

微博服务化跨语言实践

微博平台研发部 / 周晶

内容提要

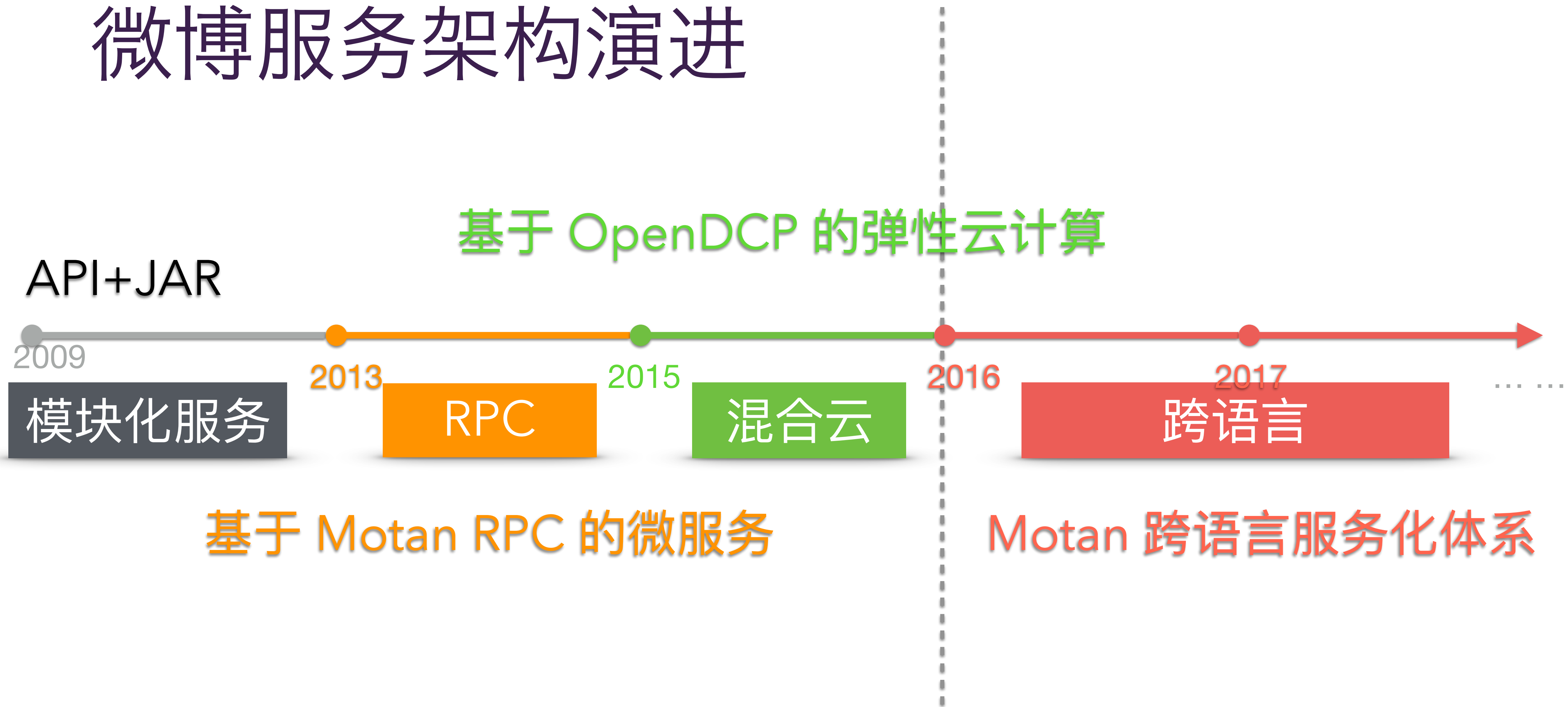
- ▷ 微博平台服务化历程
- ▷ 基于 Motan RPC 框架的微服务
- ▷ 基于 OpenDCP 的混合云微服务解决方案
- ▷ 微博服务化跨语言实践

微博平台服务化历程

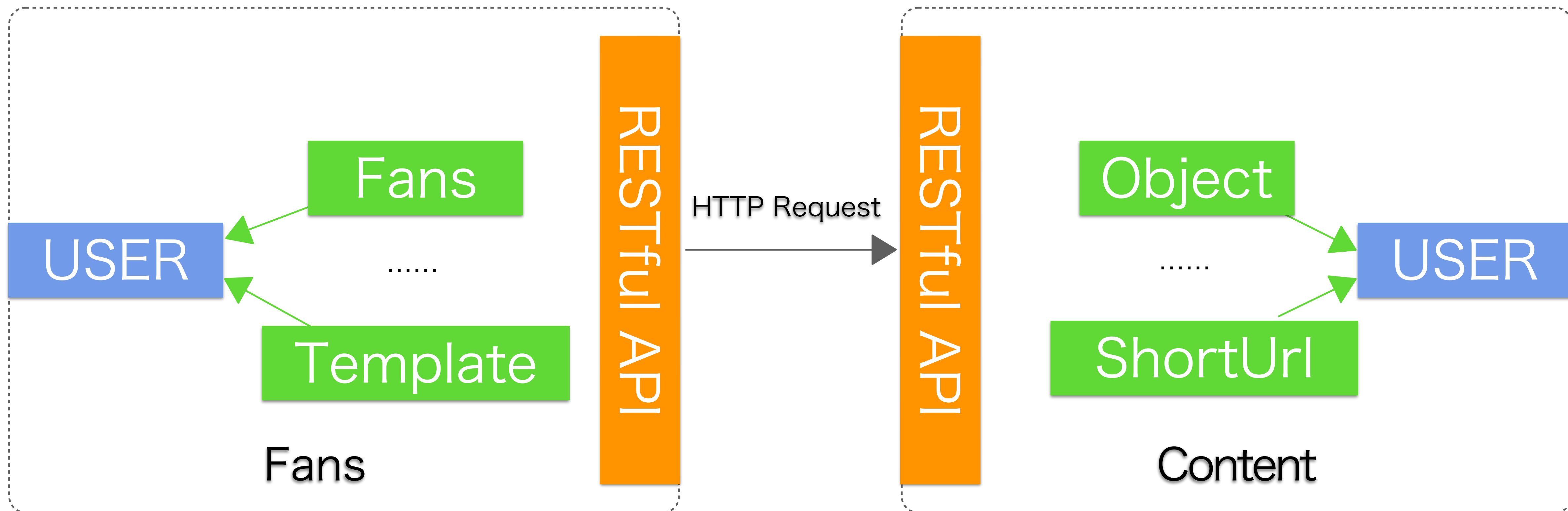
微博 RPC 服务挑战

- ▶ 用户：10亿+、DAU 1.59亿、MAU 3.61亿
- ▶ RPC服务实例： 十万+
- ▶ RPC调用规模： 万亿级 (API 百亿级)
- ▶ 框架SLA 99.999%

微博服务架构演进

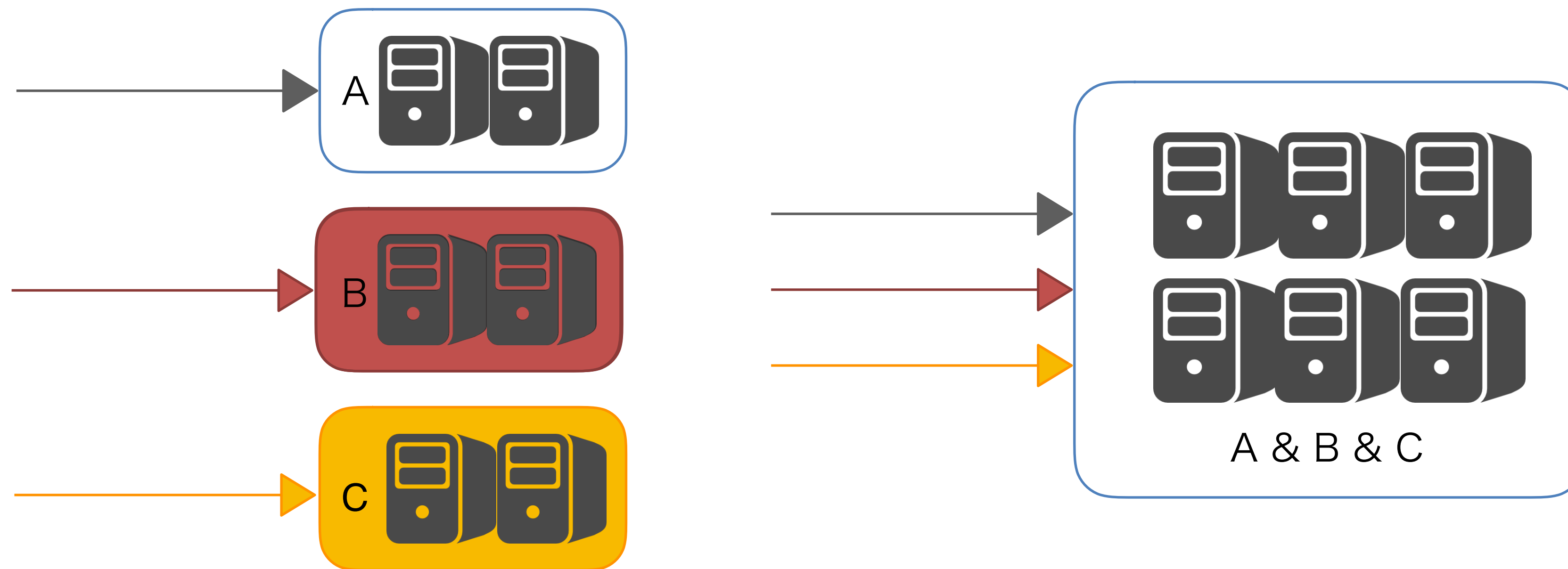


模块化交互



API	Jar
业务初期	高速发展

模块化部署



大服务池

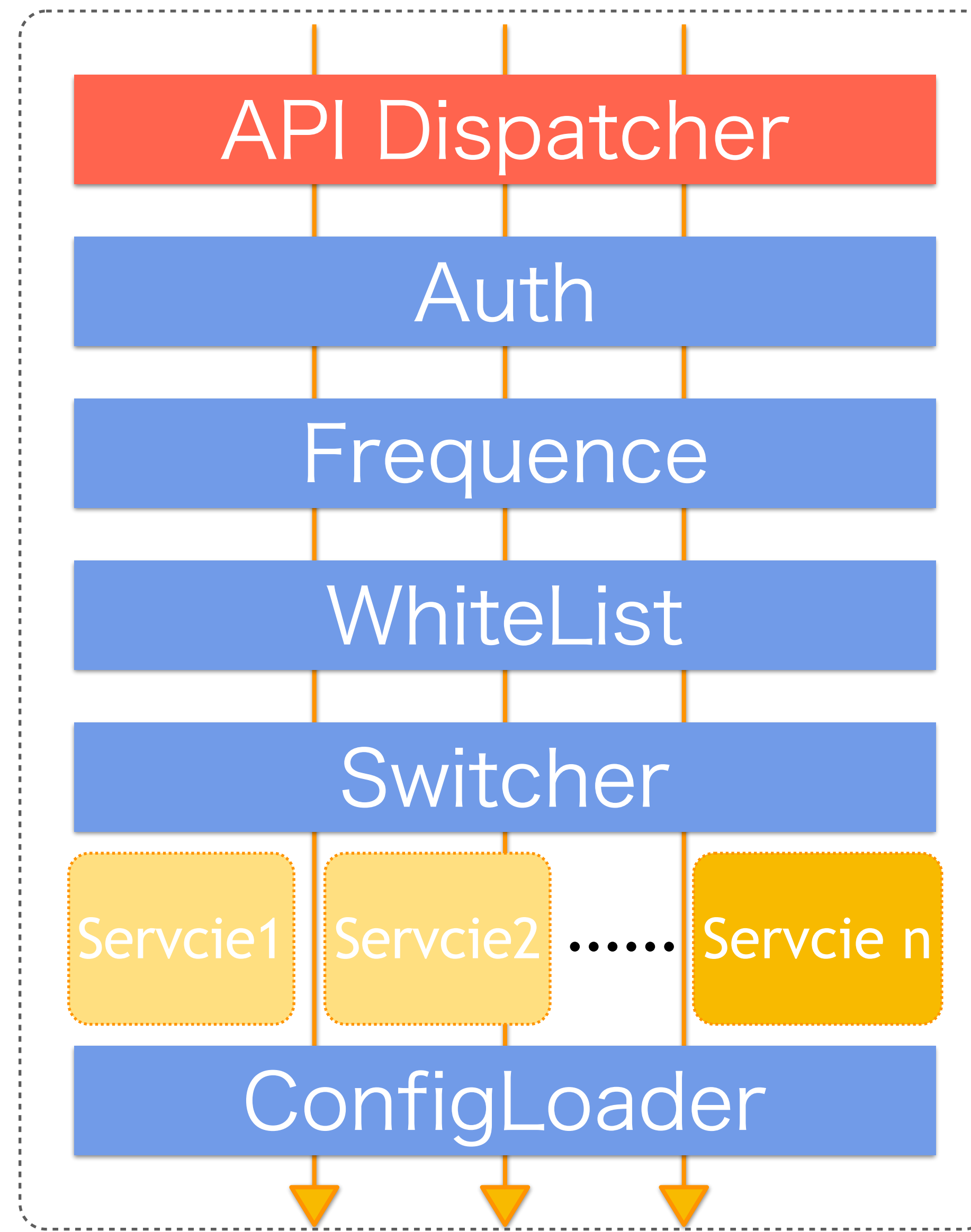
小服务池

蛮荒时代

抱团取暖

模块化请求处理

HttpRequest



Cedrus RESTful Framework

架构缺点

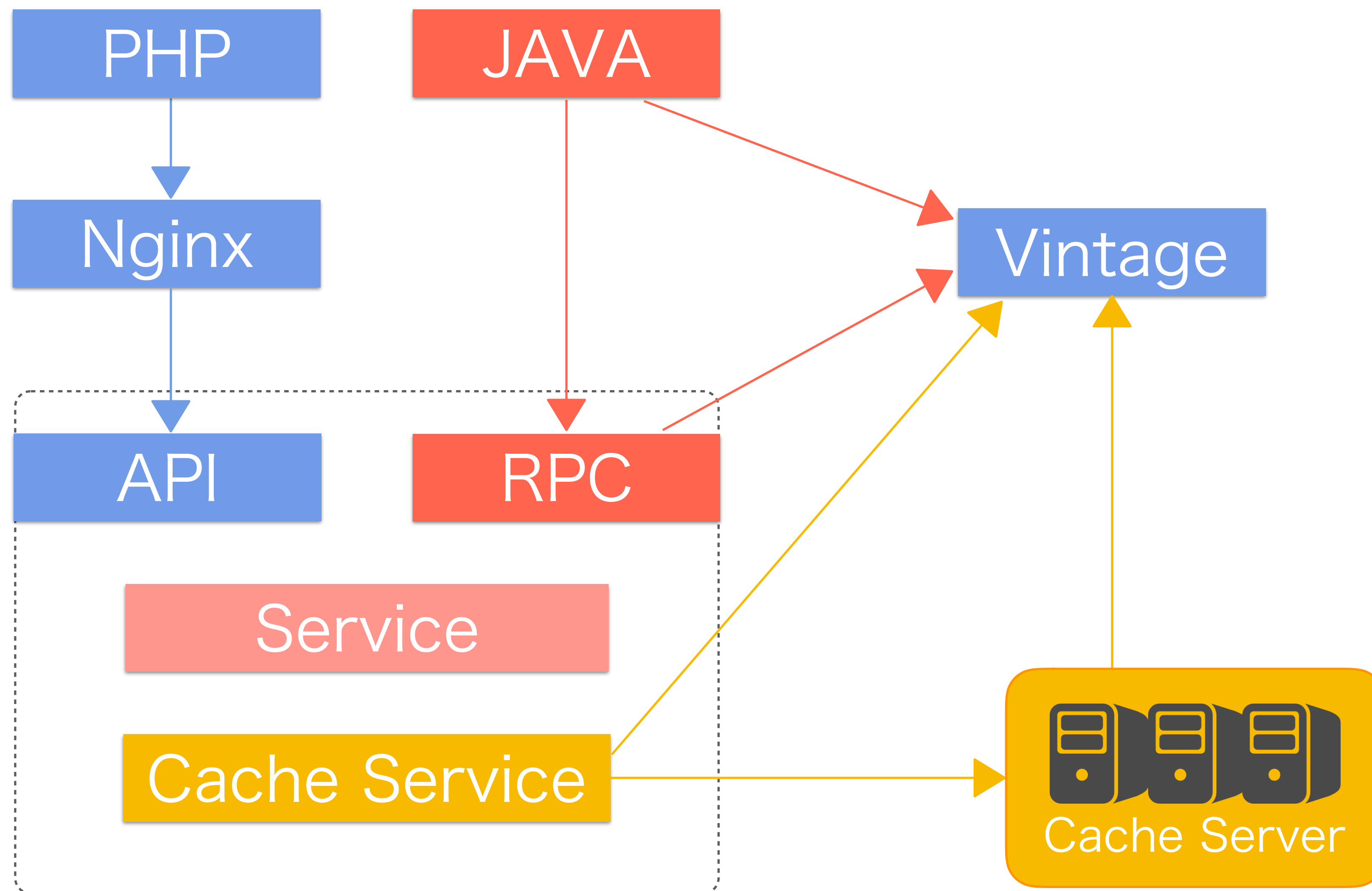
- ▷ 资源升级困难
- ▷ 服务冲突
- ▷ 项目臃肿
- ▷ 数据安全风险
- ▷ 架构调整困难

大一统

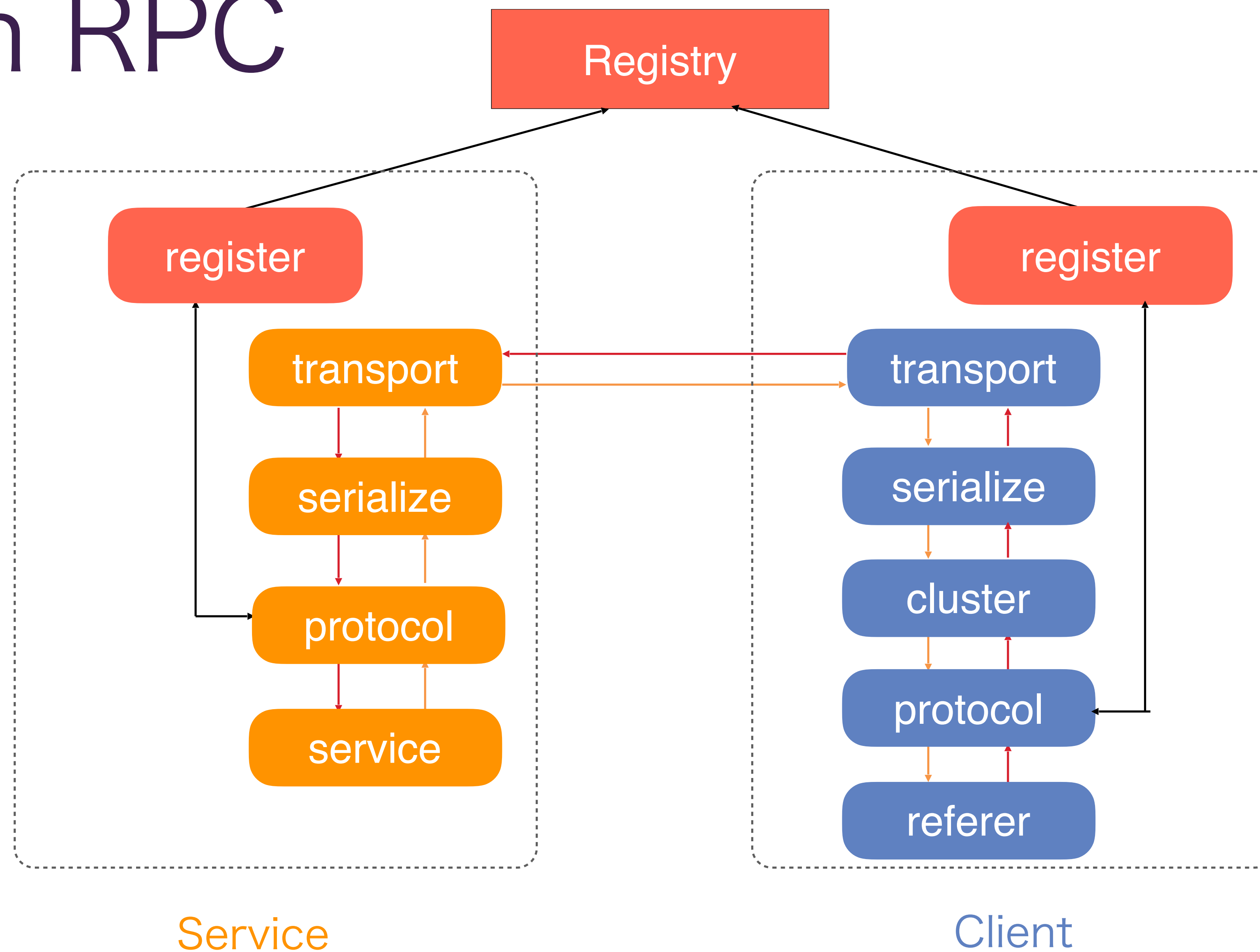
强耦合

基于 Motan RPC 框架的微服务

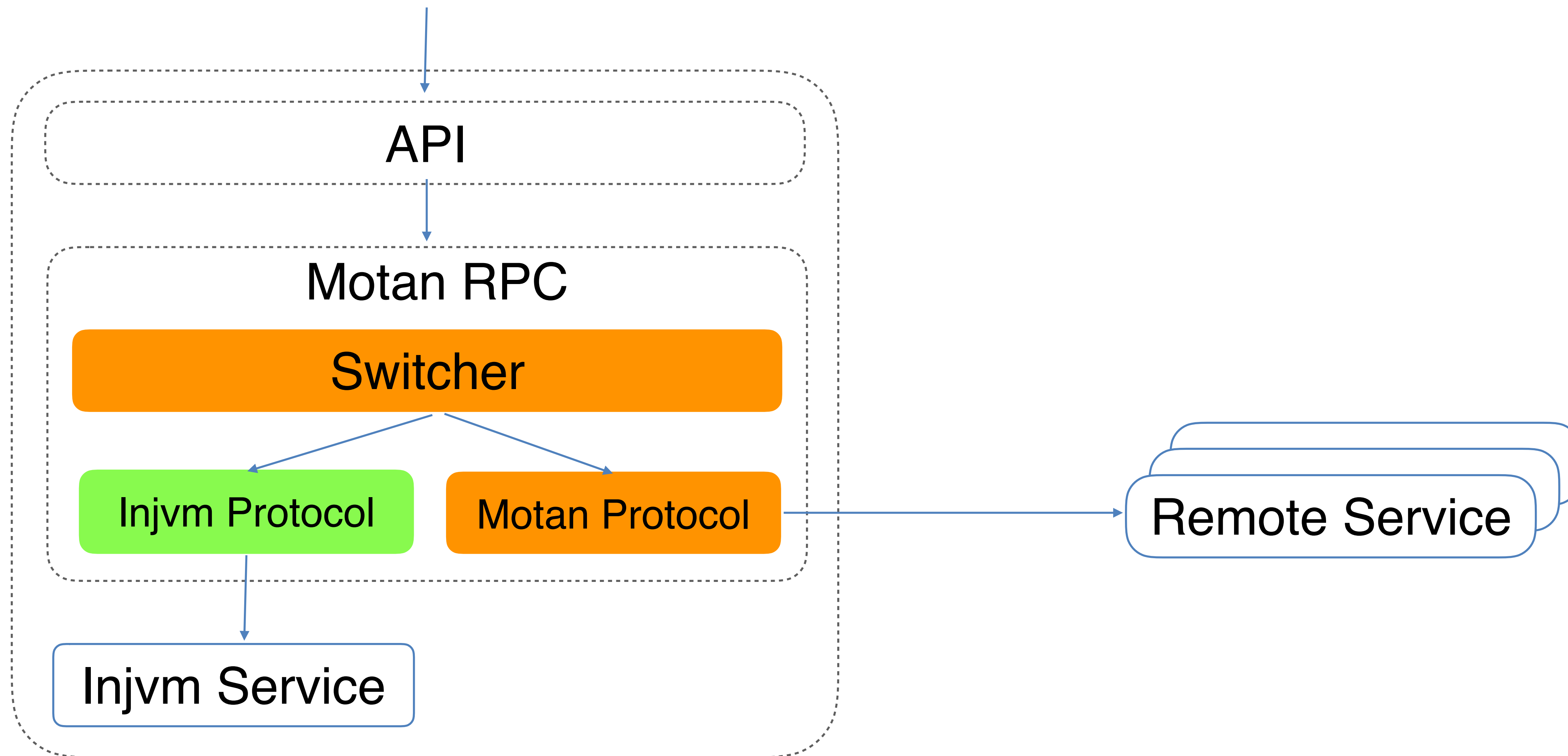
Motan RPC



Motan RPC



从模块化到微服务

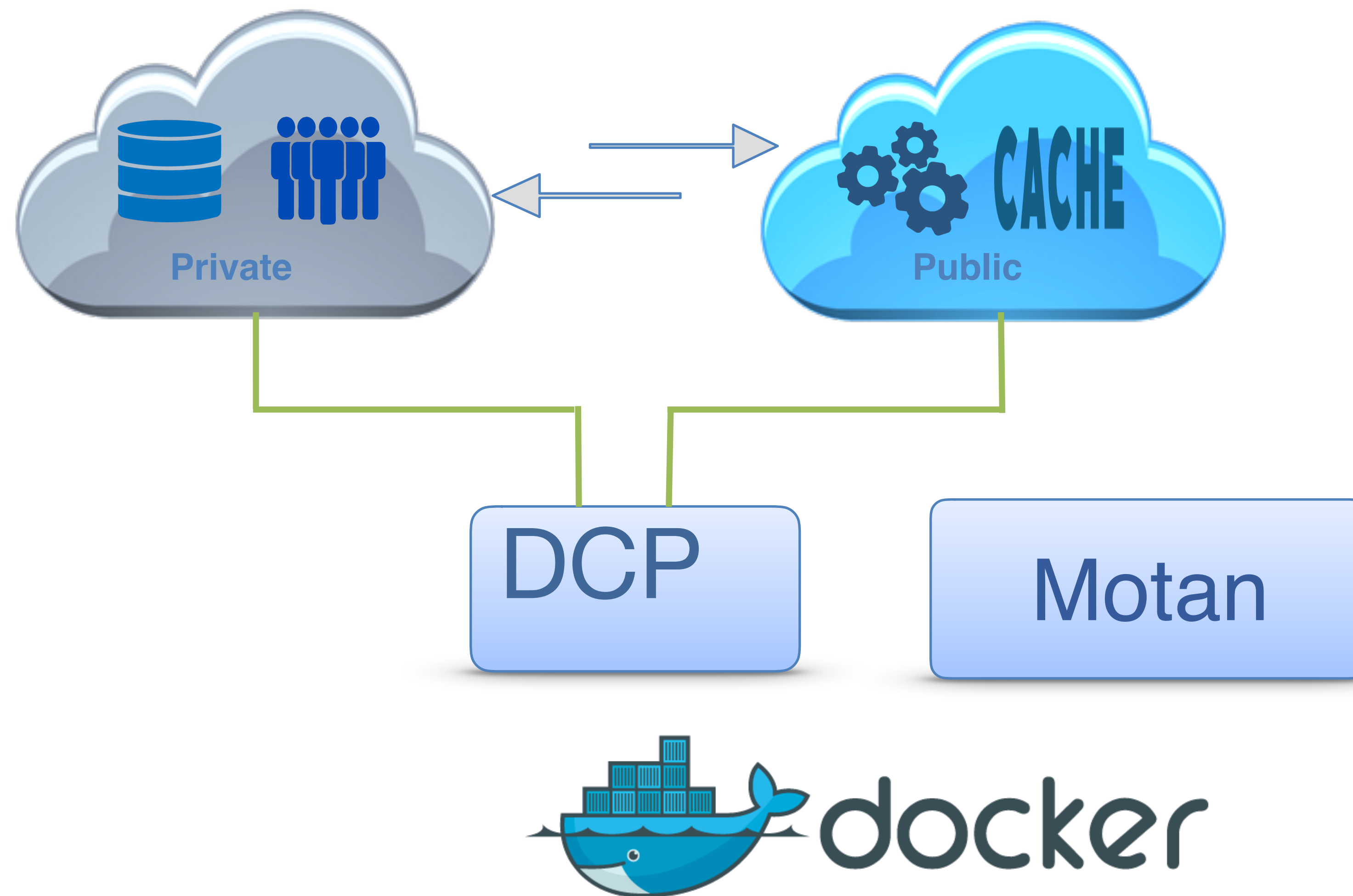


Motan为服务化打下基础

- ▷ 服务解耦
- ▷ 完善的服务治理体系
- ▷ 服务扩缩容慢，机动性不好
- ▷ 服务切分粒度太粗

基于 OpenDCP 的混合云微服务解决方案

基于混合云的架构

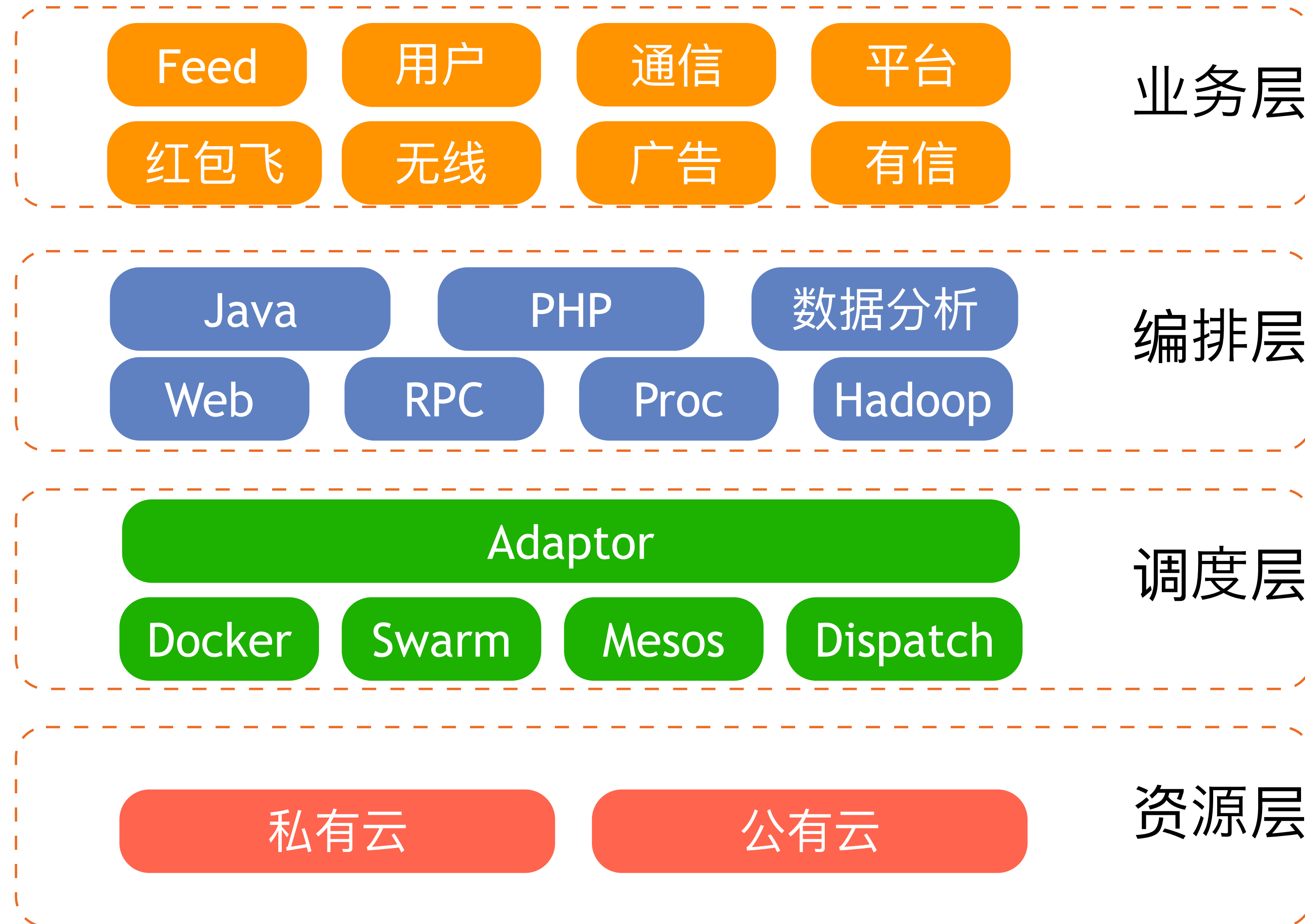


DCP (Docker Container Platform) 混合云平台简介

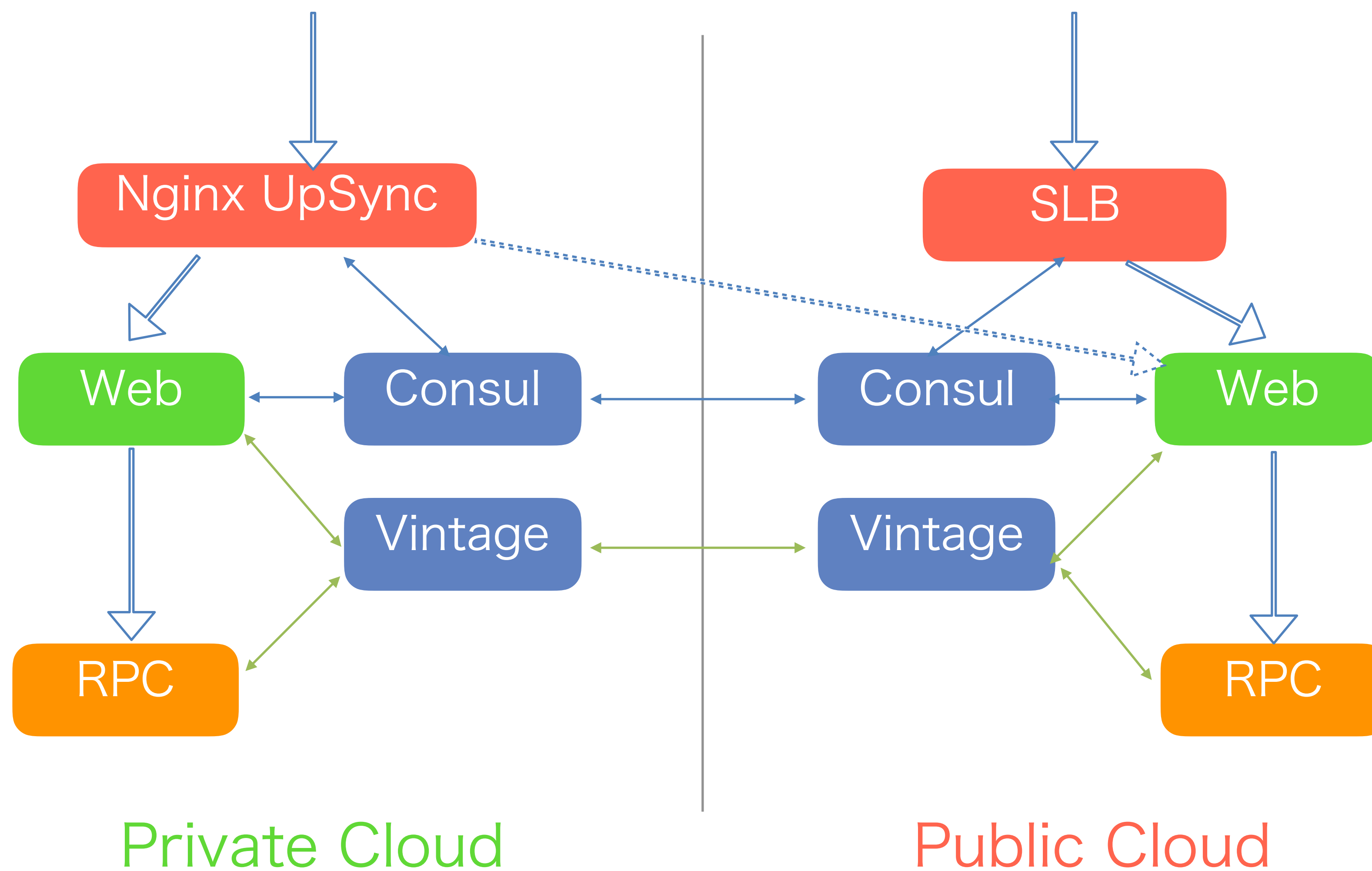
- ▷ 10分钟扩容1000+节点
- ▷ 每天600亿次的API调用
- ▷ 每天万亿次的RPC调用

- ▷ 私有云+公有云的混合云架构
- ▷ 弹性服务管理
- ▷ 服务监控
- ▷ 自动运维

基于混合云的架构

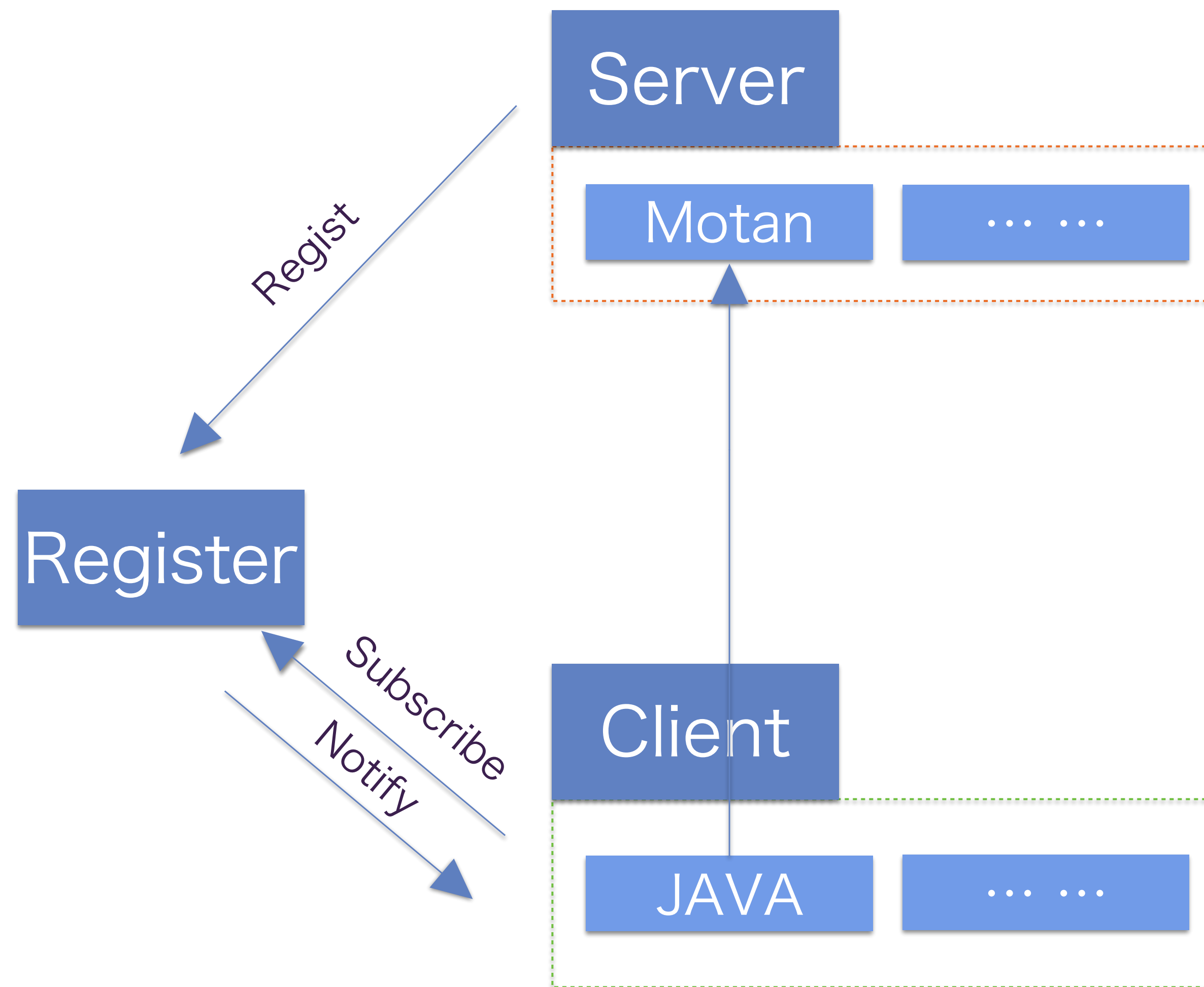


混合云服务交互流程

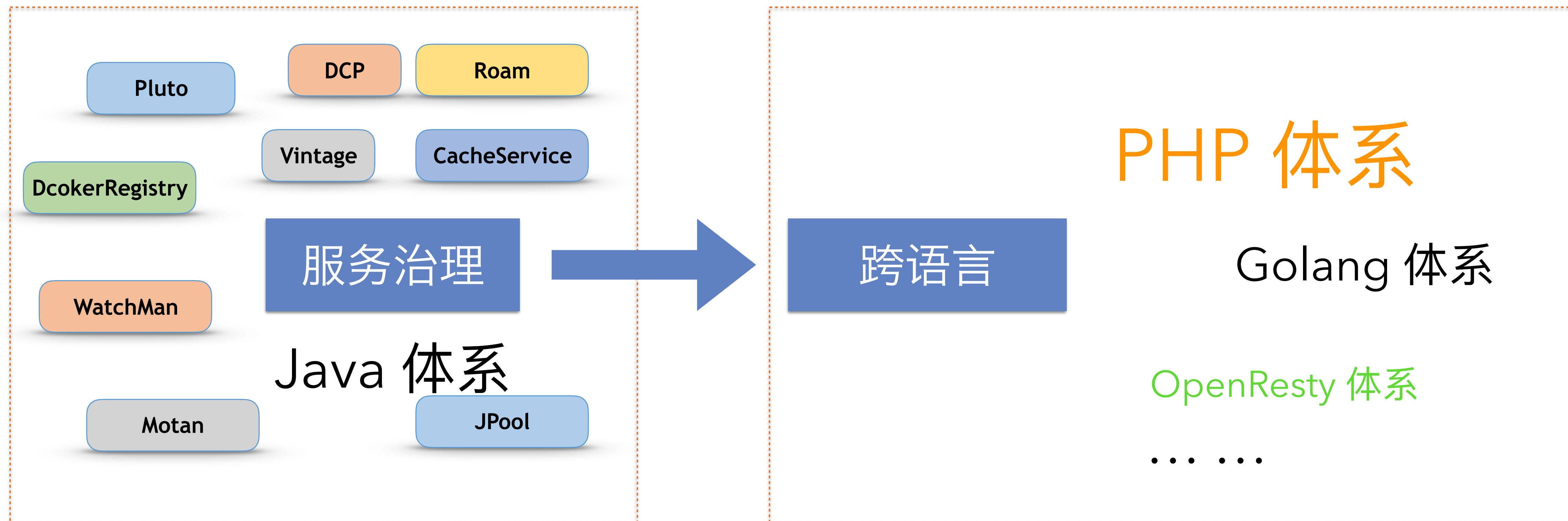


微博服务化跨语言实践

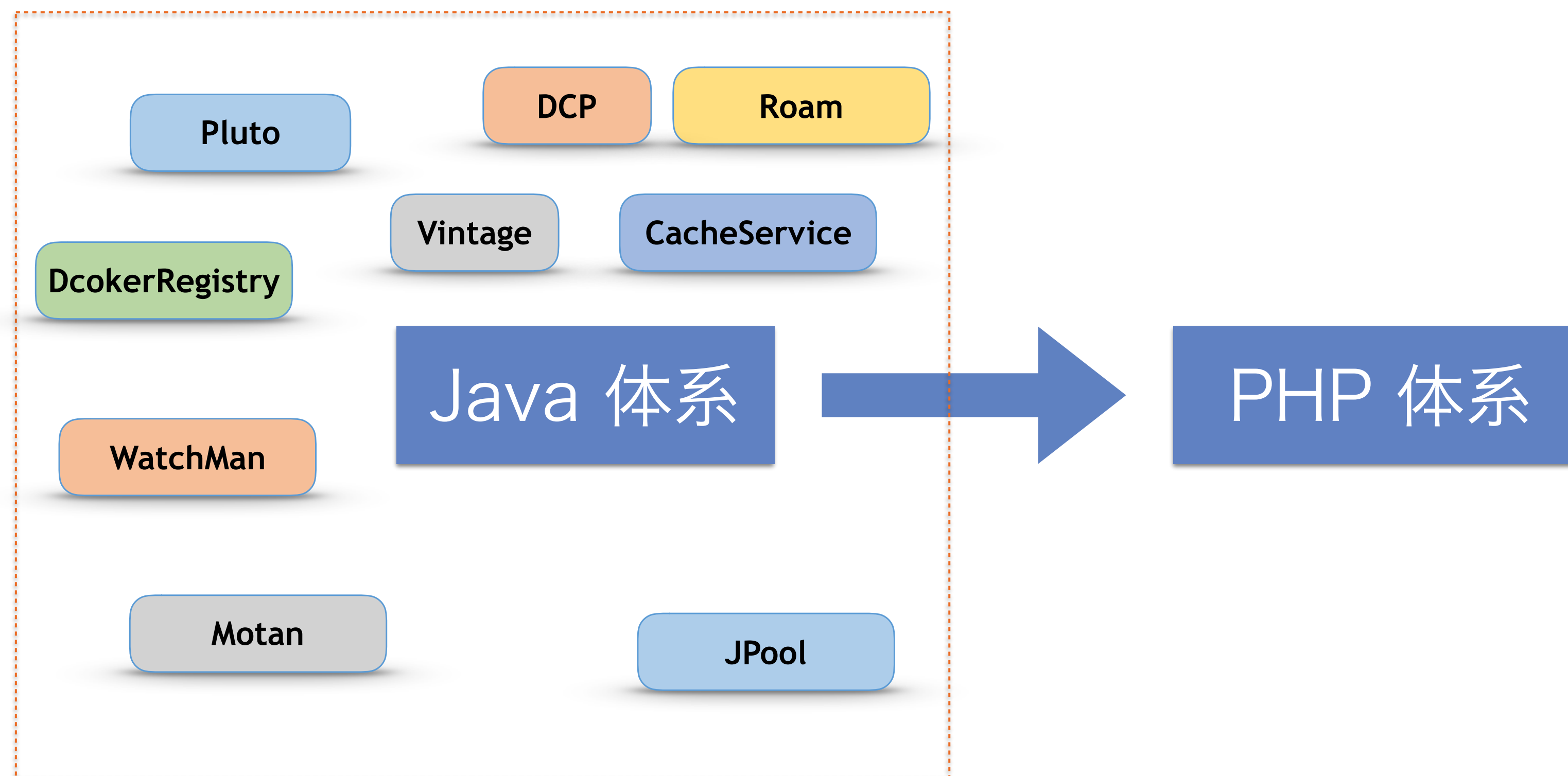
Motan 服务治理型RPC



Motan 沉淀与未来



Motan 跨语言初探



RPC 跨语言要解决的关键问题

传输(Transport)

I/O 模型

进程 / 线程模型

协议结构

Schema 和序列化

可靠性(Reliability)

易用性(Ease of use)

跨语言基础 序列化



巴别塔

跨语言基础 序列化

Json/Msgpack

Yar



PB/PB-Json

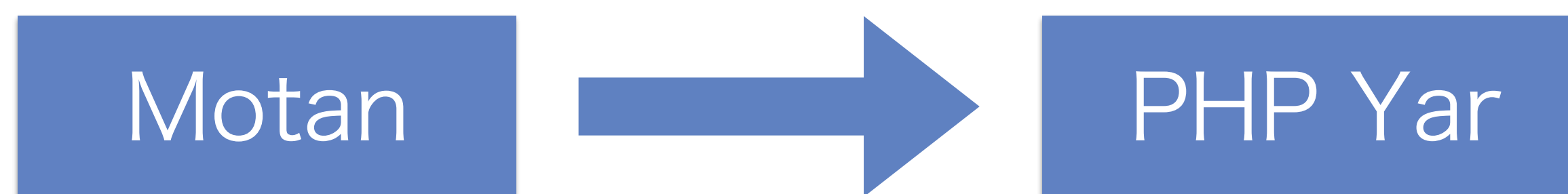
GRPC



Binary-Simple

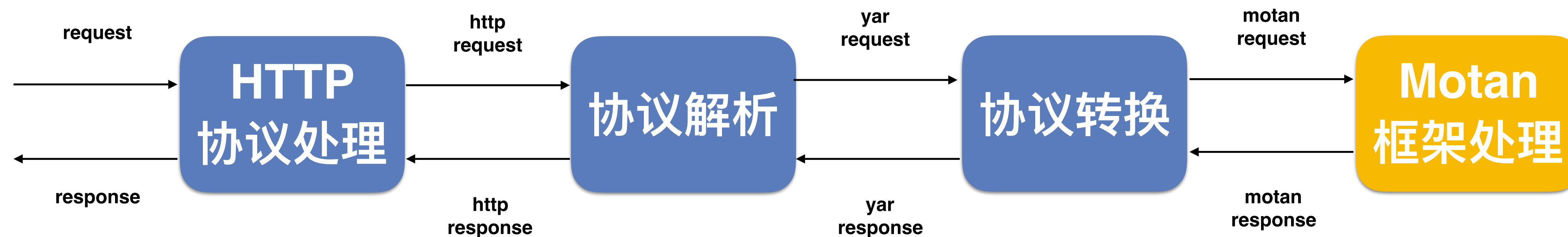
Motan2

Motan 跨语言初探 / PHP Yar

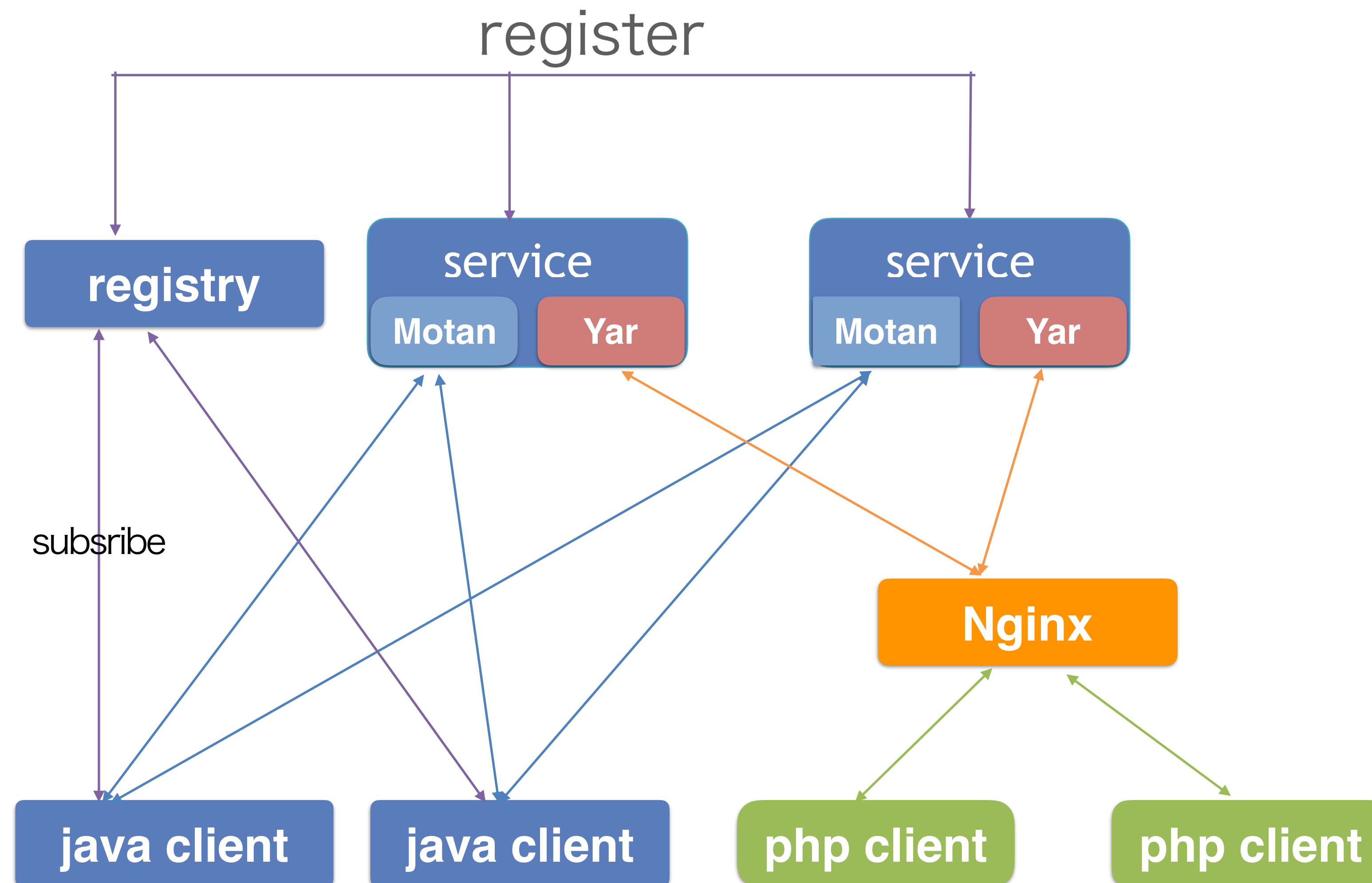


<https://github.com/weibocom/motan/tree/master/motan-extension/protocol-extension/motan-protocol-yar>

Motan 跨语言初探 / PHP Yar



Motan 跨语言初探 / PHP Yar



Motan / PHP Yar 面临的问题

- ▶ 还是 HTTP
- ▶ 只支持 PHP

HTTP 直连 与 四七层转发

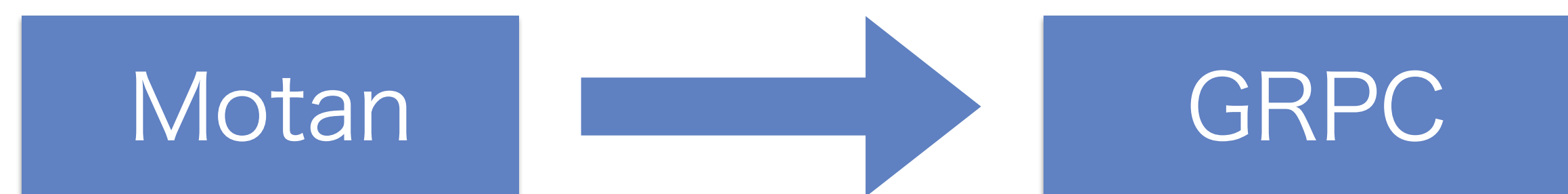
```

└─z at z in /media/psf/g/idevz/code/www/motan-client on masterXXX using
└─± wrk -t12 -c12 -d30s -T30s --latency http://localhost/httpmapi.php
Running 30s test @ http://localhost/httpmapi.php
12 threads and 12 connections
Thread Stats      Avg      Stdev     Max    +/-  Stdev
  Latency    240.21ms  139.69ms  1.41s   87.28%
  Req/Sec     5.38     2.82    10.00   62.94%
Latency Distribution
 50%    193.67ms
 75%    270.25ms
 90%    409.35ms
 99%    808.93ms
1576 requests in 30.10s, 1.04MB read
Requests/sec:    52.36
Transfer/sec:    35.33KB
    
```

```

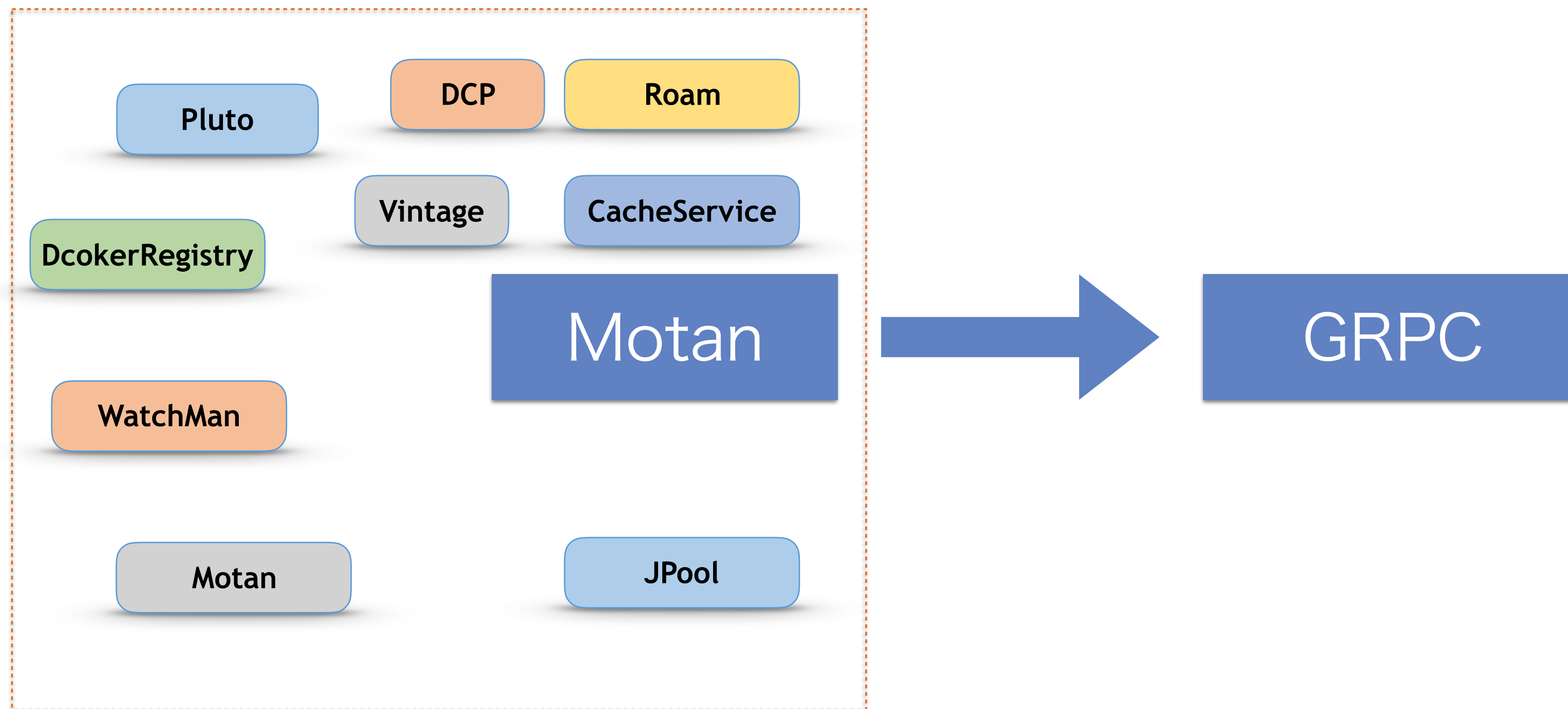
└─z at z in /media/psf/g/idevz/code/www/motan-client on masterXXX using
└─± wrk -t12 -c12 -d30s -T30s --latency http://localhost/httpmapi.php
Running 30s test @ http://localhost/httpmapi.php
12 threads and 12 connections
Thread Stats      Avg      Stdev     Max    +/-  Stdev
  Latency    108.43ms  28.40ms  338.73ms  83.68%
  Req/Sec     9.76     2.56    20.00   85.48%
Latency Distribution
 50%    102.39ms
 75%    117.57ms
 90%    138.07ms
 99%    248.07ms
3342 requests in 30.07s, 2.20MB read
Requests/sec:    111.14
Transfer/sec:    74.99KB
    
```

Motan 跨语言初探 / GRPC

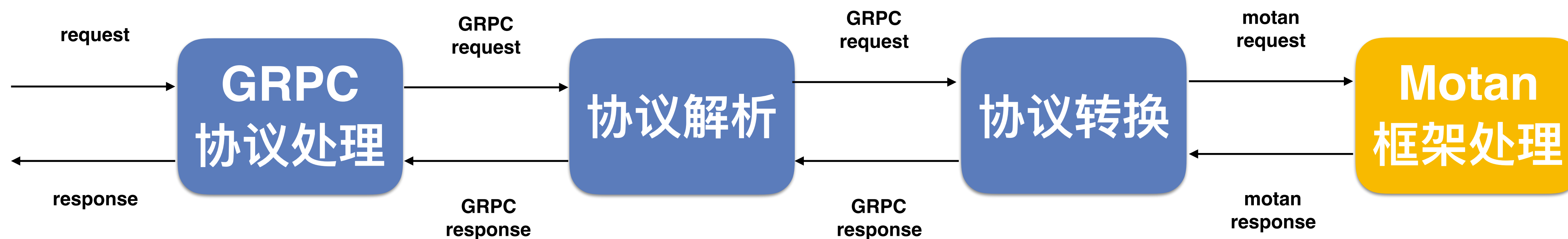


<https://github.com/weibocom/motan/tree/feature/grpc>

Motan 跨语言初探 / GRPC



Motan 跨语言初探 / GRPC



Motan / GRPC 面临的问题

- ▶ 大 PB 的实例创建与反序列化
- ▶ 当时 PHP GRPC 支持并不完善

实例创建

OpCache

发序列化

PB-Json

GRPC 大 PB

```

└─z at z in /media/psf/g/idevz/code/www/motan-client on masterXXX using
└─± wrk -t12 -c12 -d30s -T30s --latency http://localhost/mgmapi.php
Running 30s test @ http://localhost/mgmapi.php
12 threads and 12 connections
Thread Stats      Avg      Stdev     Max    +/-  Stdev
  Latency    16.62ms   4.58ms  93.86ms   91.64%
  Req/Sec    60.72     9.63   80.00    74.17%

Latency Distribution
 50%    15.76ms
 75%    17.79ms
 90%    20.19ms
 99%    34.26ms

21869 requests in 30.07s, 5.55MB read
Requests/sec: 727.29
Transfer/sec: 188.89KB
    
```

```

└─z at z in /media/psf/g/idevz/code/www/motan-client on masterXXX using
└─± wrk -t12 -c12 -d30s -T30s --latency http://localhost/mgmapi.php
Running 30s test @ http://localhost/mgmapi.php
12 threads and 12 connections
Thread Stats      Avg      Stdev     Max    +/-  Stdev
  Latency    43.91ms  10.05ms 154.92ms   82.69%
  Req/Sec    22.84     5.91   30.00    57.27%

Latency Distribution
 50%    41.39ms
 75%    47.29ms
 90%    55.85ms
 99%    81.82ms

8227 requests in 30.06s, 2.09MB read
Requests/sec: 273.64
Transfer/sec: 71.07KB
    
```

PHP PB 性能并没有优势

```

└─z at z in /media/psf/g/idevz/code/www/motan-client on masterXXX using
└─± wrk -t12 -c12 -d30s -T30s --latency http://localhost/mgmapi.php
Running 30s test @ http://localhost/mgmapi.php
12 threads and 12 connections
Thread Stats      Avg      Stdev     Max    +/-  Stdev
  Latency    252.17ms  67.84ms  796.46ms  90.85%
  Req/Sec    4.10     1.24    10.00    93.94%
Latency Distribution
  50%    239.90ms
  75%    272.50ms
  90%    308.57ms
  99%    523.63ms
1435 requests in 30.07s, 0.95MB read
Requests/sec: 47.73
Transfer/sec: 32.20KB
  
```

```

└─z at z in /media/psf/g/idevz/code/www/motan-client on masterXXX usir
└─± wrk -t12 -c12 -d30s -T30s --latency http://localhost/httpmapi.php
Running 30s test @ http://localhost/httpmapi.php
12 threads and 12 connections
Thread Stats      Avg      Stdev     Max    +/-  Stdev
  Latency    240.21ms 139.69ms  1.41s   87.28%
  Req/Sec    5.38     2.82    10.00   62.94%
Latency Distribution
  50%    193.67ms
  75%    270.25ms
  90%    409.35ms
  99%    808.93ms
1576 requests in 30.10s, 1.04MB read
Requests/sec: 52.36
Transfer/sec: 35.33KB
  
```

跨语言基石 Simple 序列化

null

00

string "hello"

01 00 00 00 05 68 65 6c 6c 6f

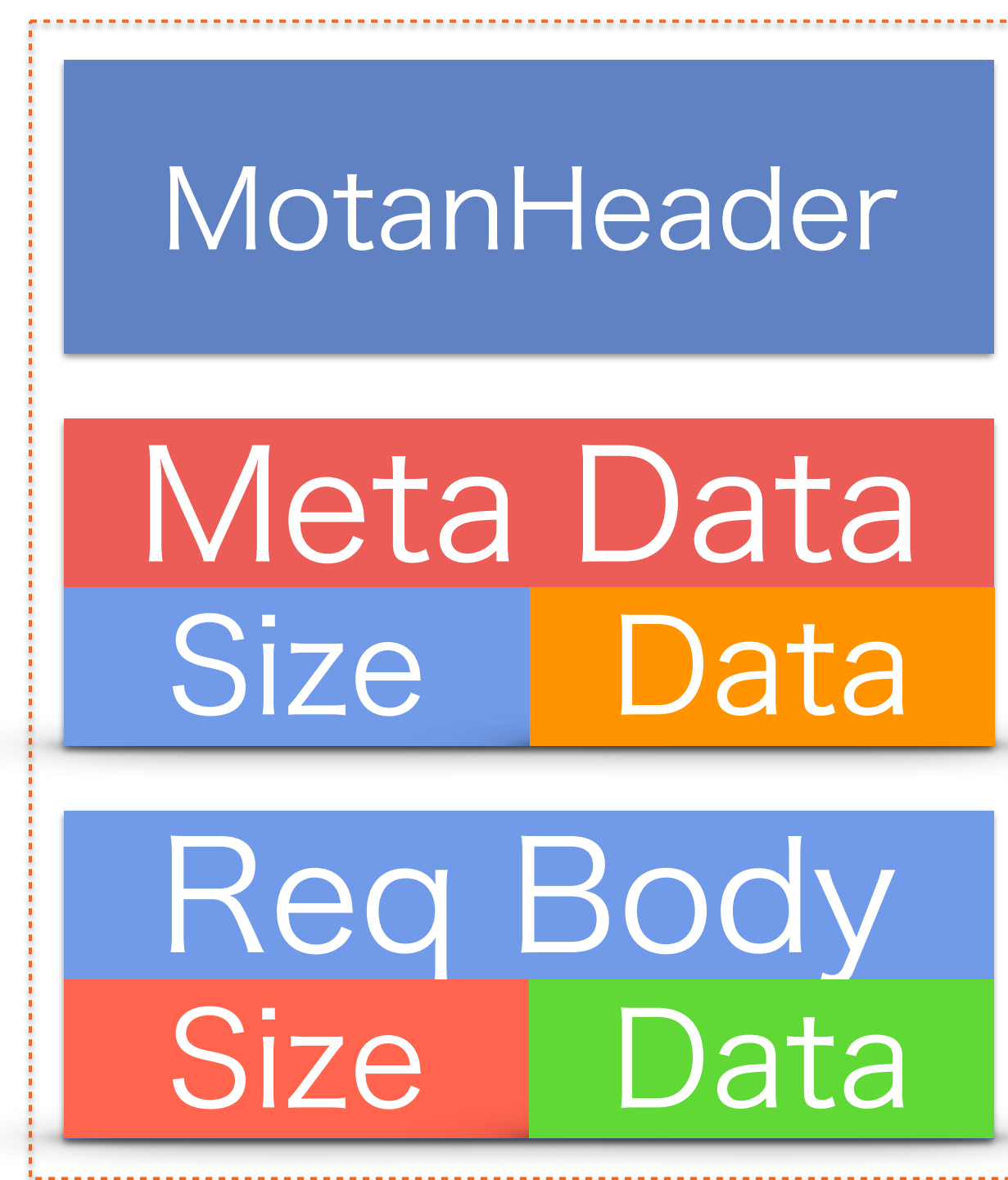
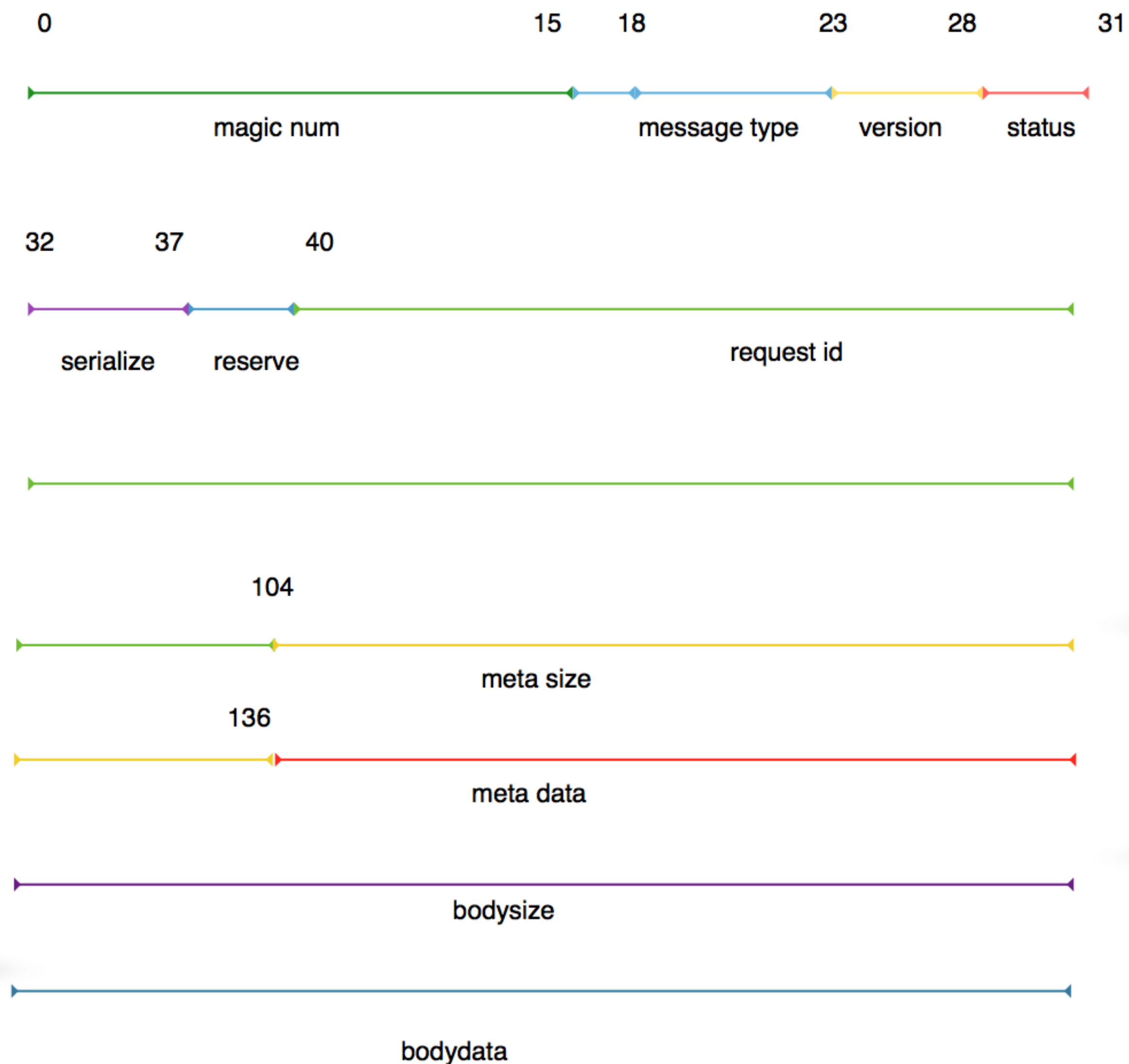
K-V

Map

map {name:ray, code: xxx}

02 00 00 00 1e 00 00 00 04 63 6f 64 65 00 00 00 03 78 78 78 00 00 00 04 6e
61 6d 65 00 00 00 03 72 61 79

Motan2 协议与 Simple 序列化

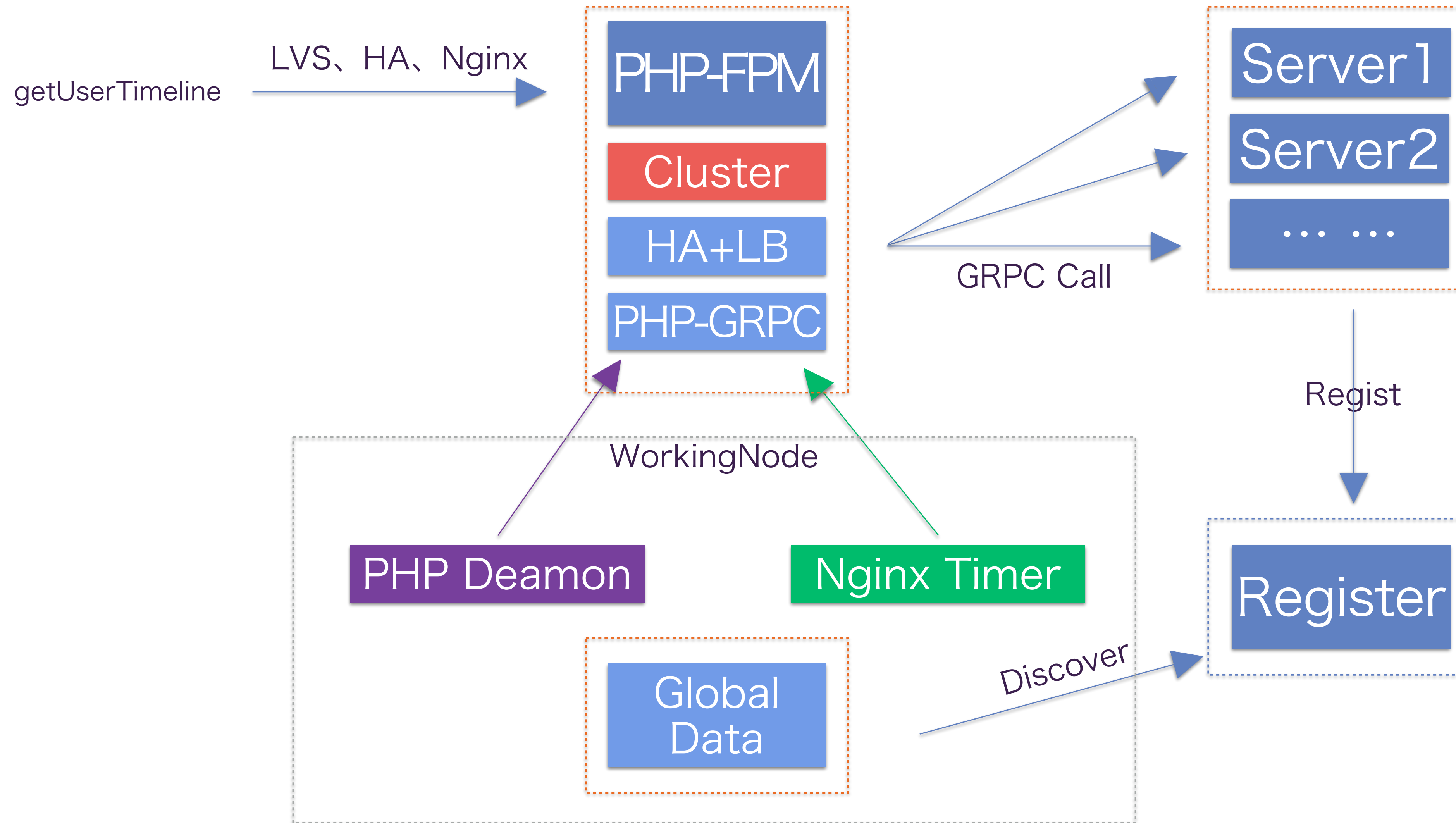


Header Payload

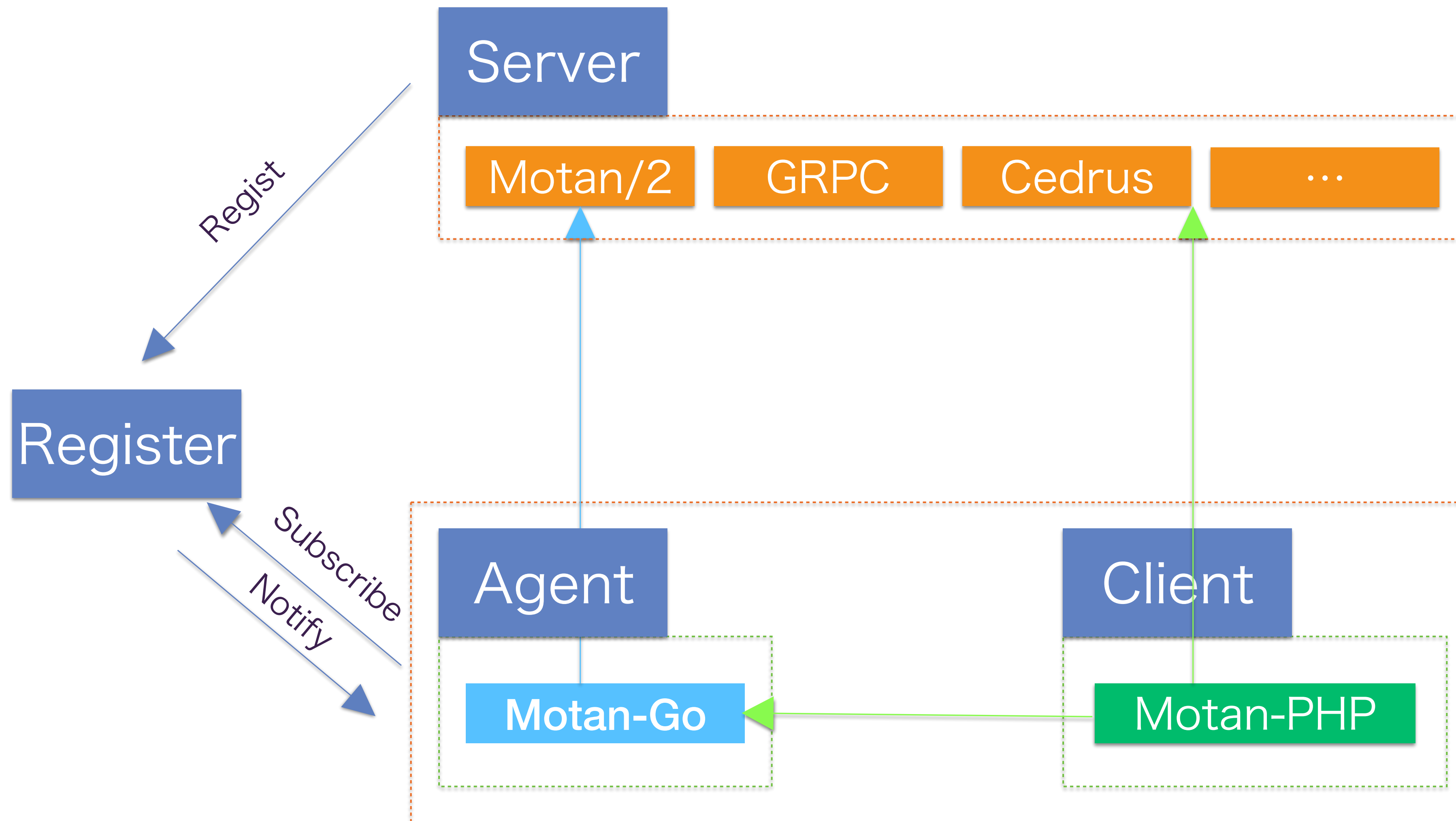
解释性语言的服务治理方案

- ▷ 常驻进程，进程间通信
- ▷ 通过 Client 端代理 / Motan Go Agent

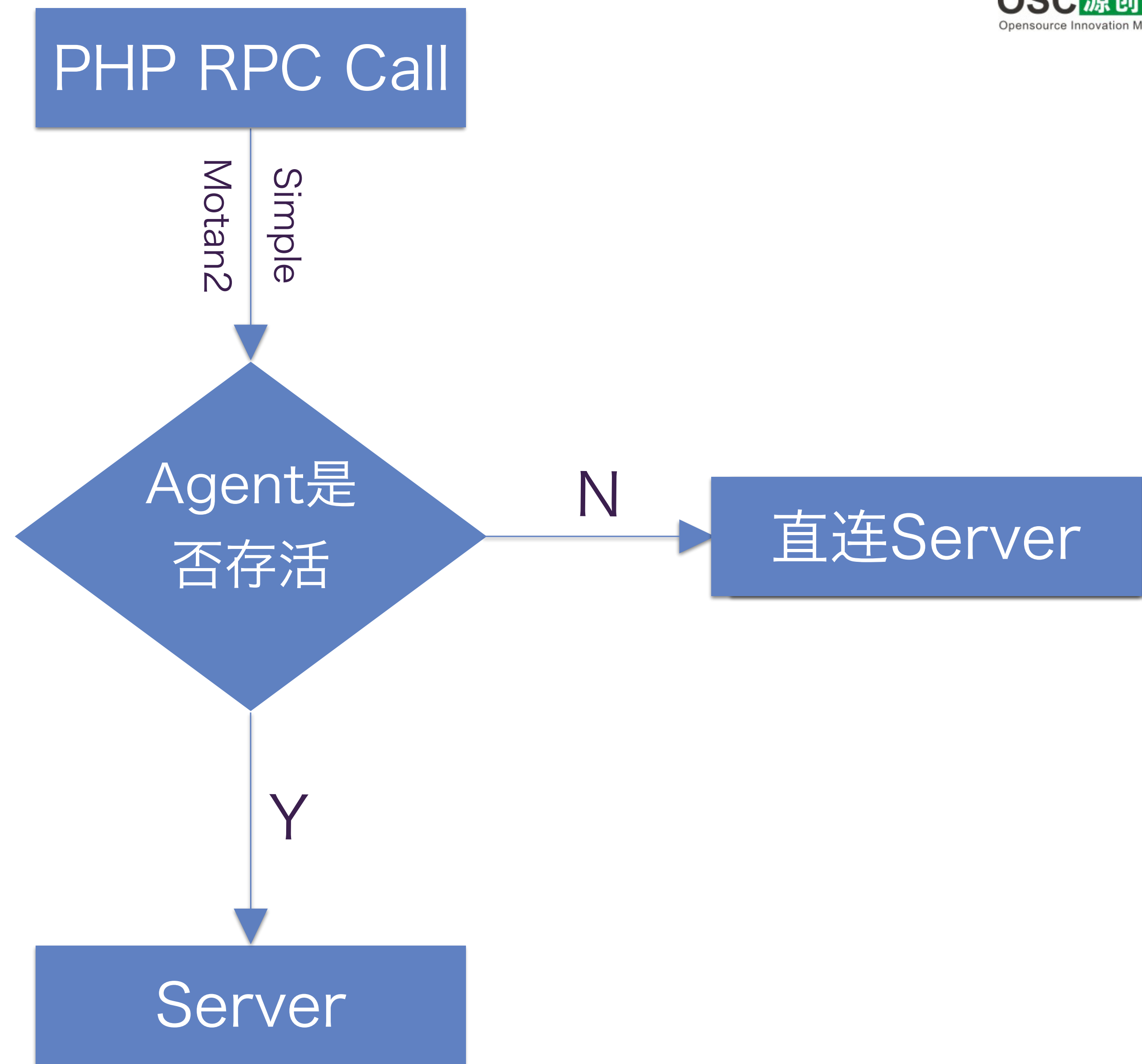
通过常驻进程



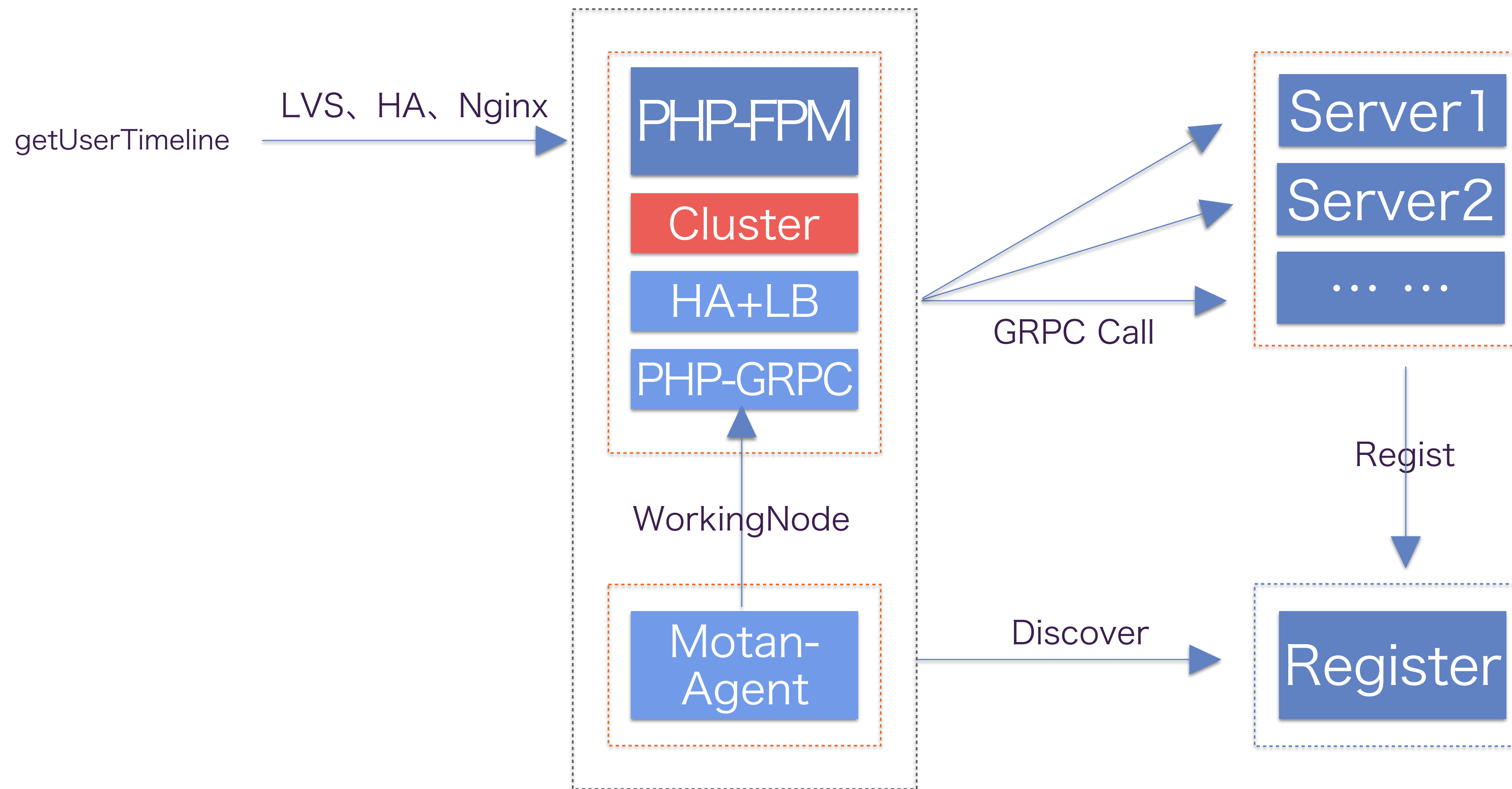
通过代理



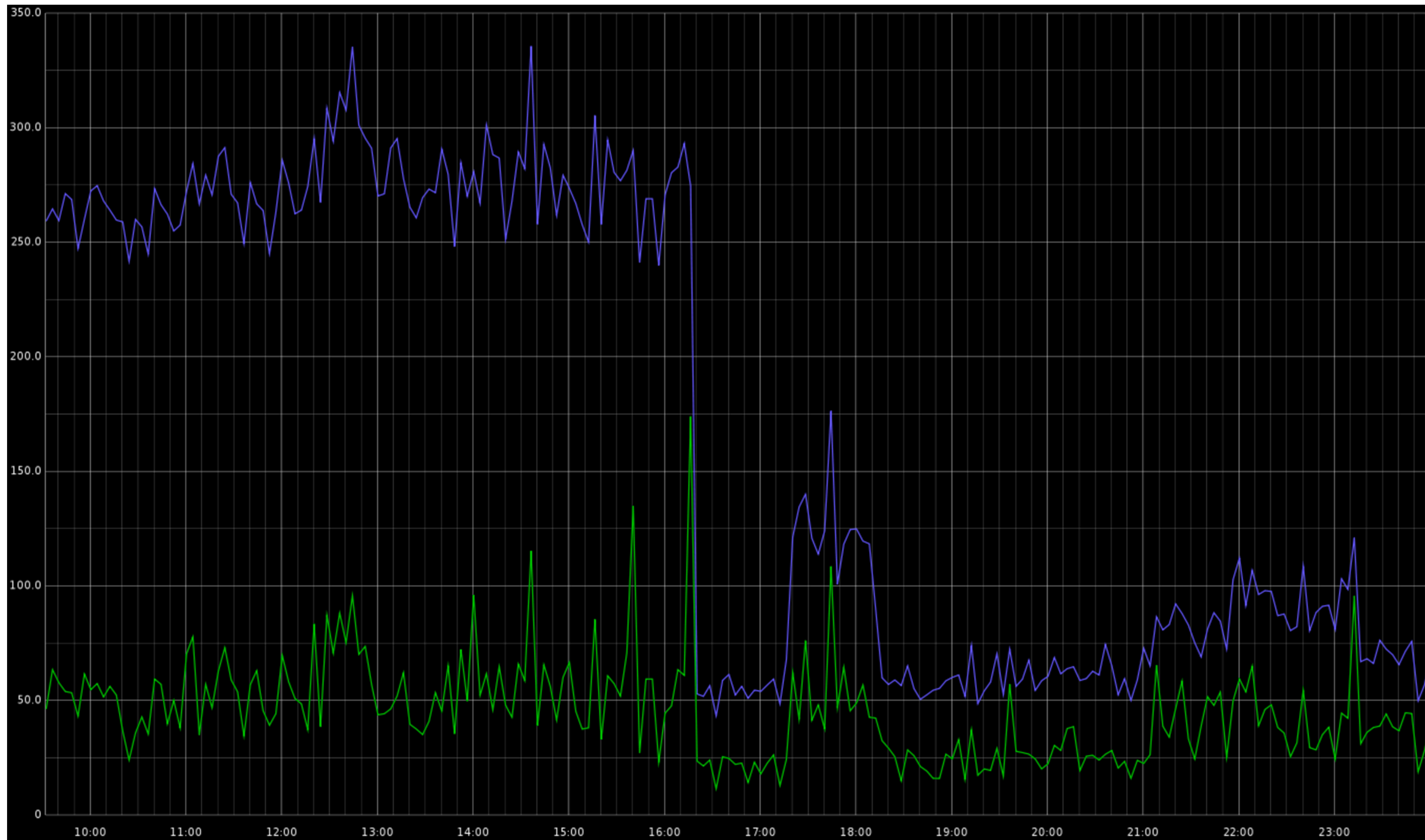
请求图示



Motan-PHP 部署



生产数据



从 RestFul API 到 Motan RPC



Motan Schema 刻画的 RPC 服务

服务协议

Path

grpc://host:port/helloworld.Greeter?group=idevz-test

目标节点

请求参数

URL

RPC描述

Motan Schema 刻画的 RPC服务

`http://host:port/query_string&group=rpc_grp&service=rpc_srv`

CURL HTTP描述

Motan Schema 刻画的 RPC 服务

motan2://host:port/service?group=idevz-test

http://host:port/.../show_batch.json?group=grp&service=srv

cedrus://host:port/.../show_batch.json?group=grp&service=srv

Motan

Srv 描述

从 RestFul API 到 Motan RPC

```
$url_str = 'http://127.0.0.1:9981/2/statuses/show_batch.json?source=2975945008&ids=4124';  
$url = new \Motan\URL($url_str);  
$cx = new \Motan\Client($url);  
$params = [];  
$headers = ['Authorization'=>'Basic ' . base64_encode('qa_test050@sina.cn:mm123456')];  
$rs = $cx->get($params, $headers);  
if (null === $rs) {  
    print_r($cx->getResponseException());  
}  
print_r($rs);  
die;
```

Server

将 Restful API 直接导出为 RPC 服务

Client

如请求 HTTP 接口般 调用 RPC 服务

降低迁移成本

欢迎持续关注微博开源

<https://github.com/weibocom>

Motan

<https://github.com/weibocom/motan>

OpenDCP

<https://github.com/weibocom/opendcp>

欢迎关注我的开源项目



580320197

<https://github.com/idevz>

