

GOPS

全球运维大会

2019 - AIOps 风向标

GOPS

深圳站

指导单位：



主办单位：



大会时间：2019年4月12日-13日

大会地址：深圳市南山区圣淘沙大酒店（翡翠店）

持续交付体系建设分享

孙雨辰 部门副总经理

目录



1

一直在路上

2

工具链建设

3

如何推动变革

4

大象也可以跳舞

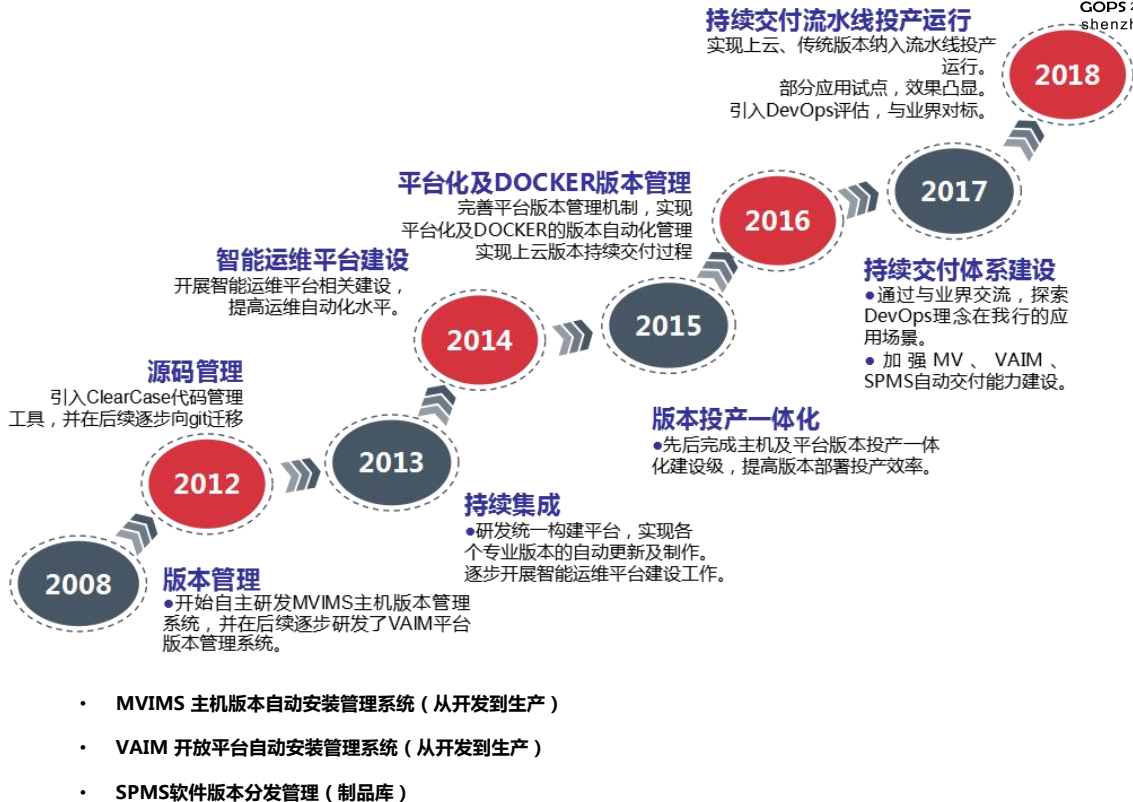
一直在路上

1. 传统瀑布型，季度版本。

- 开发、测试、投产
- 银行核心业务系统
- Z/OS主机核心

2. 月度版本，灵活投产。

- 直接发布
- 开放平台分布式系统
- 一路上云
- 流水线推进



目录

1 一直在路上

➔ 2 工具链建设

3 如何推动变革

4 大象也可以跳舞

工具链建设（源代码分支管理）

1. 计划性与随机性的选择

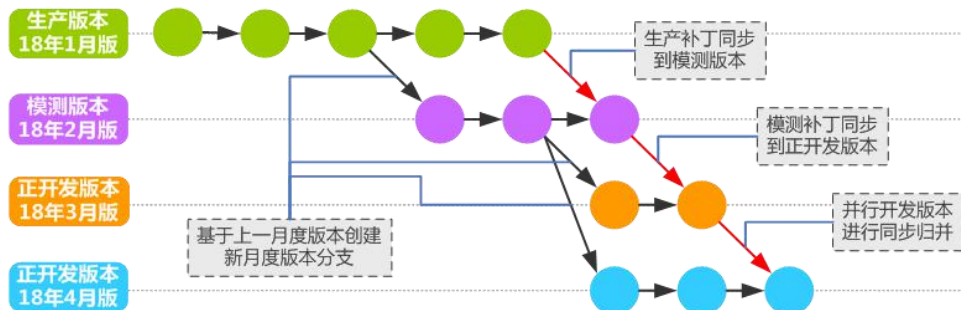
- 两种情况并存，灵活分支更加优于月度版本分支模式。
- 纳入到灵活投产的应用逐步转入到灵活月度版本特性分支模式。
- 月度版本分支复杂度低，但不灵活。灵活特性分支复杂度高，但发布灵活，发布分支和测试分支隔离性比较好，对开发团队的纪律要求比较高。源于业界的Gitflow模型。

2. 统一管理与分治

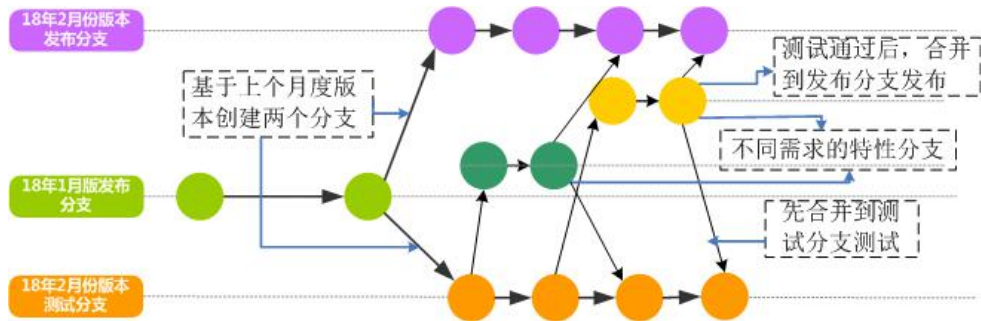
- 月度版本模式需要特定的配置管理员。
- 灵活模式没有特定的配置管理员，开发人员自己管理自己的特性分支。

3. 无论哪一款，总有一款适合你！

传统的月度版本分支模型



灵活的月度版本特性分支模型



工具链建设（元数据及数据库版本管理）

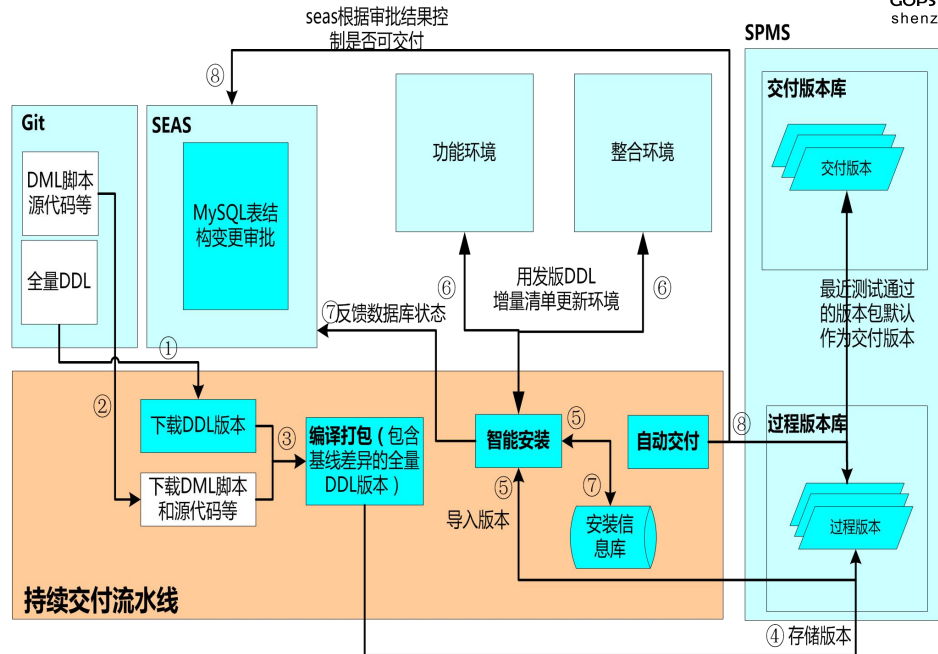
1. 主机时代计划和审批比较严格

- 约2.5万个COBOL程序、5000多张表，效率可以接受。按照单个程序、单个表的力度来管理。
- 先审批再构建、部署、测试、交付。

2. 分布式时代效率严重受阻

- XML方式管理DDL，引入开源。
- 进入持续交付流水线，自动构建、部署、测试、交付。
- 构建时，生成数据结构回写SEAS，审批不通过不能交付。

3. 人生苦短，千万不能因为审批耽误了效率！



SEAS：自研管理元数据的产品；**SPMS**：自研制品库产品。

持续交付流水线（VCDS）：自研从构建管理、部署安装、持续集成、自动交付、自动发布、自动投产部署的流程调度产品。

工具链建设（传统构建到构建云）

1. 构建是一种转换

- 源代码到目标的转换
- 数据结构到DDL的转换
- 配置文件到配置中心K-V的转换

2. 构建是一种依赖管理

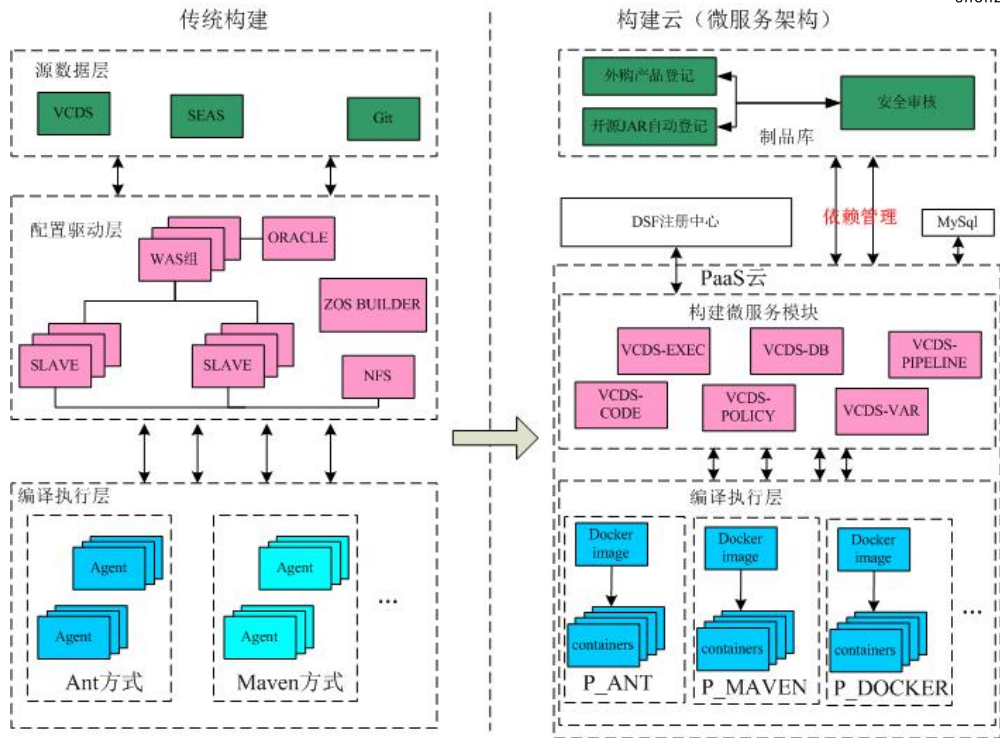
- MAVEN命令，规范化BUILD.XML扫描。
- 写入都制品库，标识关联。

3. 构建是统一的入口

- 应用服务节点模型入口
- 版本目录标准化入口

4. 构建是一种管控

- 引入黑名单开源JAR包，禁止构建。
- 代码扫描不合规，禁止构建。



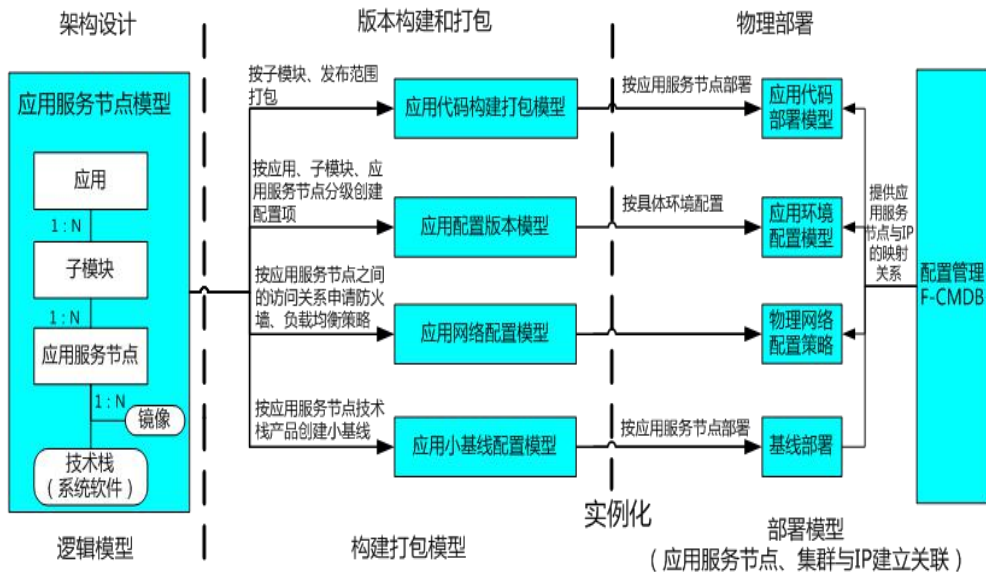
工具链建设（应用服务节点模型）

1. 车同辙、书同文

- 按照应用节点模型构建、打包、交付、部署。
- 按照应用服务节点模型管理应用级CMDB。

2. 沟通的纽带

- 开发、测试、运维不同的群组，互相沟通与交接采用相同的应用服务节点模型。



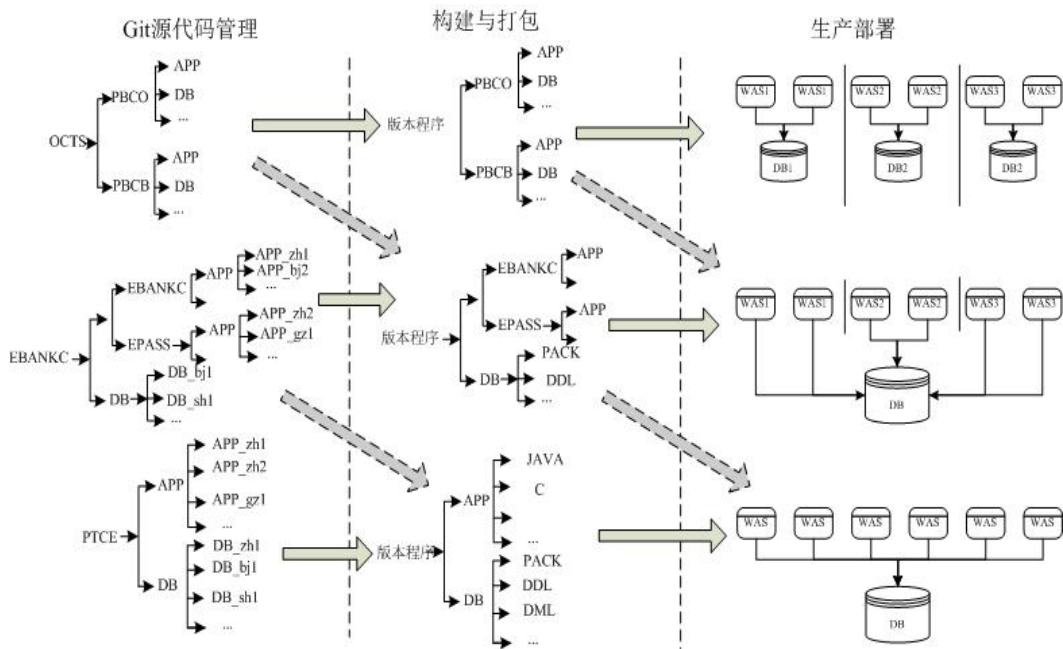
工具链建设（版本目录标准化）

1. 不同实体分类

- 程序、数据管理、文档不同分类。
- 开发按照项目维度分类，投产部署按照应用维度分类，不同应用下可以有不同项目的版本内容。

2. 平台化开发必须标准化

- 相同应用下不同的子系统及应用服务节点，各自不同的版本目录。

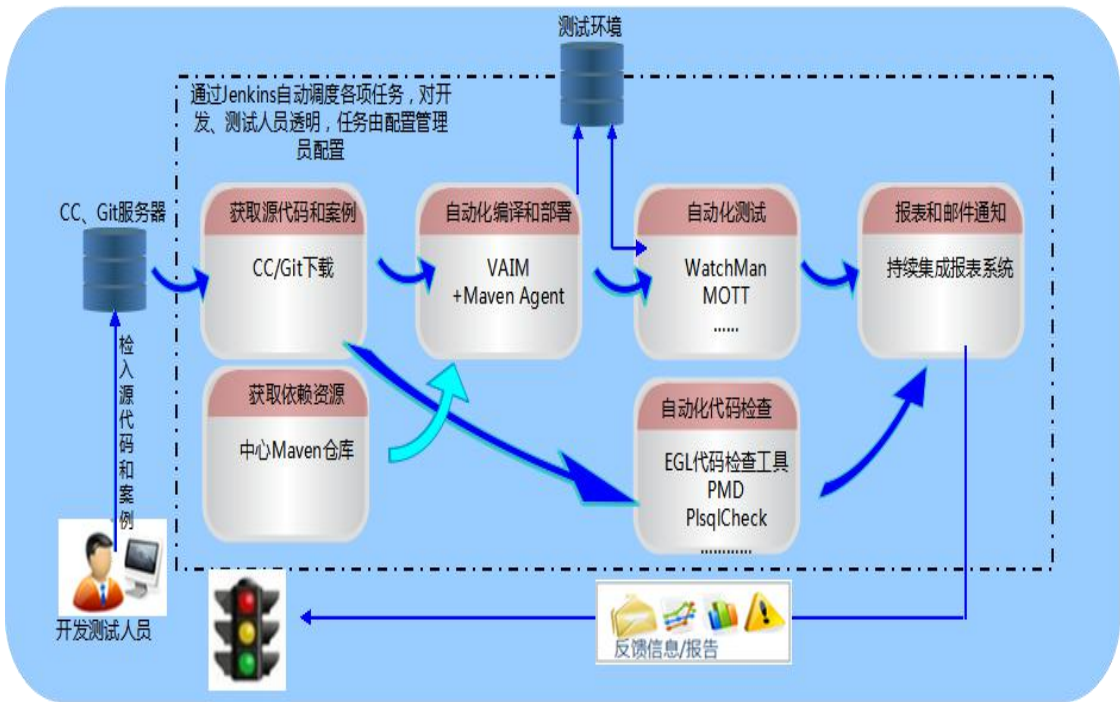


工具链建设（持续集成）

1. 各阶段作业

- 各类不同的分支策。
- 代码检查作业（COBOL、JAVA、PLSQL、EGL、COBOL）主要侧重于规范的检查。
- 版本构建及部署，调用VAIM系统。
- 环境检查及冒烟测试。
- 自动化测试。

2. 基于Jenkins的流程编排。



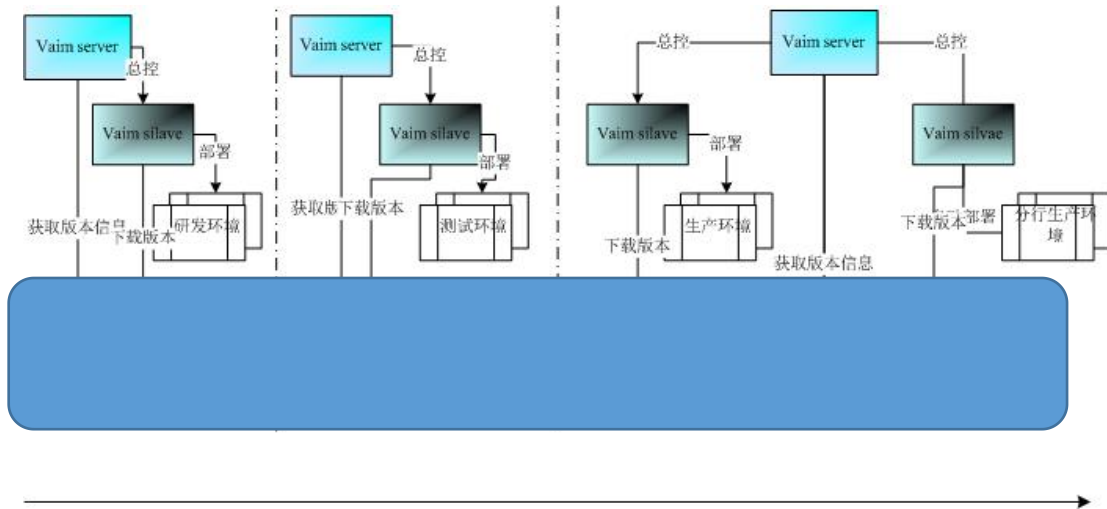
工具链建设（自动部署）

1. 自研的Server-agent产品

- WAS-SLAVE-Agent架构。
- 三中心部署，三中心互通。
- 分行、海外连接到数据中心SERVER。

2. 与制品库、容器镜像联动

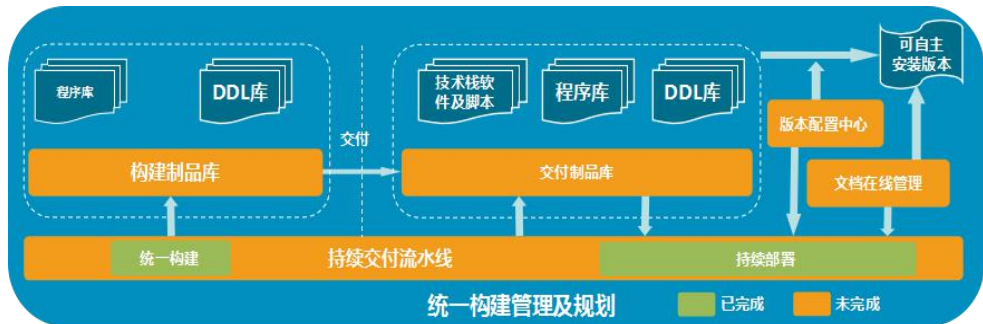
- 实现版本预下载。
- 丰富的标准组件。
- 开发团队可定制部署策略。



工具链建设（制品库及软件分发）

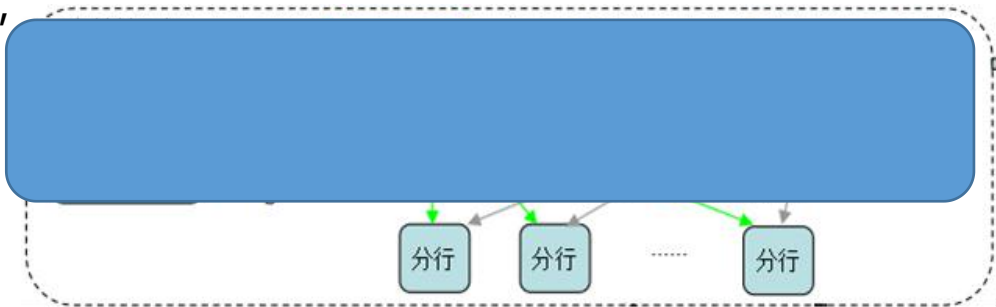
1. 多个阶段逻辑组合产品

- 以自研SPMS为核心，包括构建制品库、交付制品库，外购软件管理库、开源软件管理库、MAVEN库、容器镜像库。
- 兼具制品保存及制品分发功能，总分结构，双活节点。



2. 依赖管理信息库

- 构建扫描后，依赖关系扫描登记到制品库。



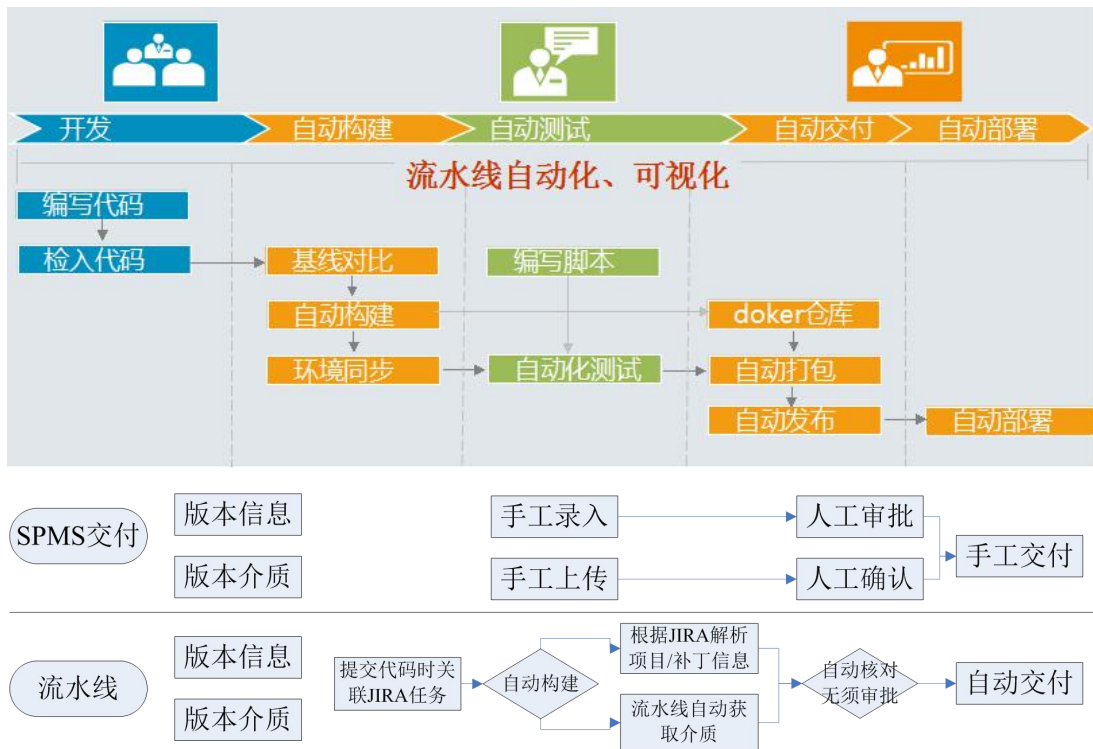
工具链建设（持续交付流水线）

1. 从构建到投产

- 串接多个子系统
- 全线可视化
- 实现线上线下联动

2. 多类型流水线

- 持续交付流水线
- 直接发布流水线
- 紧急补丁流水线
- 紧急变更流水线



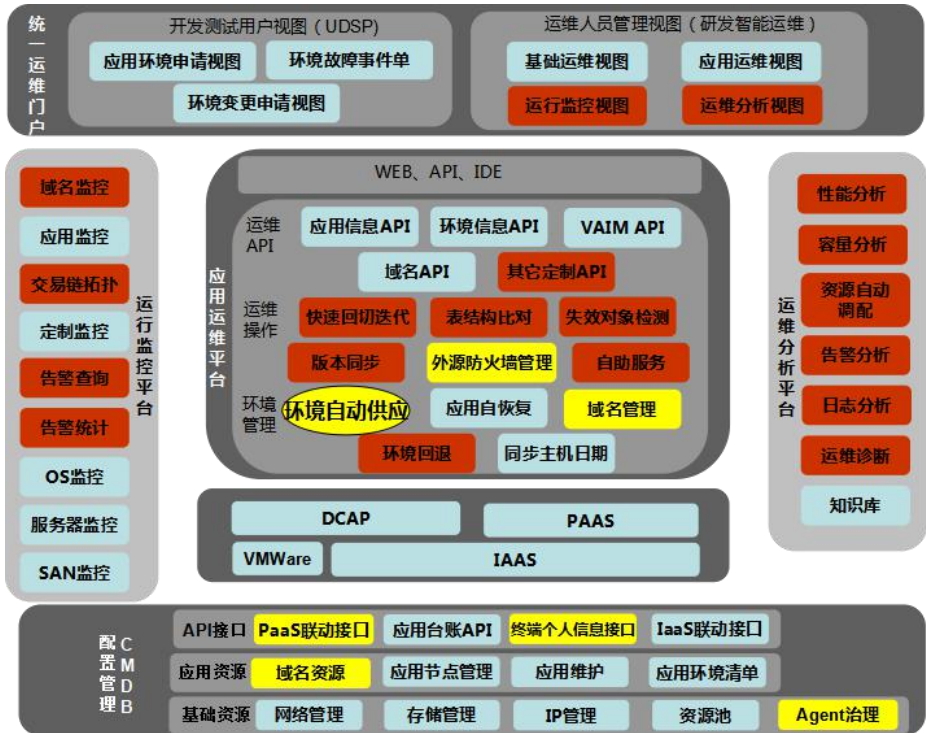
工具链建设（环境管理）

1. 内部的智能运维平台

- Server agent方式，能让agent活着不是一件容易的事情。
- 数据是基础，分为监控、分析、展现等多个模块。

2. 环境管理是资源和配置的一致性管理

- CMDB分为系统CMDB和应用CMDB。
- 能让CMDB的数据保持鲜活也不是一件容易的事情。
- 数据全部自动化收集也是一种奢望，手工也是必须有的。



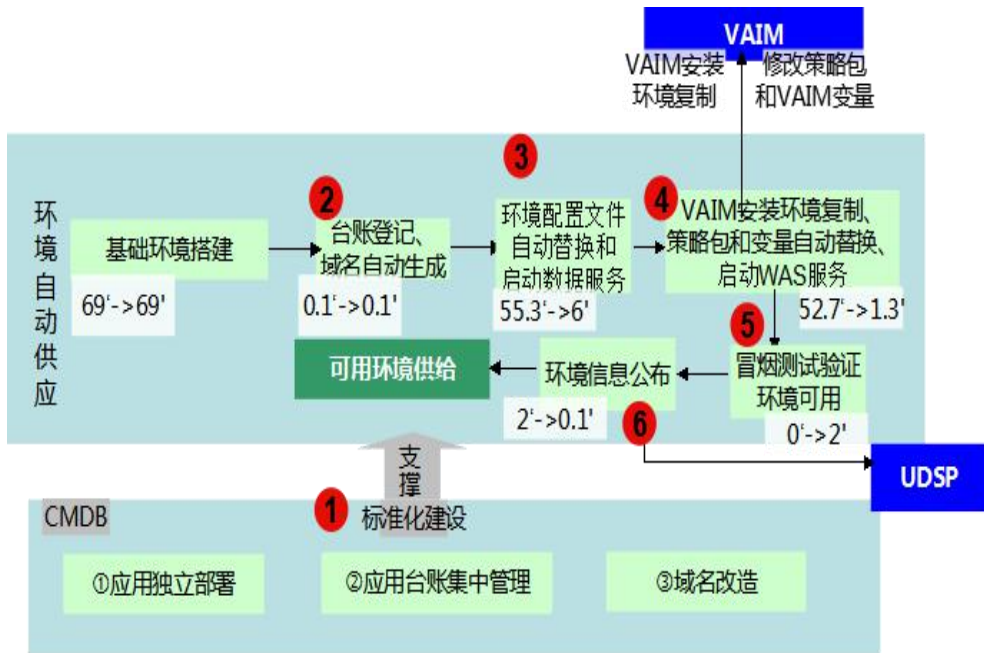
工具链建设（一键式环境克隆）

1. 多版本环境的并行开发使用

- 环境需要基于标准化建设，例如可域名访问，信息进入CMDB，独立部署等。
- 完成各类配置的自动替换。

2. 在环境自供应前的一种折中办法。

- 一种标准的版本包。
- 标准的基础环境，测试环境随时可生成。



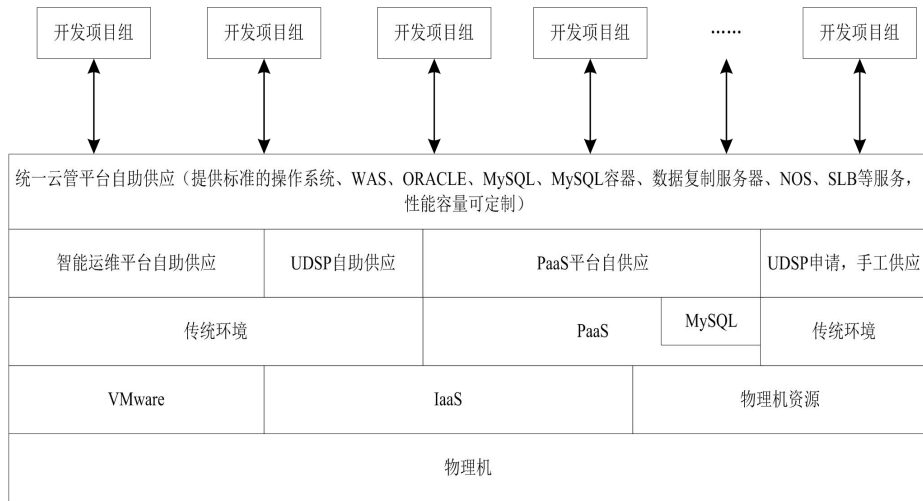
工具链建设（自服务统一管控平台）

1. 标准环境一键式供应

- 包括多个标准操作系统、中间件、数据库模板，可以实现标准环境供应。
- 基于智能运维平台及PaaS平台的操作系统或容器自供应。

2. 原来是基于多入口方式

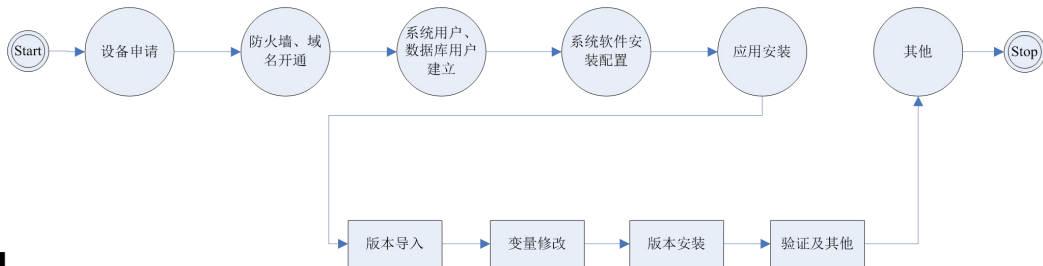
- 运维人员---智能运维平台。
- 开发人员---UDSP自助供应。
- 物理机需要手工供应。
- PaaS可完成自供应。
- IaaS为基础设施，主要供应操作系统虚拟化。



工具链建设（文档在线管理）

1. 版本文档模板化，减少错误率。

- 整合不同维度的文档，开发按照项目，运维按照应用，同应用不同项目可以随时整合。



2. 文档作为独立版本实体，在多中心流转。

- 不同的岗位可提前订阅，提前阅读了解项目情况。
- 各环节可以提前修订文档，减少交接过程。



工具链建设（配置中心建设）

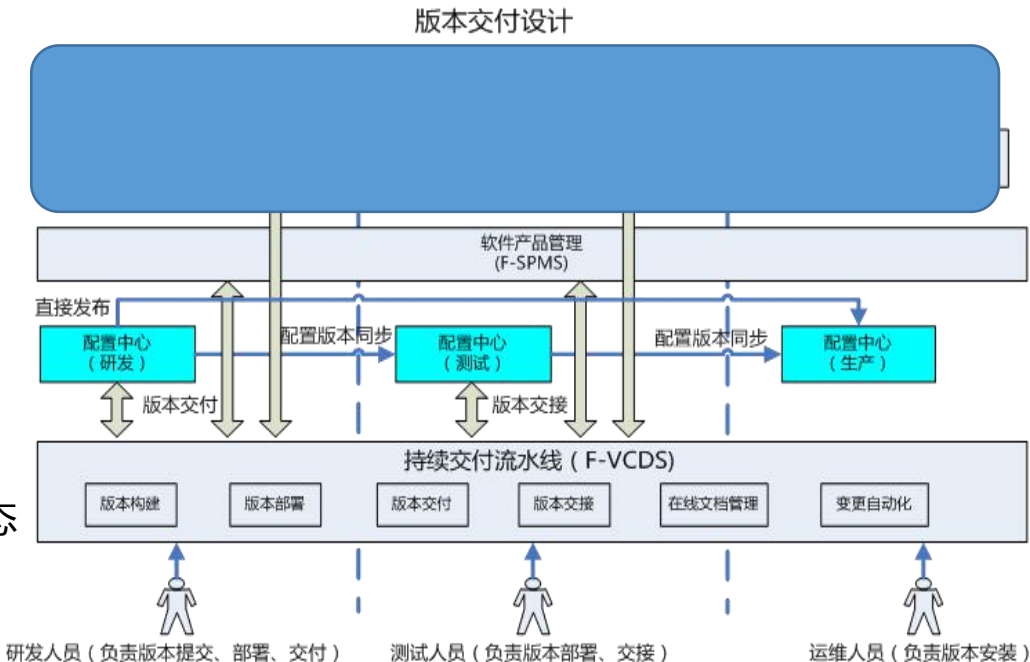
1. 配置的集中管理

- 系统软件配置
- 应用网络特性配置
- 应用软件配置

2. 同时管理静态配置和动态配置

- 部署静态配置管理，环境基线静态配置管理。
- 应用动态配置管理

3. 基于APOLLO开源工具实现



目录

1 一直在路上

2 工具链建设

 **3** 如何推动变革

4 大象也可以跳舞

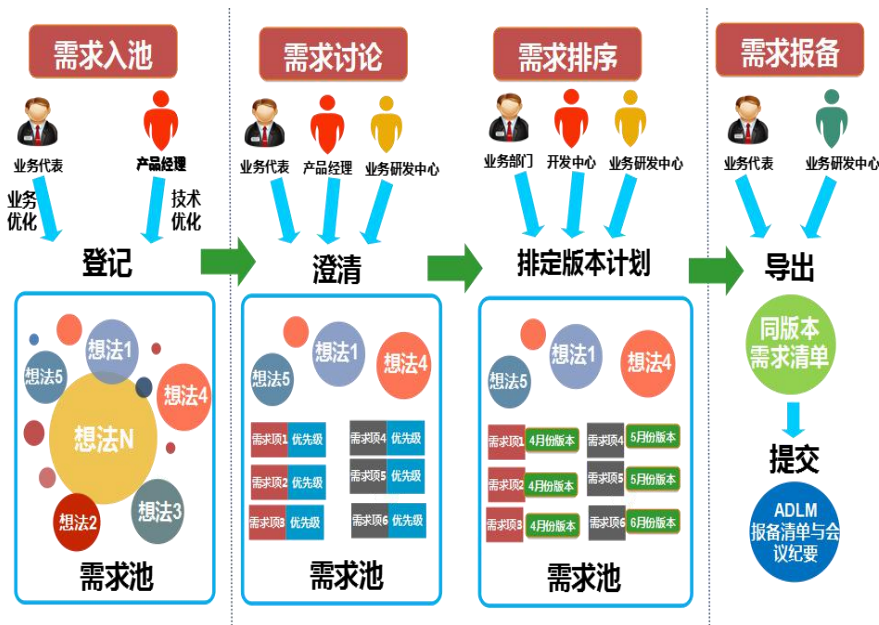
如何推动变革---开发模式改变

1. 流程太长

- 按照项目管理改变为按照产品管理。
- 需求不确定，需要反复澄清。
- 需求没有细化，版本计划周期太长。

2. 投产窗口太少

- 增加灵活投产。



如何推动变革---开发更需要改变

1. DevOps千人千面，实际情况差异很大

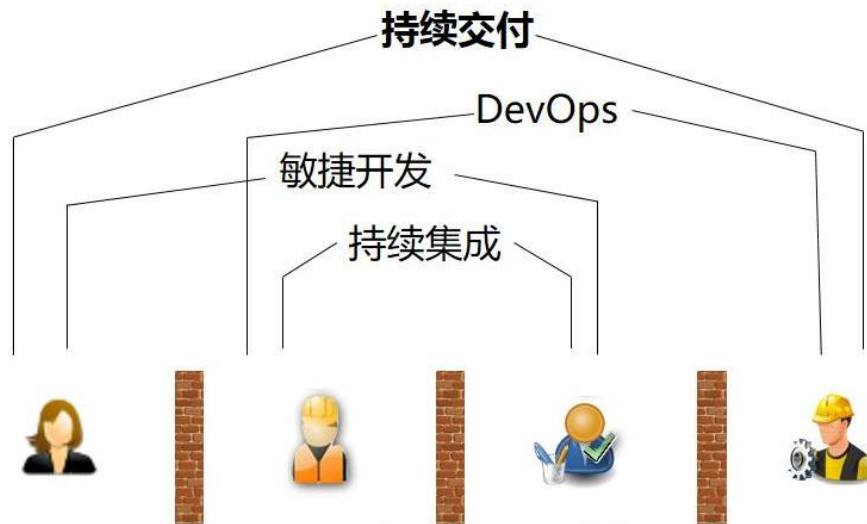
- 康威定律。
- 背景不同，历史不同，解决方法不能相同。

2. 工具链是基础，但不是全部

- 各类流程和方法也需要改变，例如：源代码管理方式，开发团队的工作习惯。

3. 实践是检验真理的唯一标准

- 论证和想法很多
- 干就完了



如何推动变革---应用架构需要优化

1. 解耦会解决很多问题。

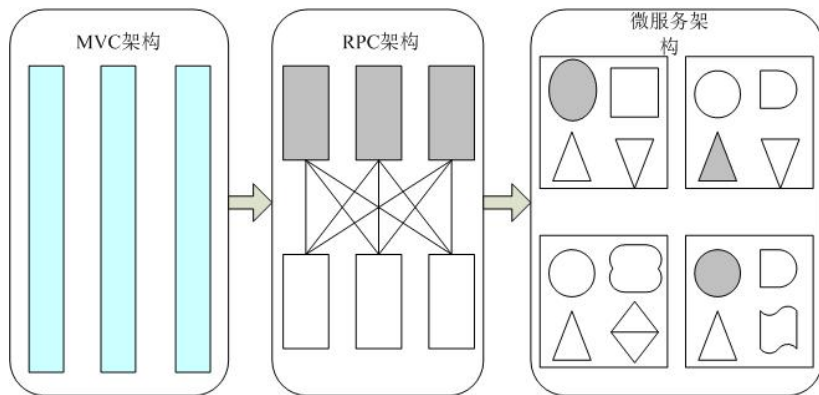
- 耦合度过高，无法做到灵活投产，应用间相互等待，会拖慢整个节奏。
- 大部分应用尤其是主机核心应用，还需要按照月度版本模式前进，多应用耦合，需要同时发布版本。

2. 独立系统很快使用持续交付方法

- 一张白纸好画图

3. 分布式微服务架构

- 各应用相对独立，比较容易改造。



目录

1 一直在路上

2 工具链建设

3 如果推动变革

➔ 4 大象也可以跳舞

大象也可以跳舞

1. 开发、测试、投产全线贯通

- 工具链不断完善，大部分可以自动化。
- 开发方式逐步改变。

2. 兼顾大版规划与持续交付

- 核心系统大版本，求稳定，少变更。
- 外围，对客系统灵活版本，多变化。

3. 兼顾上云与传统环境

- 上云非上云齐头并进最终合一。





Thanks

高效运维社区
开放运维联盟

荣誉出品

想第一时间看到高效运维社区
的新动态吗？

