

杨翊（席翁）

# 《Nacos2.0 及其生态》

# 自我介绍

杨翊 - 花名席翁

**2年+ 开源经验**

- Apache 社区
- Alibaba 社区
- Github Id :  
KomachiSion

**Apache  
ShardingSphere  
PMC**

- 路由模块
- 分布式事务
- 数据同步
- 弹性扩缩容

**Alibaba Nacos PMC**

- 服务发现模块
- 内核提升

# 目录



# Nacos 简介

Nacos /nɑ:kəʊs/ 是 Dynamic Naming and Configuration Service的首字母简称

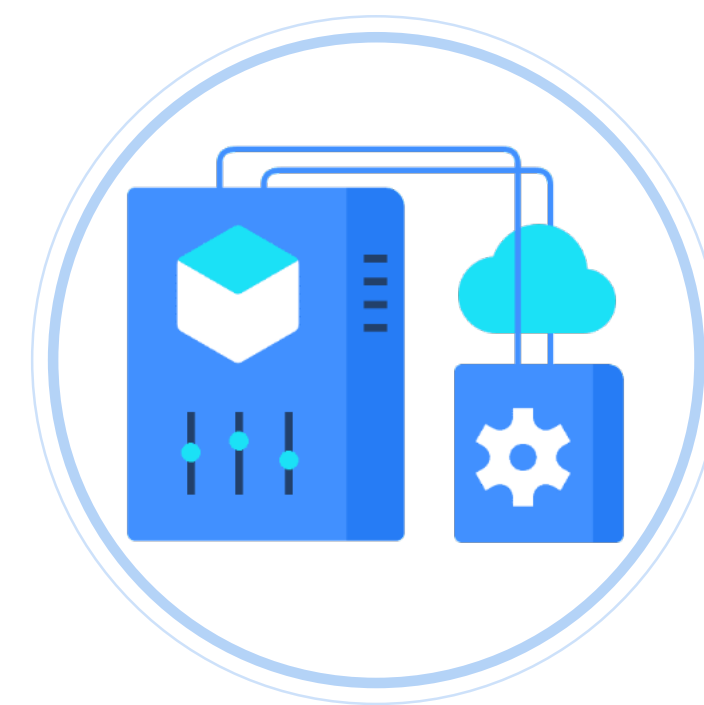
一个更易于构建云原生应用的动态服务发现、配置管理和服务管理平台



服务发现与管理



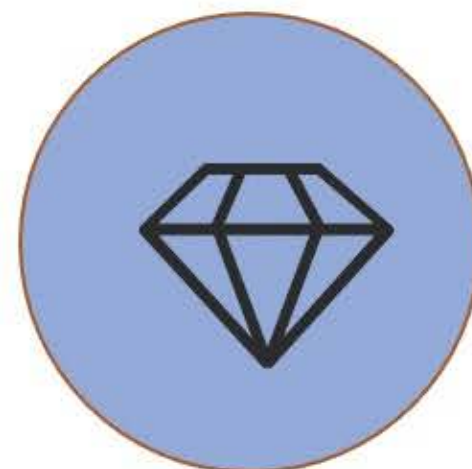
动态DNS服务



动态配置服务



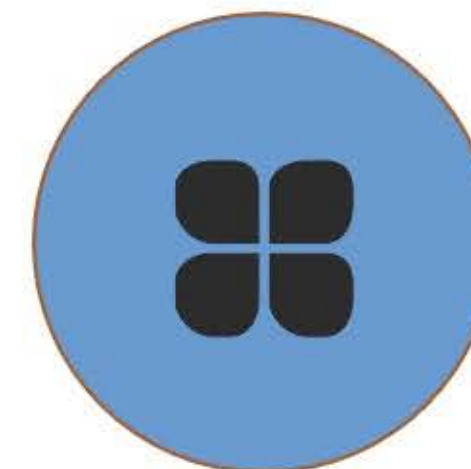
易用



稳定

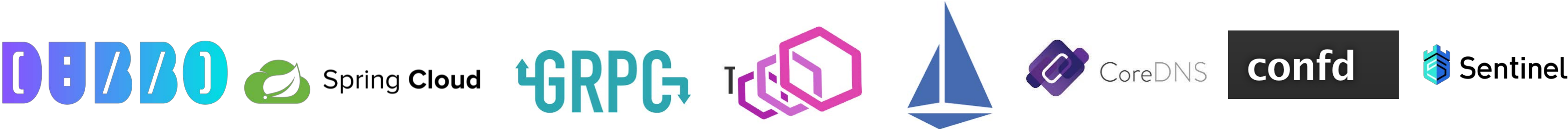


实时



规模

# Nacos 简介



NACOS.

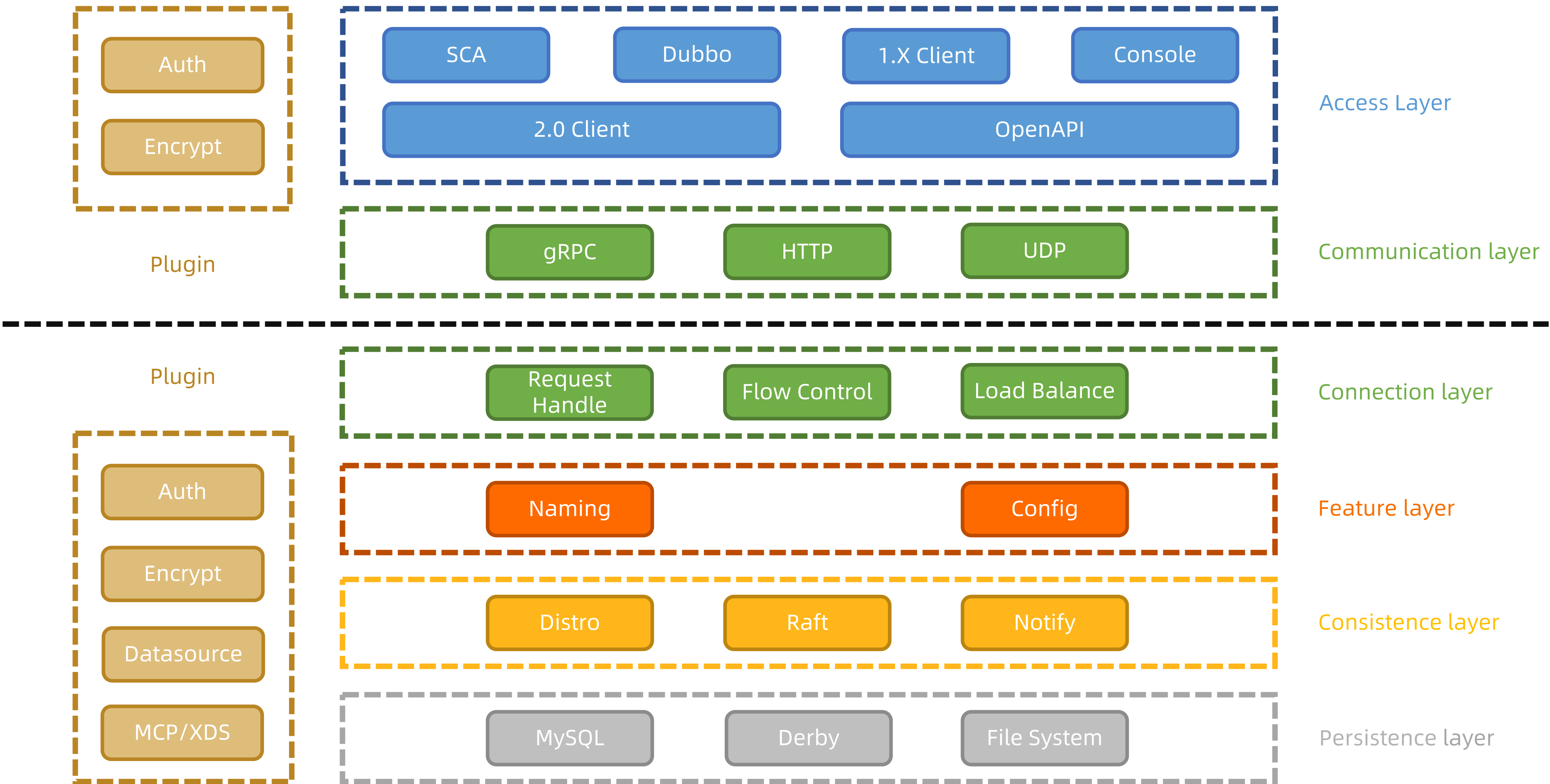
# Nacos 简介



# 目录

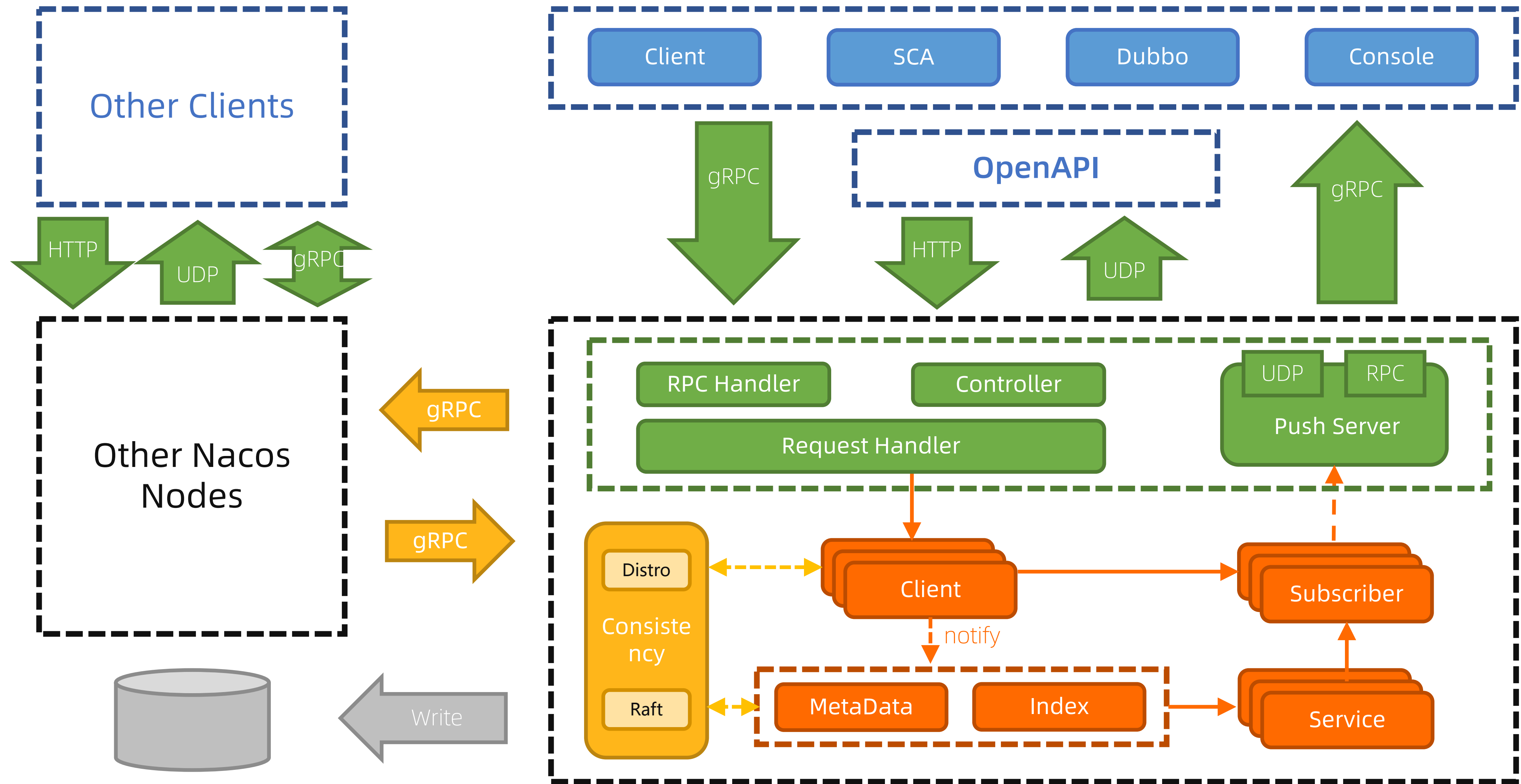


# Nacos 2.0架构

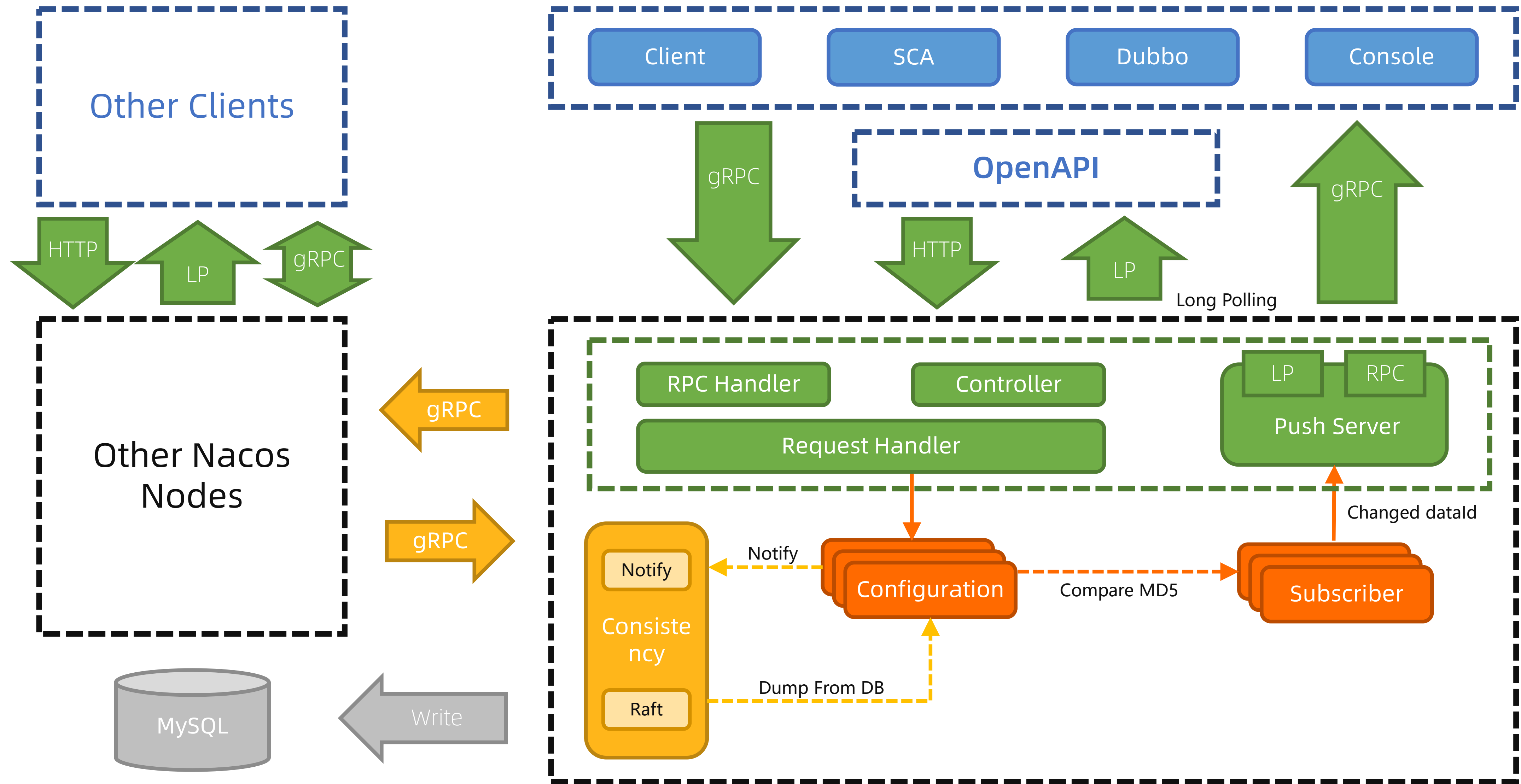




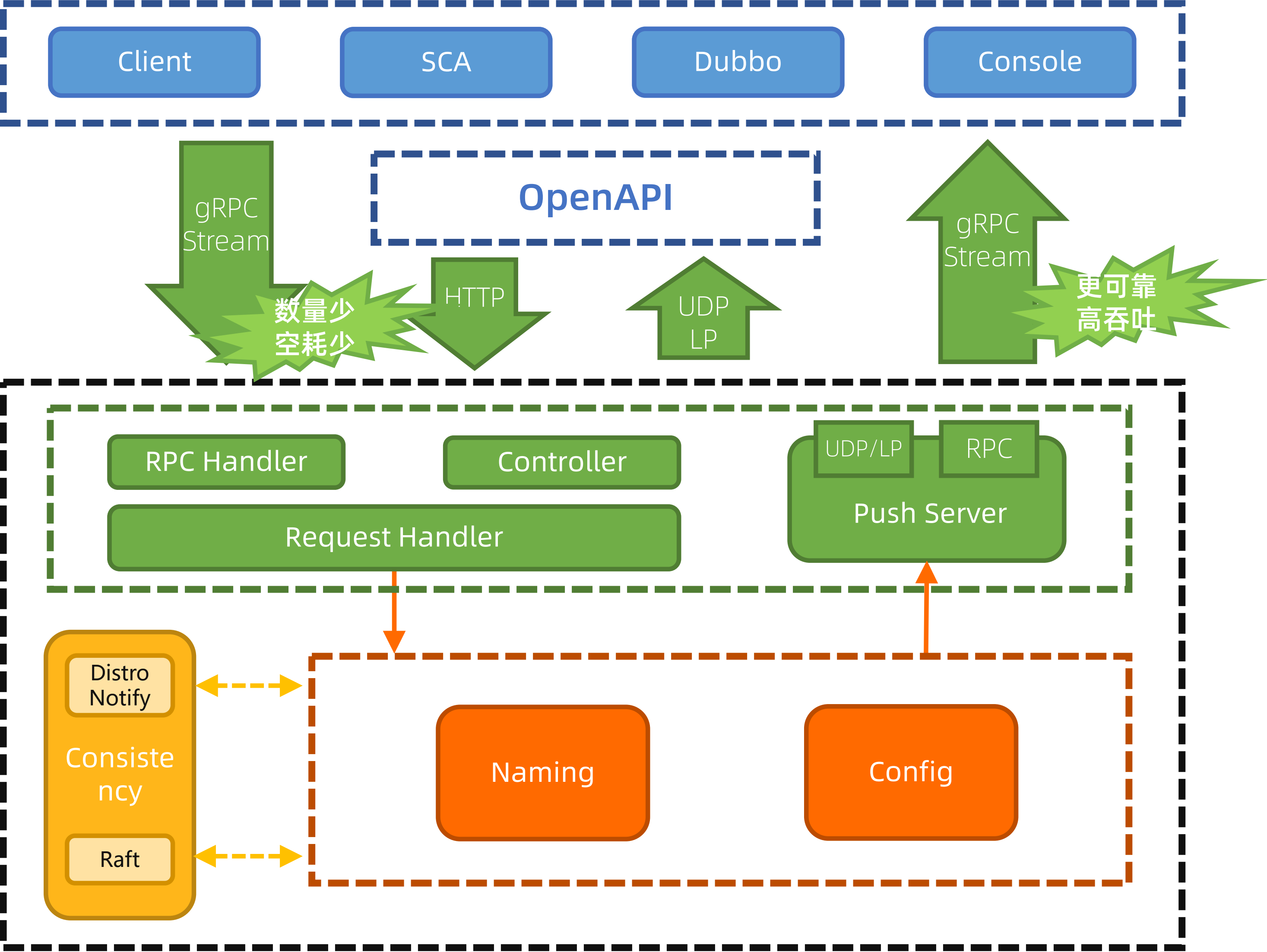
# Nacos 2.0架构下的服务发现



# Nacos 2.0架构下的配置管理



# Nacos 2.0架构优缺点



- 优点:
- 1、长连接缓解TIME\_WAIT状态连接
  - 2、流式推送更可靠，吞吐更高
  - 3、通过TCP连接快速感知服务变化
  - 4、客户端不再需要大量心跳
  - 5、真实的长连接，解决配置模块GC问题

- 缺点:
- 1、结构复杂，需要管理连接状态
  - 2、数据无状态 → 有状态
  - 3、可观测性降低

# 目录



# Nacos2.0性能—服务发现



## 服务端

指标	参数
机器	CPU 8核，内存16G
集群规模	10节点
Nacos版本	Nacos2.0.0-BETA/Nacos1.4.1

## 客户端

指标	参数
机器	CPU 4核，内存8G
集群规模	200节点
Nacos版本	Nacos2.0.0-BETA/Nacos1.4.1

# Nacos2.0性能—服务发现

-- 稳定状态测试

版本	阶段	CPU	GC	推送失败率
2.0.0-BETA	注册	23% ~ 30%	无FGC	单台服务端 (4982/62112) 8%
	稳定	3%	无FGC	无推送
	注销	10% ~ 24%	无FGC	0%
1.4.1	无法达到稳态	5% ~ 23%	FGC 1次/s	100%

## 10W级压测结论：

- 1. Nacos2.0能够较轻松地支撑该规模的客户端数及实例数。
- 2. Nacos1.X 在该规模下无法达到稳态，集群处于频繁FGC状态
- 3. Nacos2.0在大批量注册及订阅过程中，有一定比例的推送出现超时，但最终会在重试后推送成功。

为了测试出Nacos2.0和Nacos1.X的大致性能差异，又对Nacos1.X在稳定状态的场景进行了多轮测试

# Nacos2.0性能—服务发现

-- 1.X稳定状态补充测试

压力规模 (机器 * 客户端)	阶段	CPU	GC	推送失败率
200 * 250 (5w客户端, 25w服务实例)	无法达到稳态	5% ~ 23%	FGC 1次/10s	100%
200 * 100 (2w客户端, 10w服务实例)	无法达到稳态	30% ~ 40%	数分钟1次	0%
200 * 60 (1.2w客户端, 6w服务实例)	注册	20%	无FGC	0%
	稳定	10%	无FGC	0%
	注销	13%	无FGC	0%

## 补充结论：

- 1. Nacos1.X 在 5w客户端，25w服务实例的场景下，仍然频繁FGC，无法响应。
- 2. Nacos1.X 在2w客户端，10w服务实例的场景下，集群状态正常，FGC较少，但由于心跳处理速度不够，因此无法达到稳态，CPU消耗高。
- 3. Nacos1.X 在1.2w客户端，6w服务实例的场景下，能够稳定正常运行，达到稳态后，CPU较Nacos2.0还要更高。

# Nacos2.0性能—服务发现

-- 频繁变更测试

版本	压力规模 (机器 * 客户端)	CPU	GC	推送失败率
2.0.0-BETA	200 * 500 (10w客户端, 50w服务实例)	27%	无FGC	0%
1.4.1	200 * 60 (1.2w客户端, 6w服务实例)	15%	无FGC	单台服务端 (14/20499) 0.1%

## 频繁变更场景结论：

- 1. Nacos2.0频繁变更场景的系统状态和大批量集中注册时的状态没有太大差异，但推送不再超时失败。
- 2. Nacos1.X 在1.2w客户端规模下也接近集中注册时的状态，但UDP推送的不稳定性会导致在大量推送的情况下会有很小概率会推送失败，在重试和对账轮训的情况下最终依然能推送成功。

注意：由于1.X无法在10W级客户端规模下达到稳定状态，因此该频繁变更测试两个版本在不同的规模下进行，结论仅供参考



# Nacos2.0性能—配置管理



## 服务端

指标	参数
机器	CPU 8核，内存16G
集群规模	1节点
Nacos版本	Nacos2.0.0-BETA/Nacos1.4.1

## 客户端

指标	参数
机器	CPU 4核，内存8G
集群规模	200节点
Nacos版本	Nacos2.0.0-BETA/Nacos1.4.1

# Nacos2.0性能—配置管理

-- 连接容量测试

版本	客户端数	连接建立 CPU	稳定状态 CPU	GC
2.0.0-BETA	6000	40%	1%	无FGC
	12000	80%	1%	无FGC
	21000	79%	3%	无FGC
	42000	75%	2%	无FGC
1.4.1	6000	80%	60%	CMSGC 1次/4s
	12000	无法达到该量级		

## 连接容量结论：

- 1. Nacos2.0 在各个连接数均保持稳定状态，负载消耗很低。尝试的最大连接数能够达到42000个。
- 2. Nacos1.X 的长轮训连接达到6000时已经有较高的CPU消耗和大量的CMSGC。只能勉强支撑。

# Nacos2.0性能—配置管理

-- 变更推送测试

在6000长连接下测试固定TPS的推送能力

版本	TPS	CPU	GC
2.0.0-BETA	3000	30%	无FGC YCG 1次/4s
	6000	60%	无FGC YCG 1次/3s
1.4.1	3000	80%	CMSGC 1次/3s YCG 1次/3s
	6000	90%	CMSGC 1次/7s YCG 1次/1s

## 变更推送结论：

1. Nacos2.0和1.X均能在6000长连接规模下维持高TPS的配置推送，但2.0的CPU和GC表现更为优秀
2. Nacos1.X 在频繁推送的场景下，CMSGC反而有一定改善，原因是轮训不再block，不会进入老年代，YGC变得频繁，轮训上下文及Request被YGC快速回收。

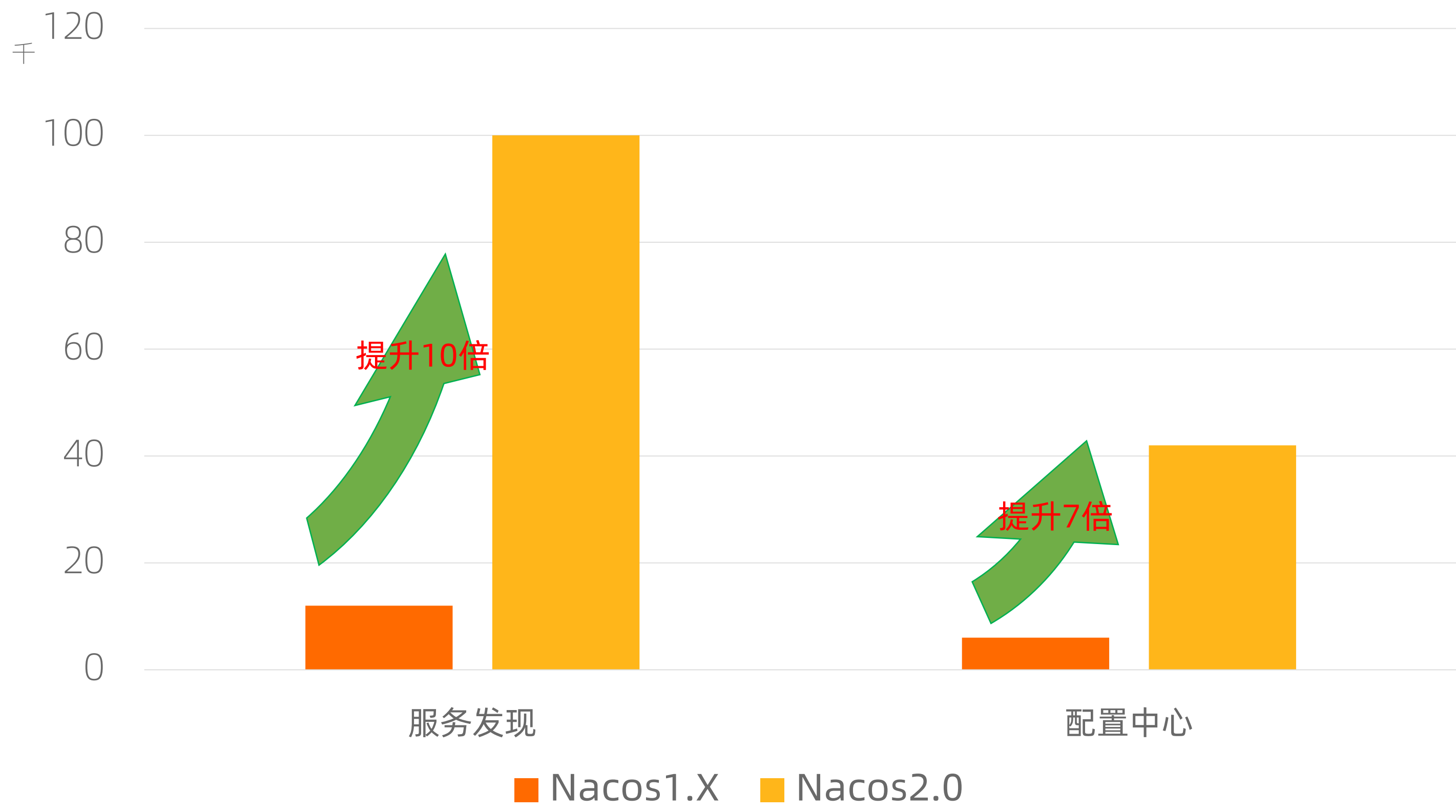
# Nacos2.0性能—结论

## 服务发现：

- Nacos2.0能够在10W级规模下，稳定运行；相比Nacos1.X版本的1.2W规模，提升约10倍。

## 配置管理：

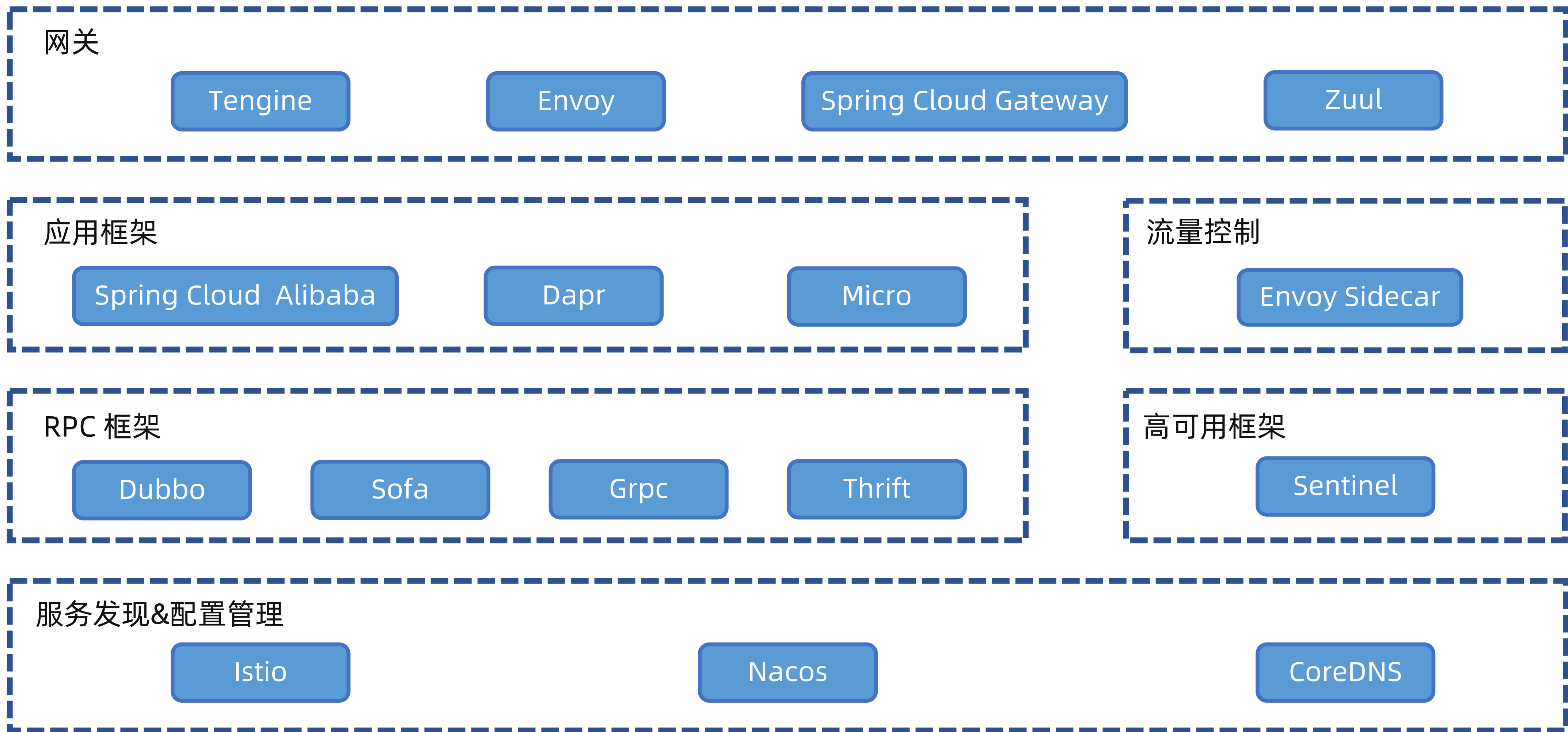
- Nacos2.0单机最高能够支撑4.2W个客户端连接；相比Nacos1.X，提升了7倍。



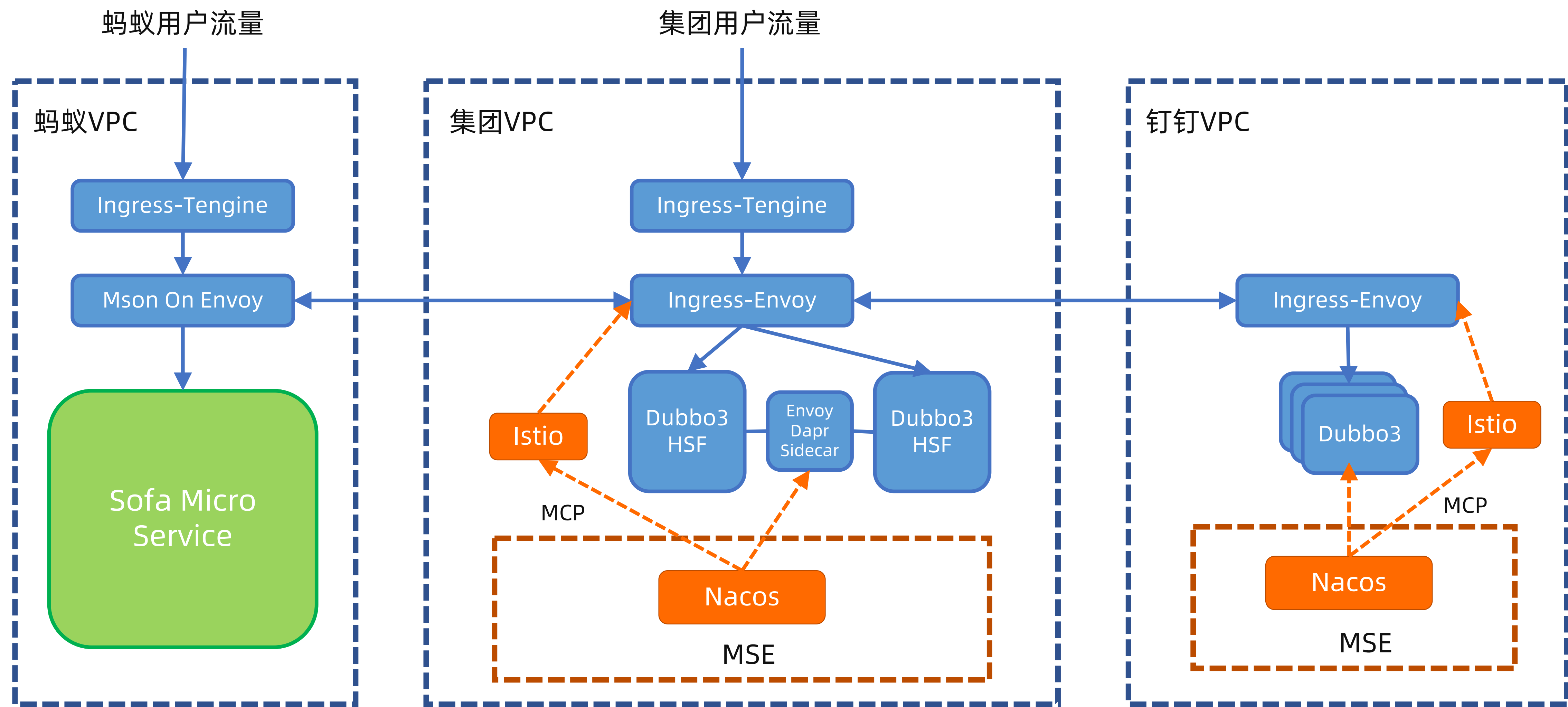
# 目录



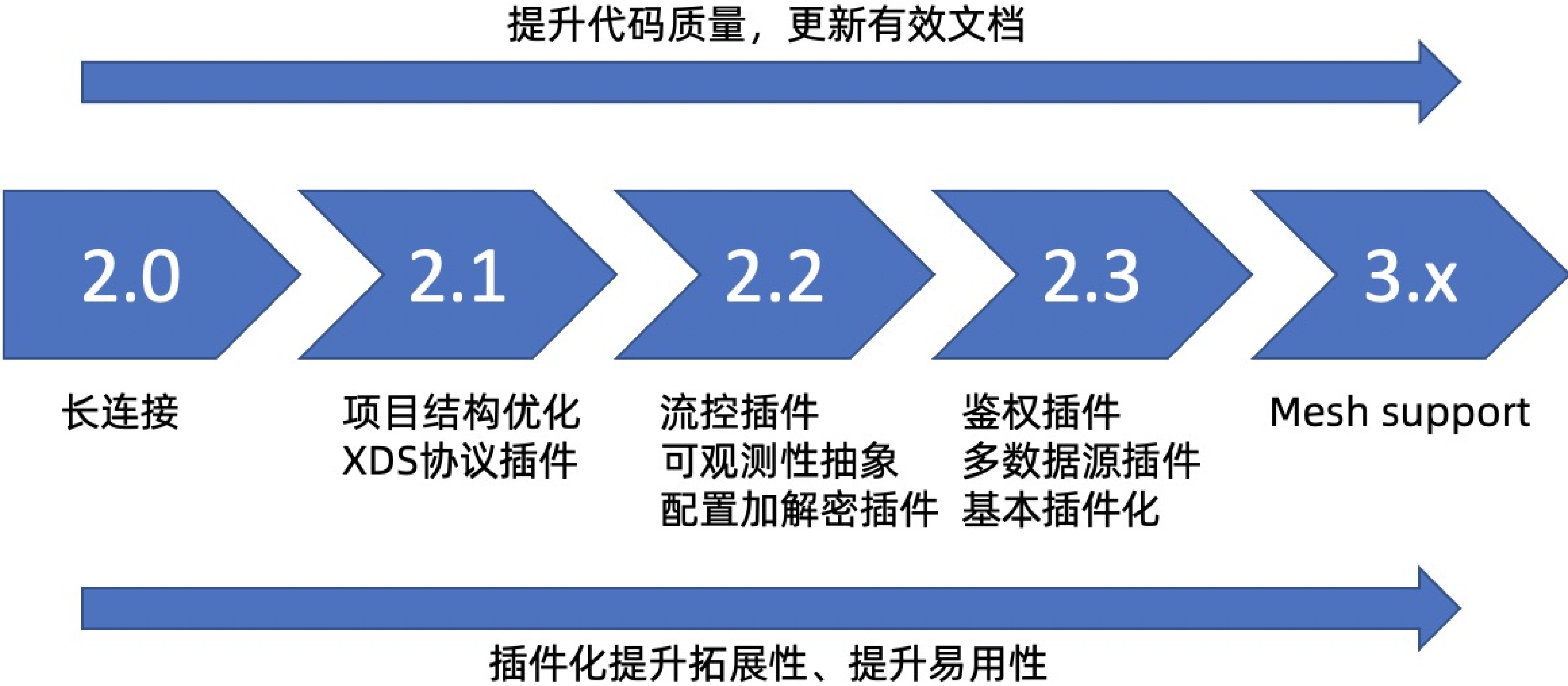
# Nacos 开源生态



# Nacos 生态阿里实践



# Nacos2.X 规划





# 加入Nacos社区

## Nacos社区群2

942 People



Scan the QR Code to Join the Group

## Nacos 2.0 用户群

189人



扫一扫群二维码，立刻加入该群。

Thanks

谢谢观看！