

金融巨头

2016年：¥100BN 收入

Global Continuous Delivery Journey

A worldwide presence in commercial and retail banking



Disclaimer: ING Bank does not have a commercial banking licence in the U.S. and therefore is not permitted to conduct commercial banking business in the U.S. Through its wholly-owned subsidiary, ING Financial Holdings Corporation (and its affiliates), it offers a full array of wholesale products such as commercial lending, corporate finance and a full range of FM products and services.



ING持续集成

Build - Continuous Integration



Code



Orchestrate



Build



Check



Store

ING持续交付

Deploy - Continuous Deployments



多语言开发

Artifactory

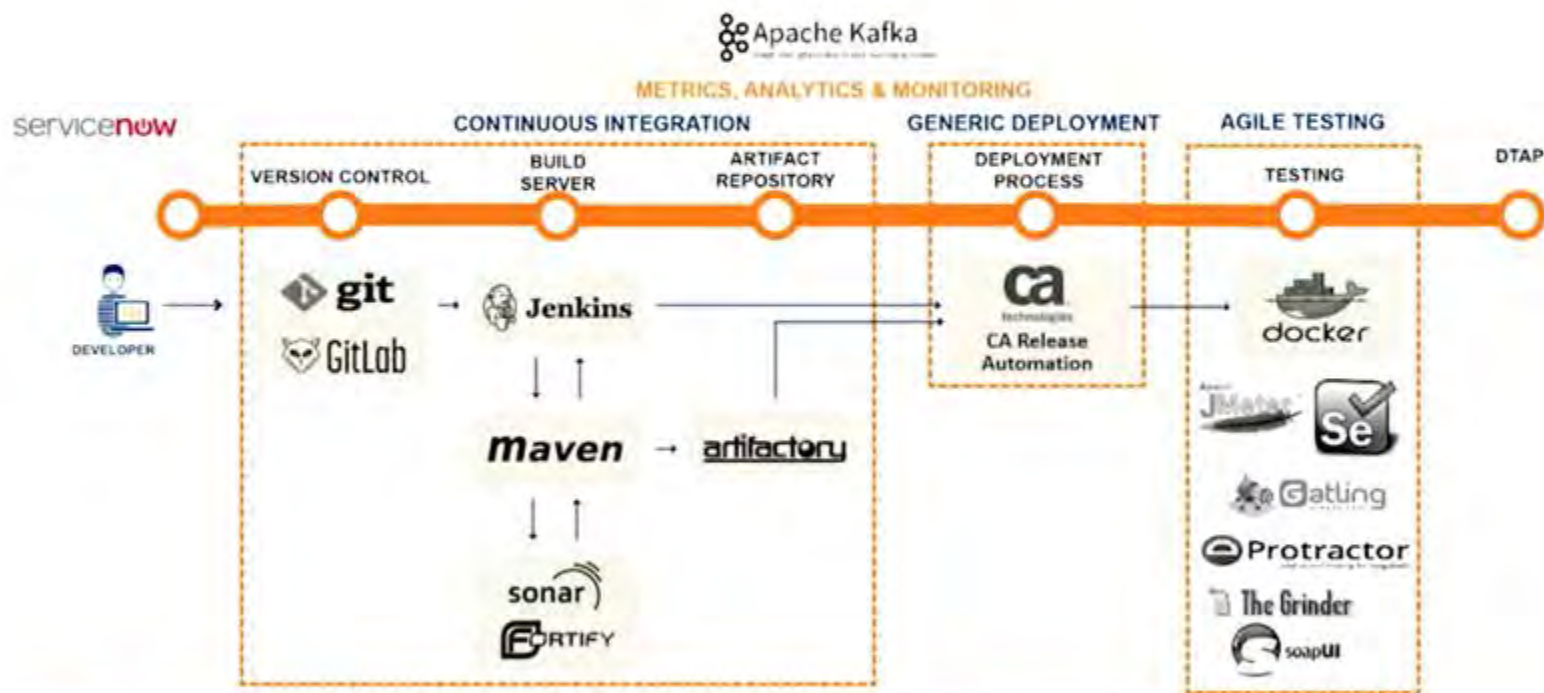
Numbers

#artifacts	3,500,000
#binaries	2,500,000
#local repos	1800
#remote repos	39
#users	4000
#groups	2600
#permissions	900
#storage	11.5 TB
#growth	1TB / month

Artifact types



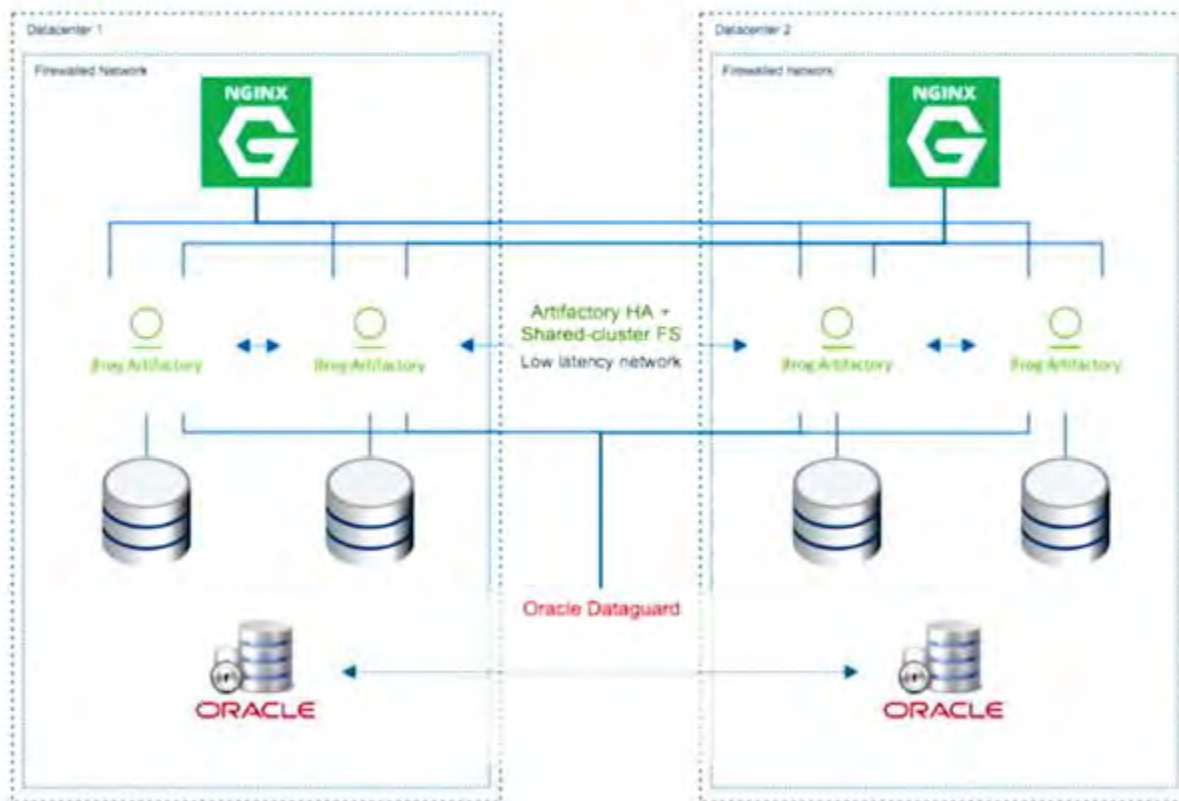
ING自定义流水线



One opinionated flow for all the (≈600) teams

高可用 / 容灾备份

Artifactory HA architecture end state



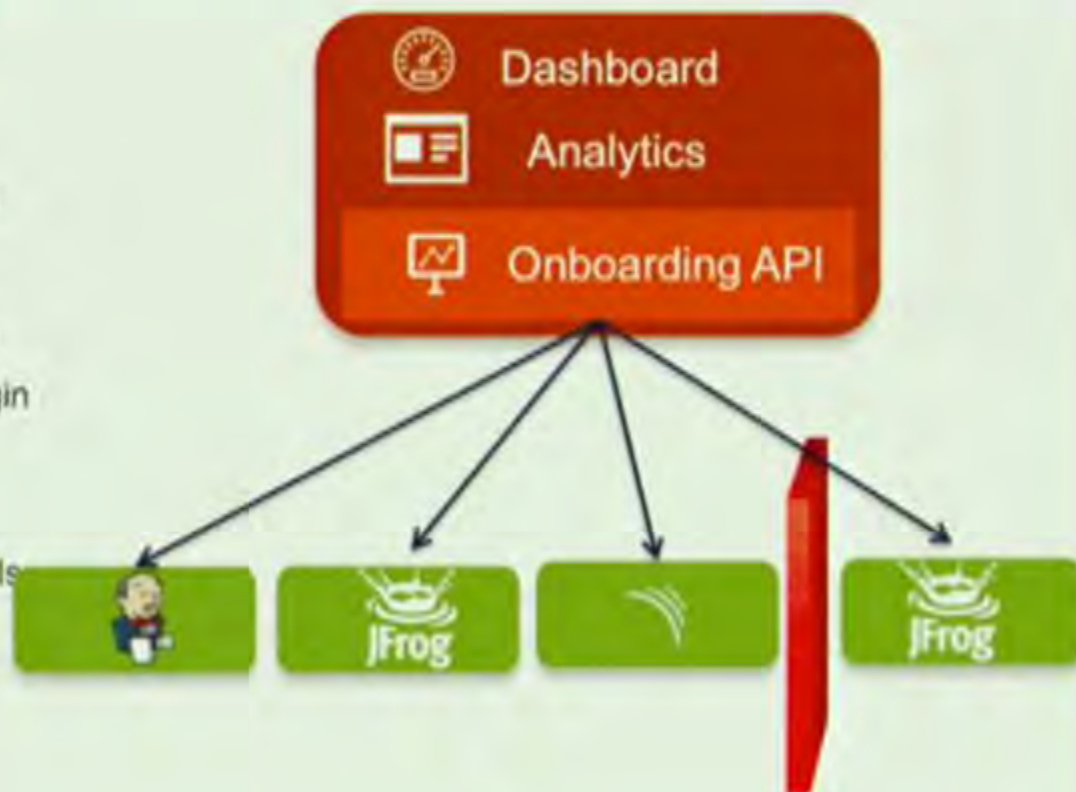


思科 Account Manager

5个人DevOps团队支持4万+ 开发者

Account Management

- Account Orchestration
 - Automated onboarding through APIs
- Extensible plugin system
 - Jenkins, Artifactory, and SonarQube
 - Other systems can be added via plugin development (Artifactory-X)
- Unified dashboard
 - Manage customer identity across tools
- CI-Stats analytics
 - Number of requests
 - Number of uploads/downloads



思科 Account Manager

Account Management

https://account.cisco.com/service/create/123

New Service

Limited fields can be edited at this time. Additional fields will be enabled in the next beta release of the account management service.

Artifactory

Jenkins

SonarQube

Team *

SAST Testing (SAST Testing)

Name *

swampup

Artifactory

Artifactory is a global service that distributes your artifacts all over the world. There is a single hostname that you should deploy to and read from, and it will resolve to the server that is physically closest to you.

Example config for your build management tool:

Maven Yum Docker

```
<code>+repo:artifactory
+repo:artifactory
+id:http://artifactory.com
url:http://repo1.maven.org/maven2/artifactory-group
+repo:artifactory
+repo:artifactory
+repo:artifactory
+id:http://artifactory.com
+id:http://artifactory.com
+id:http://artifactory.com</code>
```

Deployer User *

swampup-deployer

Deployer Password *

swampup-deployer

Users

Hosted

Remote

Virtual

+ Add

Name

思科 Account Manager

Account Management

Audit Trail

Events for Service 16003

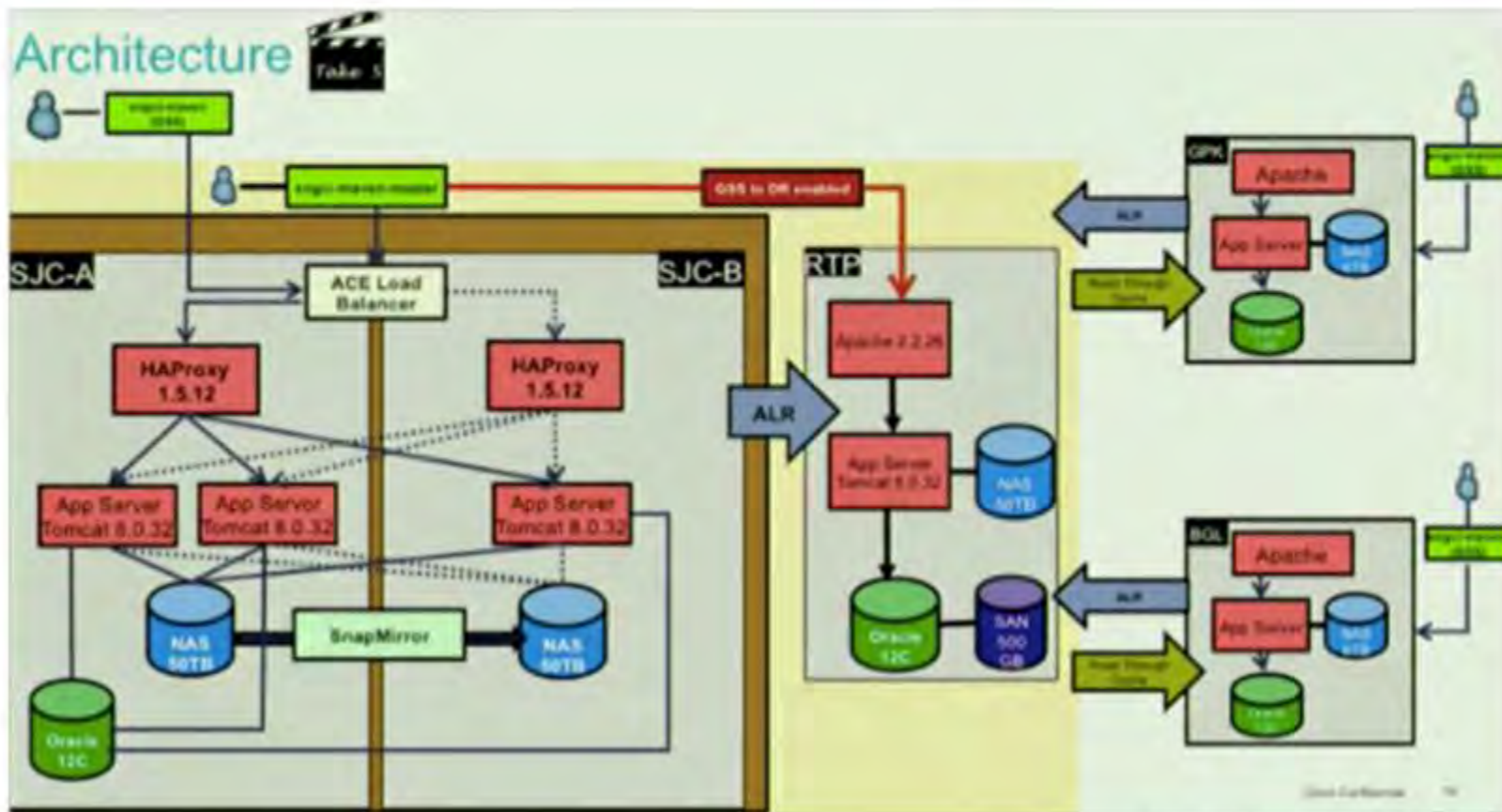


Started By	Status	Started	Completed	Log
------------	--------	---------	-----------	-----

prayyapp	SUCCESS	Apr 12, 2016 7:38:58 PM	Apr 12, 2016 7:41:28 PM	<pre>2016-04-12T18:40:14.129Z TRACE: com.cisco.accountmanagement.helpers.artifactory.RegistrationHandler: Setting replication settings to 'Satellite' for repo [[com-cisco-qa]] 2016-04-12T18:40:17.491Z TRACE: com.cisco.accountmanagement.helpers.artifactory.RegistrationHandler: Setting replication settings to 'Satellite' for repo [[com-cisco-qa]] 2016-04-12T18:40:41.754Z WARN: com.cisco.accountmanagement.helpers.artifactory.ArtifactoryHandler: Creating [[freq-dm]] on Satellite (http://repo-001.cisco.com/artifactory) 2016-04-12T18:40:41.894Z TRACE: com.cisco.accountmanagement.helpers.artifactory.SecurityHandler: Creating regular user [[freq-dm-deployer]] 2016-04-12T18:40:42.991Z TRACE: com.cisco.accountmanagement.helpers.artifactory.SecurityHandler: Creating group [[freq-dm-group]] 2016-04-12T18:40:43.340Z TRACE: com.cisco.accountmanagement.helpers.artifactory.SecurityHandler: Adding users [admin, jbaroni] to group [[freq-dm-deployer]] 2016-04-12T18:40:43.353Z TRACE: com.cisco.accountmanagement.helpers.artifactory.SecurityHandler: Setting permission target [[freq-dm-rights]] 2016-04-12T18:40:43.571Z TRACE: com.cisco.accountmanagement.helpers.artifactory.strategies.NormalStrategy: Creating local Maven repo [[freq-dm-local]] 2016-04-12T18:40:44.714Z TRACE: com.cisco.accountmanagement.helpers.artifactory.strategies.NormalStrategy: Creating local Maven repo [[freq-dm-qa]] 2016-04-12T18:40:42.148Z TRACE: com.cisco.accountmanagement.helpers.artifactory.strategies.NormalStrategy: Creating virtual Maven repo [[freq-dm-g]] 2016-04-12T18:40:50.290Z TRACE: com.cisco.accountmanagement.helpers.artifactory.strategies.NormalStrategy: Creating virtual Maven repo [[freq-dm-group-maven]] 2016-04-12T18:41:07.816Z TRACE: com.cisco.accountmanagement.helpers.artifactory.SecurityHandler: Adding repo [[freq-dm-external], [[freq-dm-group], [[freq-dm-qa]]] to production target [[freq-dm-rights]] 2016-04-12T18:41:08.001Z TRACE: com.cisco.accountmanagement.helpers.artifactory.RegistrationHandler: Setting replication settings to 'Satellite' for repo [[com-cisco-qa]] 2016-04-12T18:41:09.104Z TRACE: com.cisco.accountmanagement.helpers.artifactory.RegistrationHandler: Setting replication settings to 'Satellite' for repo [[com-cisco-qa]] 2016-04-12T18:41:28.054Z WARN: com.cisco.accountmanagement.providers.helper.ProviderHandler: CMDBS of com.cisco.accountmanagement.helpers.artifactory.Registry</pre>
----------	---------	-------------------------	-------------------------	--

思科异地DevOps

Duration (Last 30 Days) -

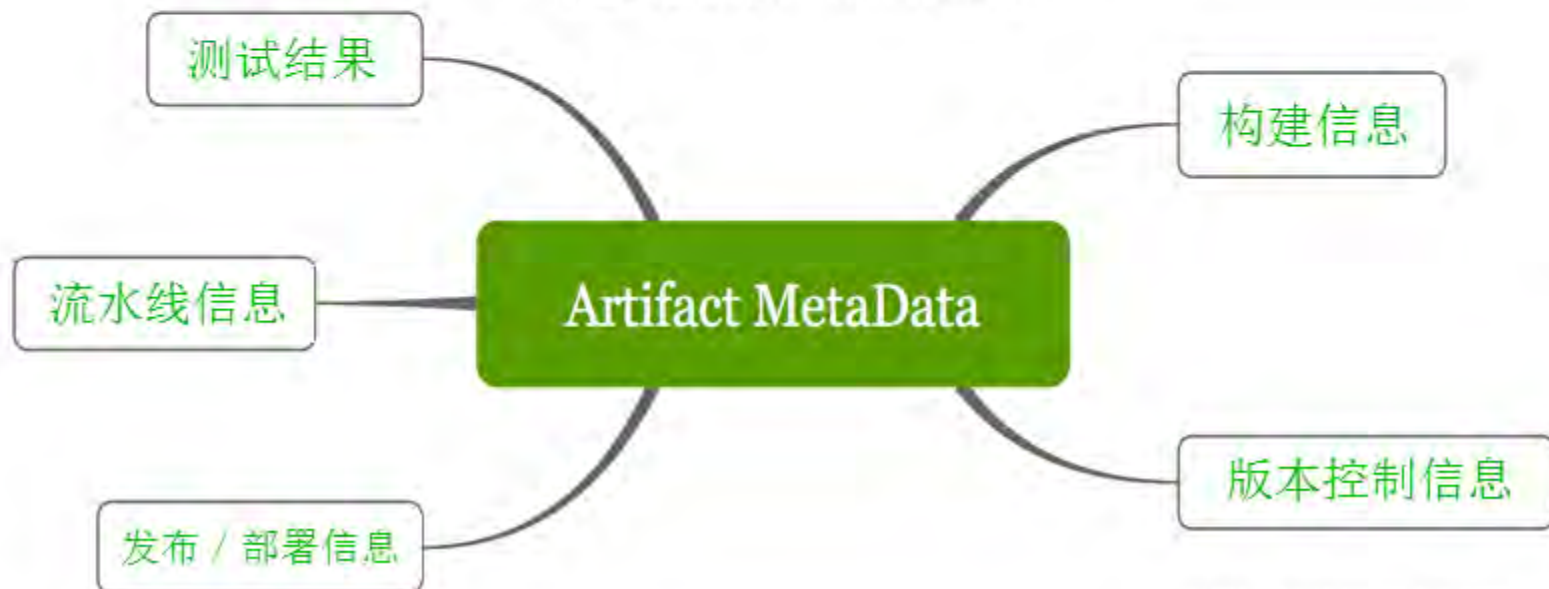


思科元数据

思科通过一个命令行能找到上亿个包里面的目标包

JFrog Artifactory制品库 + AQL查询语言

- 所有的软件包加上多样化DevOps各阶段的信息
- 根据Metadata的信息来实现自动化的快速发布
 - 提高发布产品的自信



思科元数据

Jfrog Artifactory Welcome, admin Help

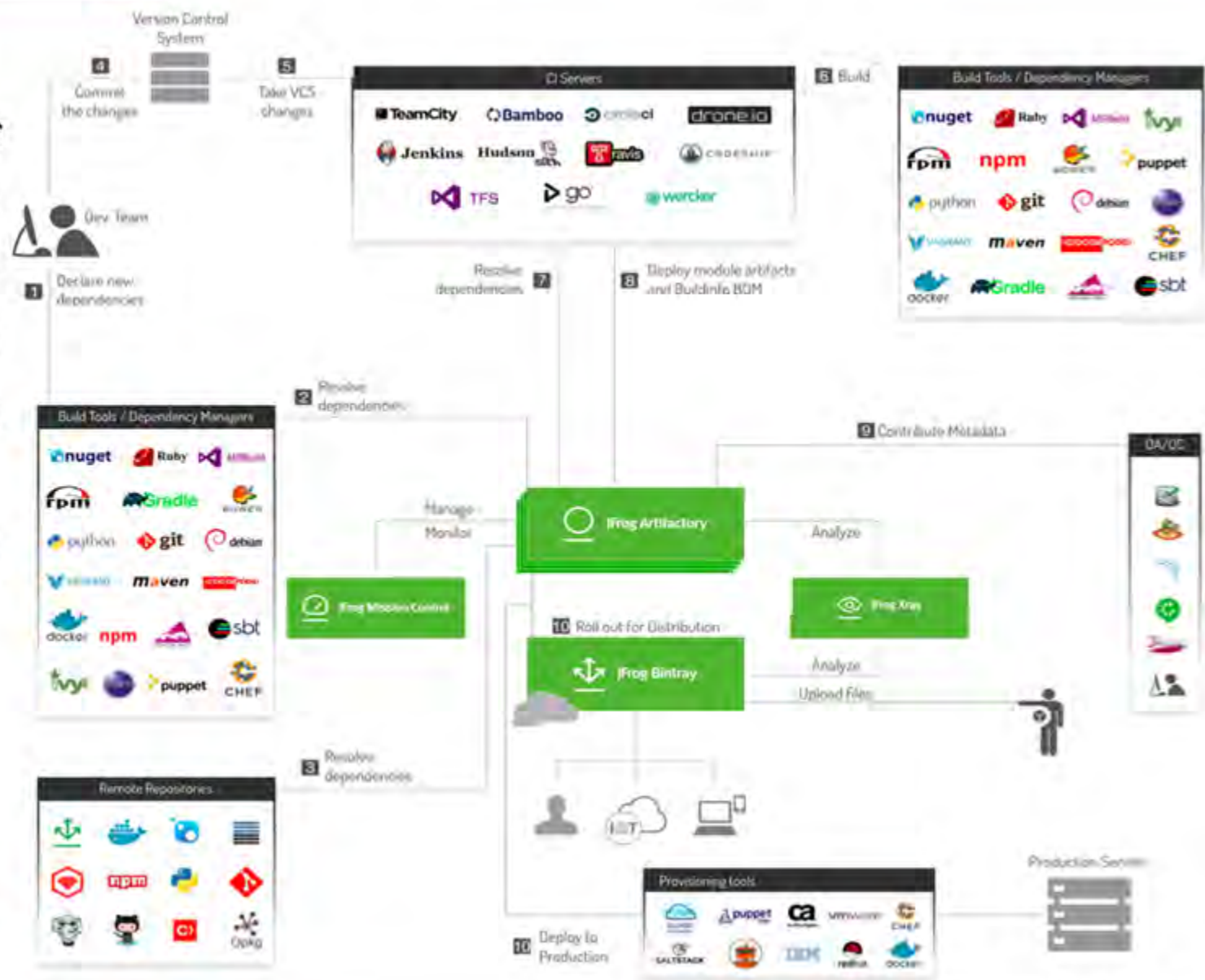
Artifact Repository Browser

multi3-6.238.war Download Actions

General	Effective Permissions	Properties	Watchers
Property		Value(s)	
qa		done	
passRate		1	
platforms		osx win386 win64 debian	
vcs.revision		251ab9ba9bea19fd4aa46622b43c8fe2da155df17	
qaType		unitTest,sonar	
testType		JUnit	
build.timestamp		1502195316210	
build.name		mvn-pro-webinar	
build.number		80	
chef.env		TestEnv1,ProdEnv1	
env		TESTING	
buildStatus		RC	
sonarUrl		http://47.93.114.82:9000/dashboard/index/jfrog:multi3	
sonarIssue		2	



- 多语言仓库
- 统一CI/CD平台
- 收集测试结果
- 设置质量关卡
- 包升级, 移动
- 统计数据, 进行评估
- 高可用, 容灾备份



JFrog DevOps



互联网 & 软件

Google Tencent 腾讯

vmware 

ORACLE



YAHOO!

JD 京东 .COM

4,000
客户

科技 & 电信

 HUAWEI

Linked in

 intel  snapo

 CISCO

 CHEF



ERICSSON 

银行 & 金融

 BARCLAYS

Bloomberg

 JPMorganCha

+100
New Logos
a month

FIS

CREDIT SUISSE 

WELLS FARGO

ING DIRECT 

工程 & 航空

ANSYS



NASA

 AIRBUS

BOS

ING



HALLIBURTON

Raytheon

零售 & 消费

COSTCO WHOLESALE

NETFLIX

GAP



95%
续费率

★macy's

TESLA

RIOT GAMES

教育 & 研究

Mit

W UNIVERSITY OF WASHINGTON

Yale UNIVERSITY

OHIO UNIVERSITY

California University of Diego

Stanford University



DevOpsOne

全开源自定义流水线



Jenkins



docker

JUnit



JFrog
杰蛙

maven



APACHE
JMeter™

sonarqube



ANSIBLE



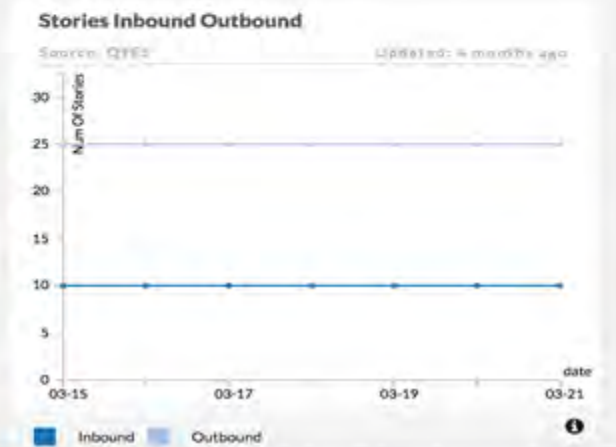
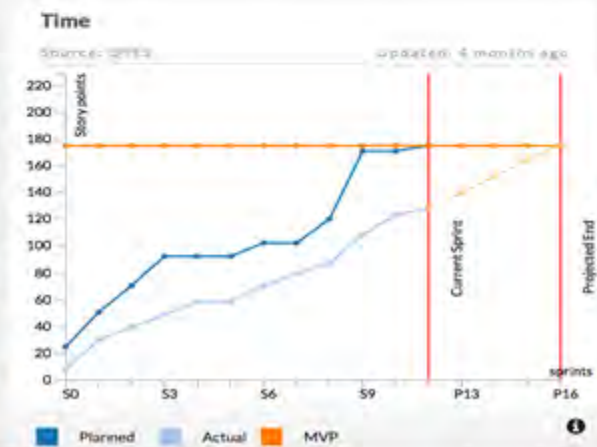
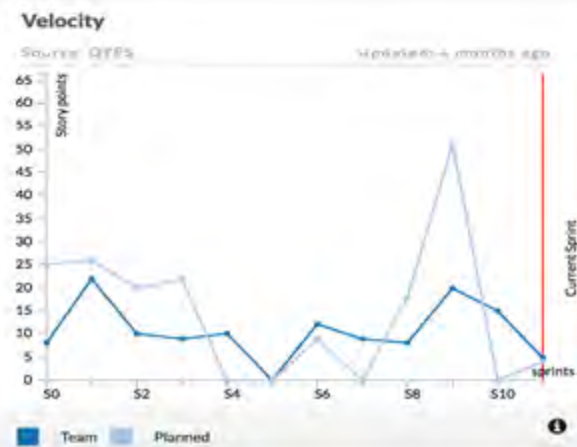
springboot-demo





Sonarcube | Issues | History | Log | Dashboard

MyApp | 2017-03-21 | Normal



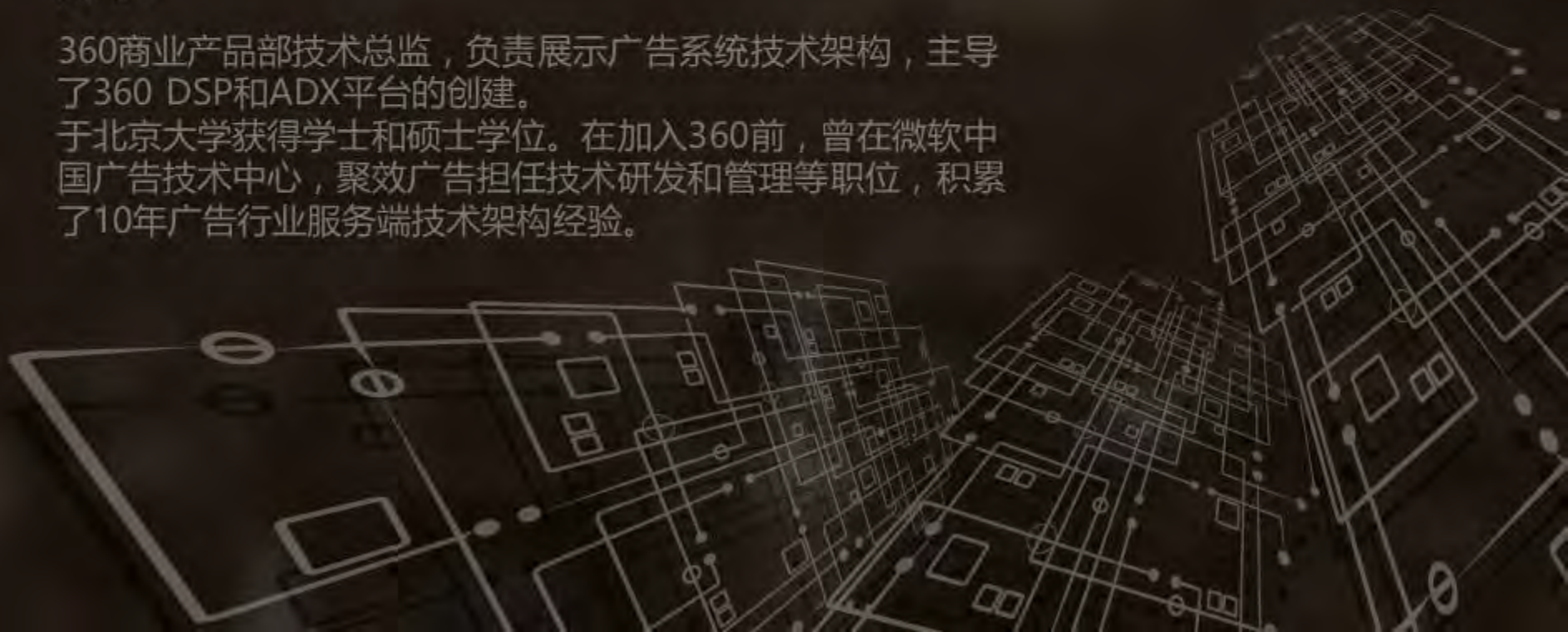
2017 Software Architecture Summit

360展示广告系统技术架构实践

陈东

360商业产品部技术总监，负责展示广告系统技术架构，主导了360 DSP和ADX平台的创建。

于北京大学获得学士和硕士学位。在加入360前，曾在微软中国广告技术中心，聚效广告担任技术研发和管理等职位，积累了10年广告行业服务端技术架构经验。



| 内容摘要

360展示广告系统介绍

系统规模扩大的挑战

高可用性管理

总结



中国程序化广告技术生态图

China Programmatic Advertising Technology Landscape

© 2017
010-59121111

RTBChina

MARKETER
广告主

PUBLISHER
媒体
CONSUMER
消费者

Super Platforms | 综合大型投放平台



DSP & DSPAN (DSP + Ad Network) | 程序化广告采购方

Ad Exchange & SSP | 程序化广告供应方

Trading Desk & Tech

采购交易平台及技术

Verification

广告验证

Measurement & Analytics

监测分析工具

Data Supplier & Data Management

数据提供和数据管理

Programmatic Creative

程序化创意

Programmatic TV | 程序化电视广告

Programmatic DOOH | 程序化数字户外广告

| 360展示广告系统介绍

335亿日均流量 | 8大定向模式 | 80万标签总量 | 1000万投放设置



360站内独家资源

PC



360导航



360搜索

站外优质合作媒体



爱奇艺

精选数百家主流媒体

移动



国内数千款TOP应用

PC

285亿

+

移动

50亿

=

一站投放

335+亿

每秒广告请求 > 40万

平均后端延迟 < 10 MS

| 360展示广告系统介绍

- 技术栈

业务服务

Zoo keeper

thrift

Nginx

QConf

Kafka

Storm

Flink

aerospike

Redis

数据分析

Hive/Pig

Spark SQL

Druid

mxnet

Map reduce

spark

Elastic Search

tensorflow

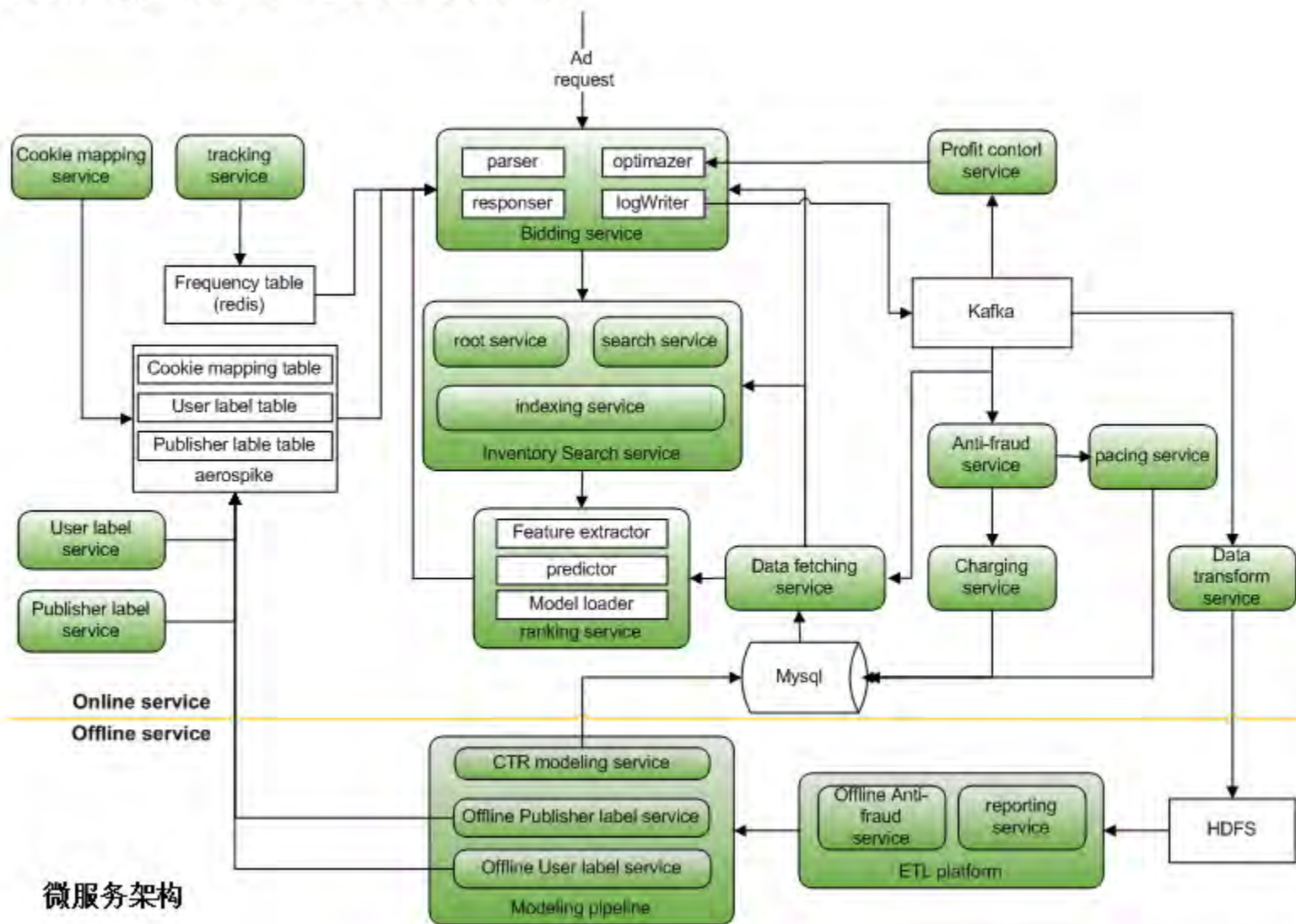
底层存储

HDFS

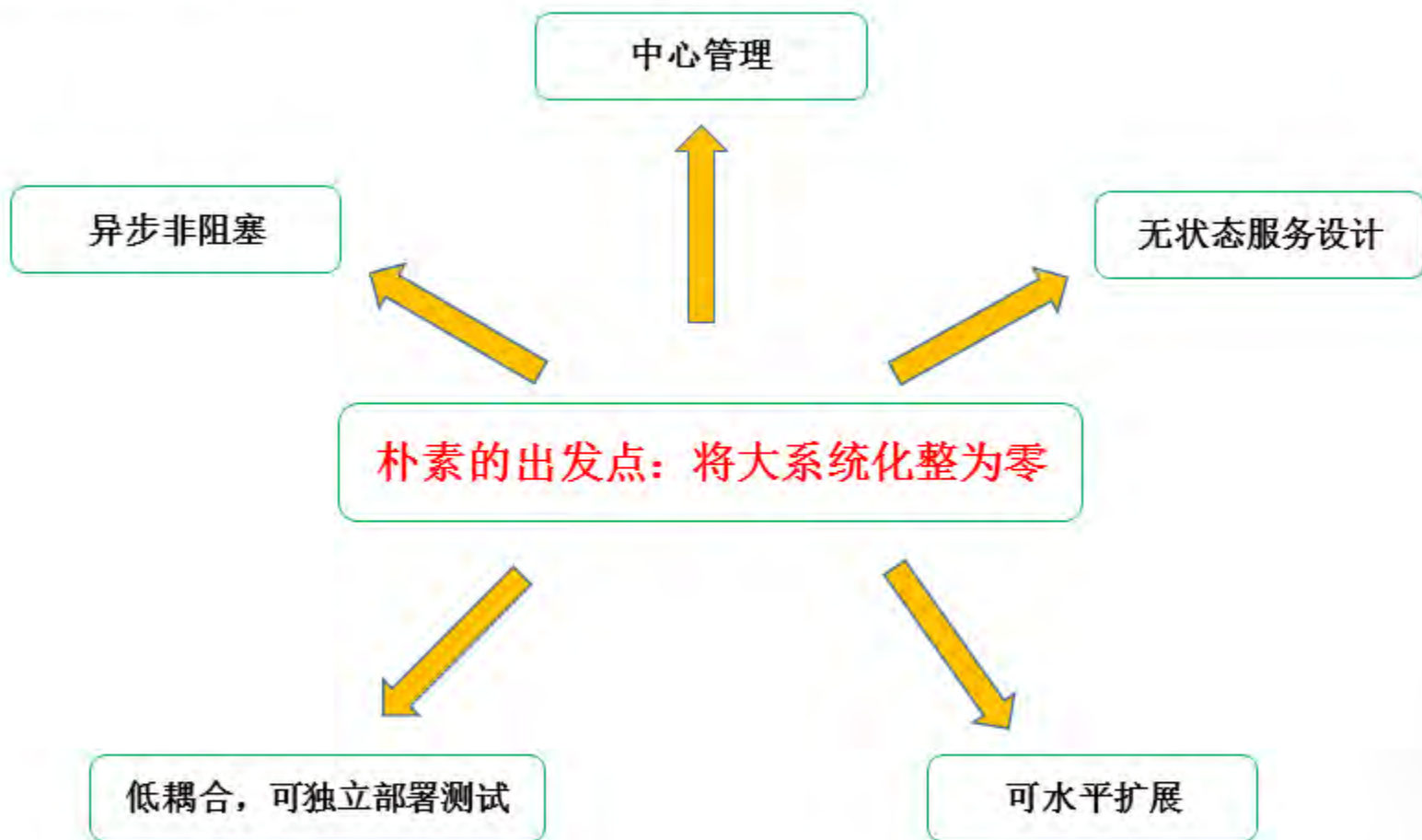
HBASE

MYSQL

360点睛实效平台架构介绍

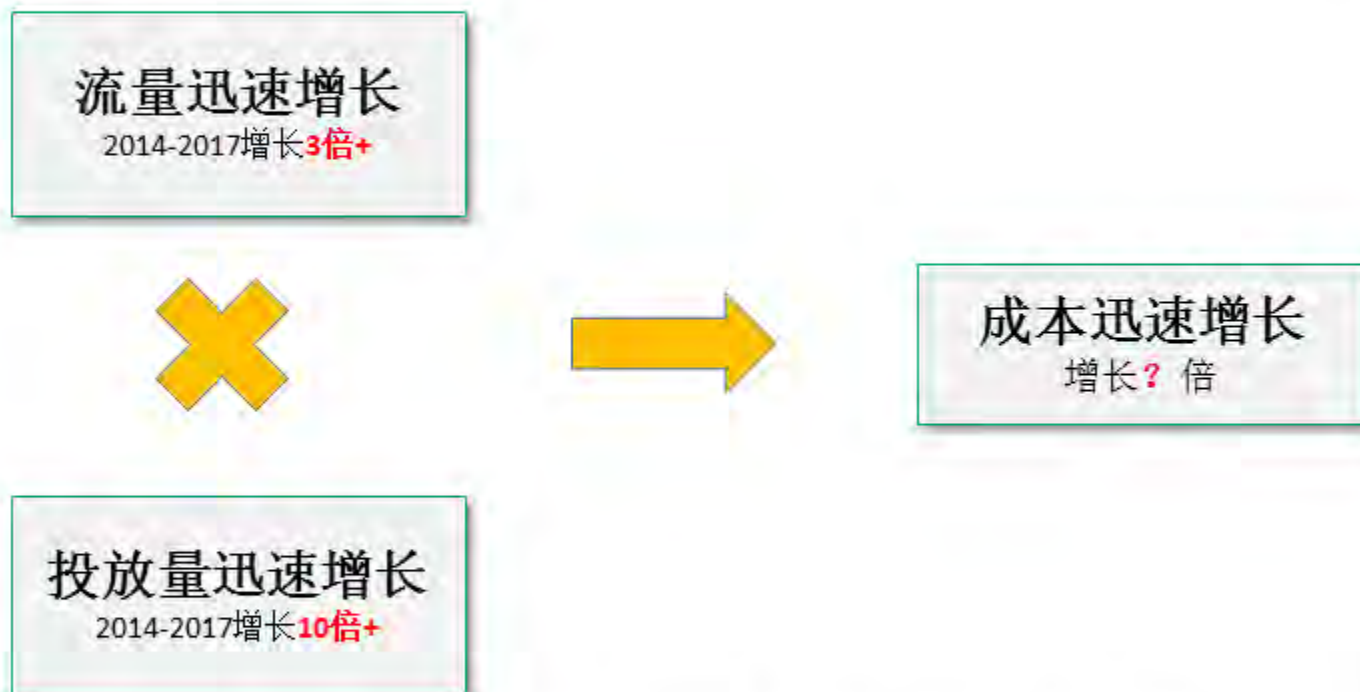


微服务架构



规模扩大的挑战

在40万的qps压力下，要在10ms内，从千万个广告投放中挑选出合适的广告



简单粗暴加机器会带来成本迅速增长需要更高效的解决方案

I 规模扩大的挑战-合理的拆分

- 合理拆分是分布式的核心
 - 业务拆分
 - 水平拆分
 - 垂直拆分
- 架构设计的重点：基于业务特点给出合适的设计方案

业务拆分+垂直拆分

VS

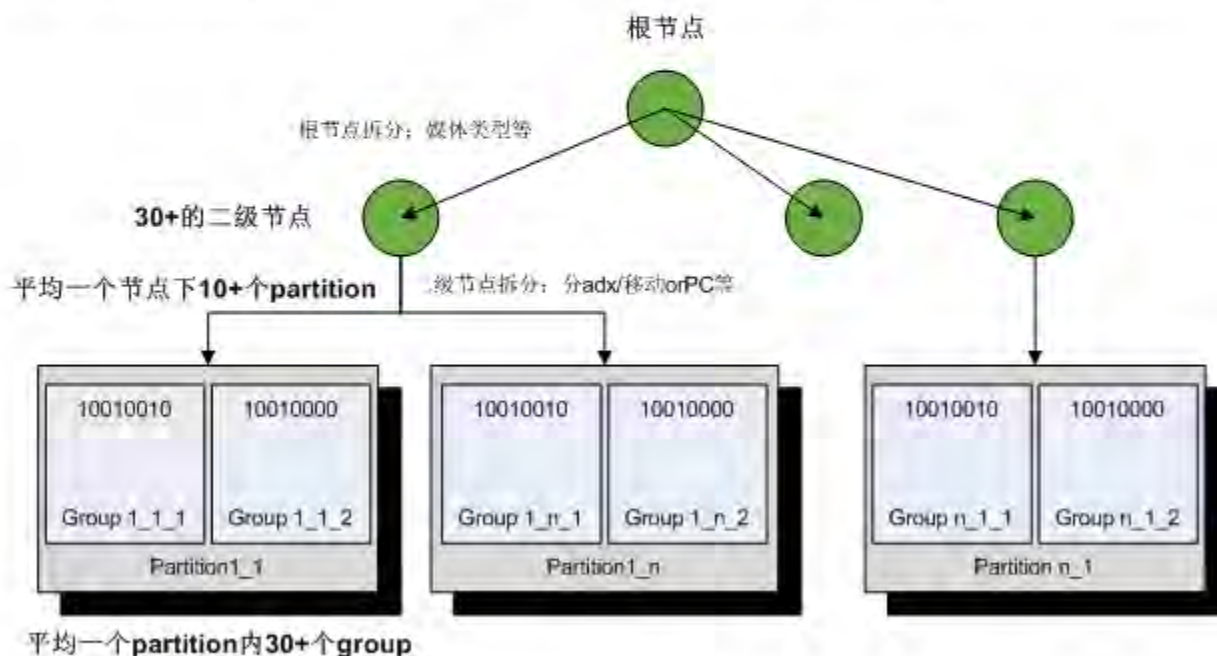
水平拆分

| 规模扩大的挑战-合理的拆分

- 展示广告业务的特点：
 - 广告投放指定的部分投放条件和媒体流量直接相关。
 - 例子1：投放指定了媒体类型（新闻网站/视频网站）等
 - 例子2：投放指定了媒体渠道（google adx/360MAX adx）
 - 例子3：投放指定了PC还是移动流量
- 展示广告业务的拆分设计思路：
 - 根据流量特点，将投放索引进行拆分
 - 当特定流量发起广告请求，能直接走到对应的索引分支进行检索，而不是在全量投放数据上进行检索

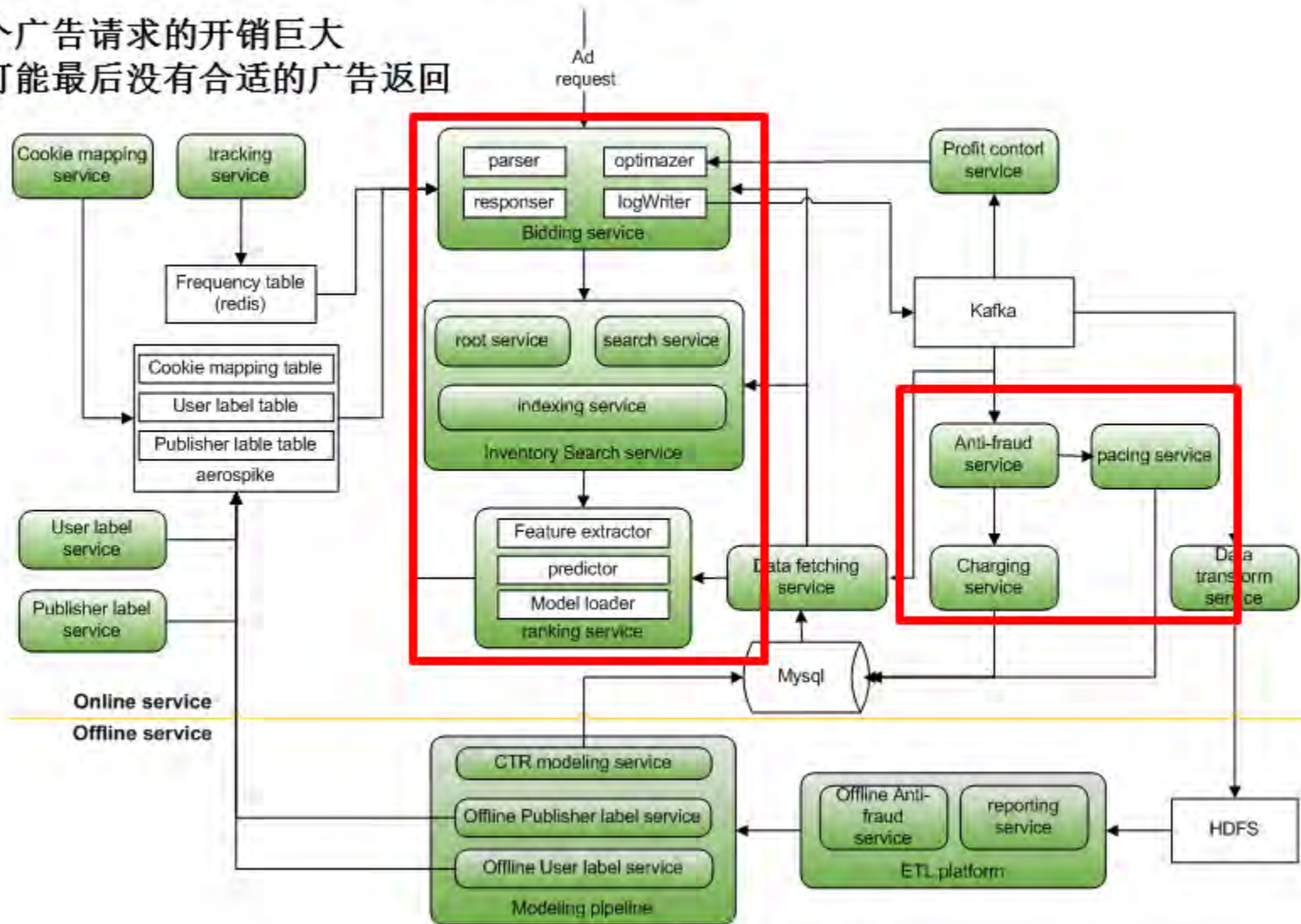
规模扩大的挑战-合理的拆分

- 展示广告投放拆分方案：
 - 根据投放和流量的关系，基于区分度做partition
 - 允许同一个投放出现在不同partition内
 - Partition内，根据广告的定向模式分group，group之间的投放不重复
 - 每个group按定向模式算出自己的bitmap（8bit代表8种定向模式）



规模扩大的挑战-合理的分层处理

一个广告请求的开销巨大
但可能最后没有合适的广告返回



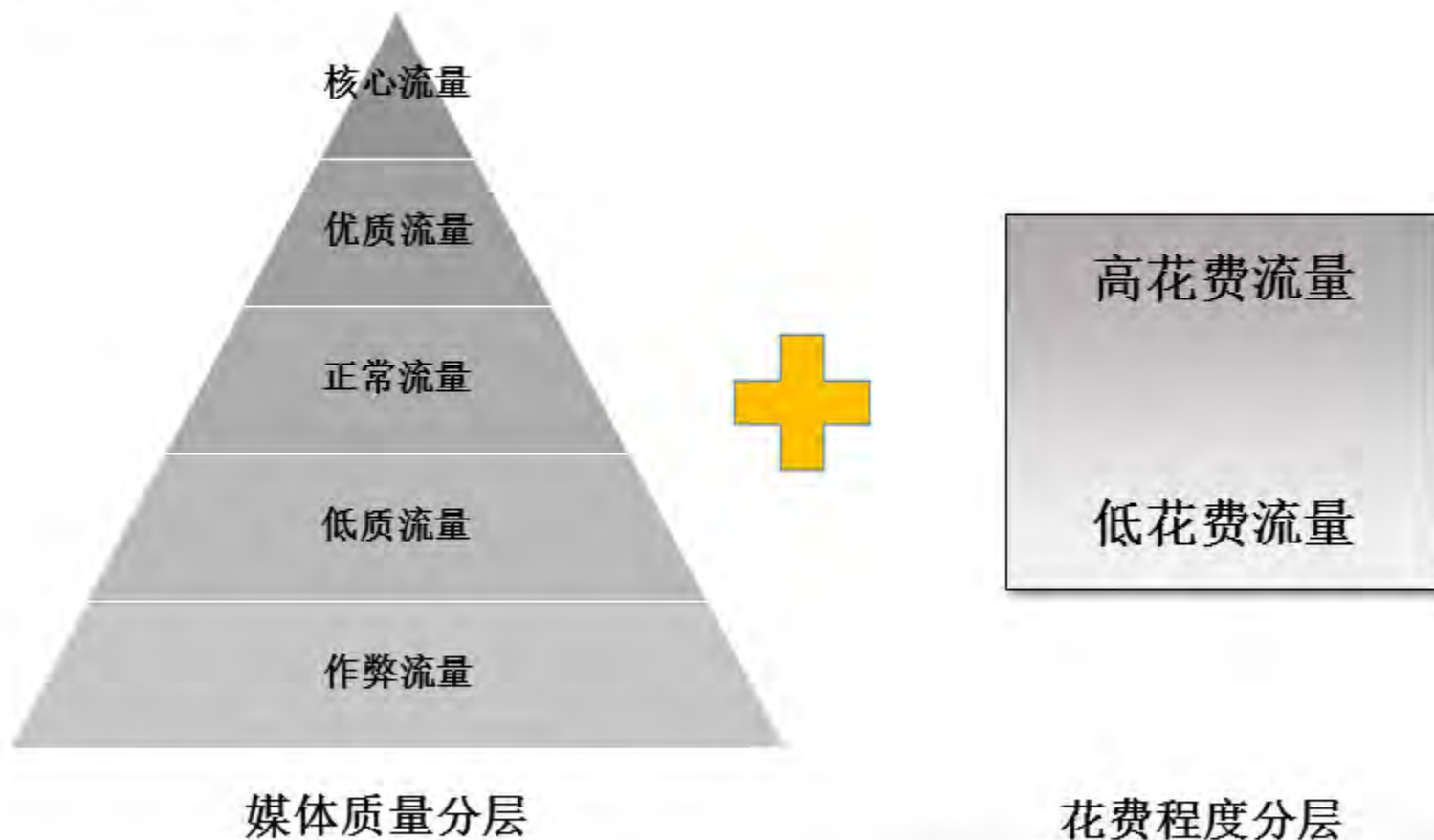
| 规模扩大的挑战-合理的分层处理

- 不是所有的流量都有一样的价值
- 不是所有的投放都有一样的价值
- 区分对待，不搞平均主义



规模扩大的挑战-合理的分层处理

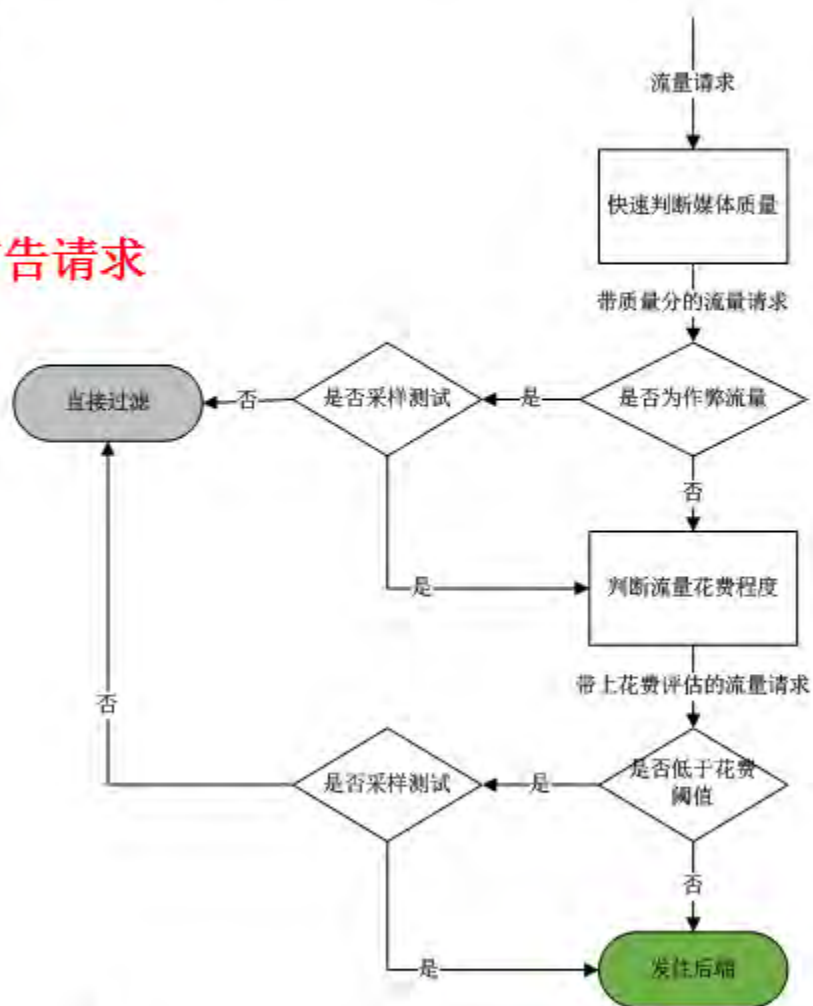
- 流量分层处理



规模扩大的挑战-合理的分层处理

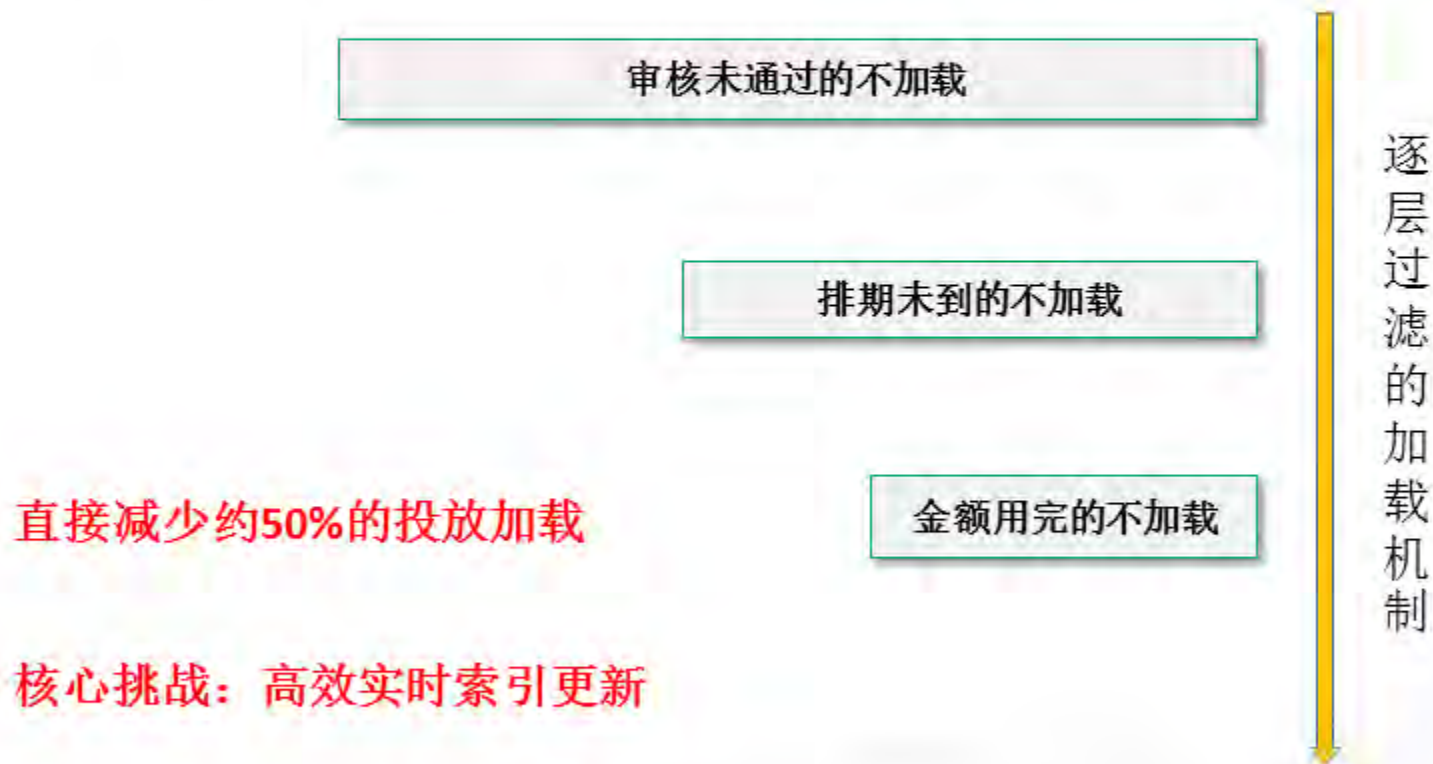
- 流量分层处理流程

直接减少约40%的广告请求
收入影响<3%



规模扩大的挑战-合理的分层处理

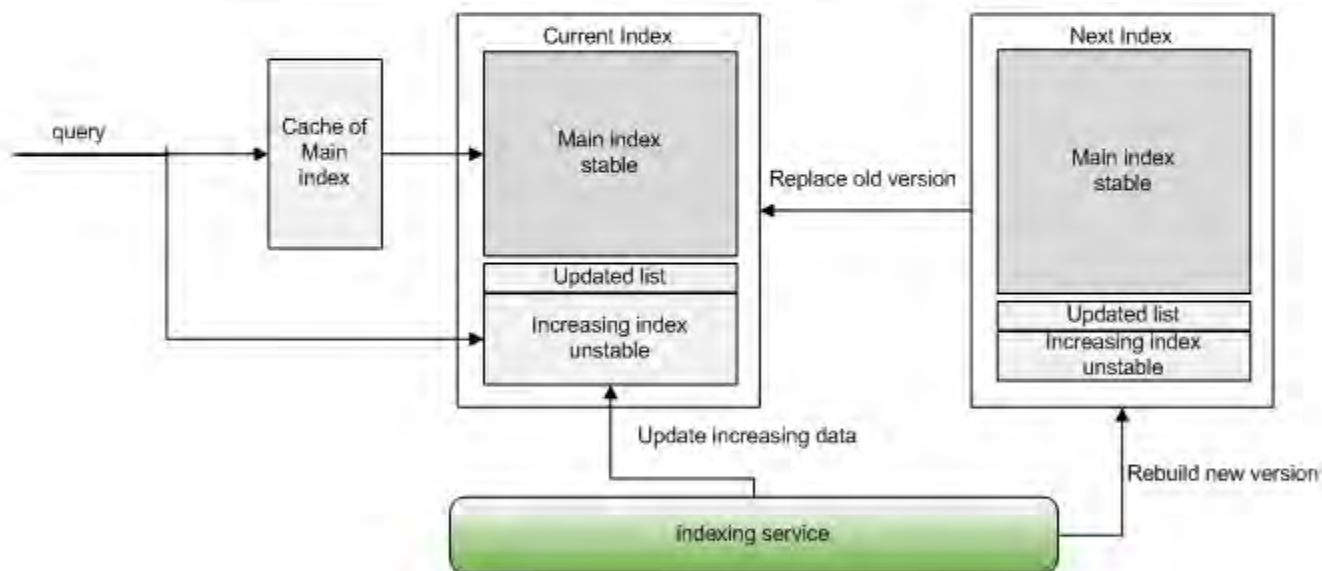
- 投放分层处理
 - 和搜索引擎不同，广告投放检索更看重实时性，不需要检索历史投放



规模扩大的挑战-合理的分层处理

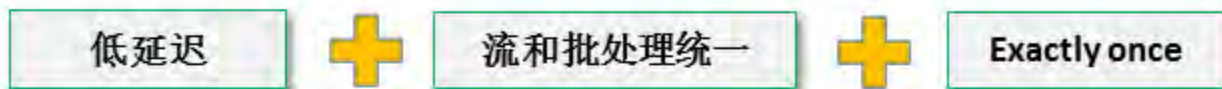
- 实时投放索引更新

- 索引分层：主索引 + 增量索引
- 检索逻辑：主索引结果 - 更新列表 + 增量索引结果



规模扩大的挑战-实时处理

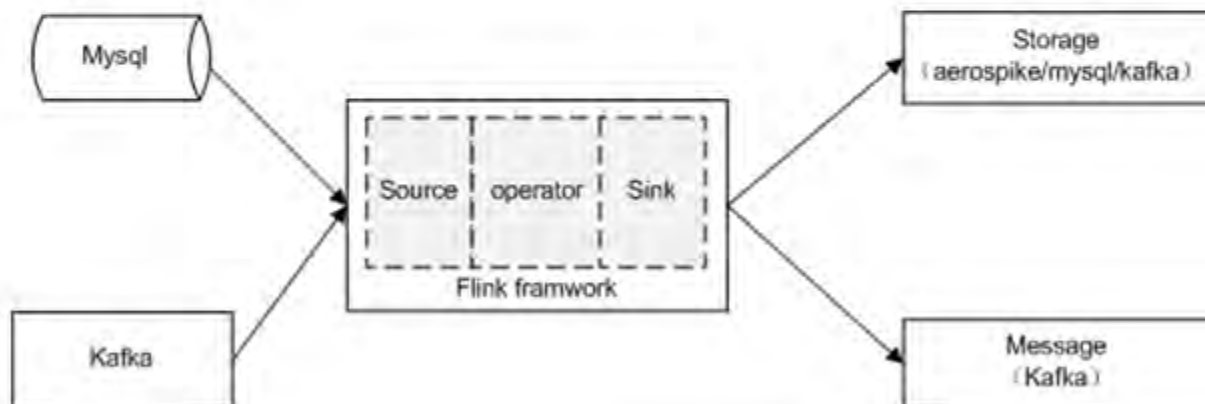
- 需求：



- 之前系统：

- Hadoop + storm + spark streaming

- 现在的方案：



| 高可用性管理

- 核心思路
 - 高可用性要在架构设计时就考虑进来
 - 基于微服务的架构，核心服务要保证自身高可用，做好隔离
 - 贯彻分层处理理念
 - 对于大型分布式系统，需要提供分布式debug能力和监控管理
- 方案：
 - 降级服务
 - 分布式debug系统
 - 分布式监控管理系统

| 高可用性管理-降级服务

- 自动降级 + 人工干预



高可用性管理-分布式debug系统

- 设计要点：

低侵入的debug lib

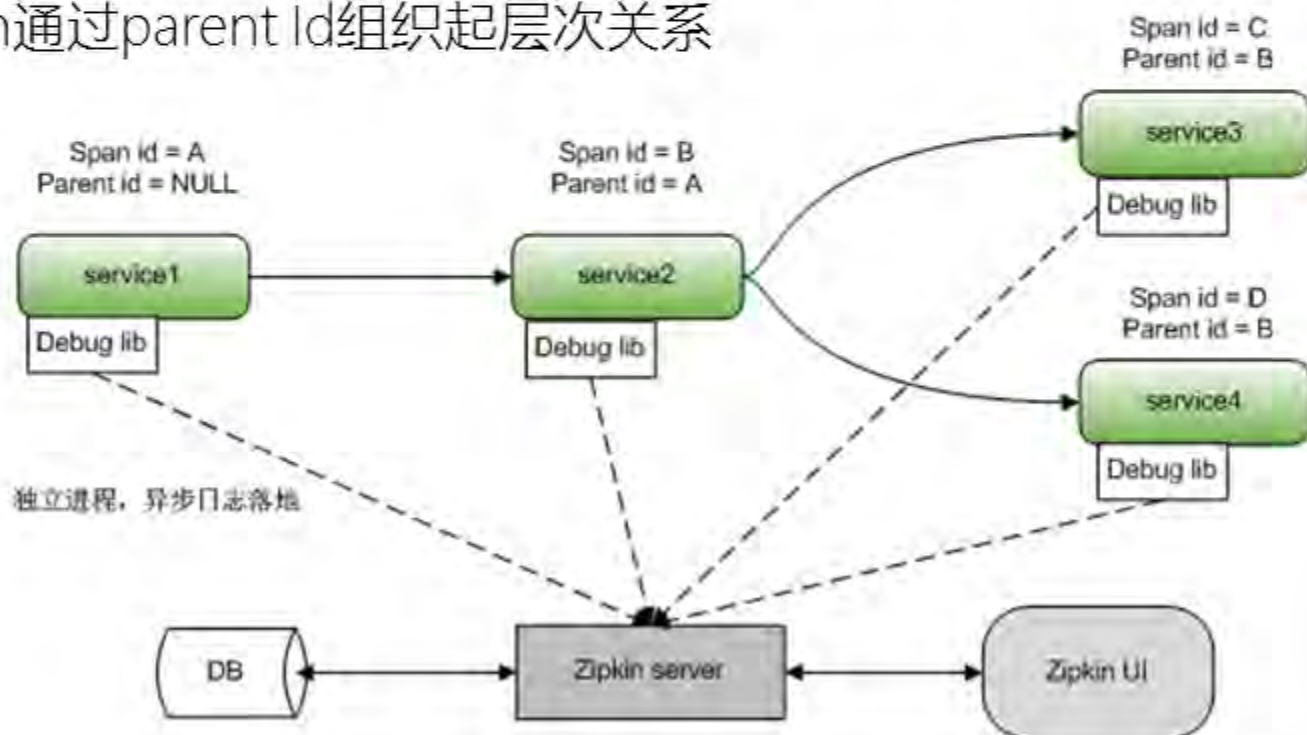


debug日志汇总串联



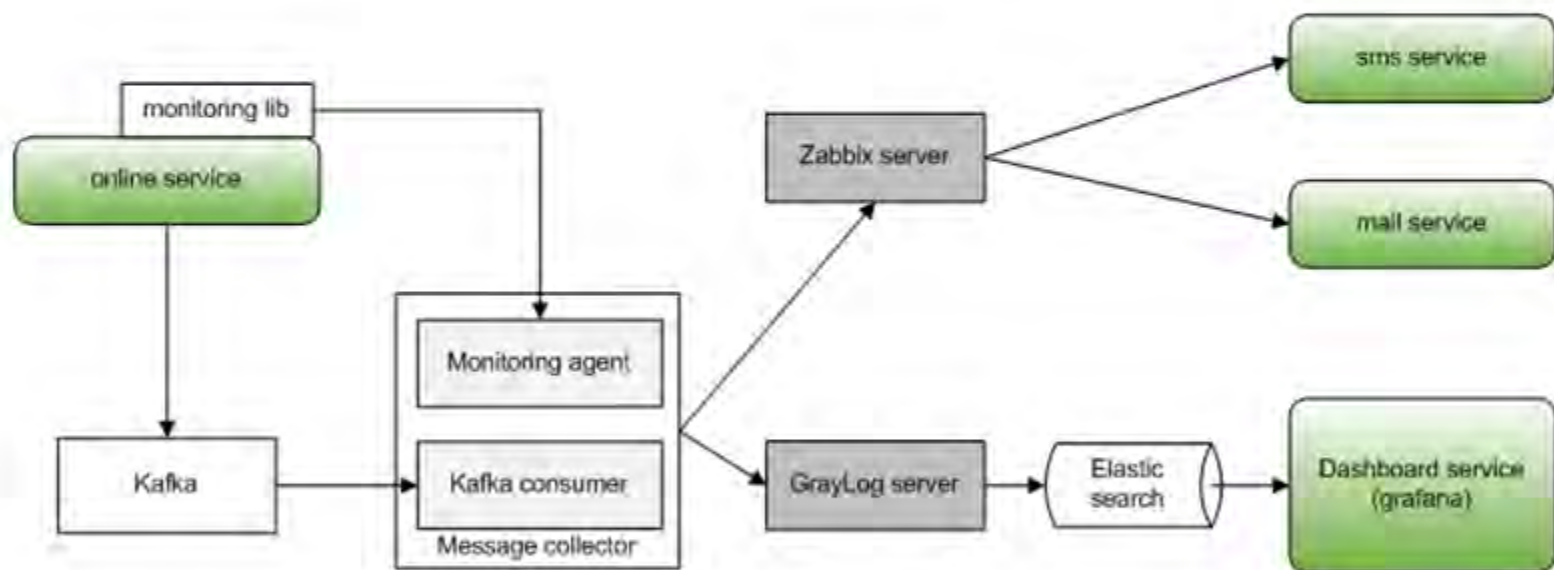
可视化展示

- 一条请求链路通过trace id唯一标识，每经过一个模块，建立一个span
- Span通过parent Id组织起层次关系



I 高可用性管理-分布式监控管理系统

- 每个线上模块嵌入一个monitoring lib，启动独立的监控服务
- Monitoring agent通过API调用获得线上模块的监控指标
- 收集和处理：Graylog + elasticsearch + Grafana + zabbix



| 总结-分布式系统设计的一些经验

整体架构思路：化整为零的微服务架构

高效应对规模增长

- 理解业务，做合理的拆分
- 分层处理
- 实时处理框架

高可用性管理

- 降级服务能力
- 分布式debug系统
- 监控管理系统

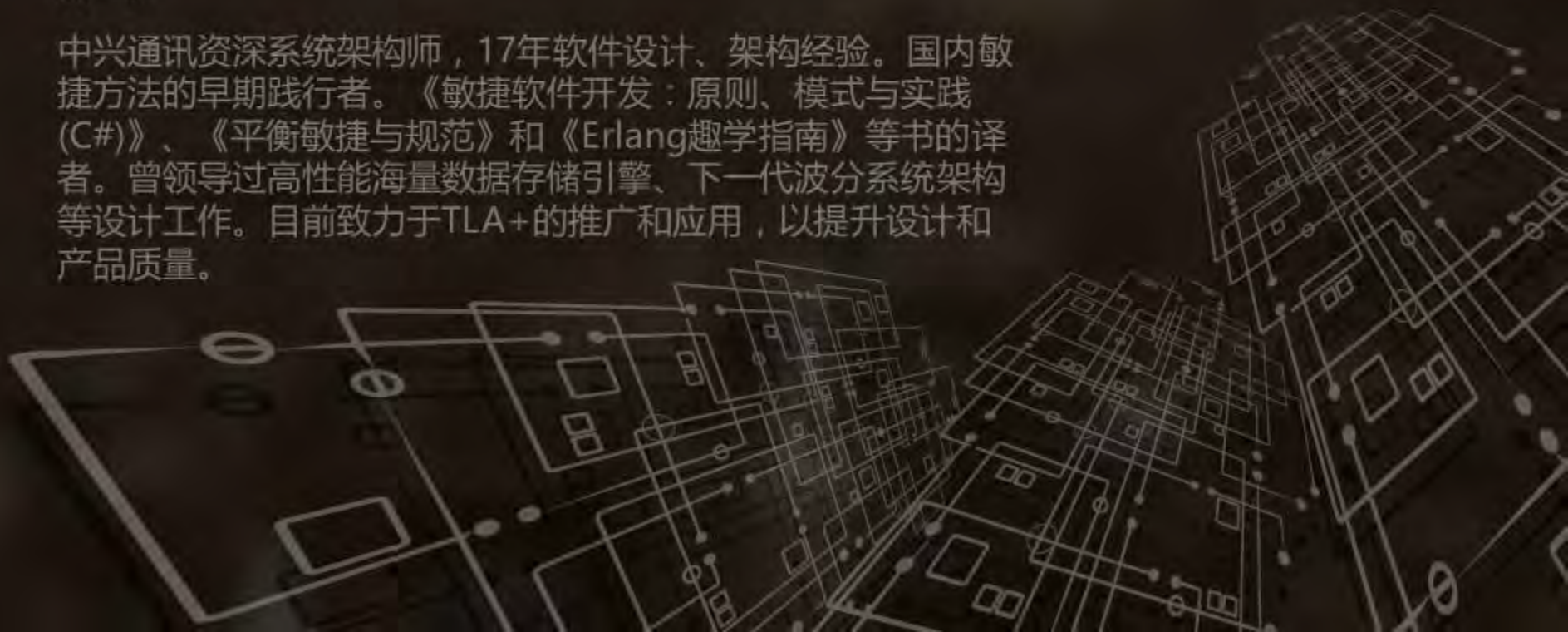
Q&A

2017 Software Architecture Summit

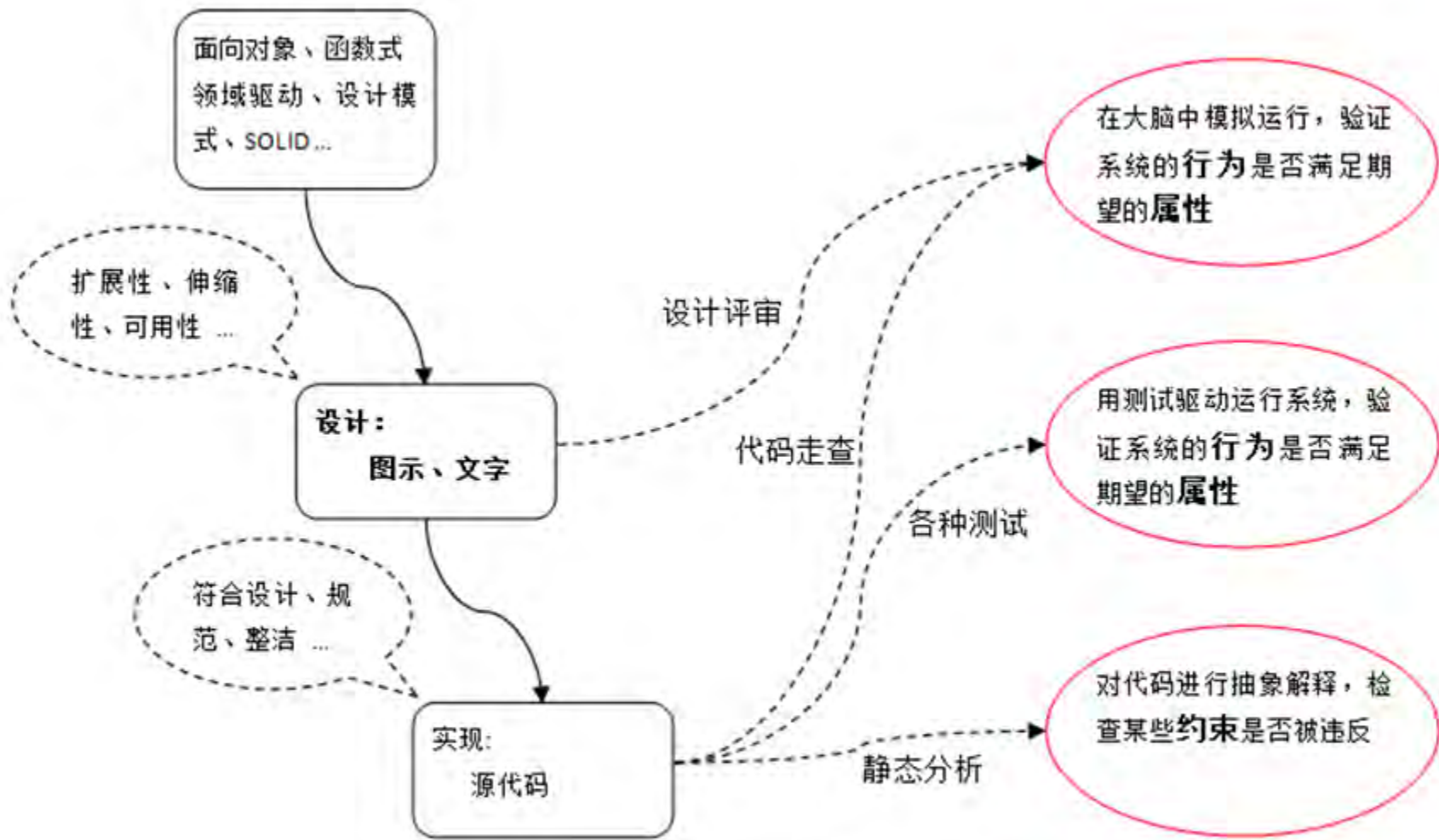
正确性驱动建模: 用TLA+设计系统

孙鸣

中兴通讯资深系统架构师，17年软件设计、架构经验。国内敏捷方法的早期践行者。《敏捷软件开发：原则、模式与实践 (C#)》、《平衡敏捷与规范》和《Erlang趣学指南》等书的译者。曾领导过高性能海量数据存储引擎、下一代波分系统架构等设计工作。目前致力于TLA+的推广和应用，以提升设计和产品质量。

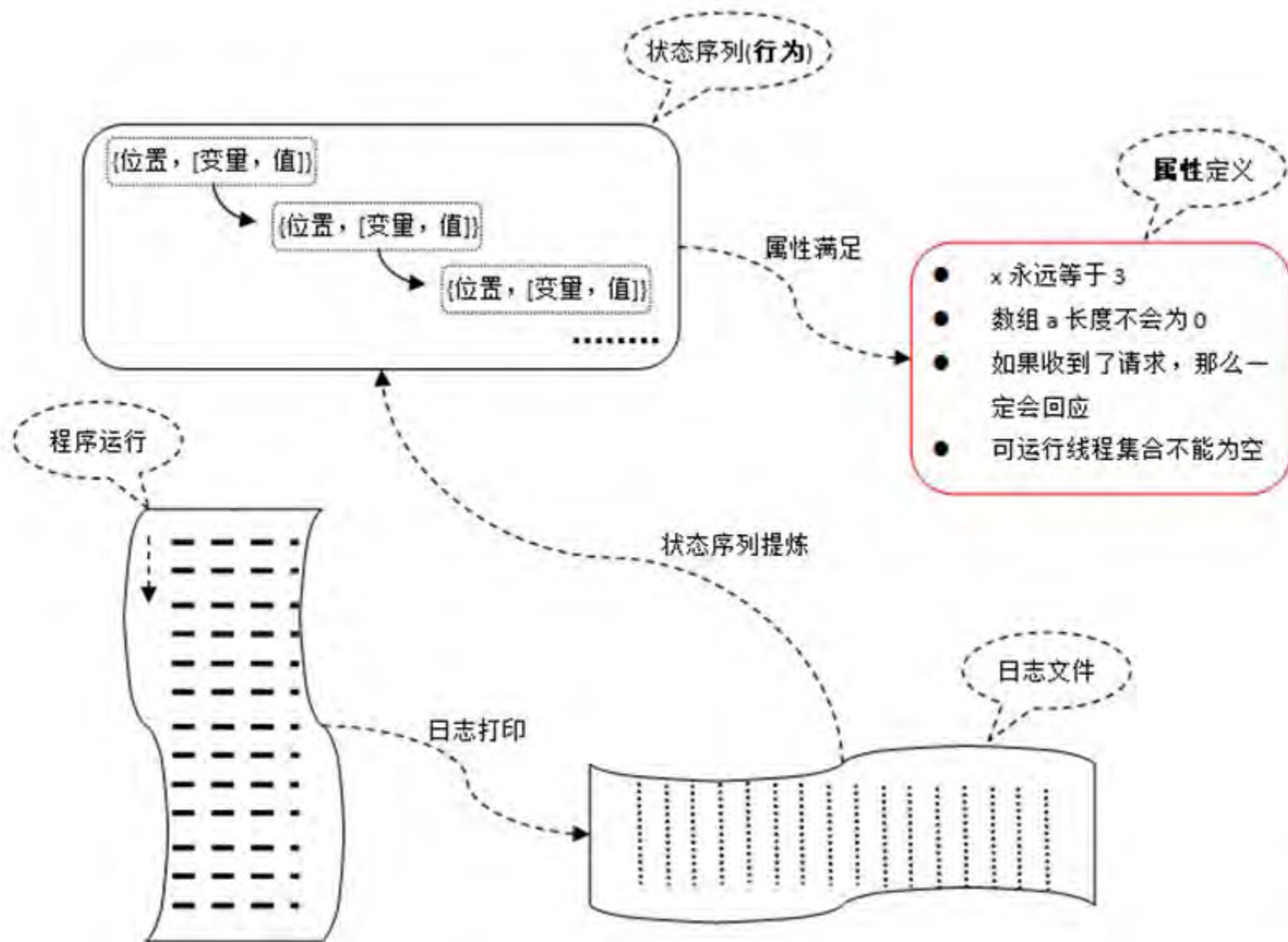


设计 & 正确性



“To a first approximation, we can say that accidents are almost always the result of incorrect estimates of the likelihood of one or more things.” - C. Michael Holloway, NASA

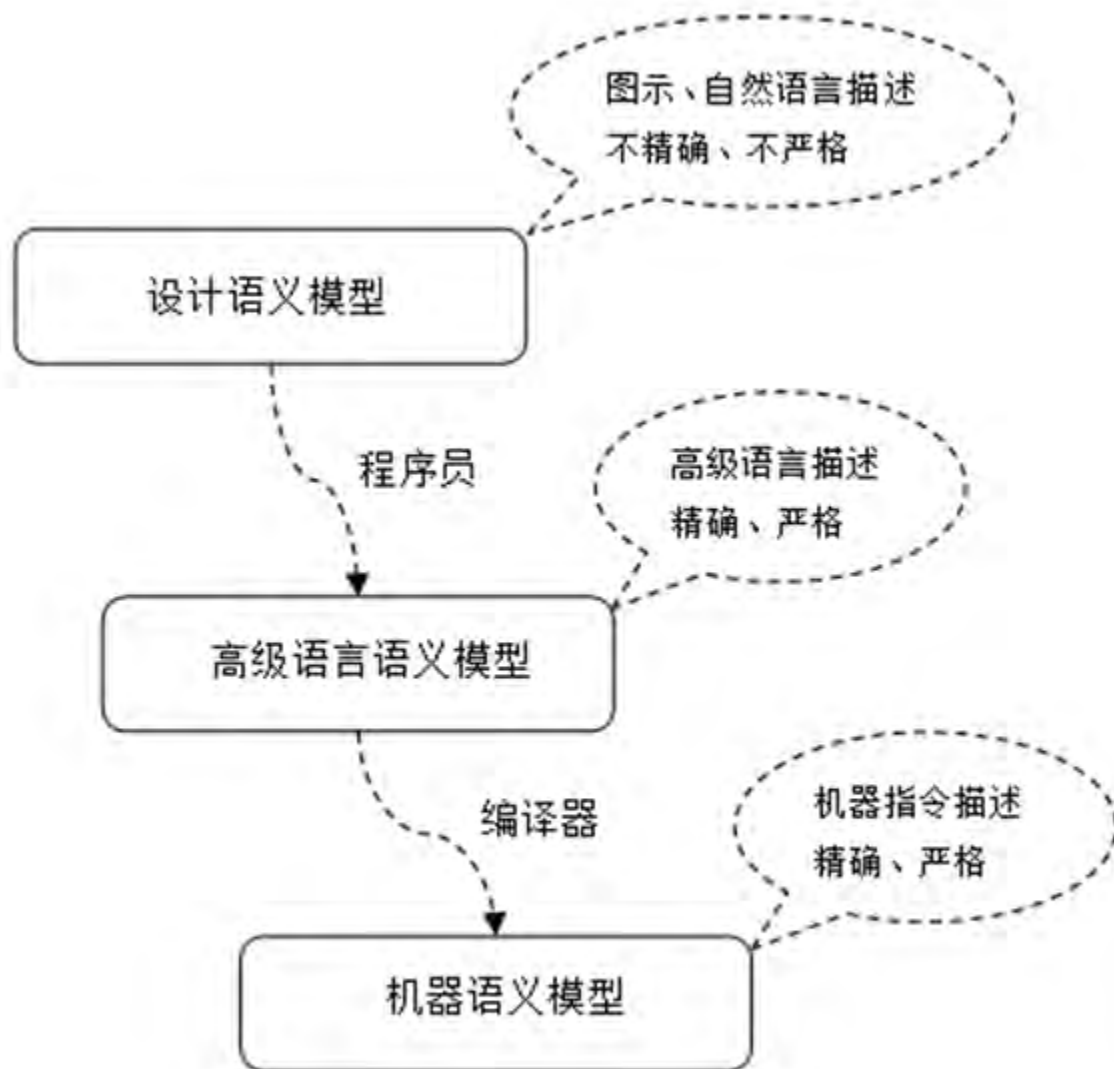
一般来讲，事故的发生几乎都是错误估计了某些事情出现可能性的结果。



繁琐、耗时、痛苦，还需要运气！

模型 & 抽象

抽象层次降低



低抽象语义模型的行为集合 是高抽象语义模型行为集合的子集

越抽象，包含的行为越多。

抽象是用不确定性忽略无关细节

Quicksort

From Wikipedia, the free encyclopedia

The steps are:

不确定性

1. Pick an element, called a *pivot*, from the array.
2. *Partitioning*: reorder the array so that all elements with values less than the pivot come before the pivot, while all elements with values greater than the pivot come after it (equal values can go either way). After this partitioning, the pivot is in its final position. This is called the *partition* operation.
3. **Recursively** apply the above steps to the sub-array of elements with smaller values and separately to the sub-array of elements with greater values.

不确定性

系统蓝图

正确性属性

环境属性、环境事件

问题固有变量、状态、行为

架构设计

风格、模式选择

中间件选择

语言、范型选择

像科学家一样思考



动态连续系统建模

提取问题**相关变量**，用**微分方程**描述变量随时间变化关系

动态**离散**系统建模

提取问题**相关变量**，用**TLA+**描述变量每一步的变化关系

Leslie Lamport



贡献：

TLA+

Paxos

Byzantine Generals

Digital Signatures

Sequential Consistency

LaTeX

TLA+的设计哲学

实用主义而非理论完美

简单：一线工程师可以很快学习和使用

实用：能够应用于具有相当复杂程度的现实软件、硬件系统。

通用：以同样的形式应用于不同种类的系统：分布式、并发、顺序、反应式、批处理等等

We are motivated not by an abstract ideal of elegance, but by the practical problem of reasoning about real algorithms. Rigorous reasoning is the only way to avoid subtle errors... .. and we want to make reasoning as simple as possible by making the underlying formalism simple

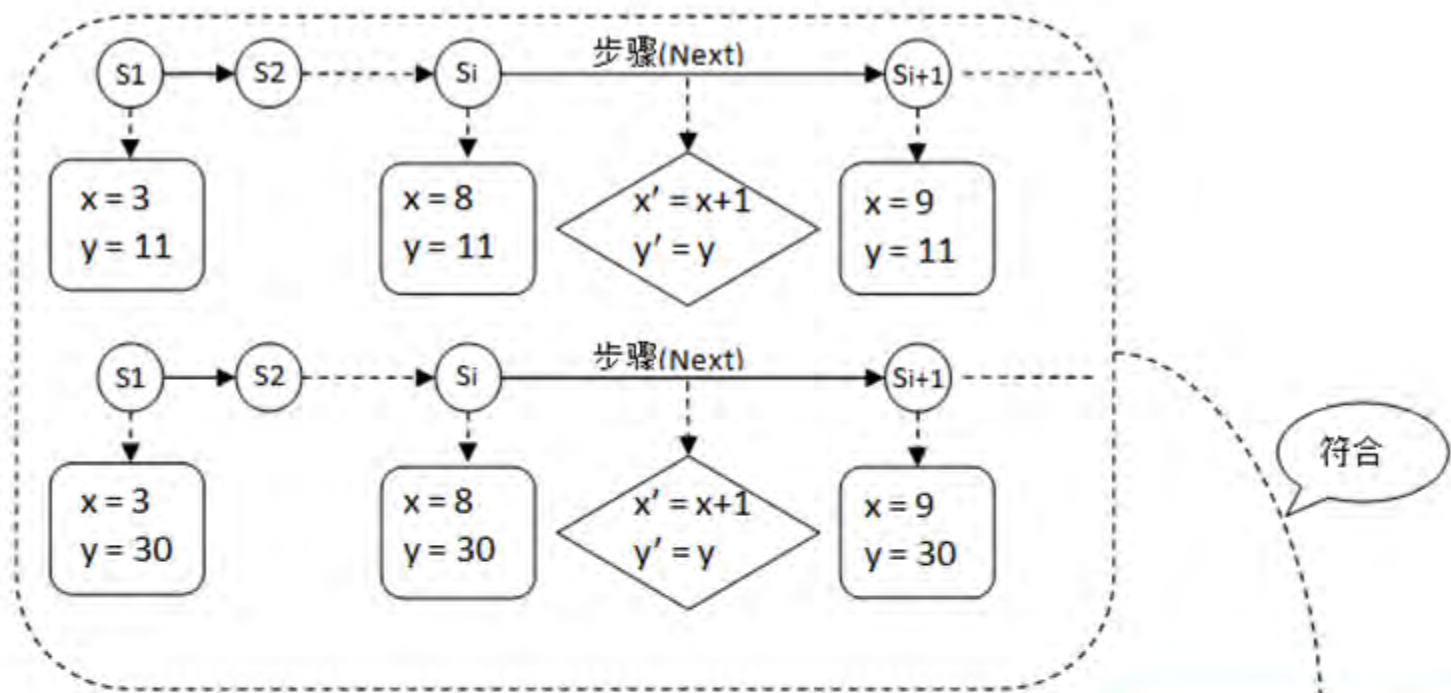
Leslie Lamport

行为 & 属性

The Standard Model An abstract system is described as a collection of behaviors, each representing a possible execution of the system, where a behavior is a sequence of states and a state is an assignment of values to variables.

The TLA+ Hyperbook, Leslie Lamport

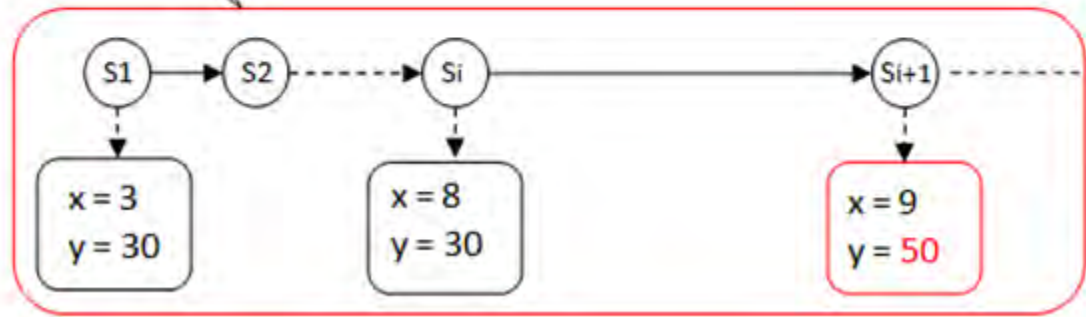
标准模型：我们用行为集合描述抽象系统，每个行为都表示该系统的一个可能执行情况，行为是状态序列，状态是对变量的赋值。



符合

Init: $x=3 \wedge y \in \mathbb{N}$
 Next: $x'=x+1 \wedge y'=y$

不符合



系统模型(行为集合)
的严格、精确定义

告诉神灯你想
要什么



过滤出

$s1 \rightarrow s2 \rightarrow s3 \dots$

...

...

简单的数学

一阶命题逻辑： $\wedge, \vee, \neg, \Rightarrow, \equiv$

基本的集合论： $\cap, \cup, \subseteq, \setminus$

一阶谓词逻辑： \forall, \exists

线性时态逻辑： $\square, \diamond, \rightsquigarrow$

期望属性的定义

Safety属性：坏的事情一定不会发生

x永远为偶数

工作线程数目永远大于0

临界区中最多只有一个线程

Liveness属性：好的事情最终会发生

如果收到用户请求，最终一定会回应

y最终会大于100

如果线程尝试进入临界区，最终一定会进入

实战时间

The Science of Programming (Monographs in Computer Science)

by David Gries · (Author)

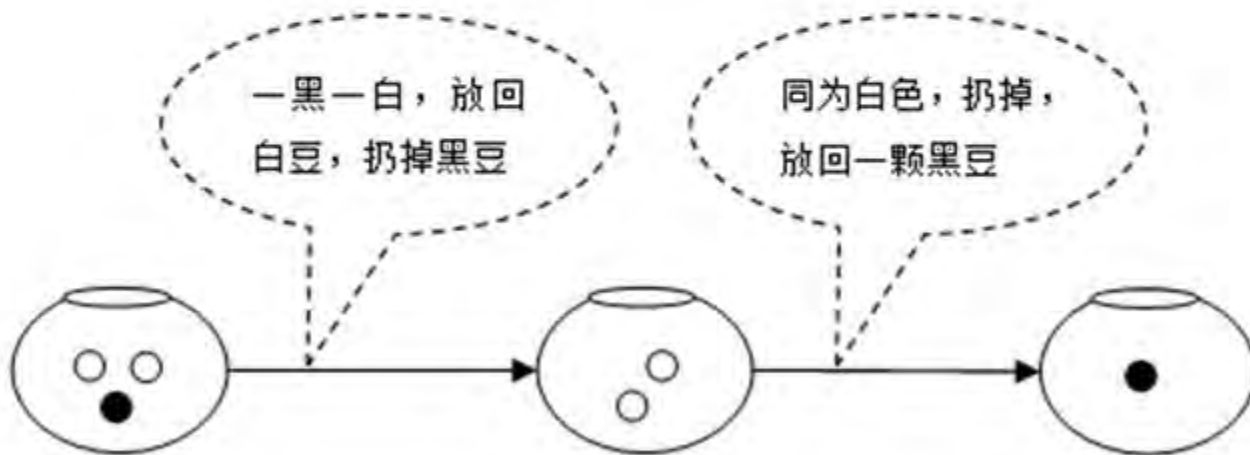
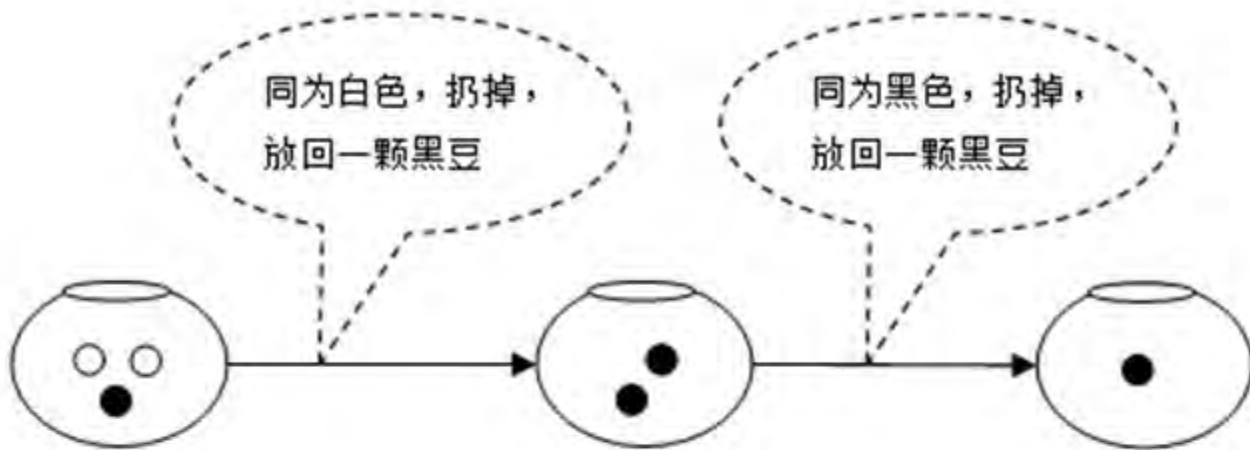
★★★★☆ · 10 customer reviews

咖啡罐问题：

有一个装有黑色和白色咖啡豆的罐子，罐子的外面还有一堆“足够多”的黑色咖啡豆。重复如下过程，直到罐子中只剩下一颗咖啡豆：

随机从罐子中取出两颗豆。如果它们的颜色相同，就把它们扔掉，并向罐子中放入一颗黑豆。如果颜色不同，就把白豆放回罐中，黑豆扔掉。

证明这个过程一定会结束。结束时，罐中所剩豆子的颜色和最初罐子中白豆和黑豆数目之间有何函数关系？



EXTENDS Integers

CONSTANT M,N

VARIABLES b,w

Init == $b = M \wedge w = N$

Next == $\vee \wedge b > 1 \quad \backslash^*$ same black
 $\wedge b' = b - 1$

$\wedge w' = w$

$\vee \wedge w > 1 \quad \backslash^*$ same white

$\wedge b' = b + 1$

$\wedge w' = w - 2$

$\vee \wedge w > 0 \wedge b > 0 \quad \backslash^*$ not same

$\wedge b' = b - 1$

$\wedge w' = w$

期望属性定义

不变性(invariant) : **$(N-w) \% 2 = 0$**

或者

时态公式 : **$[](N-w) \% 2 = 0$**

并发的本质

