GOPS 2016 Shanghai

GOPS

2016 全球运维大会

重新定义运维

上海站

会议时间： 9月23日-9月24日

会议地点： 上海·雅悦新天地大酒店

主办单位： 开放运维联盟 OOPSA Open OPS Alliance　高效运维社区 GreatOPS Community

指导单位： 数据中心联盟 Data Center Alliance

你是否同意如下所说？

- 运维只是持续交付非常后端的一个环节，运维来承担这个职责感觉有点过
- 持续交付是DevOps的核心实践，涉及多个部门，Hold不住
- 运维把部署和监控从人工变成自动化，已经足够

运维，为DevOps持续交付而生

优维科技，王津银
人称老王

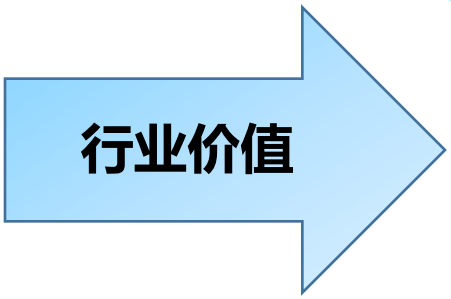
About Me

运维老王

王津银

10年运维经验（腾讯5年＋YY＋UC＋运维创业经验）

2年开发经验，无惧于开发的技术交流和PK



http://www.easyops.cn

隔壁老王

优维科技创始人，致力于DevOps运维推广

互联网运维杂谈作者>2.5w人

应用运维标准规范组联席组长兼神秘

运维需要跨界的突破，重行定义运维价值

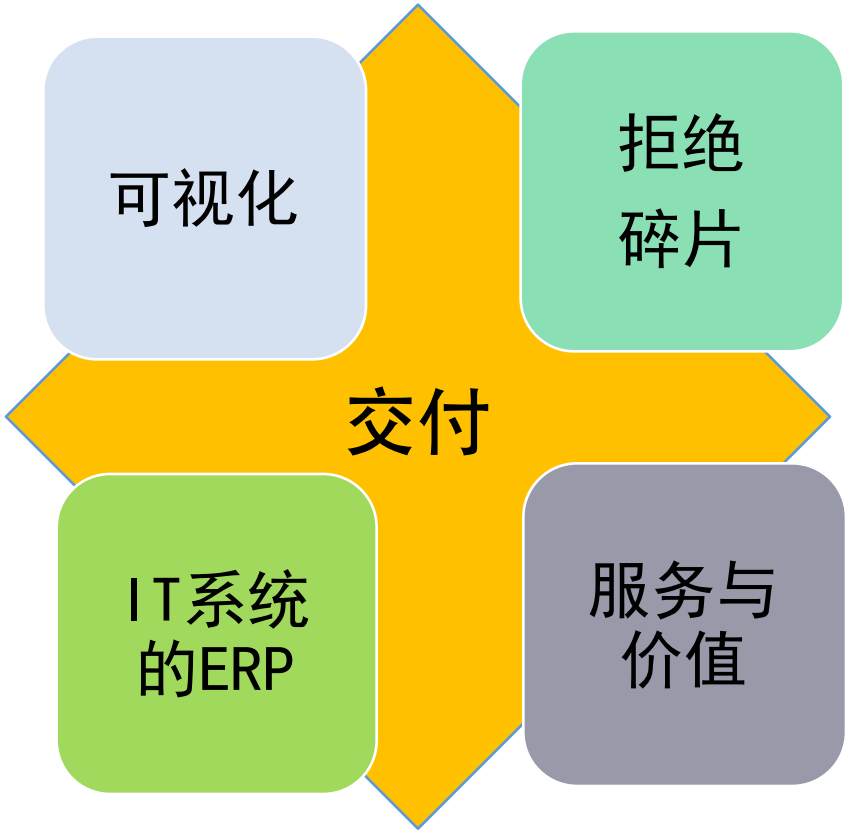
目录

Contents

运维的本质探讨

- 职能分析
- 惨状分析
- YY状态分析

运维的本质探讨



- 交付，运维及服务，运维及交付。
- 可视化，可视化整个运维过程，甚至是It过程。
- 拒绝碎片，产品、组织碎片带来都是运维高成本的核心原因。
- 服务与价值，传递给客户的能力。
- It系统的ERP，运维平台的定位。

目录

Contents

持续交付的组织要素

- 研发只会写代码，测试用例全靠测试同学维护，更别提线上部署规范、架构和性能优化
- 测试无法和研发形成有效沟通，测试止于黑盒，浮于功能，覆盖点不全面；
- 运维和研发老死不相往来，线上故障排查困难重重，逻辑架构想不清，理还乱，更别提数据挖掘和深度优化

不要期望组织先改变，而是技术先行

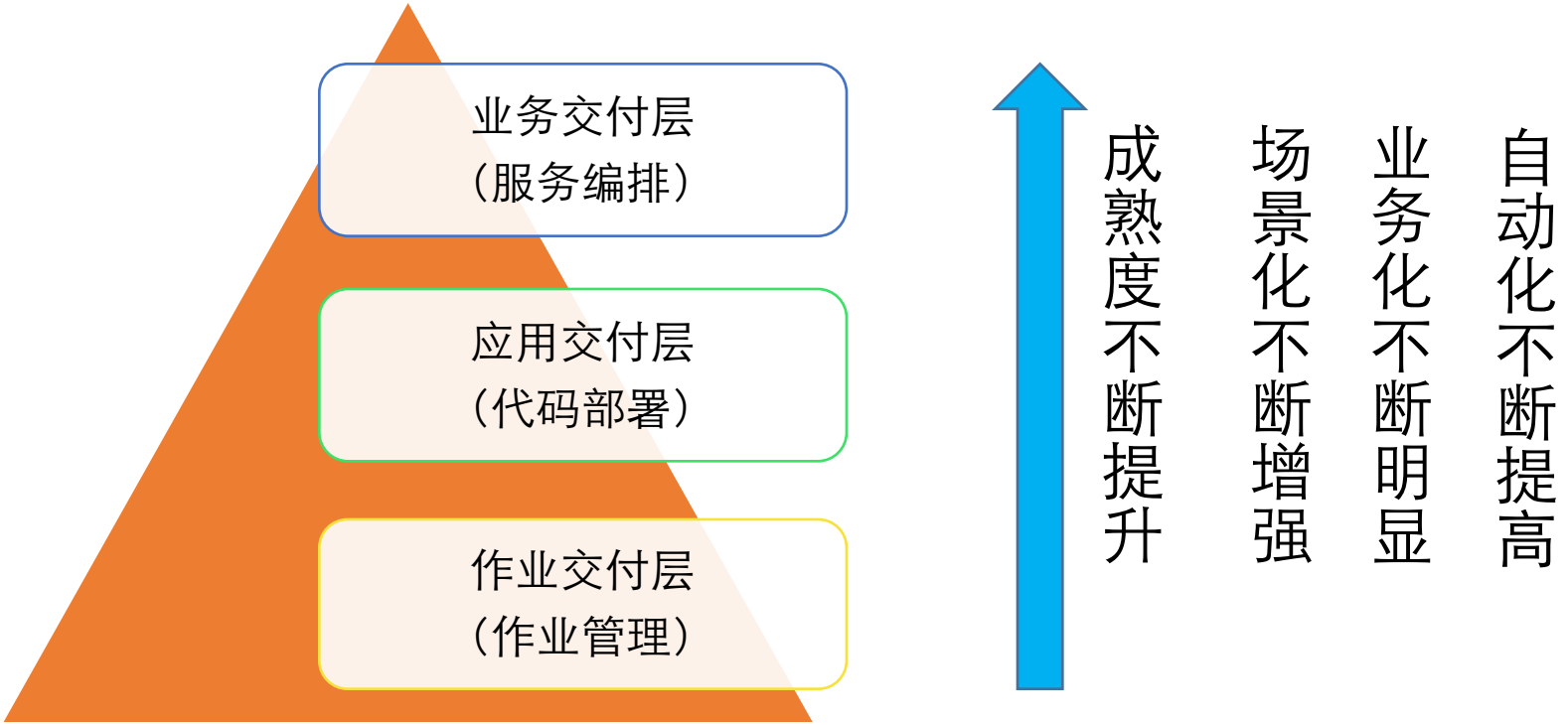
持续交付的标准化要素

去除职能差异化和技术能力差异化:

- 角色要素：产品/运营/研发/测试/业务运维
- 场景要素：版本操作/资源交付/业务交付
- 技术要素：硬件、应用、中间件
- 能力要素：海量/中小企业

标准化是一致性的体现

持续交付的平台要素



平台化是一致性的唯一保证和手段

持续交付的架构要素

巨石化架构

- 代码依赖而非依赖
- 升级中断而非平滑
- 服务耦合而非服务治理
- 配置和程序耦合而非分离
- 决策集中而非分布式

服务化架构

- 全局交付价值链优化
- 交付过程工具化，pipline化
- 单服务升级而非全平台升级
- 代码和配置各自升级
- 灰度发布

目录

Contents

持续交付的整体架构

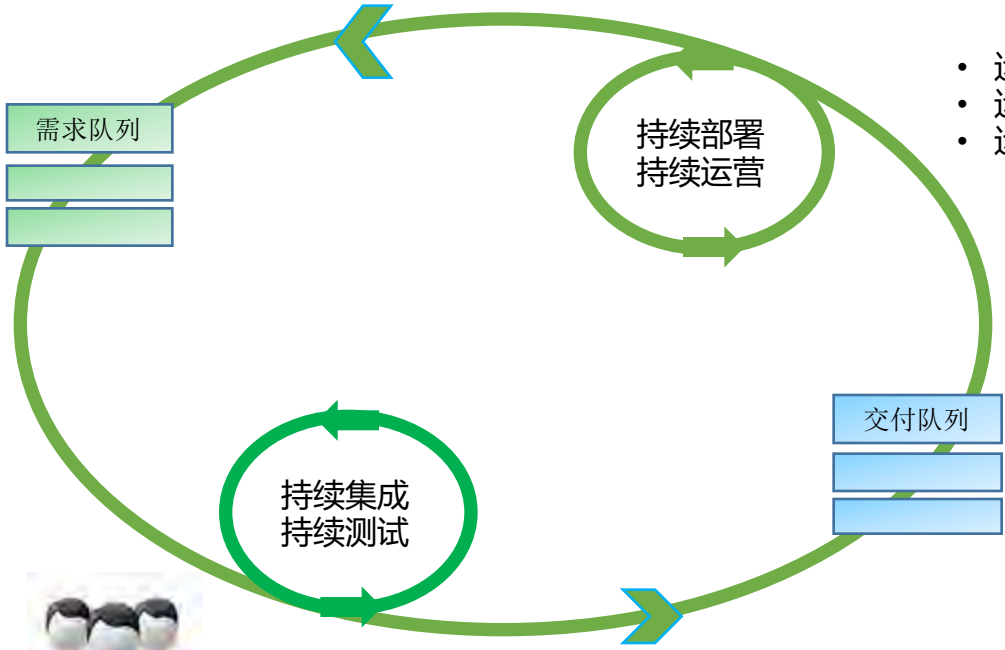
1、持续交付的目标是实现价值交付，内在目标是降低风险、提高质量。

2、可靠可重复的流水线式持续交付的基础，在该流水线上实现持续集成、测试、部署、反馈等端到端流程整合。

3、实现持续交付需要有交付能力管理过程与技术、团队与实施。

持续交付的流水线实现

- 用户反馈产生的需求
- 运行持续反馈产生的需求

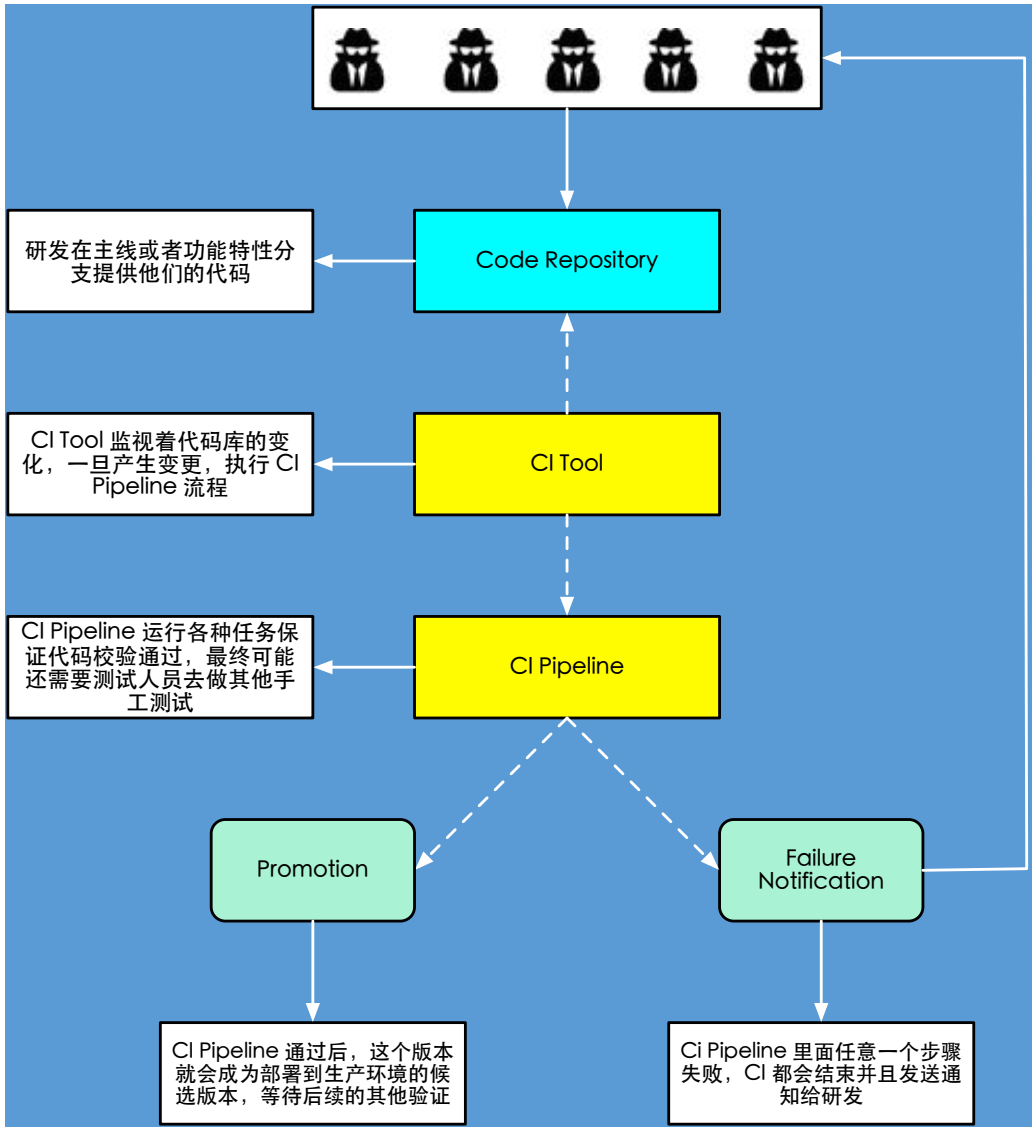


- 运维标准化
- 运维平台化
- 运维PaaS化

- 人工构建库

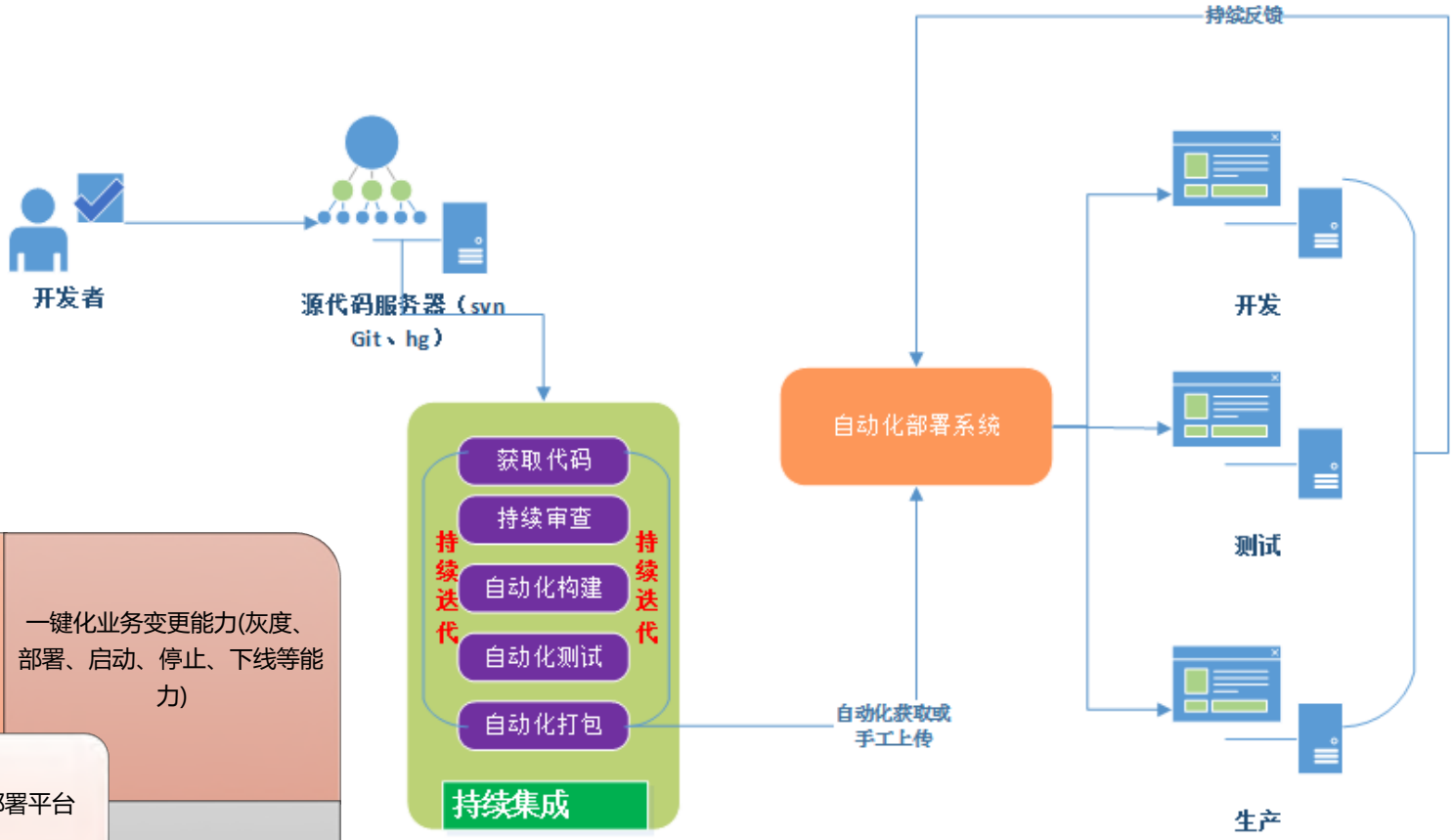
- 技术架构服务化
- 持续集成与测试
- 用户验收测试驱动研发
- 冒烟测试和探索性测试

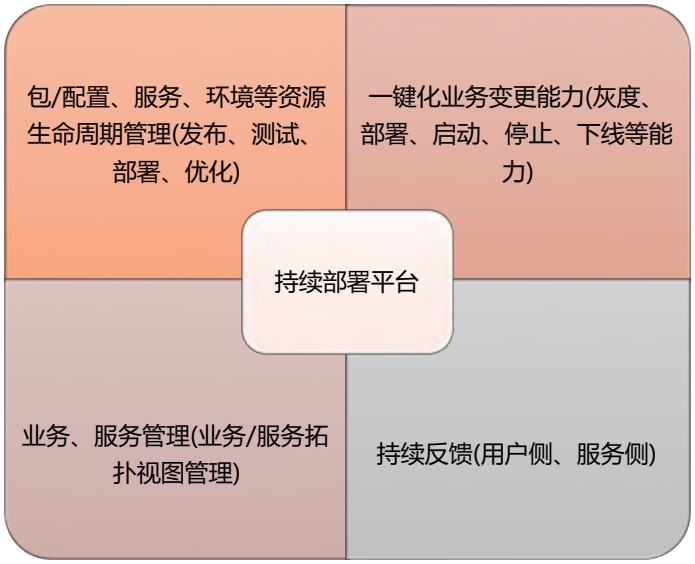
持续交付至持续集成



- 越早集成，Bug修复成本就越低。
- CI需要自动化和持续部署系统对接
- CI可以集成代码校验、单元测试、UAT等
- CI的标准化工具平台Jenkins

持续交付之持续部署

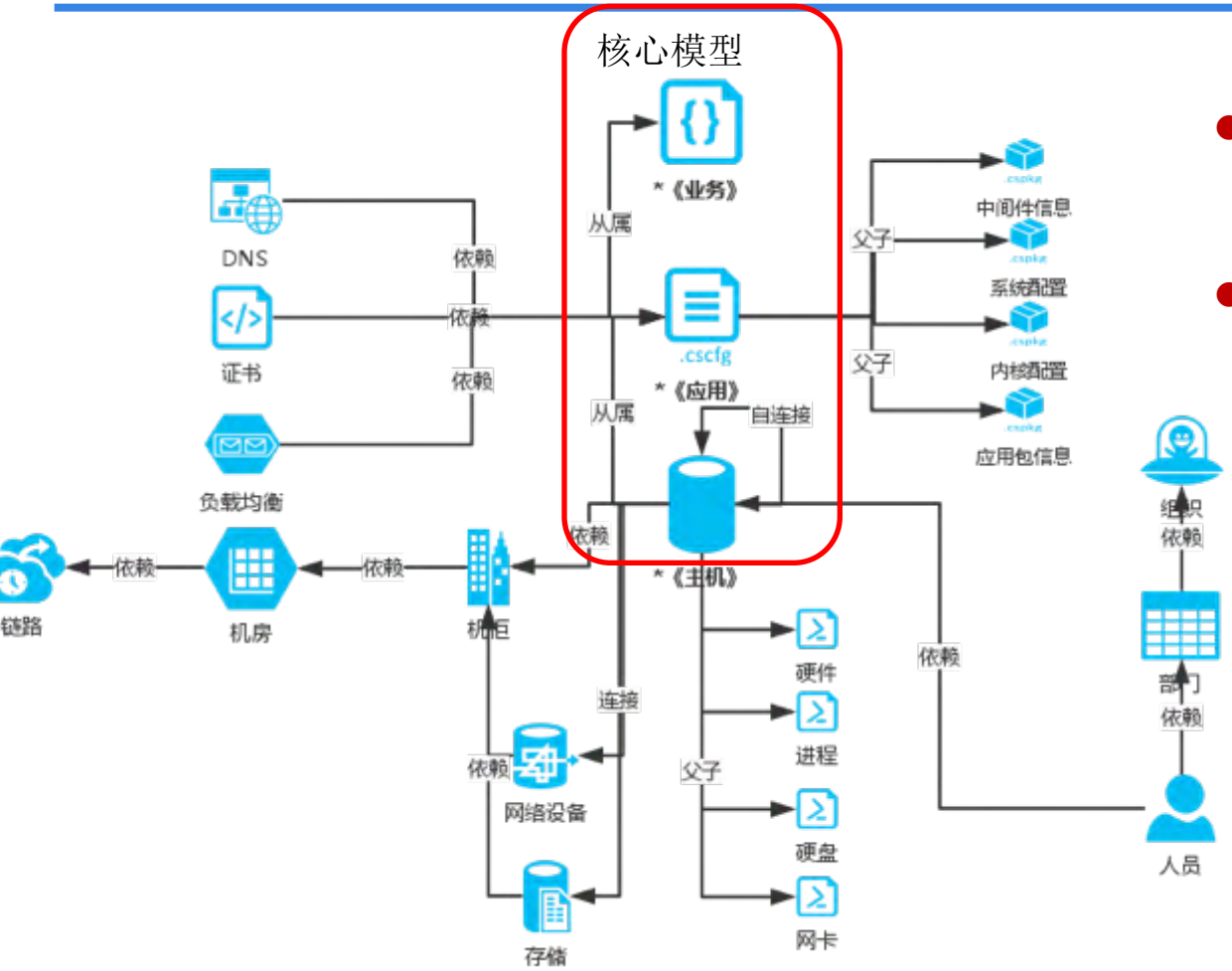

持续反馈
开发者
源代码服务器（svn
Git、hg）
获取代码
持续审查
持续迭代
自动化构建
自动化测试
自动化打包
持续集成
自动化部署系统
自动化获取或
手工上传
开发
测试
生产


包/配置、服务、环境等资源生命周期管理(发布、测试、部署、优化)
一键化业务变更能力(灰度、部署、启动、停止、下线等能力)
持续部署平台
业务、服务管理(业务/服务拓扑视图管理)
持续反馈(用户侧、服务侧)

持续交付的平台架构


流程管理
权限管理
运维统一门户
运维服务目录
研发者服务目录
任务中心
状态中心
调度编排系统
web类业务
语音类业务
游戏类业务
互联网金融业务
金融类业务
运维服务层API（由各个服务平台提供）
ITIL服务管理（基础）
HelpDesk
事件管理
问题管理
突发事件管理
变更管理
配置管理
IT服务管理（高级）
容量管理
可用性管理
连续性管理
...
运维者服务平台
机房管理平台
交换机管理
LVS管理
FS管理平台
DNS管理平台
作业平台/配置管理
研发者服务平台
分布式NoSql存储
分布式Cache服务
统一RDS服务
统一文件、块、对象存储
名字服务
分布式消息队列服务
持续交付平台
持续构建
持续审查
持续测试
持续发布
持续反馈
...
统一监控平台
业务主动监控平台
服务级监控平台
应用级监控平台
平台服务级监控平台
基础设施级监控平台
ITOA分析平台
业务质量分析平台
机房质量平台
容量分析平台
日志分析平台
应用性能管理
用户体验管理
CMDB统一接口层(API)
配置管理CMDB
基础资源管理
服务器
网络设备
机架
存储
机房
IP
域名
虚拟机
对象关系
业务信息管理
应用包
包配置
crontab
进程端口
流程
工具
权限
系统配置
对象关系
逻辑资源
组织管理
代理商
人员管理
角色管理
服务商
运营商
地址管理
权限管理
IaaS，基础设施平台
OpenStack
公有云
私有云
服务器
网络
存储
机房

持续交付之IT资源管理

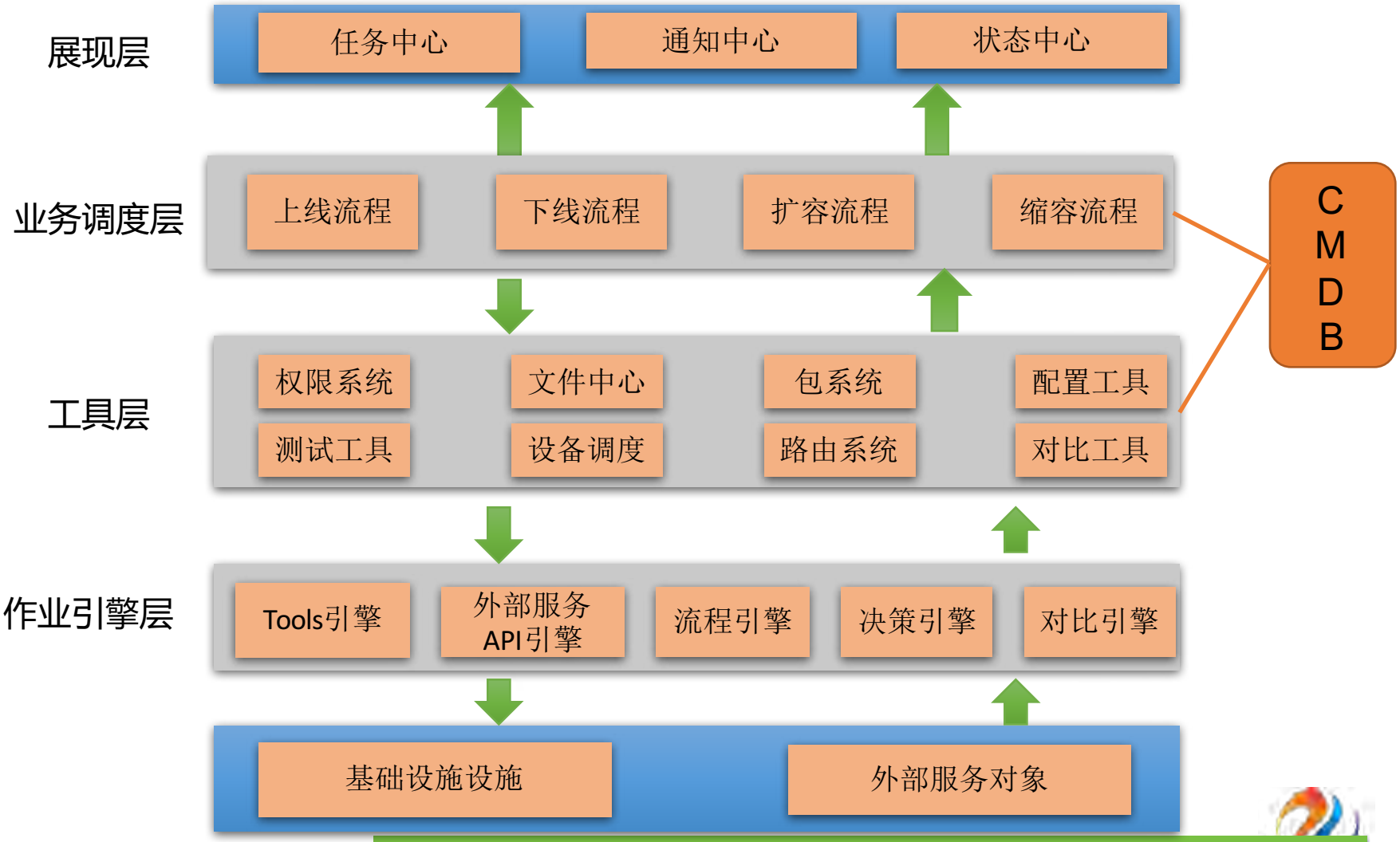


- CMDB分核心模型和扩展模型
- 建立以应用为中心的资源管理模型

CMDB管理的套路

1、CMDB名字应该改一下，叫IT资源管理
2、CMDB分核心模型和扩展模型
3、CMDB对象关系要简化
4、不要太迷信自动发现
5、CMDB要领导支持，团队理解一致
6、云计算的概念层次就是CMDB的层次
7、CMDB是你的资源及组织管理的快照

持续交付之持续部署平台



持续交付之标准化模型

	部署路径	属主	权限 (目录、文件)	代码	配置	日志	...
JDK	√	√	√	×	×	×	
JBOSS -EAP	√	√	√	×	×	×	
公共组件	√	√	√	√	√	×	
业务层	√	√	√	√	√	√	

➢ 构建 X 轴(属性)

- 所谓的 X 轴，是指企业IT系统每一层技术栈应该遵循的标准

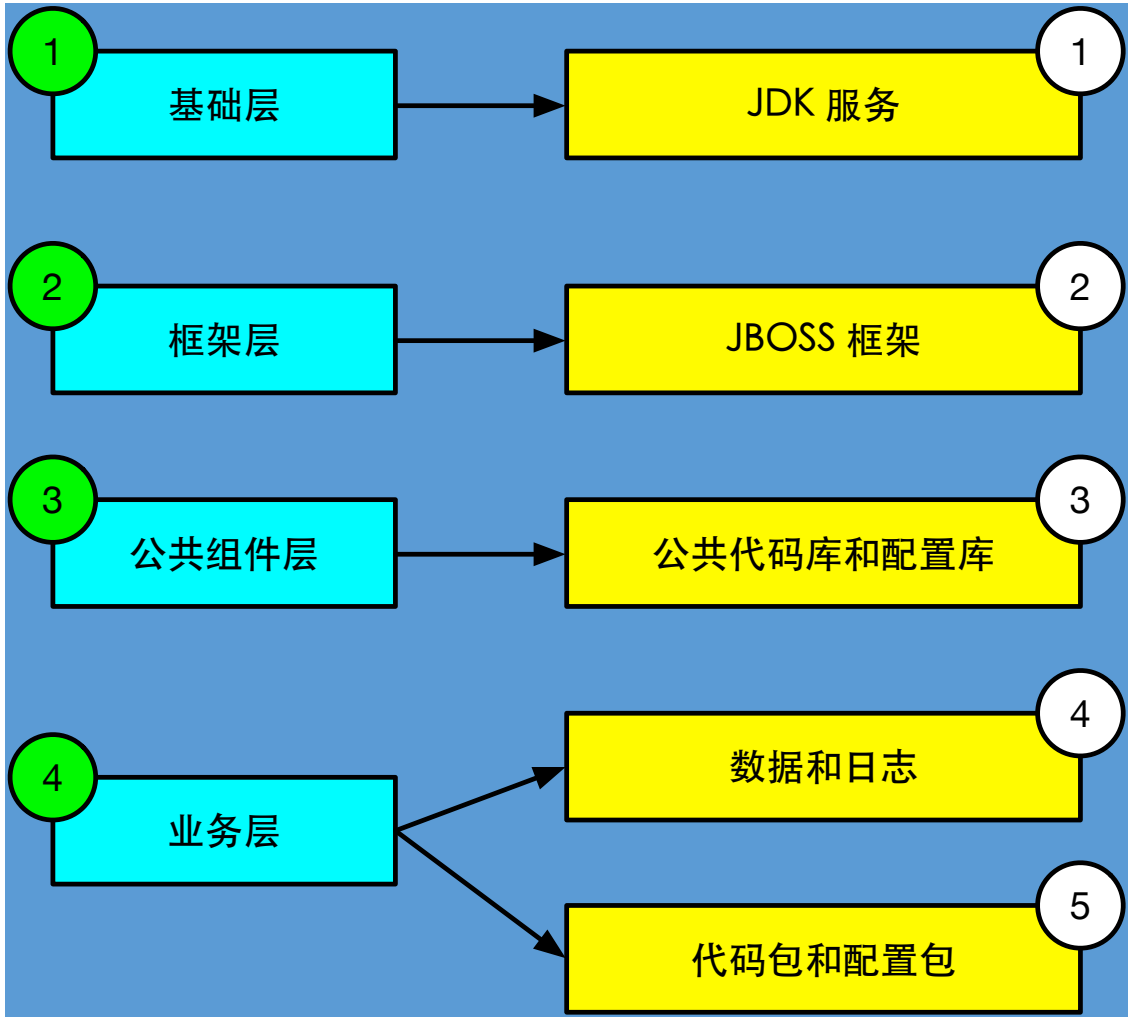
对每一层的技术栈进行深度分析，构建出实体应该具有的属性

➢ 梳理 Y 轴(实体)

- 所谓的 Y 轴，是指企业IT系统所遵循的技术栈

从企业的技术栈入手，从宏观上把需要标准化的实体梳理清楚

持续交付之标准化模型



- 从底到上梳理你的对象
- 该对象可标准化什么
- 对象的分离
- 对象内子对象的分离

持续交付之应用标准化模型

启动脚本

统一的启动脚本，通过传入参数来匹配不同的业务组件

实体隔离

不同的实体部署上必须隔离

数据隔离

数据需要写到数据目录或者数据卷上

日志实践

日志写到数据或者日志卷上；

规范的输出级别、内容格式、日志种类、轮替周期和定期清理

代码和配置

代码包无状态，一次打包，多环境流转；

配置包环境相关，甚至可以实现配置中心

目录

Contents

持续交付能力成熟度模型

能力维度 / 成熟度	持续管理和持续集成	环境管理	部署管理	发布管理	测试管理	数据管理（DBA）	配置管理	架构管理
5级 - 优化管理级	团队定期碰头，讨论集成问题，并利用自动化 / 更快地反馈和更好的可视化来解决问题。	更有效地管理所有环境，可以使用更新的技术来准备环境，比如docker。	准备工作全部自动化。如果适当的话，使用虚拟化和容器技术来完成环境的快速部署。	运营和交付团队定期沟通来管理风险，缩短循环周期。	生产环境回滚很少，缺陷可以立即发现并修复。	在两次发布之间建立了数据库性能和部署流程的反馈回路。	定期验证配置管理是否支持高效的合作，快速开发和可审计的变更管理流程	**所有技术架构中涉及到的组件都是服务化的。技术架构中的接口也都是服务化封装的。可以无感知变更**
4级 - 量化管理级	构建度量收集，可视化并采取行动。构建不能长时间处于失败状态	所有环境的管理都纳入了运维监控能力范围，提供了容量 / 可用性 / 一致性等多种监控手段。	精心计划的部署管理，对发布和回滚流程进行测试。对于变更的对象要有强制校验的能力，避免变更过程异常。	环境与应用程序的健康性得到监控，并有前瞻性管理；变更前置时间得到关注和监控。对于变更的对象要有强制校验的能力，避免变更过程异常。	质量度量和趋势跟踪。对于非功能需求进行了定义和度量	每次部署都进行数据库的升级和回滚测试。对数据库进行监控和优化。	开发人员每天至少提交到主干一次。只在需要发布时才会拉分支。	架构的服务化能力可以从监控 / 管理 / 调用能力 / 调用状态等多个角度衡量。
3级 - 已定义级	每次代码提交都进行自动化构建和测试。依赖关系被很好的管理，脚本和工具得到很好的重用。	开发 / 测试 / 生产甚至其它各种环境的标准化和规范化能力是一致的。建立了标准的环境管理过程。	软件部署使用全自动自助服务一键式过程。使用相同的过程向各种环境部署。	定义并执行变更管理和审批过程。监管及合规的条件得到满足。	自动化的单元测试盒验收测试。后者是测试人员写的。测试是开发过程一部分	数据库变更作为部署流程的一部分自动执行。	库和依赖被很好的管理。变更过程决定了版本控制的使用规则。	对关联的技术架构服务变更可以通过其提供的API接口来调用完成
2级 - 可管理级	定期的自动化构建与测试。任意一个构建都可以自动化过程重新从源版本控制库上构建	对应用涉及到的环境是有清晰的职责定义和owner定义，并在线可视化管理。	自动化部署到几种环境中。新环境的创建成本非常低。所有的配置都放置在外部，并做版本控制。	痛苦且不频繁，但能可靠地发布。从需求到发布可以做到部分可追踪	自动化测试是用户故事开发的一部分	对数据库的变更使用自动化脚本完成，而这些脚本与应用的版本相对应	重建软件所需的内容都进行了版本控制，包括：源代码 / 配置 / 构建和部署脚本 / 数据迁移	所有服务的变更是可以通过webconsole完成的，可视化完成
1级 - 初始级	软件构建是手工过程。没有对产物和报告进行管理	几乎没有环境管理的能力，应用的环境分布是混乱且不可知的。	软件部署是手工过程。针对具体环境生成二进制包。环境准备是手工的。	不频繁且不可靠的发布	开发完之后，才做手工测试	数据库变更没有版本化，且手工进行	没有使用版本控制，或者提交不频繁	毫无架构设计准则，架构设计是无序和混乱的

持续交付的运维观

1、持续接收到持续交付，运维的核心转变

2、运维掌握了最好的持续交付切入点：CMDB和持续交付

3、交付的最终评价：质量、效率、成本

4、持续交付是打破部门墙的核心实践

5、持续交付的本质：标准化+平台化+服务及面向用户的价值

运维的理解

持续交付的运维观

6、基于交付链(Dev/Test/Ops)的全局优化，而非局部(Ops)优化

7、运维的问题不是仅仅运维侧的问题，是一个IT问题

8、运维离用户最近，你代表用户，就有最强的驱动力

9、跨界由此而生

运维的坚持

DevOps管理专家

Thanks

优维欢迎您的加入，上海、广州、深圳！

GOPS2016 全球运维大会更多精彩

GOPS2016

全球运维大会·北京站

2016年12月16日-17日

北京国际会议中心