

Architect

SACC

2022 中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2022

· 激发架构性能 点亮业务活力

云上会议 网络直播 | 2022年10月27-29日

IT168.com

ChinaUnix.net

ITPUB

# B站运维数仓建设和数据治理实践

bilibili

SRE 资深研发工程师

袁帅



## 分享嘉宾: 袁帅 (villager)

B站基础架构/业务SRE 资深研发工程师

- 2020年加入B站
- 擅长基础架构、Devops、SRE等领域架构设计和研发
- 先后从事过大数据运维平台，B站Tidb平台化建设，运维作业平台及CMDB资产管理平台的研发工作
- 目前专注于SRE CMDB元信息平台的研发和治理工作



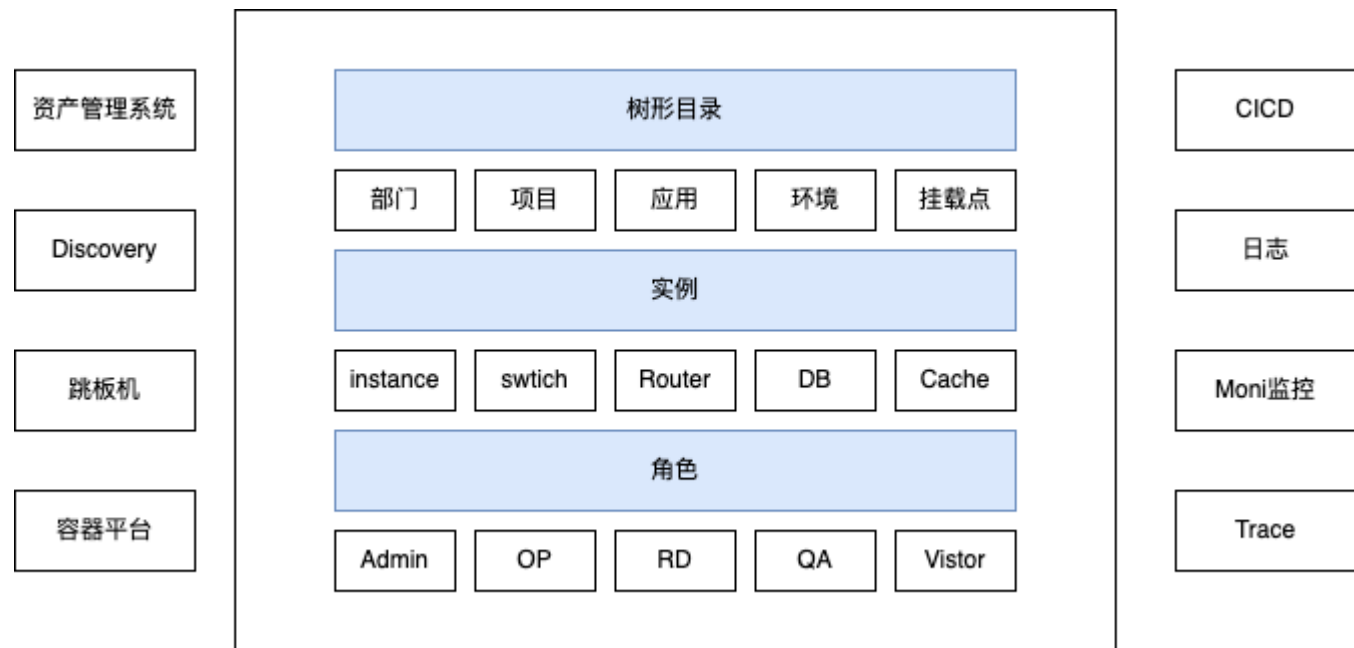
# 目录

- 01 背景：B站SRE体系是如何构建元数据和使用的
- 02 一步步，服务树沦为一个毫无边界的元数据中心
- 03 B站运维数仓建设思路
- 04 关注数据质量
- 05 总结

# B站SRE体系是如何构建元数据和使用

## PART 01

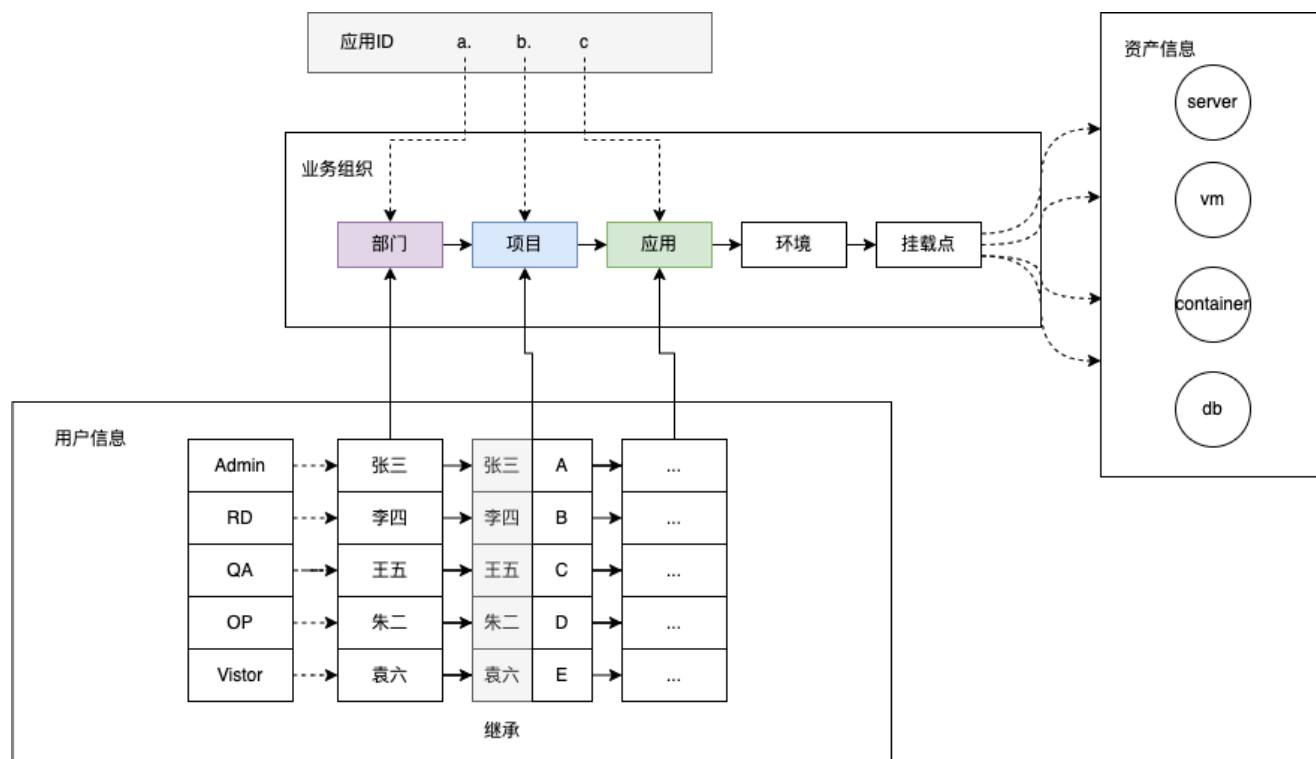
# 服务树的诞生



## 主要功能:

- 服务（树形）目录：以应用的视角组织服务的树形目录，细化和统一资源隔离
- 全局应用ID：为应用分配全局唯一Appid, 周边平台业务数据组织，聚合均已Appid标识
- 实例管理：提供应用部署相关的元信息（IP，机房，云厂商等）
- 节点角色管理：基于内部统一用户，托管个系统角色用户管理功能，实现鉴权

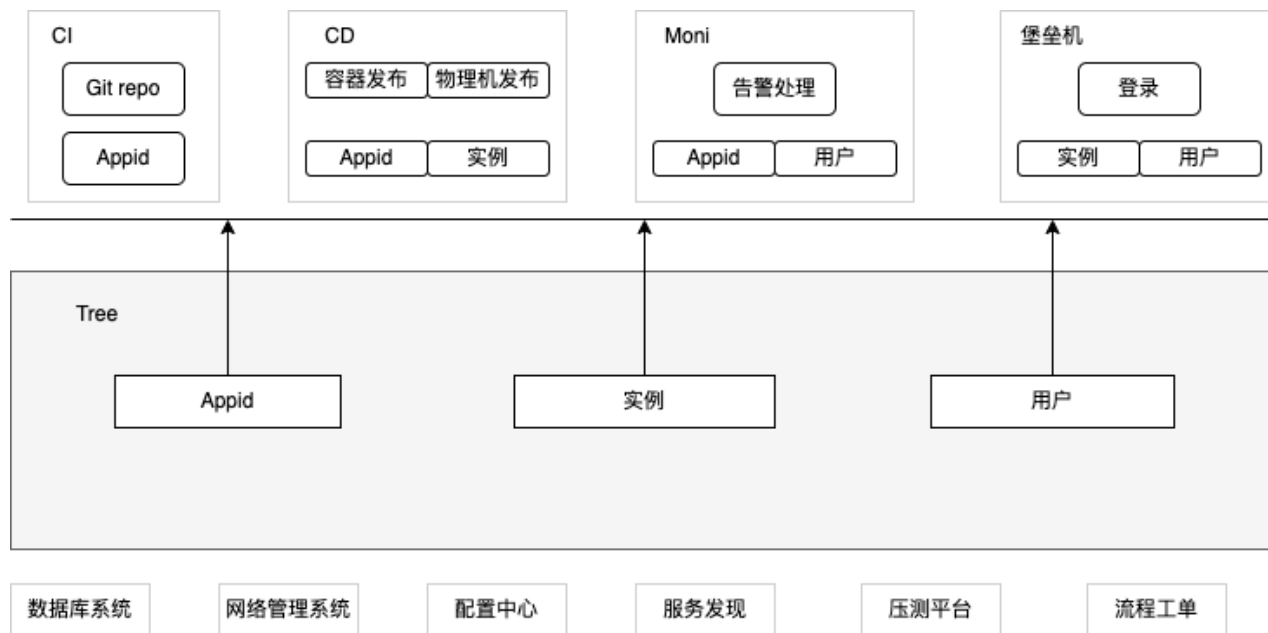
# 元数据构建



## 如何构建数据

- 按照部门.项目.应用建立应用组织关系, 提供全局应用ID
- 管理业务和资产的关联关系, 实现机器故障时的业务感知
- 提供RBAC模式业务人员角色
- 自上而下的角色继承关系

# 元数据使用



## 如何使用数据

- Appid作为平台资源管理核心
- 以CI平台为例，创建应用构建信息，编译构建产出围绕Appid建设，CD发布时根据应用选择制品，选择实例
- 周边平台的资源都围绕Appid建设，应用提供资源和人员信息关联



# 一步步，服务树沦为一个毫无边届的数据中心

## PART 02

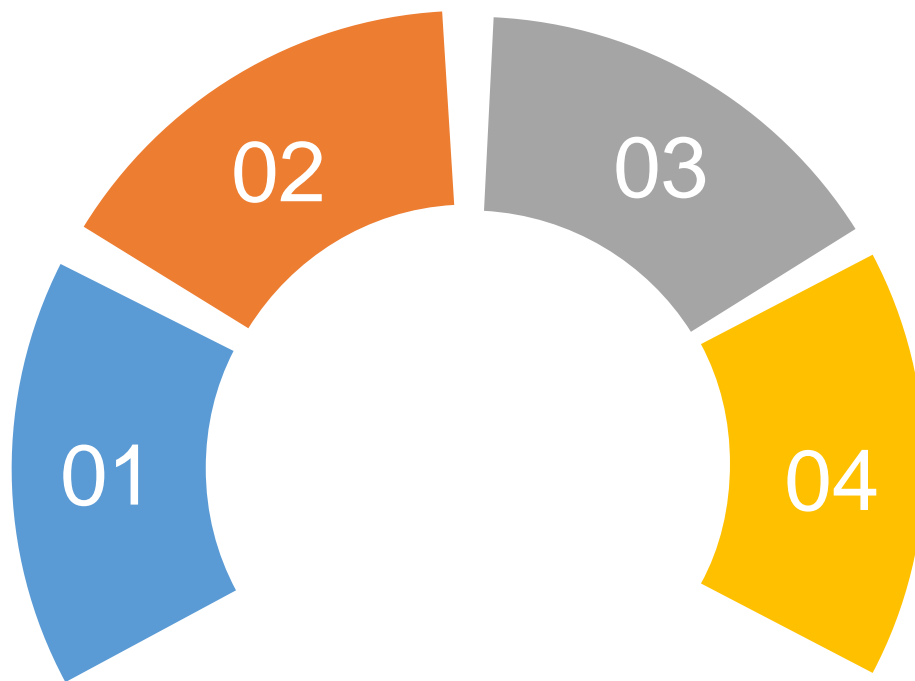
# 核心问题

## 权限管控不足

RBAC场景无法满足细粒度、精细化的权限管控。直接或间接导致周边平台乱用服务树角色，导致角色人员混乱，无法治理

## 数据治理不足，灵活性不足

数据直接关联关系缺失，围绕应用做的资产关系缺乏统一的规范注册上报;不能很好的支持配置项的自定义和扩展，导致接入新资源时缓慢



## 自动化能力缺失

传统的资源，自动化能力严重不足，很多物理机、云主机销毁后无法感知，导致数据准确性低

对于容器类信息，同样面临该问题

## 消费场景考虑欠缺

围绕应用建设的运维数仓，本身从设计上没有考虑应用变动所带来的消费场景，周边平台对接完后发现应用改动后，缺乏感知

# OpenAPI混乱

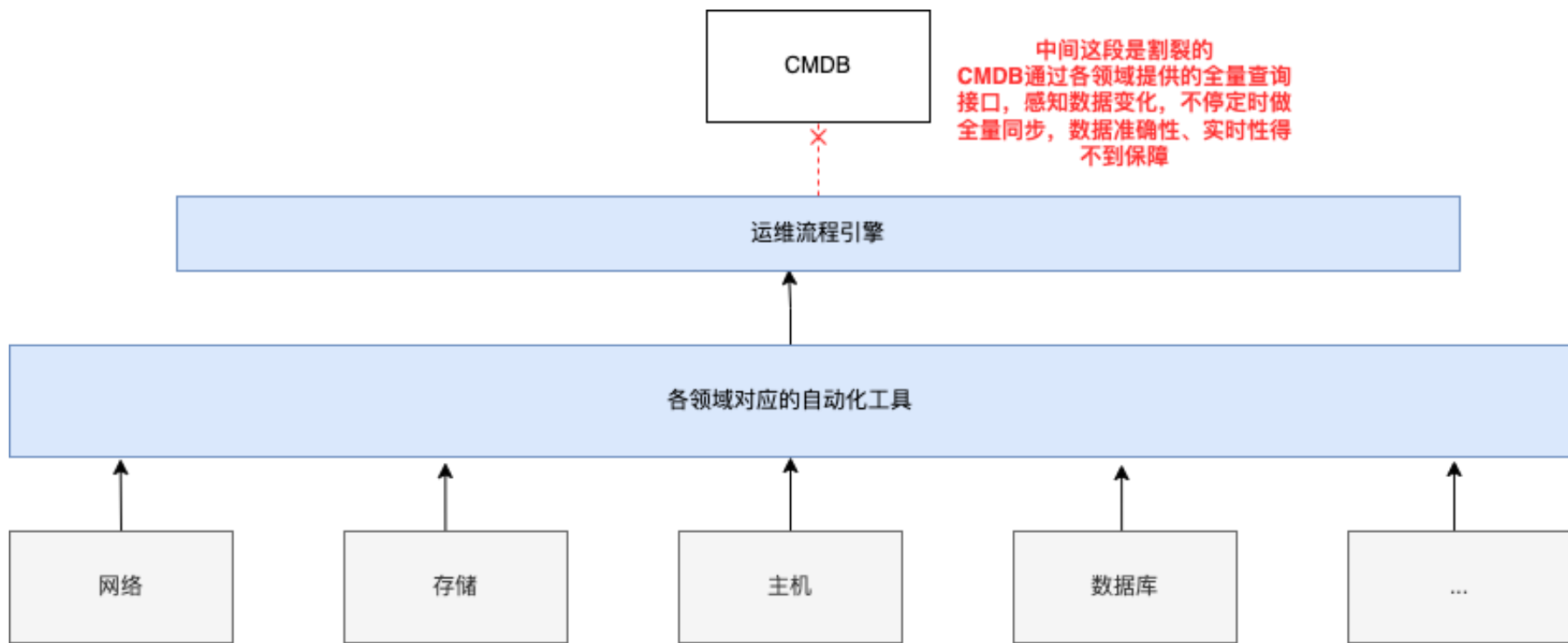
缺乏规范和制度，快速迭代，快速满足需求

平台调用方缺乏治理和维护

接口替换代码不处理，接口文档长期不维护

消费场景太多，平台定制接口太多，很多功能可以复用接口

# 数据更新依赖全量查询

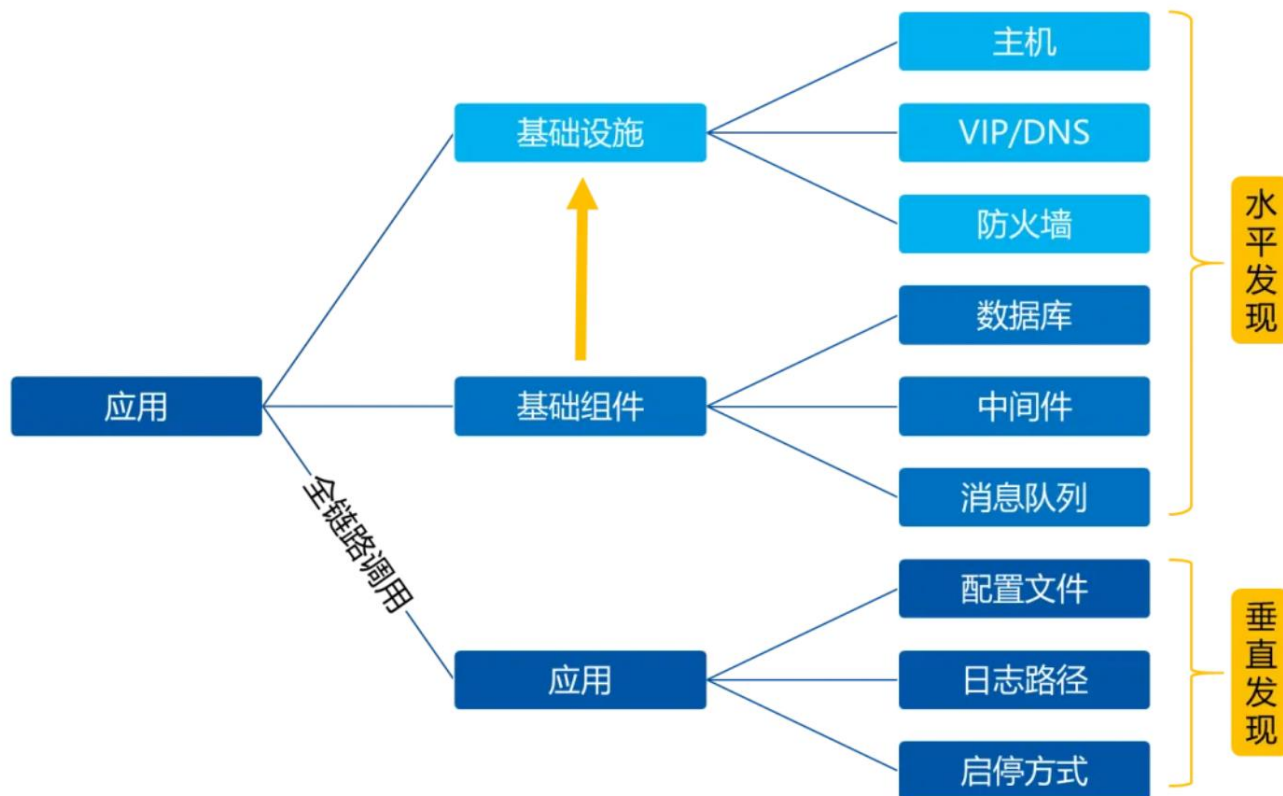




# 非标的使用方式太多

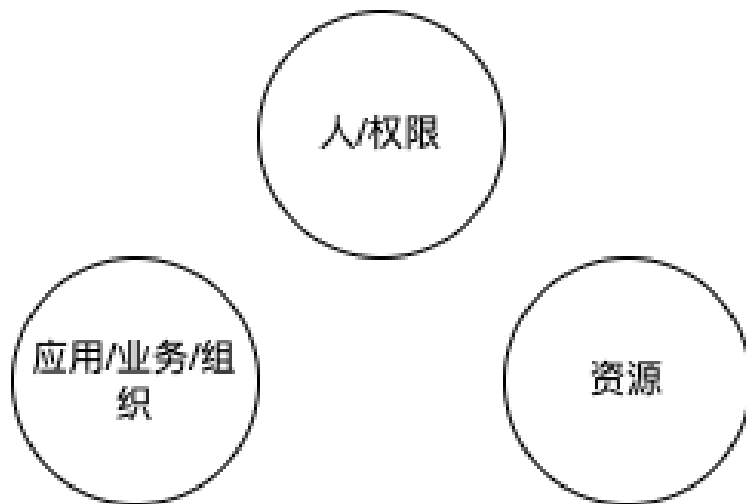


# 无法应用画像



- 应用缺乏应用资源注册
- 应用上下游依赖关系没有梳理

# SRE团队治理数据无从下手



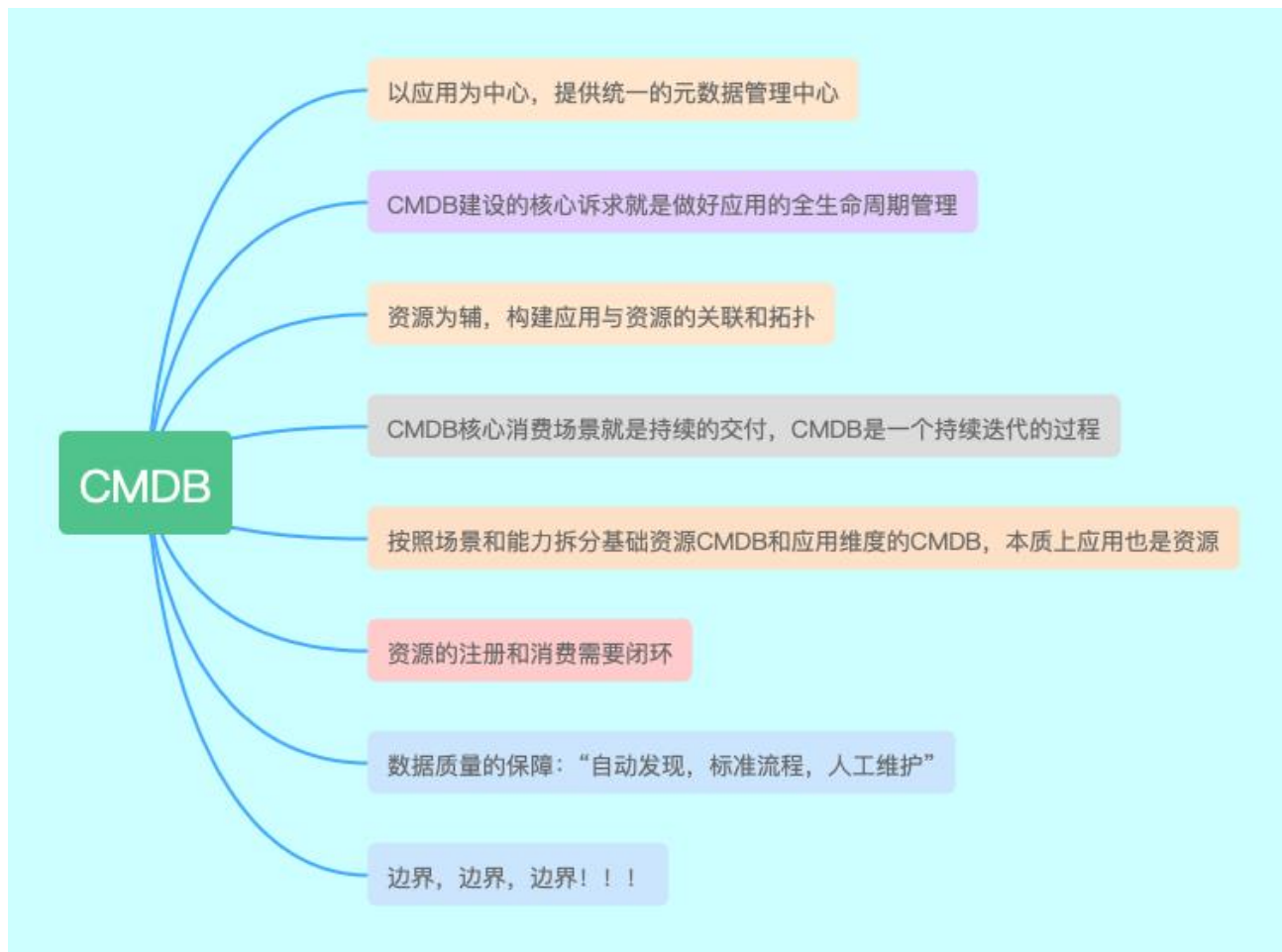
- 最小权限安全控制原则，服务树权限申请审批没有限制导致人员、权限信息混乱
- 应用无法迁移
- 应用无法交接
- 应用下所使用的资源，无法做关联，覆盖不全

# B站运维数仓建设的整体思路

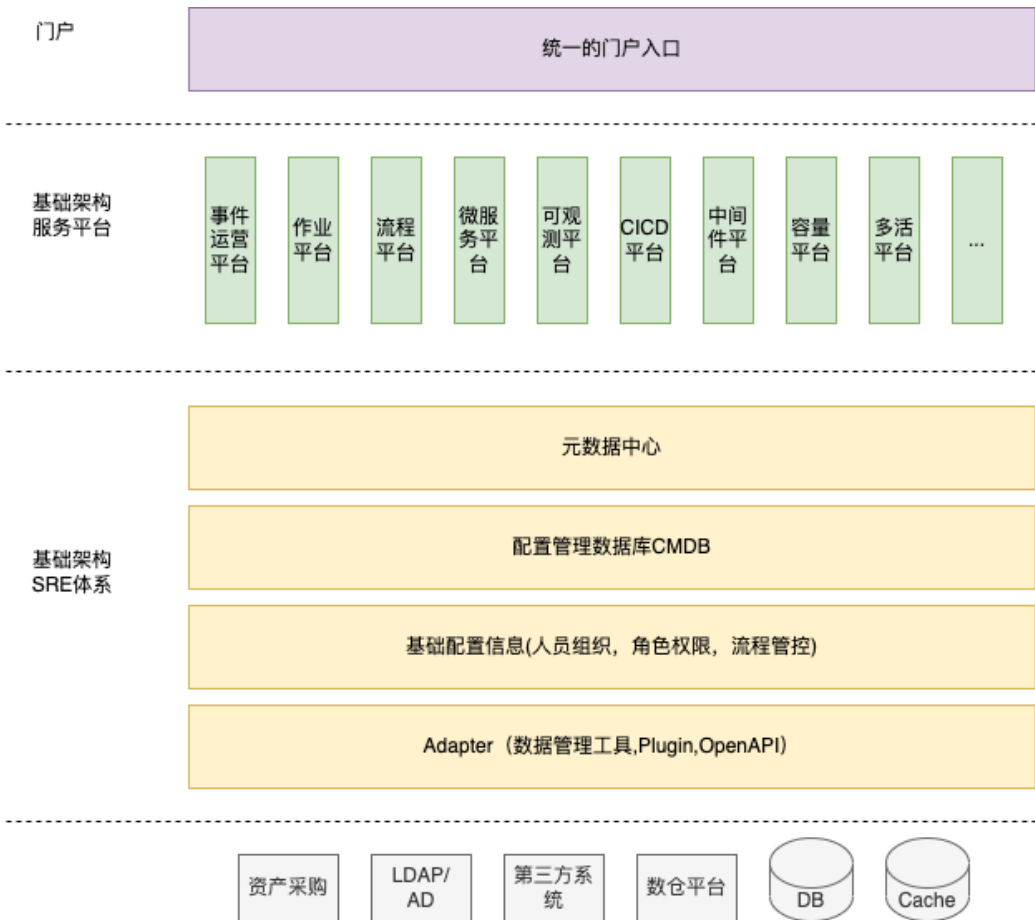
## PART 03



# 明确目标和准则

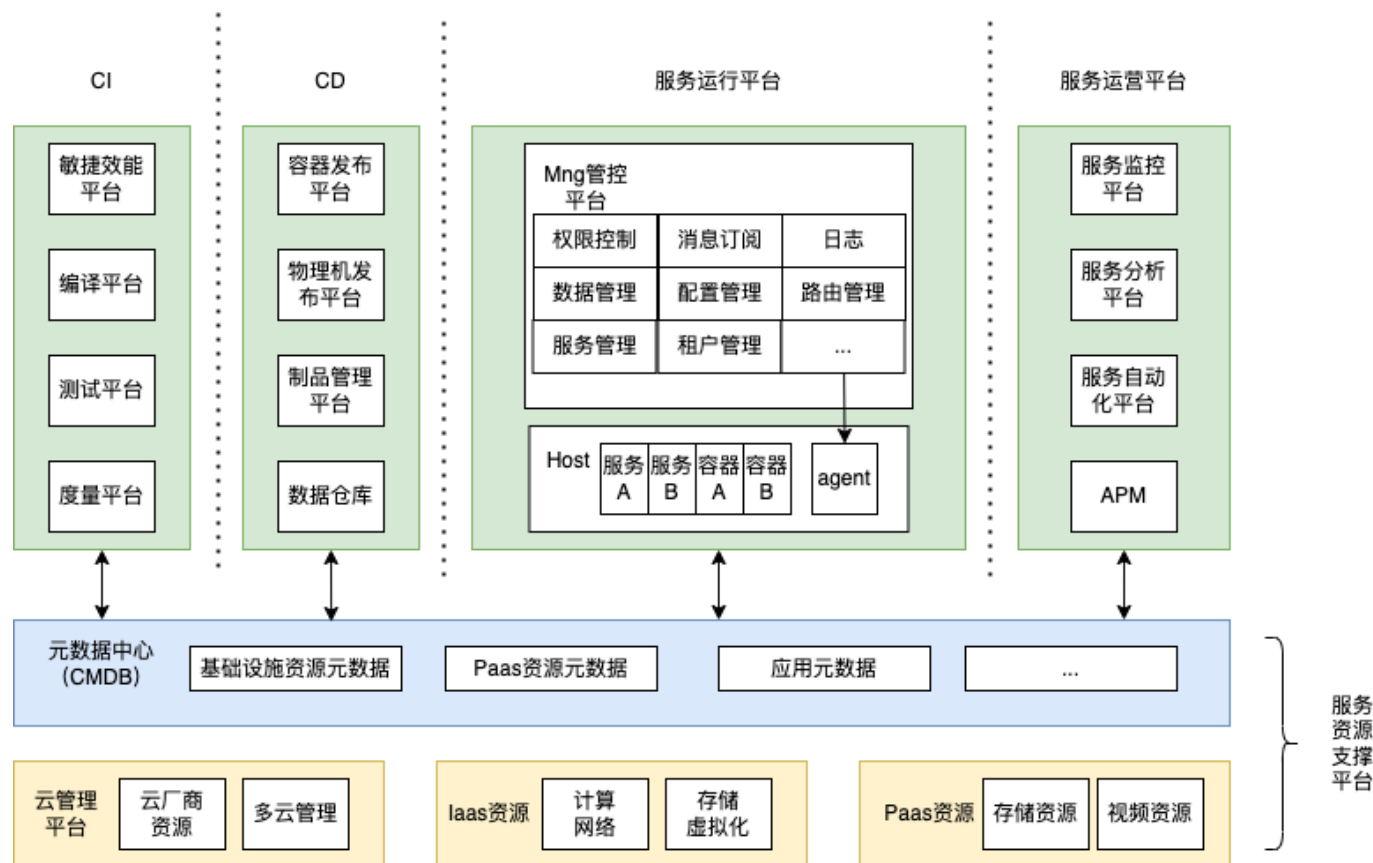


# 元数据中心-在SRE体系中的定位



- 元数据中心作为SRE体系中核心的服务存在
- 元数据中心的数据来自CMDB，支持基础架构服务平台运转
- 元数据中心为端到端的服务运行时过程提供数据支撑

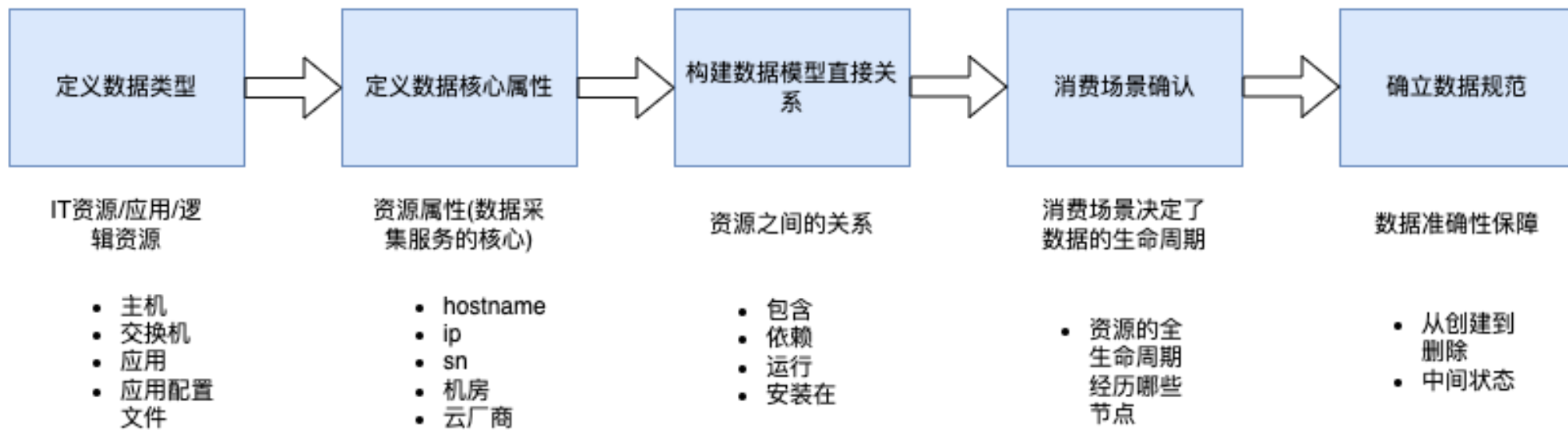
# 服务在运行期间与元数据中心的关系



- 元数据中心为服务全生命周期内提供数据基础
- 元数据中心也是作为服务的支撑平台存在的，贯穿服务始终

# 元数据中心如何建立数据模型

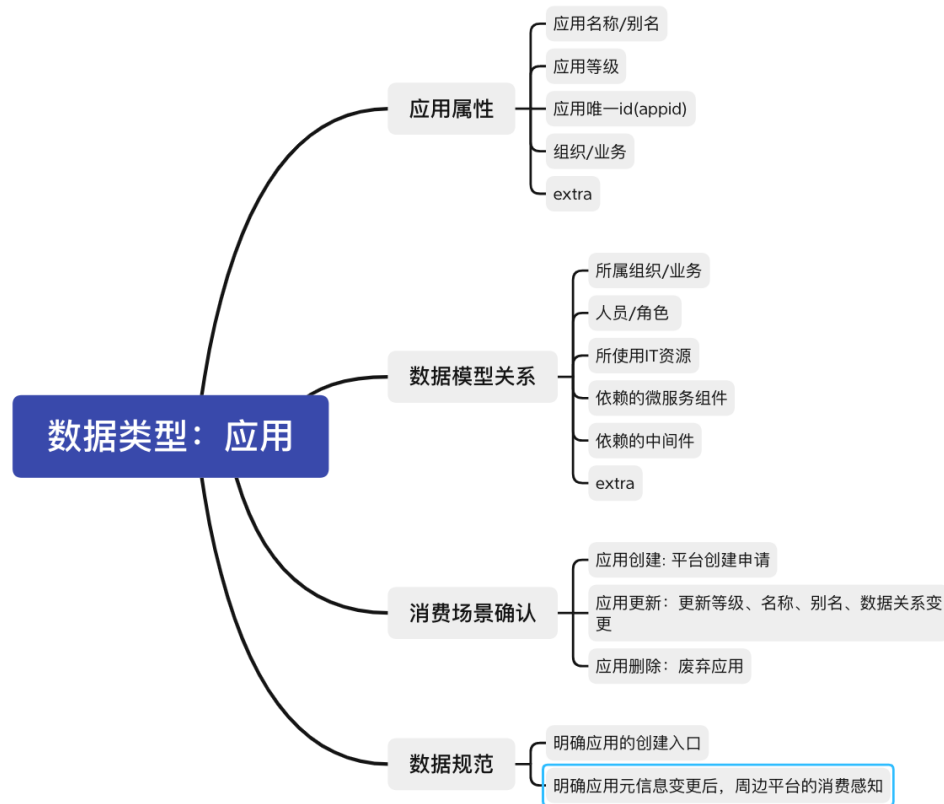
CI配置模型梳理过程



以数据全生命周期为出发点，确定属性，理清关系，确定消费场景，自动化流程保障数据准确性



# 举例：应用数据模型的创建

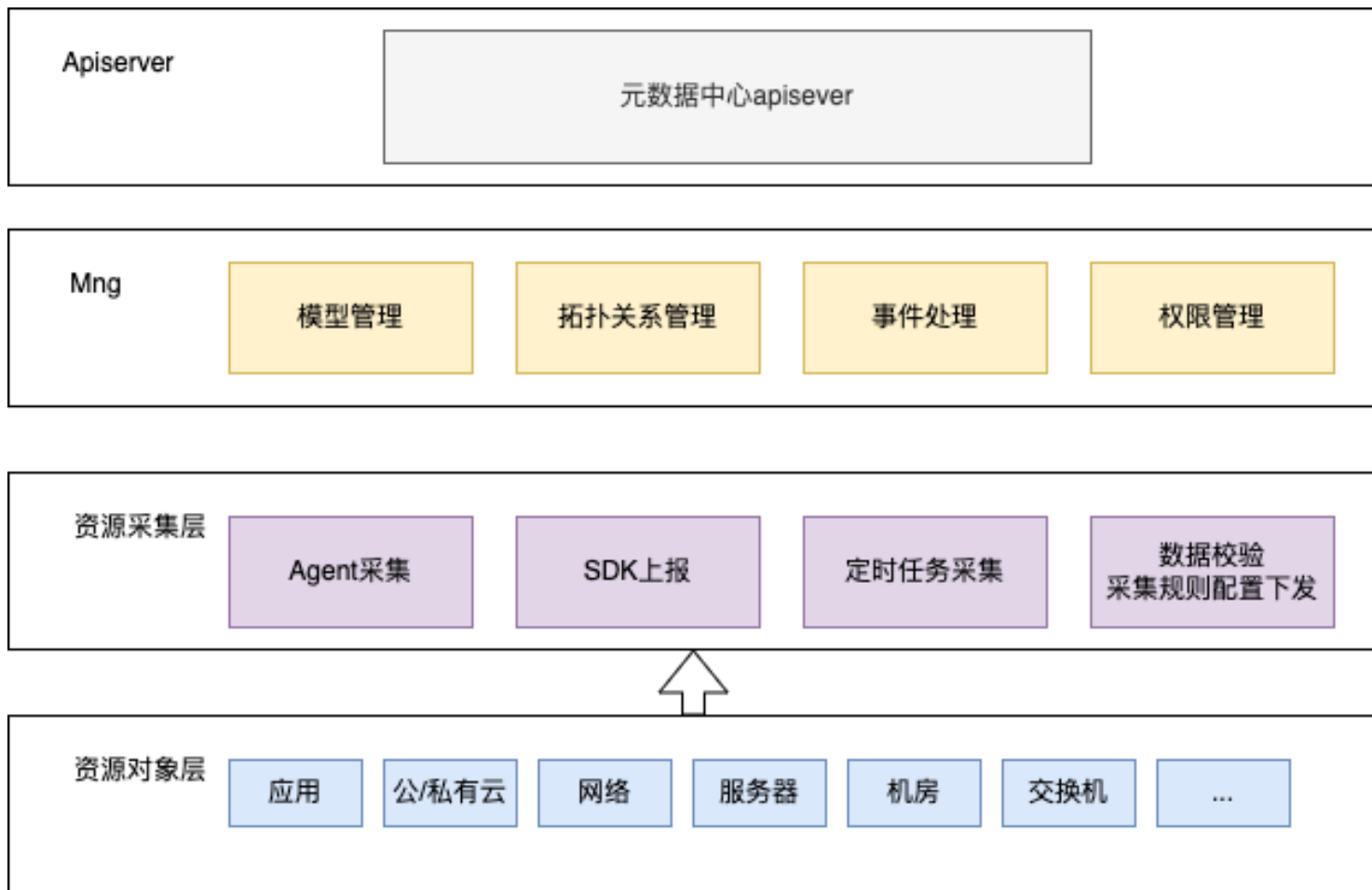


创建思路基本按照流程一步步细分

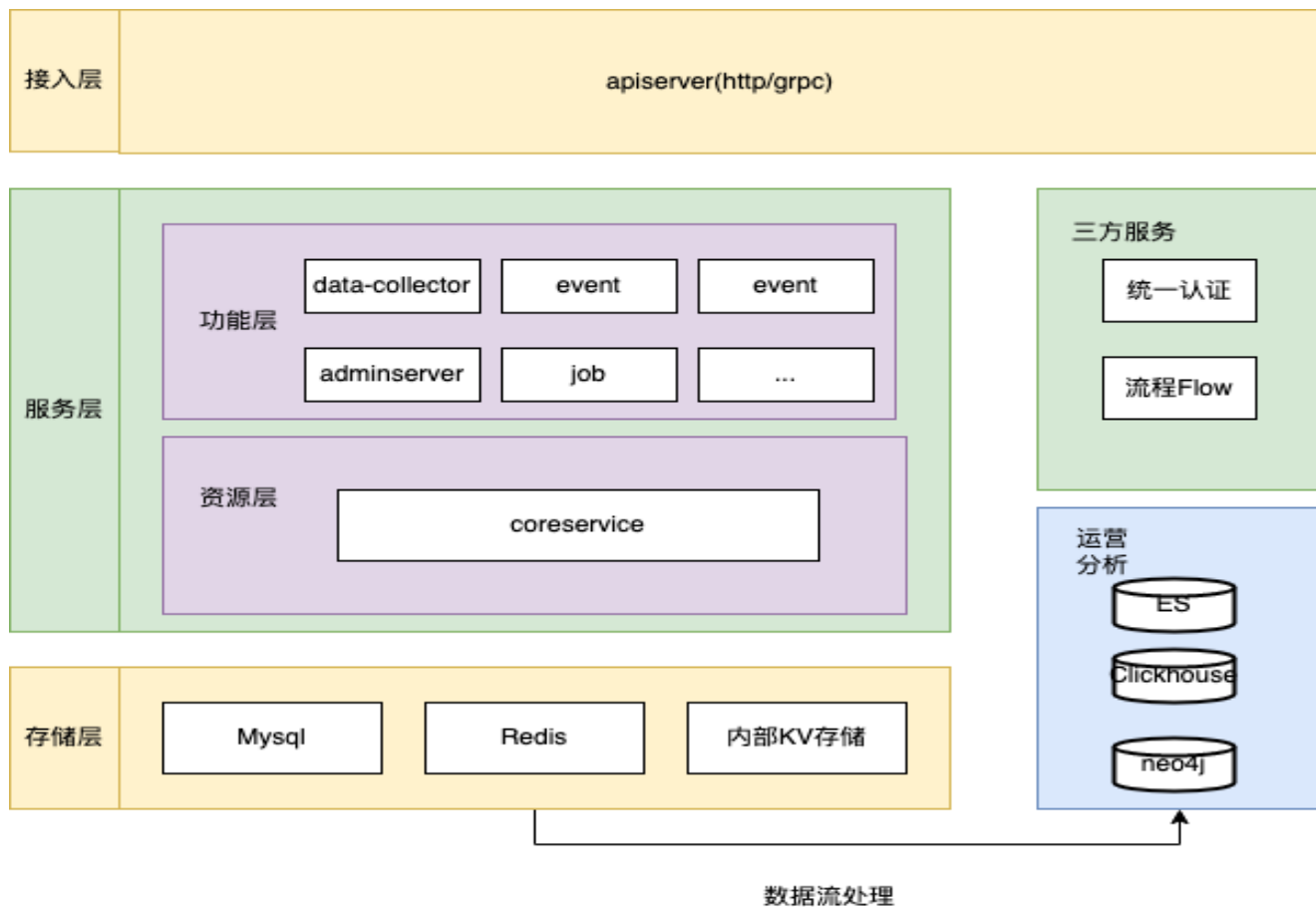
自定义一个数据模型需要产品、研发一起界定

数据规范需要各方遵守

# 元数据中心产品架构

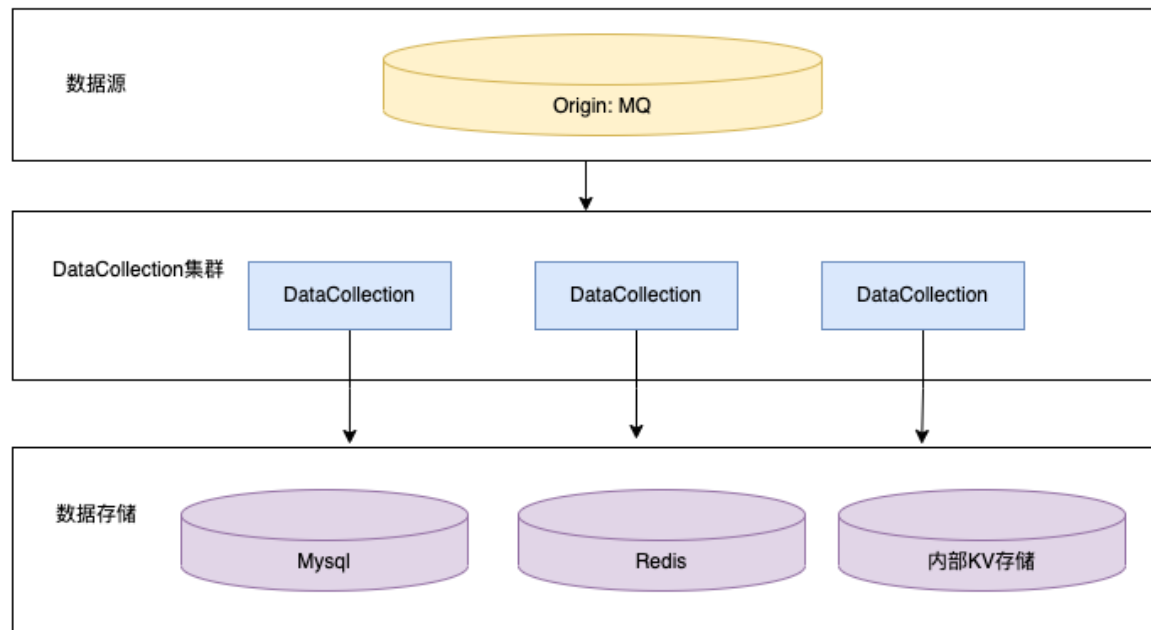


# 元数据中心技术架构



# 元数据中心数据采集

Data-Collector 数据采集模块



- 数据源(Origin): 采集的数据节点采集器会将数据发送到指定的内部消息队列里, DataCollection订阅指定Topics接收采集数据
- DataCollection集群: 分布式, 连接Topics消费数据
- 数据存储: 数据经DataCollection处理最终落在Mysql, Redis和内部KV存储上



# 原始数据关联关系分析

## Relation Model

关系	说明	示例
依赖	depend on	应用依赖配置
包含	include	k8s集群中包含组件
运行	run in	容器运行在宿主机上
安装在	install in	数据库安装在服务器上
连接	connect with	服务器连接在交换机上

把CI和CI之间的关系种类列出来，前提需要产品、业务方充分沟通

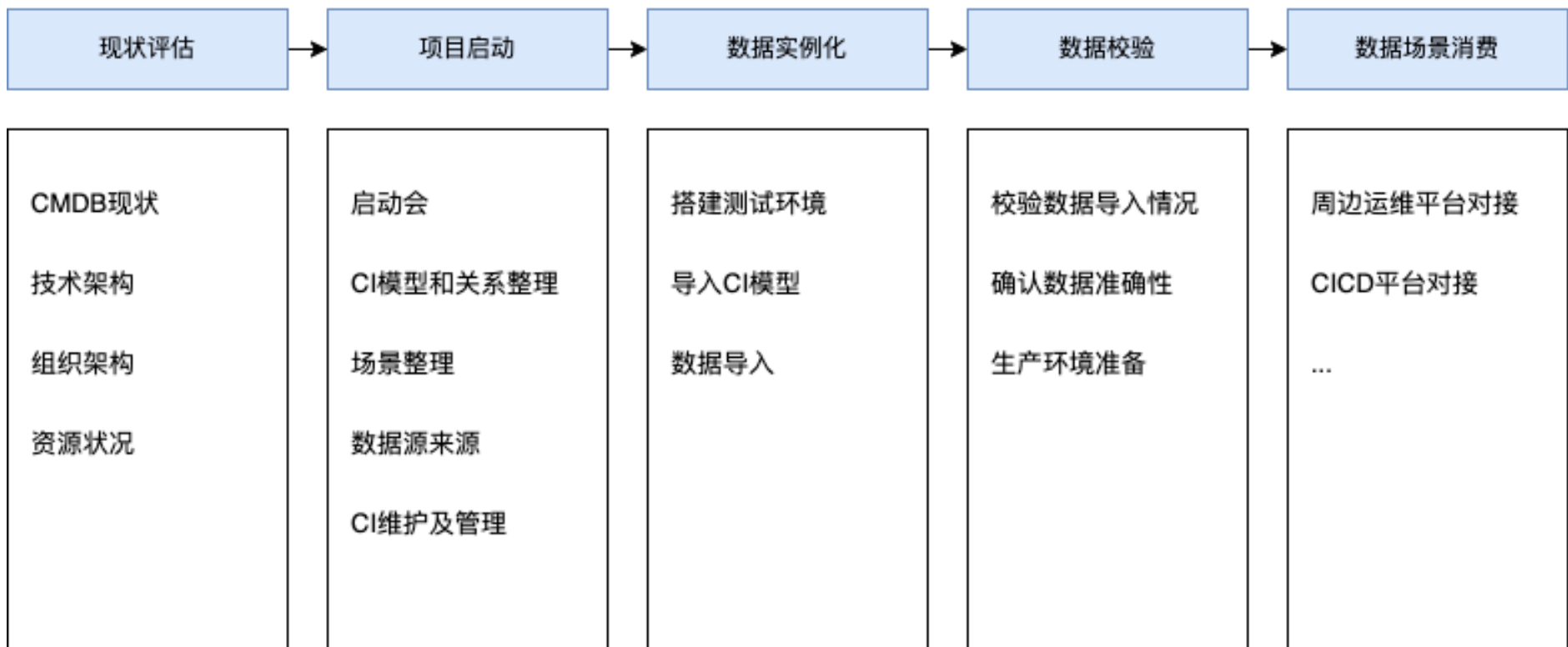
以应用为例：  
应用运行在物理主机上，应用的DB安装在服务器上，服务器链接网络交换机，机柜支撑物理设备。。。

按照梳理的关系分析整理出节点类型和边类型

最后定义出以应用视角的图模型

# 如何落地

B站元数据中心落地实施流程



# 关注数据质量

## PART 04

# 制度规范

## 规范要求

明确定义CMDB平台的作用，  
以及与其它业务系统间的关系

明确定义资源的管理过程以及  
责任人和责任平台

明确定义资源的基线标准以及  
偏差管理办法

从服务业务场景的视角来规划  
和建设配置管理能力

## 流程要求

能够真实反映资源状况

能够完整的包含所有的资源信  
息以及资源间关系

全局唯一的权威数据源

数据能够被用户及系统方便，  
及时和高效的获取

## 组织要求

成立统一的配置管理能力建设  
主体

各个业务团队明确配置消费和  
完善的责任

形成配置管理讨论、优化和需  
求收集的机制

## 平台要求

逐步实现配置自动发现，自动  
维护

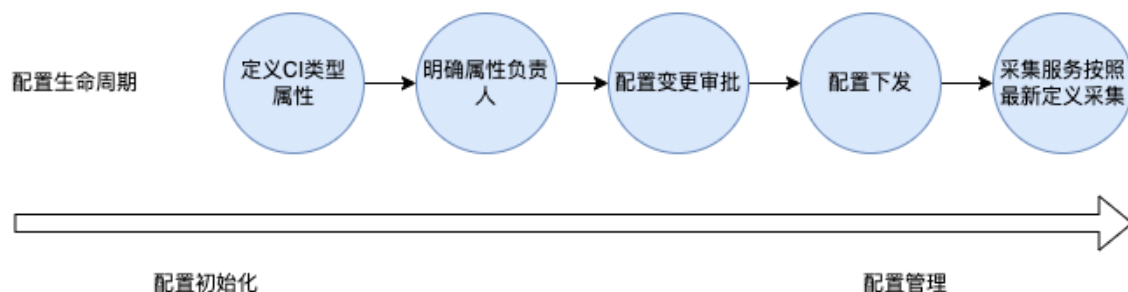
实时跟踪资源的状态及配置变  
化

模型灵活，能够根据业务需求  
实时扩展和调整

配置可视化，能够支持资源问  
题的分析和快速定位

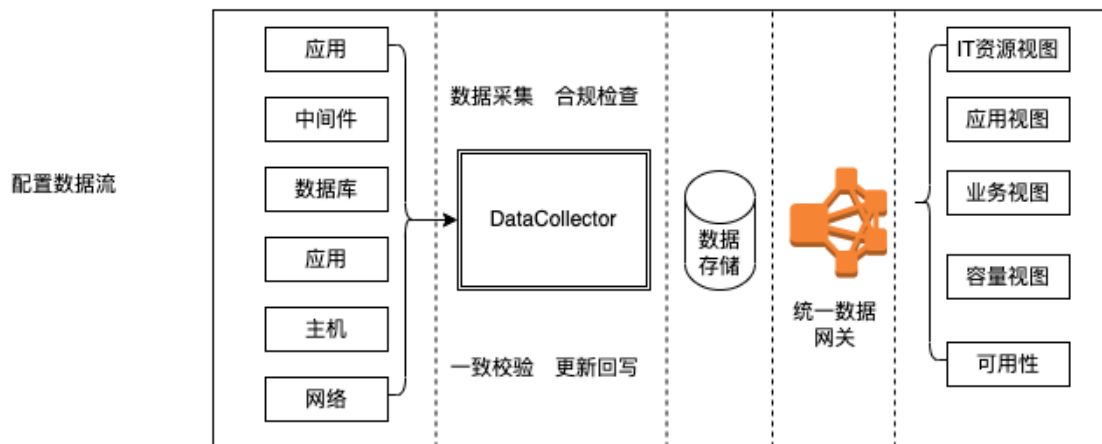
# 打造数据全生命周期闭环

聚焦数据闭环和数据可靠性建设



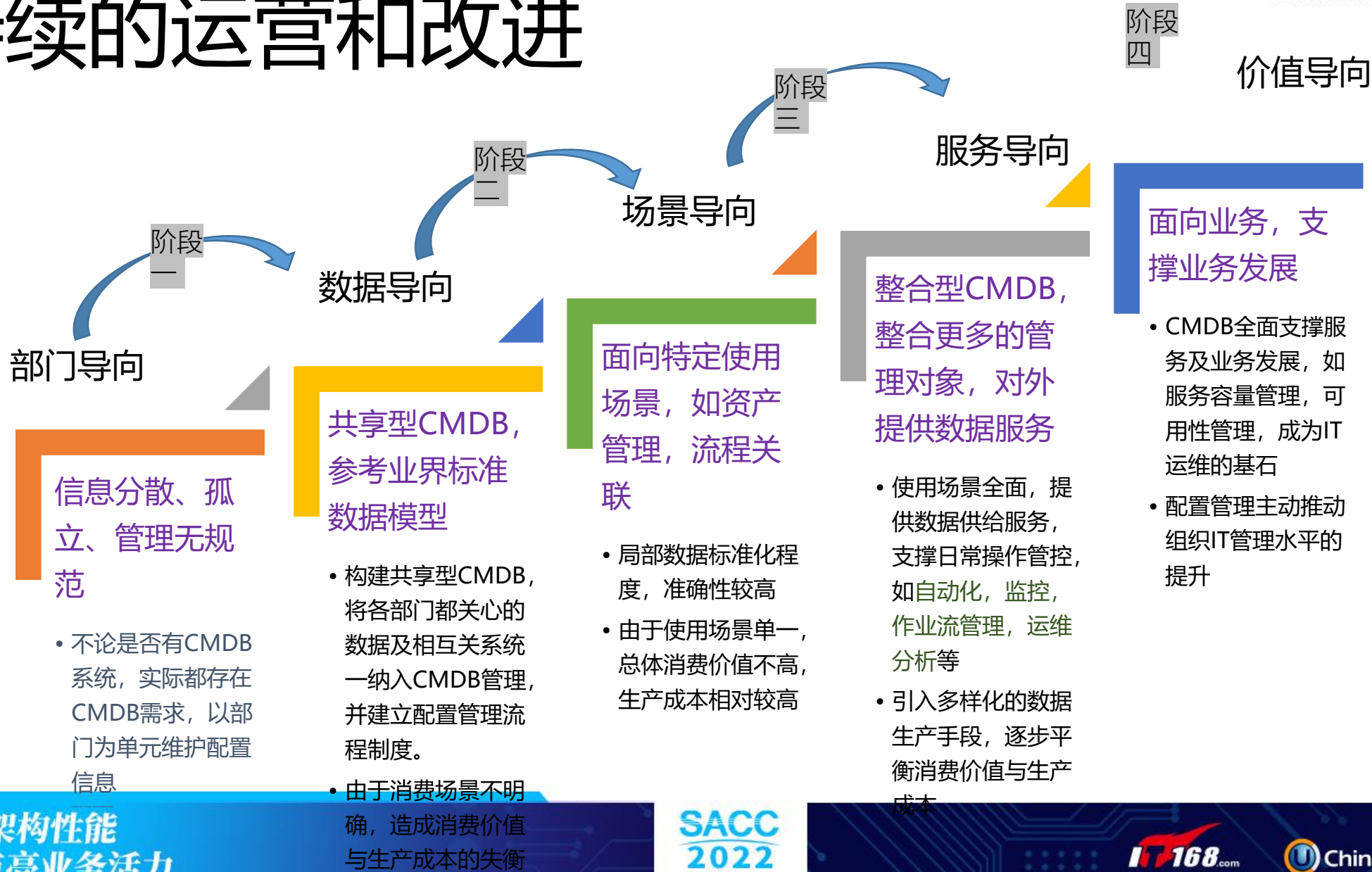
闭环:

- 依托运维流程引擎闭环，针对资产类数据建设重构
- 期间存在短期没有闭环的流程，我们对此类上报的数据没有匹配上后，将此类数据列入异常数据中，生成数据报表，反推
- 保留所有变更记录





# 持续的运营和改进



# 总结

## PART 06

# CMDDB的建设的四大阶段



# Thanks



哔哩哔哩技术

微信扫描二维码，关注我的公众号





THANKS

Architect