

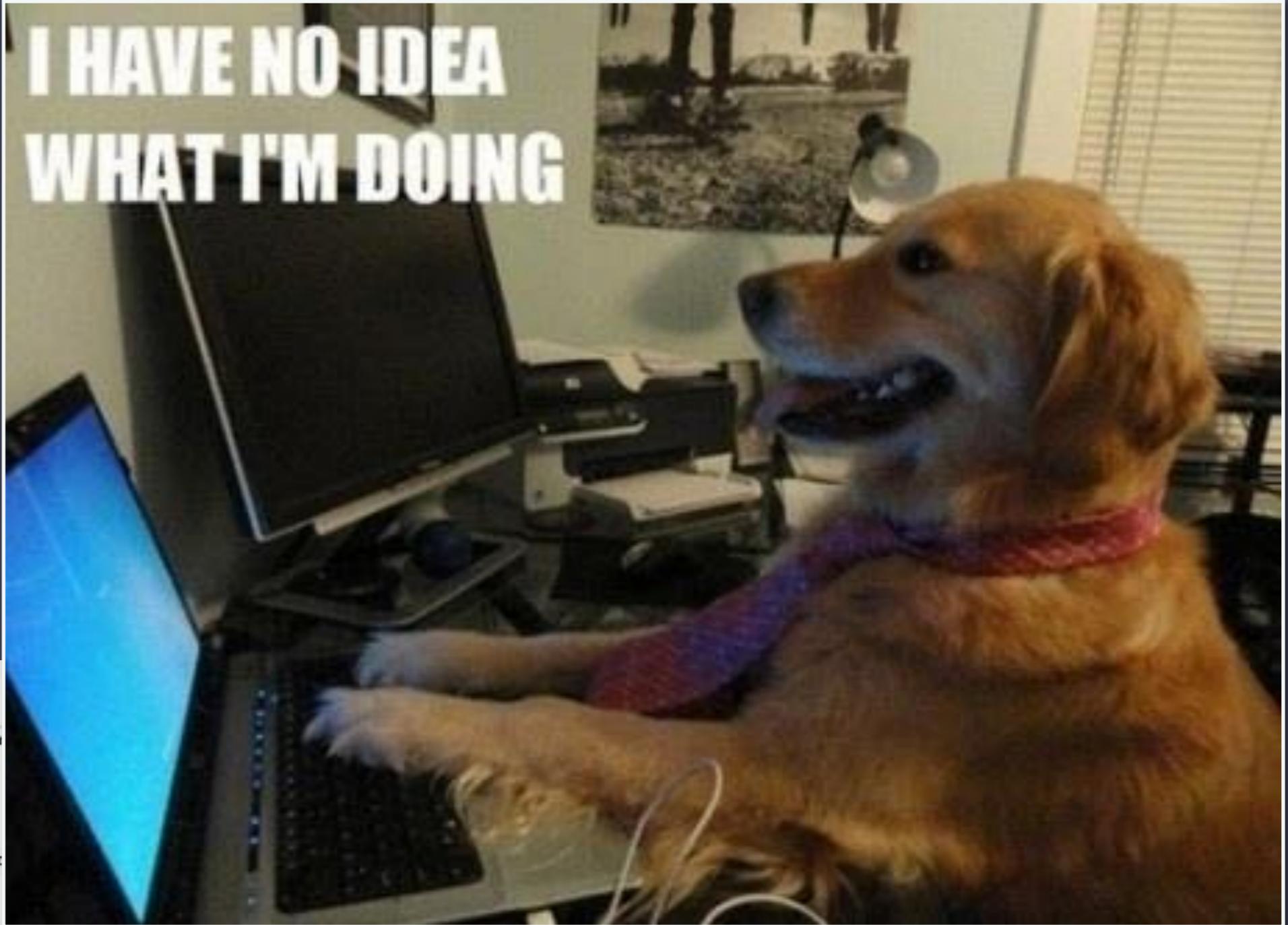
# Microservices in Practice

李昊 2017/03/18

没有干货



I HAVE NO IDEA  
WHAT I'M DOING



Google

what is a micros  
what is a microsoft account  
what is a microscope  
what is a microservice  
what is a microsite  
what is a microstate  
what is a microscope used for  
what is a microsecond  
what is a microsd card  
what is a microsoft word  
what is a microsoft

1. a microcomputer.
2. a microprocessor.

adjective

1. extremely small.  
"a micro bullet area"

Report offensive query



Translations, word origin, and more definitions

Feedback

“

*...the microservice architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms...*

*These services are built around business capabilities and independently deployable by fully automated deployment machinery. There is a bare minimum of centralized management of these services...*

-- Martin Fowler & James Lewis (2014)

# Microservice Architectural Style

- 细粒度的服务
- 独立进程
- 围绕业务建模
- 轻量级通信
- 去中心化管理

“

*...the microservice architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms...*

*These services are built around business capabilities and independently deployable by fully automated deployment machinery. There is a bare minimum of centralized management of these services...*

-- Martin Fowler & James Lewis (2014)

# Microservice Architectural Style

- 是一种风格
- 不是新科技
- 不是SOA换了个壳
- 不是一套有章可循的标准实现

“

*Technology is the answer, but what was the  
QUESTION*

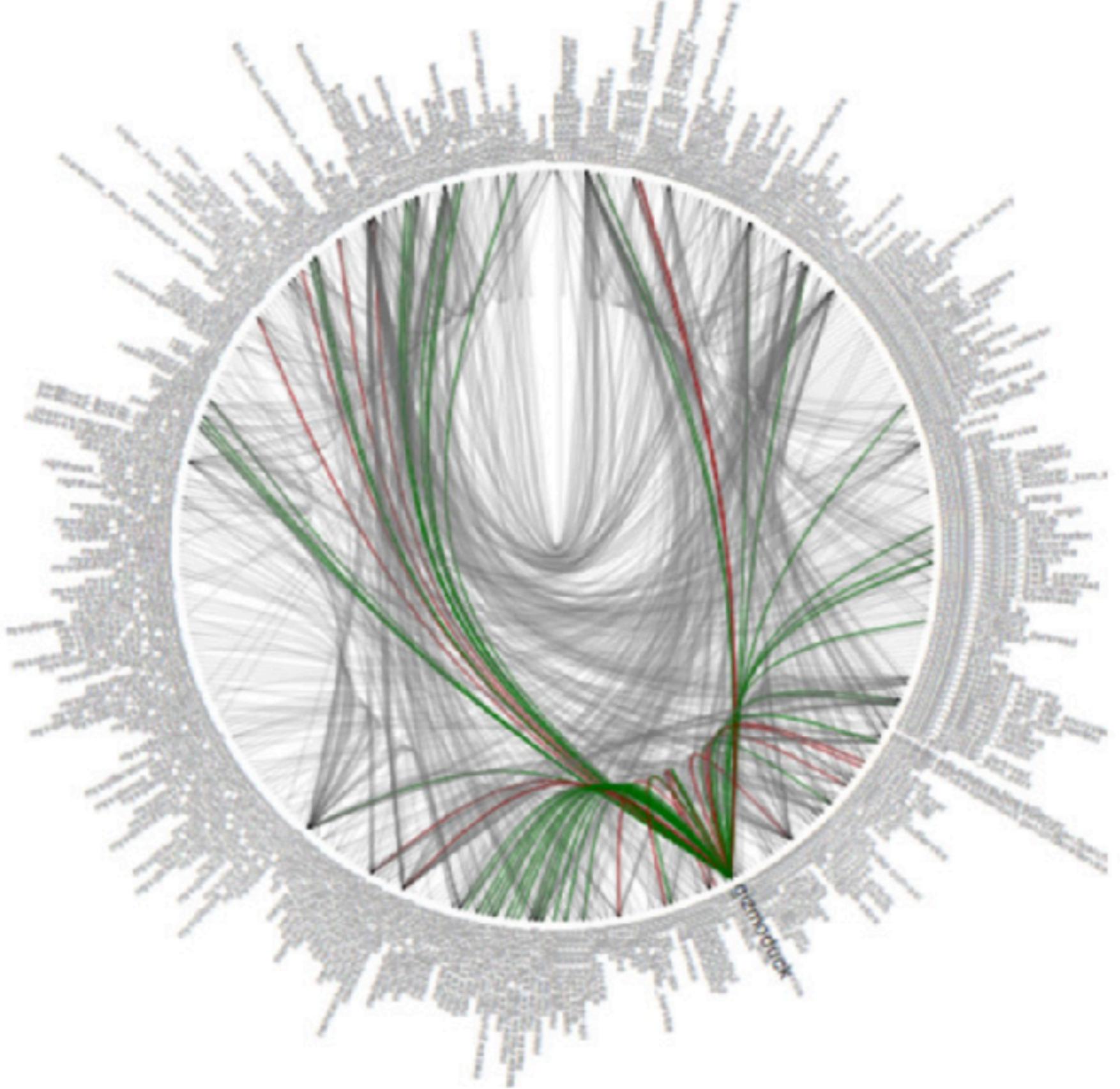
-- Cedric Price (1965)

# Monolithic System



# Distributed System



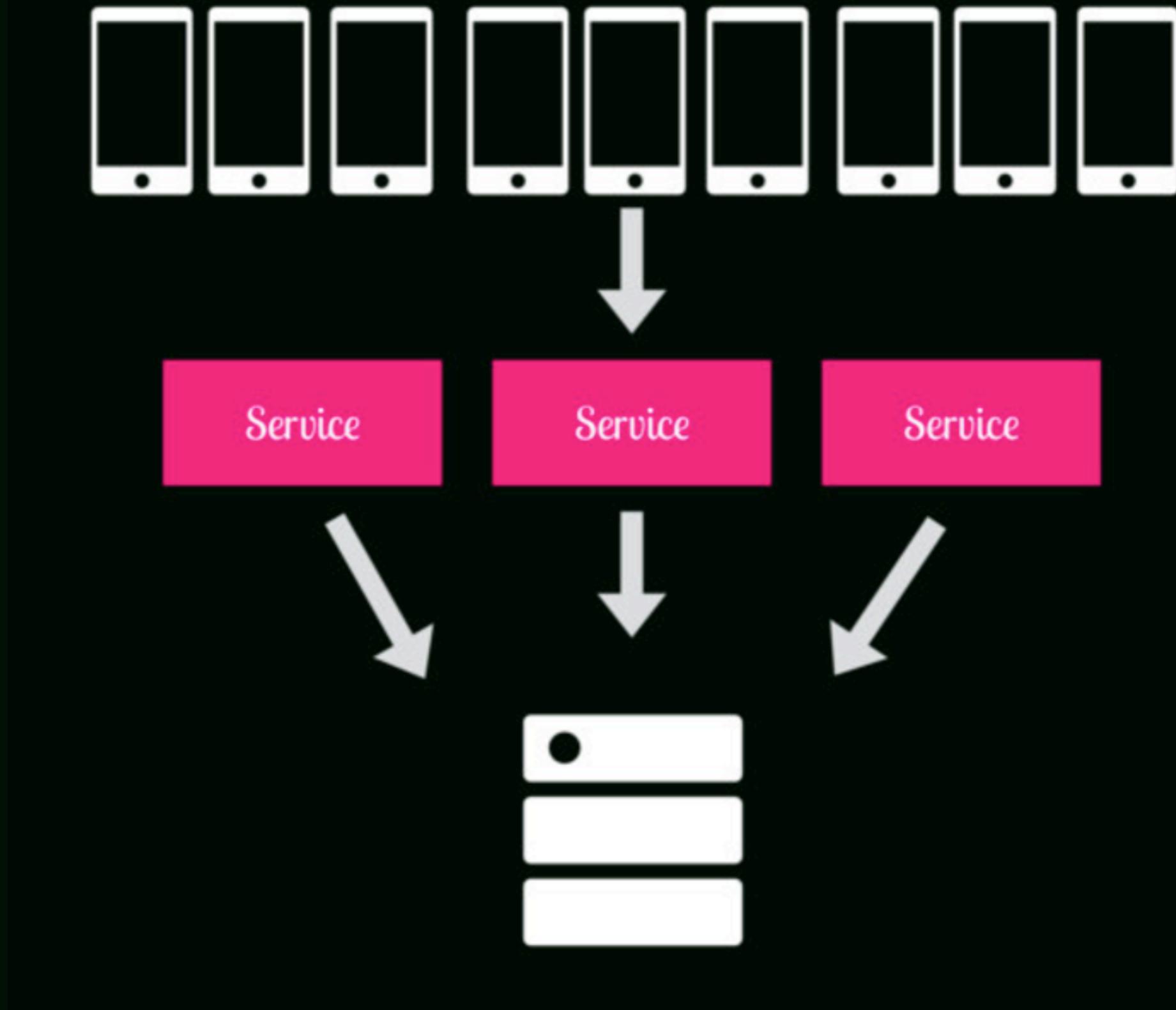


I know you are not Twitter

“

*A distributed system is one in which the failure of a computer you didn't even know existed can render your own computer unusable.*

-- Leslie Lamport



所以你也在写分布式系统

# The Hardest Part

Asynchrony

Partial Failure

- Nondeterminism
- There is no NOW

- Everything breaks
- FLP, CAP, Spanner?

# 解决分布式系统的什么问题？

- 移动互联网时代提供服务的需要
- 大型单体系统难以维继：维护 / 演进
- 硬件缺失，软件趟路
  - 必要的组件（服务发现，断路器）
  - 有用的方法（DDD，SAGA，CQRS）
  - 日益成熟的新技术（容器，No-SQL）

# 所以什么是微服务？

在分布式系统建设中，为了应对需求的快速变更，在高增速的大公司内部构建高效、自治、响应迅速的“创业风格”团队的一些尝试



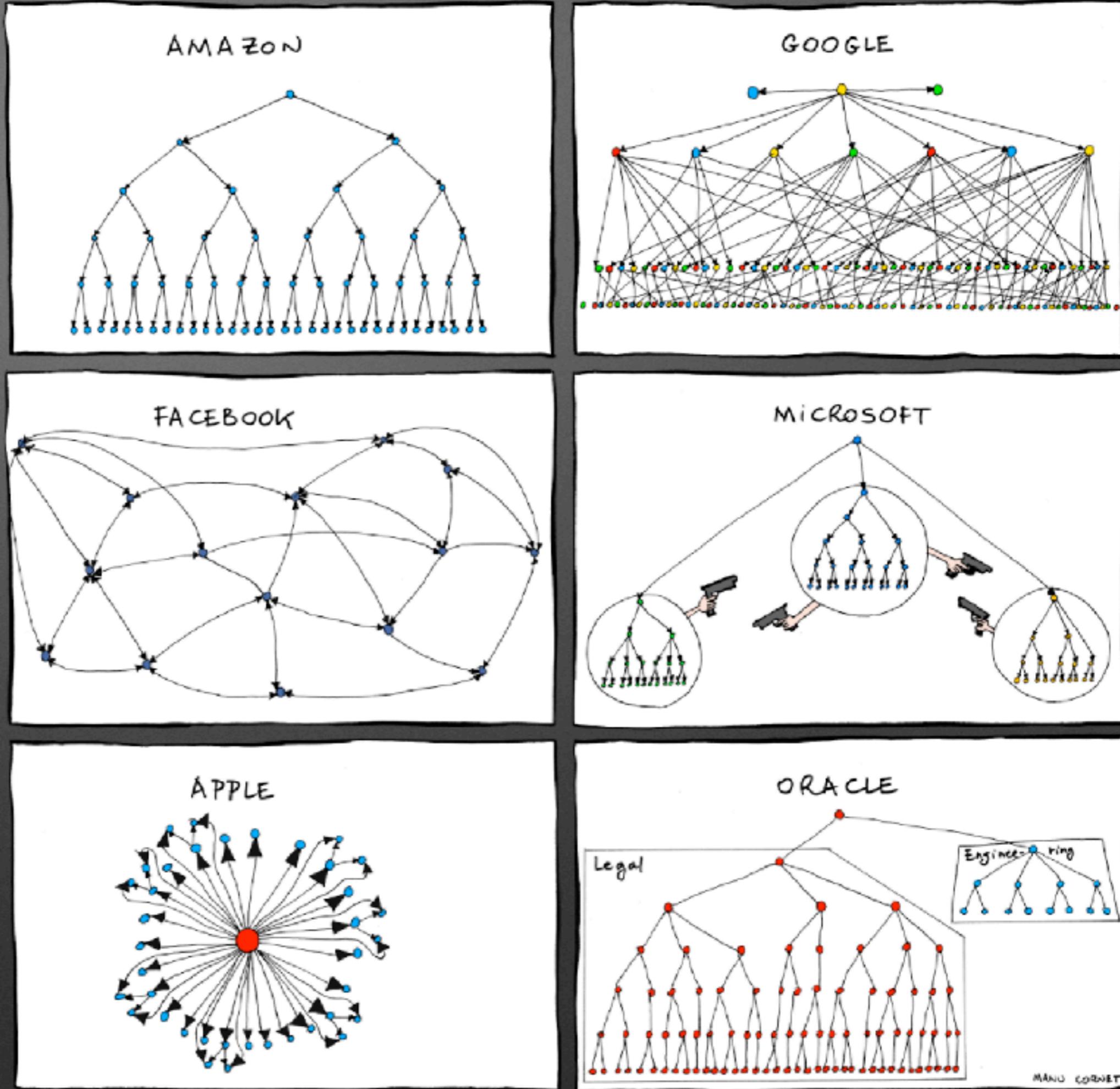
# 如何实施



# 人是根本

- 有多少研发？里面有多少懂运维？
- 有多少团队？他们互相信任吗？
- 有多少产品？多少业务系统？
- 哪些需要纳入微服务化的计划？
- 如何去在内部销售基础设施？

# 康威定律



# 一个实例

“

DBA是公司唯一有操作生产数据库权限的人，一个对我自己的服务的简单数据库操作必须等他有空的时候来执行，这实在是太荒唐了

-- SWE

# 一个实例

“

那些开发根本不去找root cause，也不懂执行的SQL有什么影响，老是跑到生产环境上，从history里面找sql语句执行，太不靠谱了

-- DBA

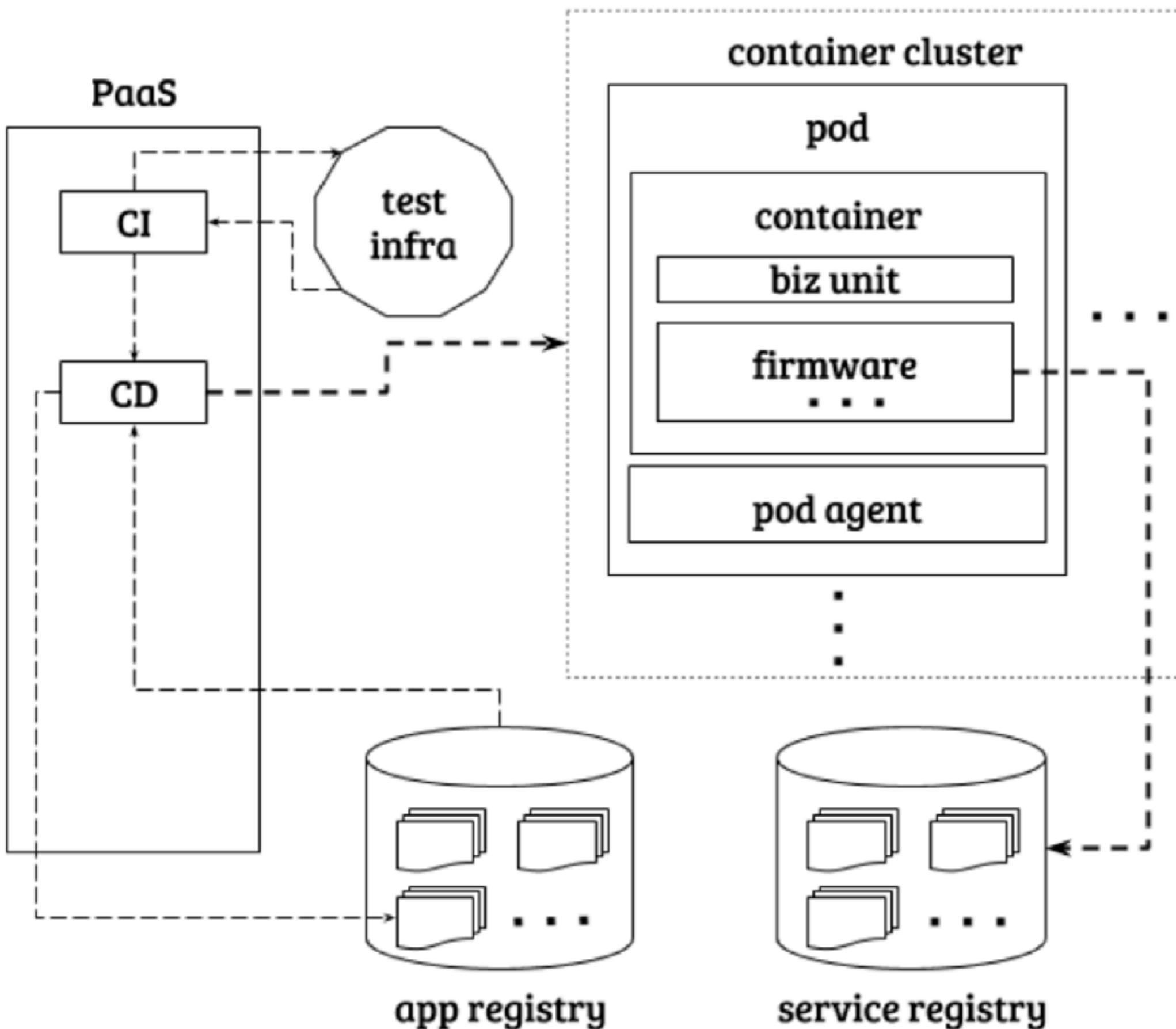
# 一些心得

- 专职的基础设施团队负责
- 挑选粒度合适，边界清楚的服务开刀
- 提供标准化模板，提供迁移指南
- 去中心化不是多样化
- You build it, you run it

# 流程

**Continuous Development**  
**Continuous Integration**  
**Continuous Deploy**

# 基于自建容器平台的CI和CD



# 容器管理平台

货车帮登录平台

开发测试环境(西信机房)

帮助

yuwen

总览

应用

应用管理

域名管理

发布统计

模板

集群

主机管理

业务线管理

容器管理

仓库管理

系统

日志审计

参数设置

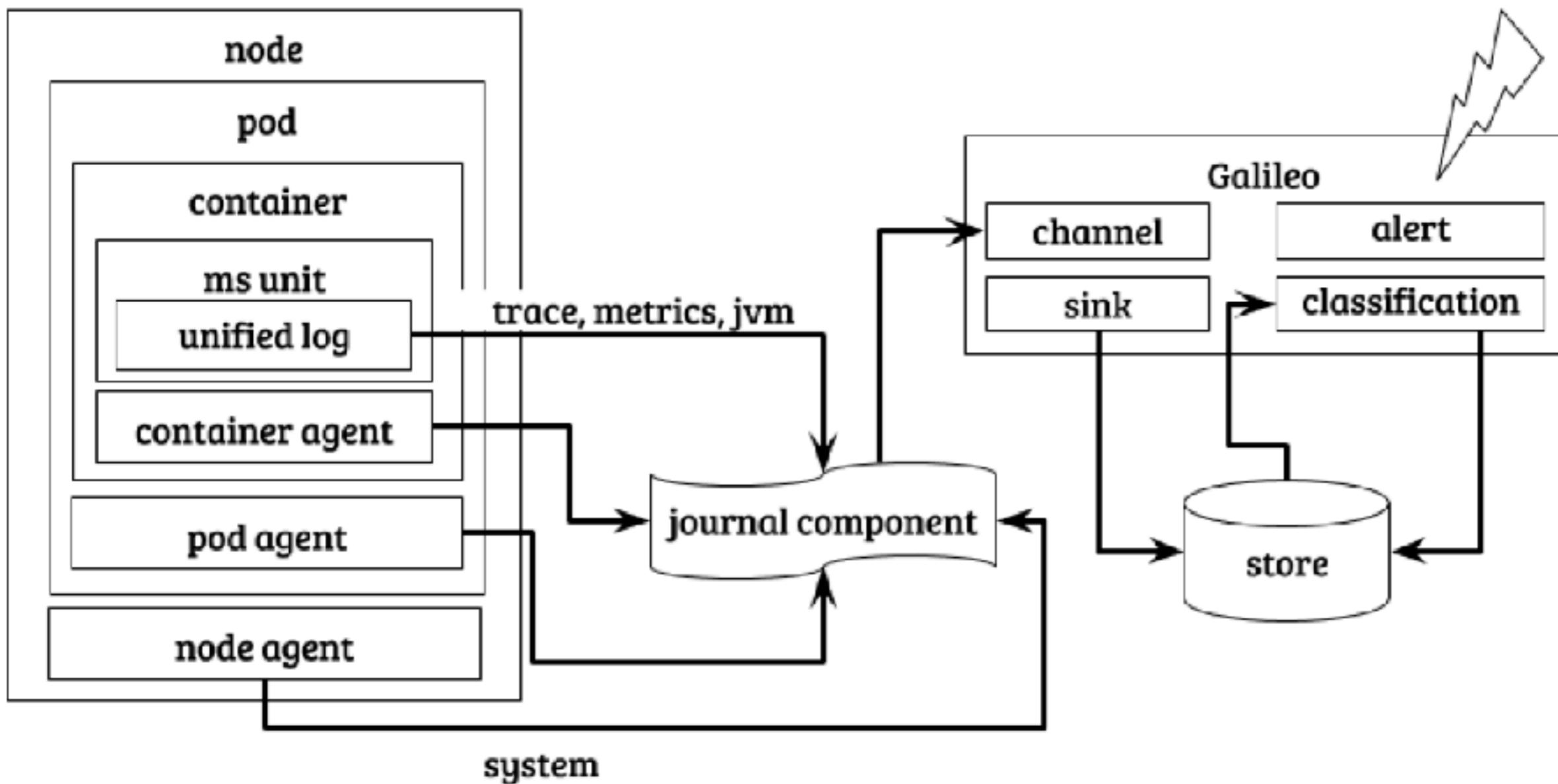
用户权限

集群管理

容器ID	容器名称	版本	镜像	应用名称	业务线	部署类型	NODE IP	POD IP	状态
bc06c0b70ba3	apoloanserver	1.1.2	dev.docker.56qq.cn:5000/gps-gpc:v1.2	loan	金融&支付	dev	10.2.1.12	172.16.17.3	running
4eb201ff9c17	cmssqj	1.19.54	dev.docker.56qq.cn:5000/dns-dnssapi-001:v1.1.9	dns	基础设施	qa	10.2.1.28	172.16.80.31	running
0612801943fb	cjudge	1.0.7	dev.docker.56qq.cn:5000/galileo.qjudgev2015.2.21.705	monitor	基础设施	qa	10.2.1.46	172.16.49.32	running
352283a65fe5	infci	1.5.36	dev.docker.56qq.cn:5000/mic-infci:v1.5	infci	基础设施	qa	10.2.1.27	172.16.53.18	running
015e64699ee1	cc	1.14.21	dev.docker.56qq.cn:5000/ce-cc:v1.14	cc	基础设施	qa	10.2.1.27	172.16.53.8	running
376c533085e2	gaci	1.15.1	dev.docker.56qq.cn:5000/galileo-gacis:v1.5	galileo	基础设施	dev	10.2.1.12	172.16.17.26	running
bd58f8a43473	cloudmaster	1.19.11	dev.docker.56qq.cn:5000/ump-cloudmaster:v1.1.9	ump	基础设施	dev	10.2.1.27	172.16.53.32	running
bda282766fd4	collector-p-ms2	1.4.0	dev.docker.56qq.cn:5000/galileo.collector-proxy-ms2:v201704-1	collector-p-ms2	基础设施	dev	10.2.1.27	172.16.53.4	terminated

共144条 1 2 3 4 5 ... 18 >

# 自建监控平台Galileo



# Galileo



# Galileo

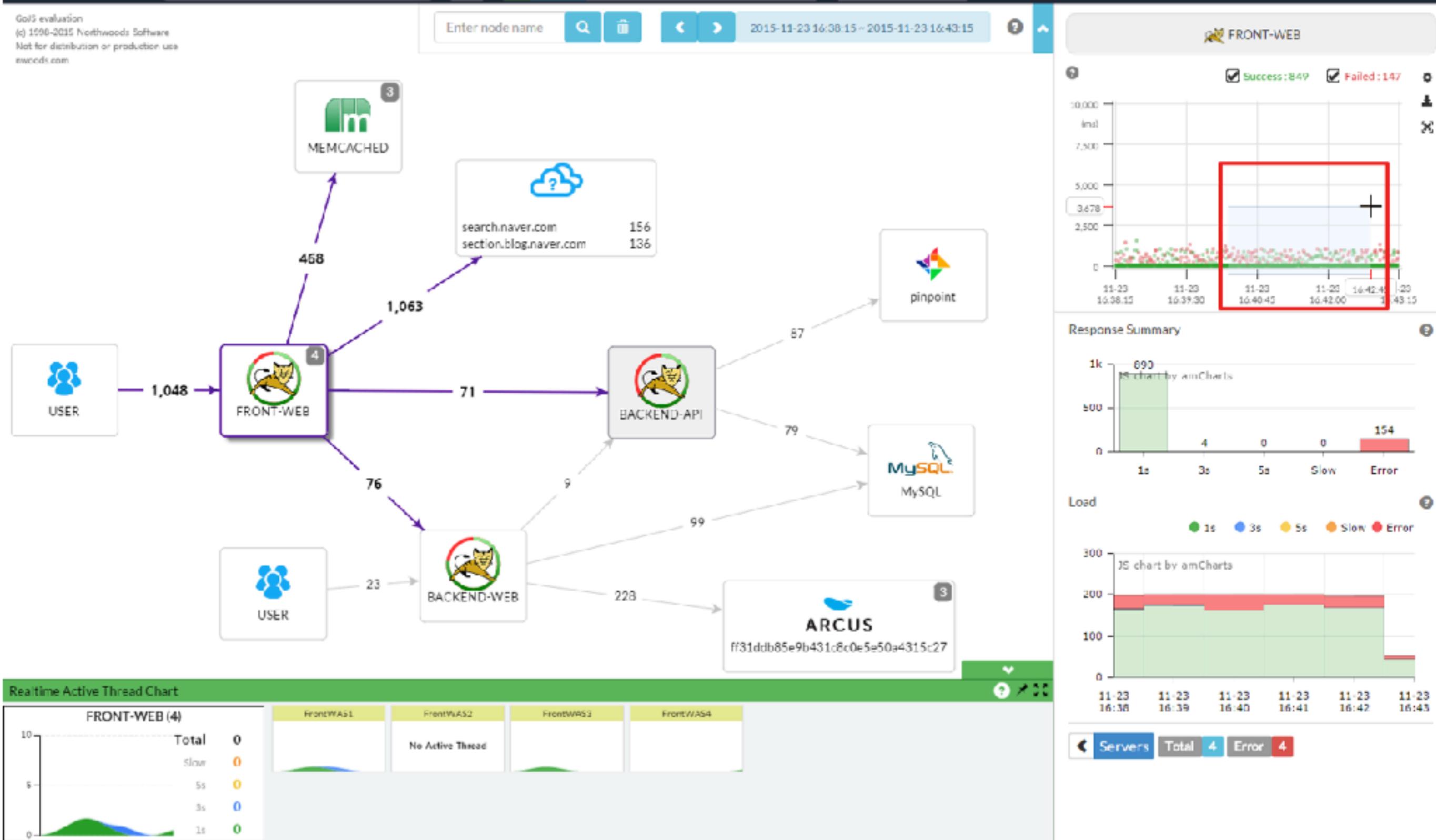
#	Start Time	Path	Res.(ms)	Exception	Agent	Client IP	Transaction
46	11/23 16:40:01 445	/backendapi.pinpoint	154	↳	ApiWAS1	10.101.55.177	FrontWAS2^1447735076405^686834
9	11/23 16:42:35 864	/backendapi.pinpoint	15	↳	ApiWAS1	10.101.55.177	FrontWAS3^1446197325899^6583581
7	11/23 16:42:42 193	/backendapi.pinpoint	13		ApiWAS1	10.101.52.71	FrontWAS3^1446197325899^6583589
36	11/23 16:41:01 441	/backendapi.pinpoint	13		ApiWAS1	10.101.55.177	FrontWAS3^1446197325899^6583453
49	11/23 16:39:49 171	/backendapi.pinpoint	12		ApiWAS1	10.101.55.177	FrontWAS1^1447735076452^175174
50	11/23 16:40:01 445	/backendapi.pinpoint	154	↳	ApiWAS1	10.101.55.177	FrontWAS1^1447735076452^175174

Application:/backendapi.pinpoint TransactionId:FrontWAS2^1447735076405^686834 AgentId:ApiWAS1 ApplicationName:BACKEND-API

Call Tree	Server Map	Timeline	Mixed View	nelo	Self >= 1000	Q ↕	Complete	C					
Method			Argument		Start Time	Gap(ms)	Exec(ms)	Exec(%)	Self(ms)	Class	API	Agent	Application
↳ Tomcat Servlet Process			/netspider.pinpoint		16:40:01 801	0	885	██████████	0		TOMCAT	FrontWAS2	FRONT-NEB
REMOTE_ADDRESS			127.0.0.1										
invoke(Request request, Response response)					16:40:02 801	0	885	██████████	0	StandardContextValve	TOMCAT_METHOD	FrontWAS2	FRONT-NEB
doGet(HttpServletRequest request, HttpServletResponse res					16:40:02 801	0	885	██████████	4	FrameworkServlet	SPRING	FrontWAS2	FRONT-NEB
demo()					16:40:02 802	1	881	██████████	599	DemoController	SPRING_BEAN	FrontWAS2	FRONT-NEB
execute(HttpServletRequest request, ResponseHandler resp					16:40:02 802	0	58	■■■■■	14	CloseableHttpclient..	HTTP_CLIENT_4	FrontWAS2	FRONT-NEB
open(Route route, HttpContext context, HttpPal section.blog.naver.com					16:40:02 802	0	2	■■■■■	2	AbstractHttpClientCon...	HTTP_CLIENT_4	FrontWAS2	FRONT-NEB
execute(HttpServletRequest request, HttpClientConnection /					16:40:02 804	0	34	■■■■■	34	HttpRequestExecutor..	HTTP_CLIENT_4	FrontWAS2	FRONT-NEB
http.status.code			200										
http.io			write=0, read=34										
connect()			http://section.cafe.naver.com/		16:40:02 852	50	2	■■■■■	2	URLConnection	30K_HTTPCONN	FrontWAS2	FRONT-NEB
execute(HttpServletRequest request, ResponseHandler resp					16:40:03 443	589	158	■■■■■	1	CloseableHttpclient..	HTTP_CLIENT_4	FrontWAS2	FRONT-NEB
↳ IOException			Internal Server Error										
open(Route route, HttpContext context, HttpPal dev-pinpoint-workload003.rcl:8080					16:40:03 443	0	1	■■■■■	1	AbstractHttpClientCon...	HTTP_CLIENT_4	FrontWAS2	FRONT-NEB
execute(HttpServletRequest request, HttpClientConnection /backendapi.pinpoint					16:40:03 444	0	186	██████████	156	HttpRequestExecutor..	HTTP_CLIENT_4	FrontWAS2	FRONT-NEB
http.status.code			500										
http.io			write=0, read=156										
↳ Tomcat Servlet Process			/backendapi.pinpoint		16:40:03 445	1	154	██████████	0		TOMCAT	ApiWAS1	BACKEND-API
REMOTE_ADDRESS			10.101.55.177										
invoke(Request request, Response response)					16:40:03 445	0	154	██████████	144	StandardContextValve	TOMCAT_METHOD	ApiWAS1	BACKEND-API
doPost(HttpServletRequest request, HttpServletResponse res					16:40:03 445	0	16	■■■■■	2	FrameworkServlet	SPRING	ApiWAS1	BACKEND-API
↳ NestedServletException			Request processing failed; nested exception is java.sql.SQLException: ORA-00001: unique constraint (SYS_C001044) violated										
backendapi()					16:40:03 446	1	8	■■■■■	9	DemoController	SPRING_BEAN	ApiWAS1	BACKEND-API
↳ RuntimeException			com.mysql.jdbc.driver.CUBRIDException: Syntax: Unique constraint violation										
getConnection()					16:40:03 446	0	8	■■■■■	9	BasicDataSource	DBCP	ApiWAS1	BACKEND-API
setAutoCommit(boolean autoCommitFlag) false					16:40:03 446	0	2	■■■■■	2	ConnectionImpl	MySQL(MySQL)	ApiWAS1	BACKEND-API
list()					16:40:03 448	0	1	■■■■■	8	MemberServiceImpl	SPRING_BEAN	ApiWAS1	BACKEND-API

# Galileo

GoJS evaluation  
(c) 1996-2015 Northwoods Software  
Not for distribution or production use.  
[www.northwoods.com](http://www.northwoods.com)



# Config Center

# Code Review

reviewboard.56qq.com/r/

Review Board 2.5.7

My Dashboard New Review Request All Review Requests Users Groups

ALL REVIEW REQUESTS Hide closed

Summary	Submitter	Posted	Last Updated
Submitted: refactor: move the codes about "S1002" to dedicate package ".biz.s1002"	caixu	一月 6th, 2017, 12:49 p.m.	1 day, 1 hour ago
Submitted: feat(server): 客户端优化图片加载。	caixu	一月 6th, 2017, 9:13 p.m.	1 day, 4 hours ago
Submitted: test(ut): add unit test support and rename previous as integration-test	qintao	一月 6th, 2017, 10:35 a.m.	1 day, 4 hours ago
Submitted: feat(audit): display special lines in register audit view	zhang1wei	一月 6th, 2017, 6:59 p.m.	1 day, 5 hours ago
管理端自己生成合同	wangzhenquan	一月 6th, 2017, 9:32 a.m.	1 day, 5 hours ago
Submitted: feat(opt):re use redis connection in one request context	luofan	十二月 22nd, 2016, 9:26 p.m.	1 day, 5 hours ago
Submitted: feat(opt):re use redis connection in one request context	luofan	十二月 22nd, 2016, 9:28 p.m.	1 day, 5 hours ago
Submitted: feat(core):[TAPD-1008446] updated	luofan	十二月 26th, 2016, 5:36 p.m.	1 day, 5 hours ago
Submitted: refactor: refactor the exception log.	caixu	一月 5th, 2017, 5:37 p.m.	1 day, 17 hours ago
Submitted: refactor[server,core]:1008476~1008478 consignor register function refactor for AP#?optimize register k2 consignor function#?繁杂	jiangpeirong	十二月 26th, 2016, 10:57 p.m.	1 day, 18 hours ago
Submitted: feat[server,core]: TAPD-1008365 ,register k2 consignor function	jiangpeirong	十二月 26th, 2016, 9:13 p.m.	1 day, 18 hours ago
Submitted: refactor[core,server]: TAPD-1008790,refactor common user register function	jiangpeirong	一月 5th, 2017, 5:37 p.m.	1 day, 18 hours ago
Submitted: refactor[core,server]: TAPD-1008790,refactor common user register Function	jiangpeirong	一月 5th, 2017, 7:56 p.m.	1 day, 18 hours ago
Submitted: feat[core,server]: TAPD-1008792,add a query consignor base-info Interface for app [AP]	jiangpeirong	一月 4th, 2017, 6:17 p.m.	1 day, 18 hours ago
Submitted: feat: add new roster files for H1004 and put the dev test uids to whitelist	caixu	一月 4th, 2017, 8:01 p.m.	1 day, 19 hours ago
Submitted: feat(server): add a mobile white list for test.	caixu	一月 5th, 2017, 10:33 a.m.	1 day, 19 hours ago
Submitted: feat: TAP-1008827 优酷SSO touch session 接口的网络流量	caixu	一月 5th, 2017, 1:37 p.m.	1 day, 19 hours ago
Submitted: feat(RewardLogController): add API to query reward log	qintao	一月 5th, 2017, 5:10 p.m.	1 day, 20 hours ago

1 2 3 4 > » 7 pages

# Testing

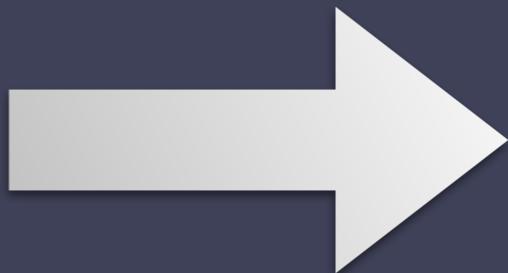
# Hard things are hard

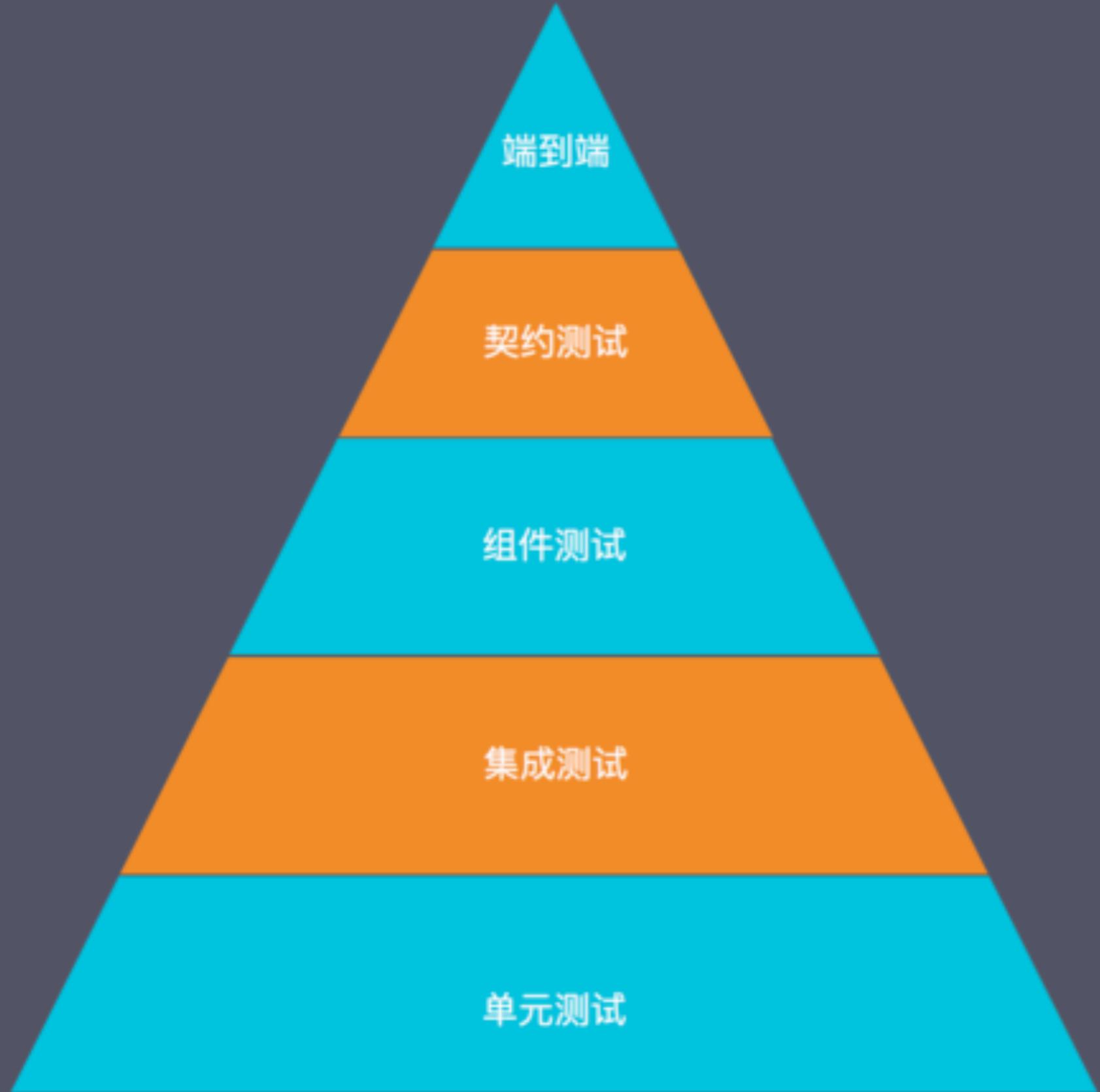
Asynchrony

Complexity

Partial Failure

Large State  
Space





# **Simple Testing Can Prevent Most Critical Failures: An Analysis of Production Failures in Distributed Data-Intensive Systems**

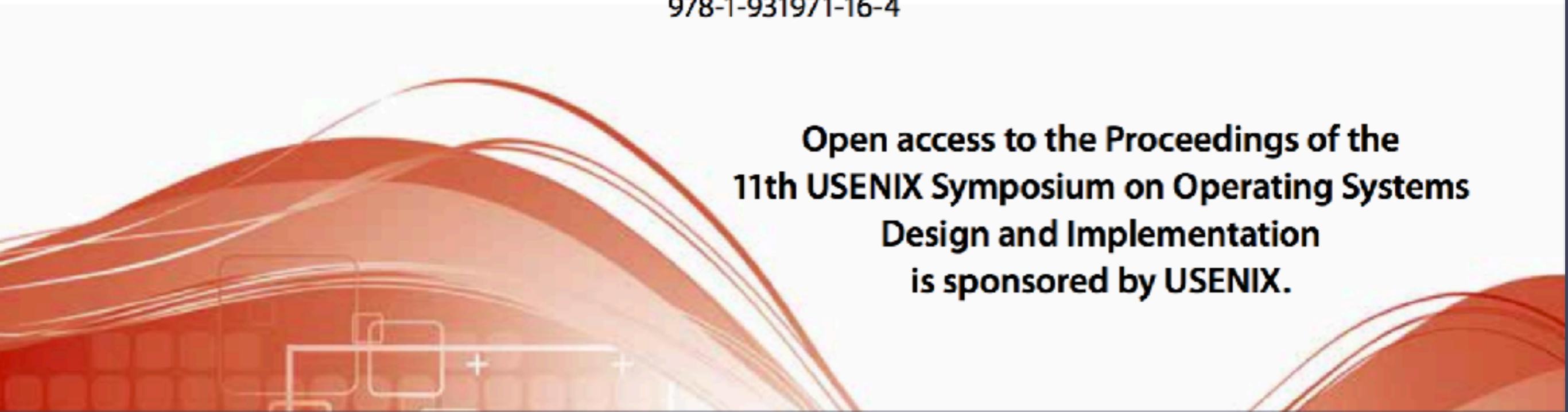
**Ding Yuan, Yu Luo, Xin Zhuang, Guilherme Renna Rodrigues, Xu Zhao,  
Yongle Zhang, Pranay U. Jain, and Michael Stumm, University of Toronto**

<https://www.usenix.org/conference/osdi14/technical-sessions/presentation/yuan>

**This paper is included in the Proceedings of the  
11th USENIX Symposium on  
Operating Systems Design and Implementation.**

**October 6–8, 2014 • Broomfield, CO**

**978-1-931971-16-4**



**Open access to the Proceedings of the  
11th USENIX Symposium on Operating Systems  
Design and Implementation  
is sponsored by USENIX.**

# Advanced Quality Assurance

- Formal Verification
- Property-based Testing
- Canaries
- Fault Injection
- Game Days

“

*A distributed system is one in which the failure of a computer you didn't even know existed can render your own computer unusable.*

-- Leslie Lamport

# 一些心得

- 流程也要考虑实际业务场景设定
- 引导人关注错误指标的就是坏流程
- 对经过检验的最佳实践要认真执行
- 流程不是银弹，还是人最重要

技术！

“

*Technology is the answer, but what was the  
QUESTION*

-- Cedric Price (1965)

..... 1974 ..... 1975 ..... 1976 ..... 1984 .....



RFC 674



RFC 684



RFC 707



Implementing  
Remote  
Procedure Calls

..... 1988 ..... 1989 ..... 1991 ..... 1994 ..... Present .....



A Critique of the  
Remote  
Procedure Call



RFC 1094  
NFS



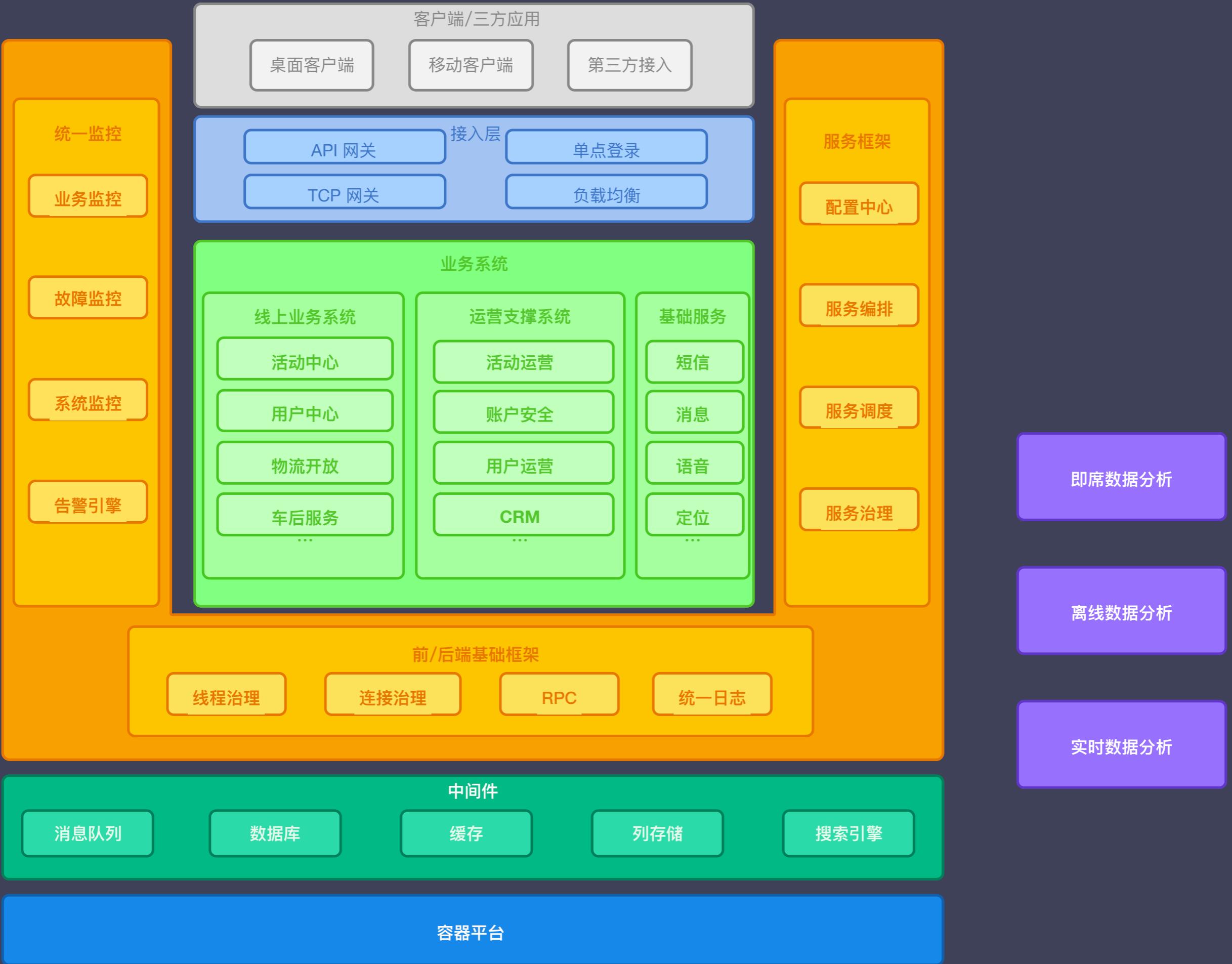
CORBA



A Note on  
Distributed  
Computing

Future .....

不仅仅是选RPC框架



# 隔离

- separation
- isolation
- insulation

# 隔离

- 数据中心隔离
- 物理节点隔离
- 容器隔离
- 进程与线程的隔离
- 依赖隔离

# 技术选型

- 架构演进要基于业务和团队的演进周期
- 技术选型的主要目标是降低复杂度
- 不轻易自己造轮子：维护成本很高
- 举棋不定时选最无聊的技术

# IPC Protocol

- REST API
- RPC

# Service Configuration Discovery

- ZooKeeper
- Eureka
- Etcd
- Consul

# 可以深入的话题

- 可度量的持续交付
- 可度量的持续测试
- 可度量的物理资源
- 可度量的服务性能
- 可度量的业务指标
- 统一的监控与告警
- 前端的工程一体化
- 隔离、定位与恢复
- 路由、限流与分流
- 管理、编排与扩容
- 流程代码化、自动化
- 辅助工具的开发

# Q&A?