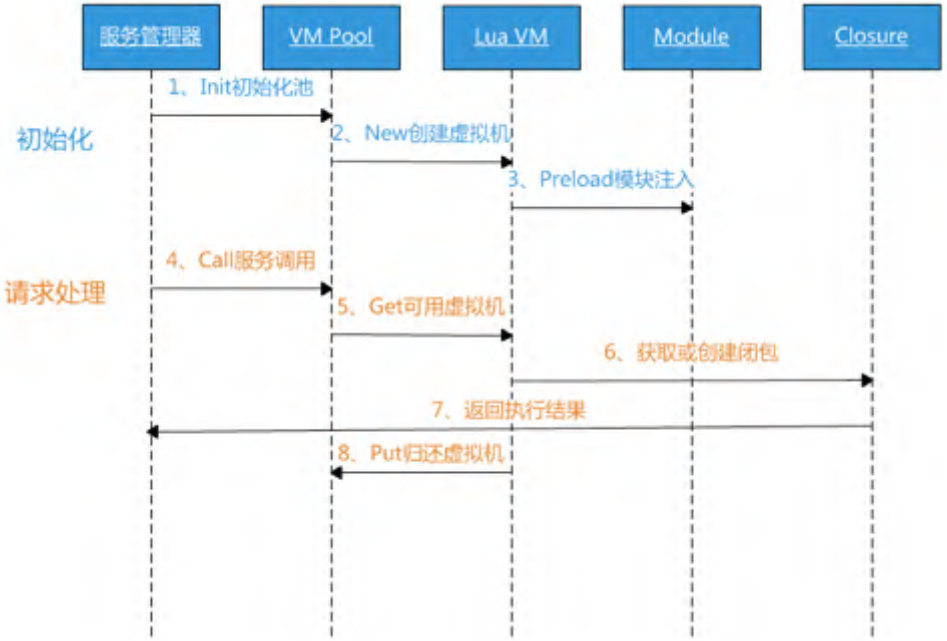
嵌入式Lua虚拟机

Gopherlua

VM Pool

Closure 隔离

即写即用



One VM One Service, Write Once Run Anywhere

服务IDL

key: sum

服务唯一标识

服务标题: 求和-可用性测试

服务标题

服务描述: 计算两数之和

服务描述

服务入参: {"a":{"title":"数字1","type":"int"},"b":{"title":"数字2","type":"int"}} 配置

服务入参

服务返回结果: {"sum":{"title":"求和结果","type":"int"}} 配置

服务返回结果

处理规则

处理器ID: lua

处理器ID

处理器规则:

```
local a = params.a
local b = param.b

local tlib = require "tlib"
local num, err = tlib.sum(a, b)

if err ~= nil then
    return nil, {code = 500, msg = err, data = params}
end
return {sum = num}
```

处理器规则

服务等级协议SLA

服务测试

a: 1

数字1

b: 2

数字2

测试

输出结果

powered by ace


```
1  {
2      "ext": {
3          "client": "[illegible]36287",
4          "request_time": "2017-03-27 16:33:44.792647298",
5          "response_time": "2017-03-27 16:33:44.792781966",
6          "server": "[illegible]:10008"
7      },
8      "id": "20170327163344-58d8ce68b8644",
9      "result": {
10         "sum": 3
11     }
12 }
```


腾讯游戏 Tencent Games

微服务划分

策略

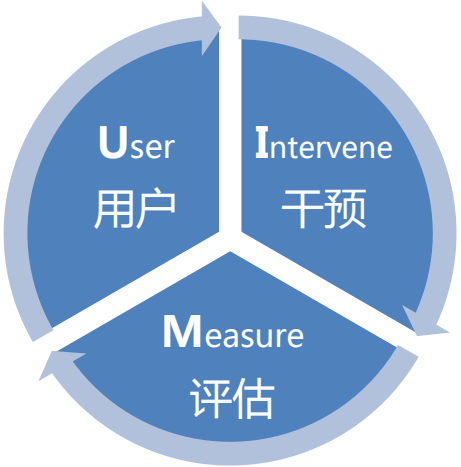
- 服务编排、运营策略

特征

- 离线标签类画像，批量选取用户群

指标

- 离线及实时用户指标，单用户业务个性化属性数值



触达

- 短信、Tips、邮件、站内信通知

营销

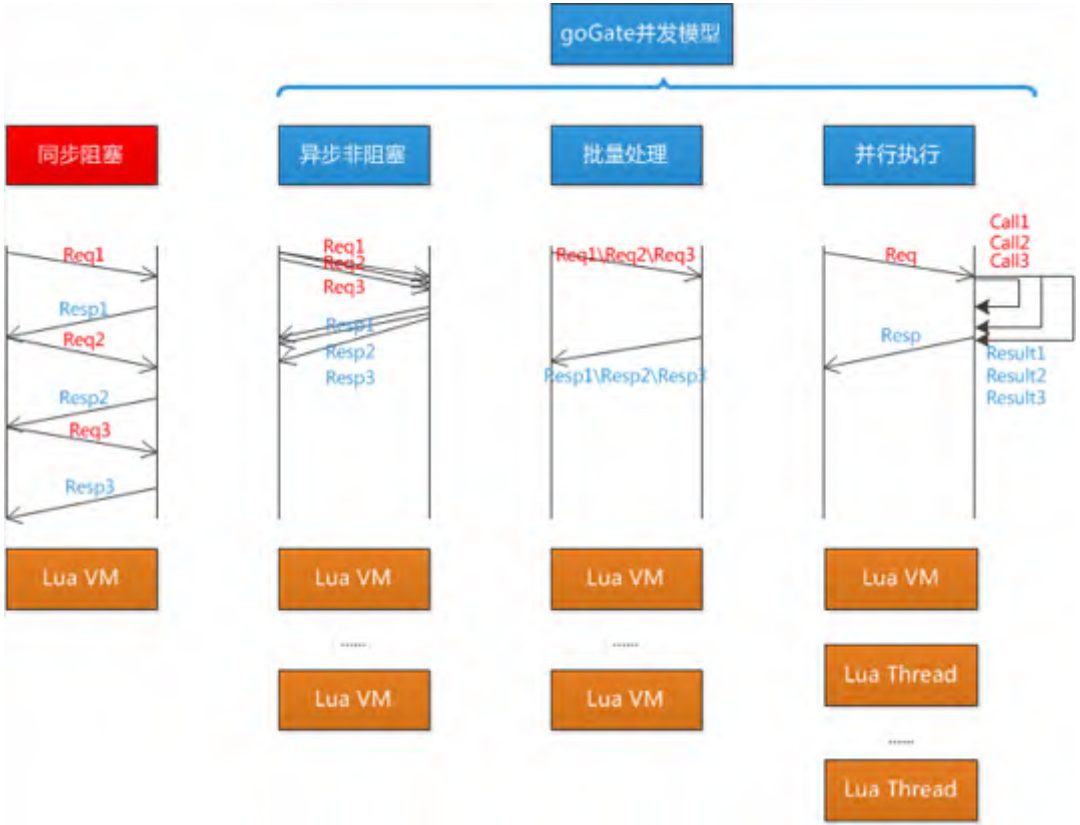
- 礼包发放、积分赠送、体验资格

微服务实践

服务注册发现	• CDB + CKV / ETCD
负载均衡	• CL5 / LVS CAE自动伸缩容
鉴权	• ID / Token / IP
流控	• Atomic + Inmem + Redis、令牌桶
降级	• 轻重分离、单元化部署、容错
日志监控告警	• 实时上报、缓存汇聚/本地文件、ELK
运行时类库	• Bind Golang to Lua

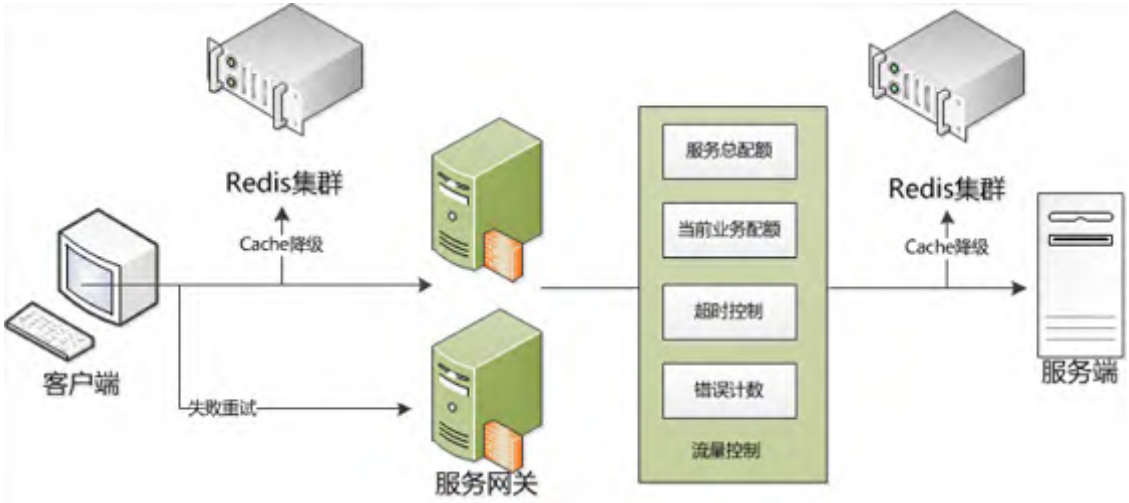
并发模型

- 异步Async
- 批量Batch
- 多核并行Parallel
- Lua协程绑定Go程
- IO阻塞自动切换



高可用

- 负载均衡
- 寻址
- 限流
- 缓存
- 降级
- SLA保证



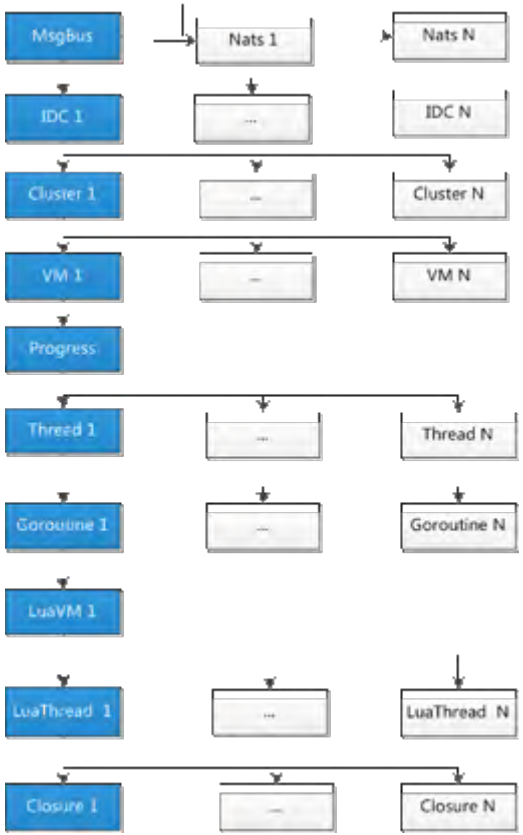
并行执行单元

消息总线

屏蔽本地网络差异

微执行单元

水平伸缩



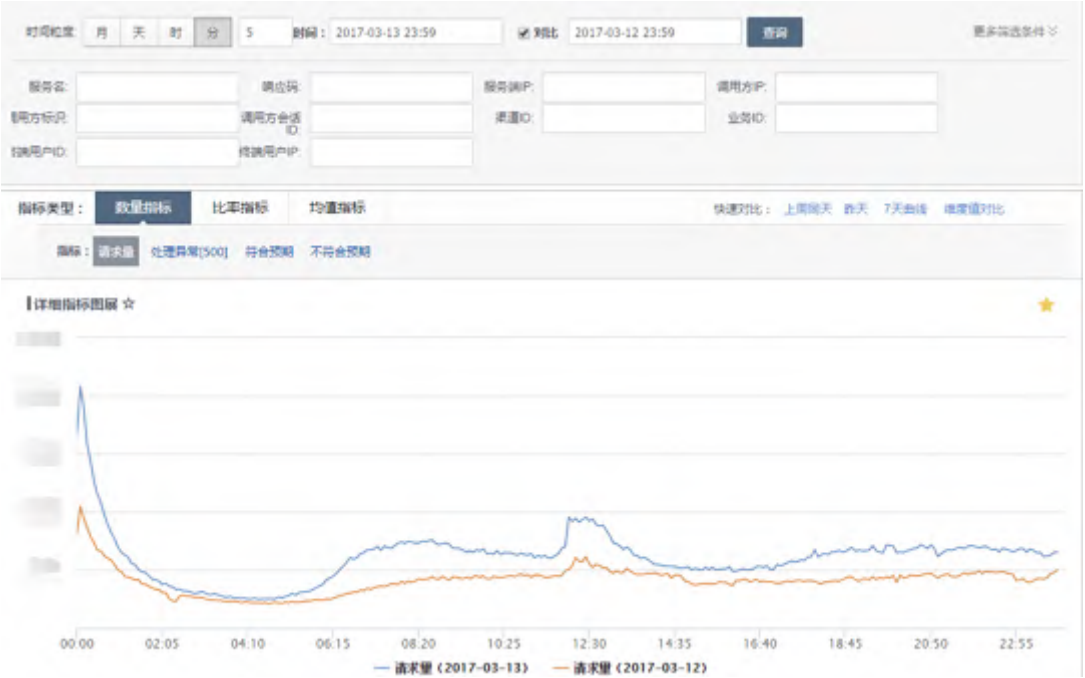
运营监控

旁路实时上报

自定义告警策略

收敛算法

海量日志查询



DSL加速敏捷

微服务编排

Node
API调用

Graph
条件分支

Flow
编排策略

```
local flow = function(user_id, gift_id)
    local ret_1 = rpc_call("get_user_level", user_id)
    if ret_1.level > 1 then
        local ret_2 = rpc_call("send_gift", user_id, gift_id)
        return ret_2.status
    end
end
```

DSL设计

```
flow:set_params({                                   --设置流程入口参数
    user_id = {},
    gift_id = {},
}):set_returns({                                    --设置流程返回值
    status = "{{status}}"
}):set_node("node_1", {                             --设置节点node_1
    args = {user_id = "{{user_id}}"},               --节点入参
    params = {user_id = {}},                        --节点形参
    returns = {level = {}},                         --节点返回值
    bind = {level = "level"},                       --绑定到流程上下文的值
    caller = "get_user_level",                      --服务名
}):set_node("node_2", {
    args = {user_id = "{{user_id}}", gift_id = "{{user_id}}"},
    params = {user_id = {}, gift_id = {}},
    returns = {status = {}},
    bind = {status = "status"},
    caller = "send_gift",
}):set_entry(                                       --设置流程的入口节点
    "node_1"
):set_graph(                                        --设置节点之间的关系图
    "graph_1", {                                    --从节点node_1指向node_2，如果上下文的level字段大于1
        "node_1", "node2", {"gt", "{{level}}", 1},
    }
)
```

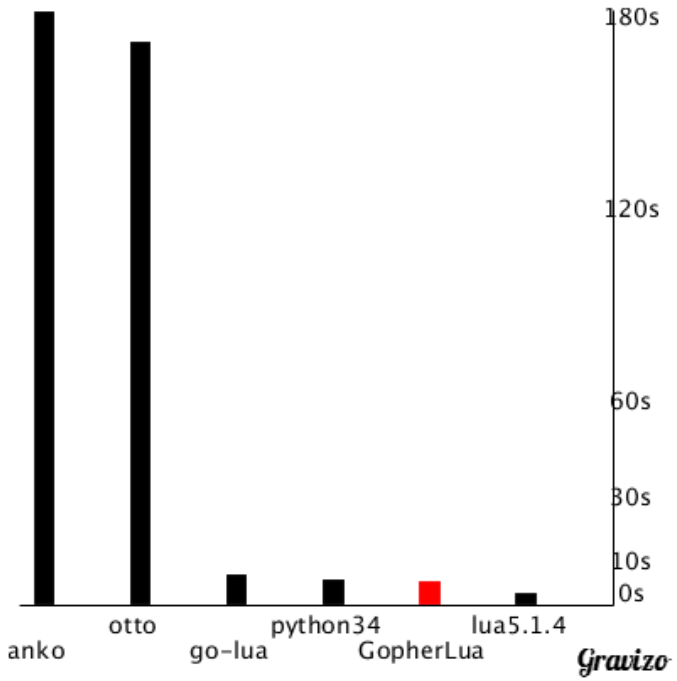
腾讯游戏 Tencent Games

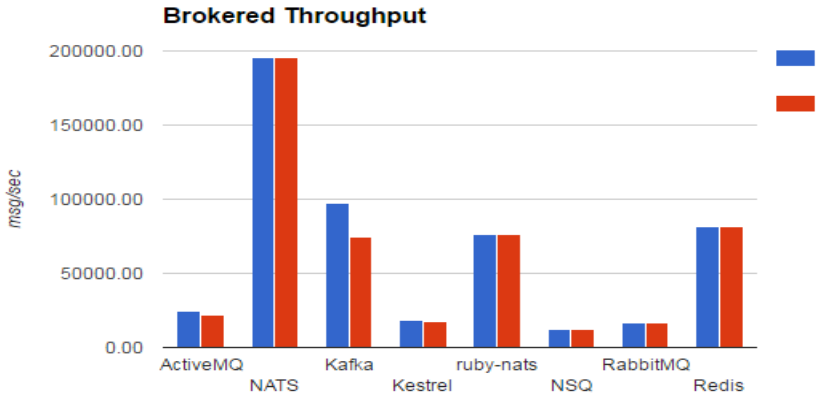
应用落地

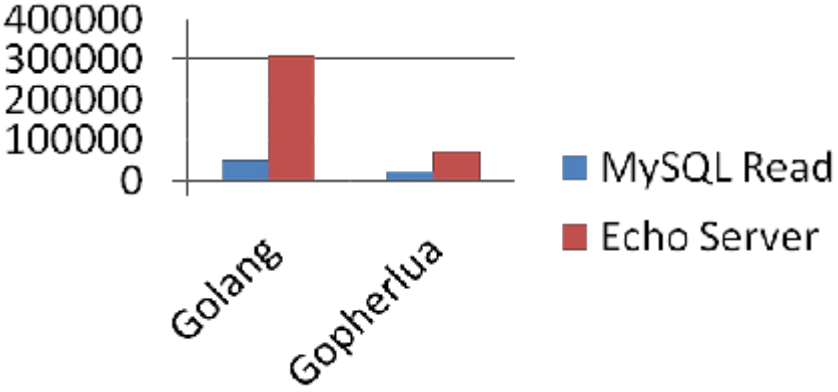
开发外包	• 正职开发核心服务，流程编排外包
运营自助	• 运营填写表单，活动一键上线
流程复用	• 策略嵌套、一键复制
自动测试	• 自动Mock、分支覆盖
模型沉淀	• 流程固化，沉淀运营模型

基准数据

基准数据


180s
120s
60s
30s
10s
0s
anko
otto
go-lua
python34
GopherLua
lua5.1.4
Gravizo


Brokered Throughput
msg/sec
200000.00
150000.00
100000.00
50000.00
0.00
ActiveMQ
NATS
Kafka
Kestrel
ruby-nats
NSQ
RabbitMQ
Redis

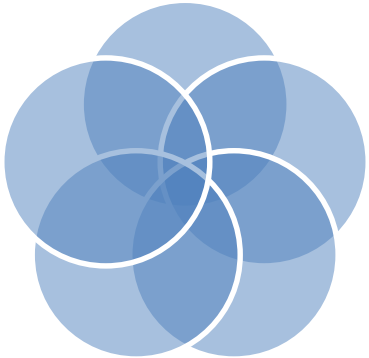

400000
300000
200000
100000
0
Golang
Gopherlua
MySQL Read
Echo Server

总结及展望

总结及展望

服务网关
快速集成

微服务
敏捷迭代



服务编排
流程沉淀

技术驱动
业务优化

执行单元
并发并行

关注QCon微信公众号，
获得更多干货！

Thanks!

INTERNATIONAL SOFTWARE DEVELOPMENT CONFERENCE

主办方 Geekbang. InfoQ