

DOIS

DOIS | 2019 · 北京站  
DevOps 落地，从这里开始

# DevOps 国际峰会

暨 GNSEC 全球新一代软件工程师大会

指导单位： 云计算开源产业联盟  
OpenSource/Cloud Alliance for Industry (OSCAR)

主办单位： DevOps时代

 高效运维社区  
GreatOPS Community

时间：2019年7月5日-6日

地址：北京悠唐皇冠假日酒店



# BizOps : 应用运维为业务创造更多价值



封铨贤

中国民生银行

十年IT从业经历，长期从事商业银行核心系统的运维工作，参与了两代银行核心系统的建设。现专注于分布式核心系统的运维支撑体系建设，以及DevOps在银行业的应用实践。曾担任某国际软件厂商的技术咨询顾问，为多家世界500强企业实施了企业信息化项目。

# 目录

## CONTENT

1

BizOps的基本理念

2

BizOps的关键原则

3

BizOps的应用实践

4

BizOps的典型案列



# PART 01



## BizOps的基本理念

BizOps(应用运维)并不是DevOps延伸到BizDevOps (业务开发运维一体化)

# 这些年出现的新名词



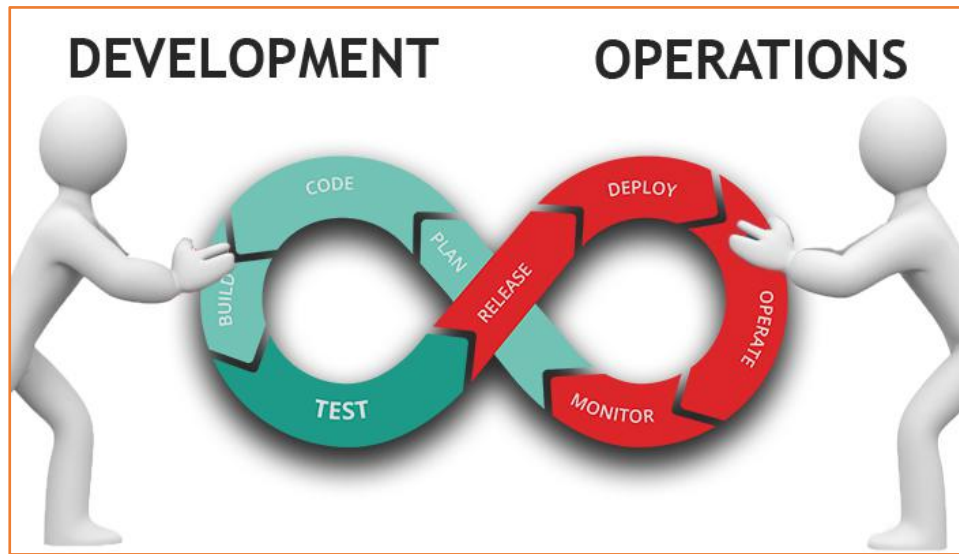
# DevOps关注实施态

## DevOps关键要素

- 价值流
- 部署流水线
- 版本控制
- 自动化配置管理

## DevOps强调效率

- 持续集成
- 持续部署
- 持续交付





# BizOps(应用运维)：我们不生产新名词，它一直在这里

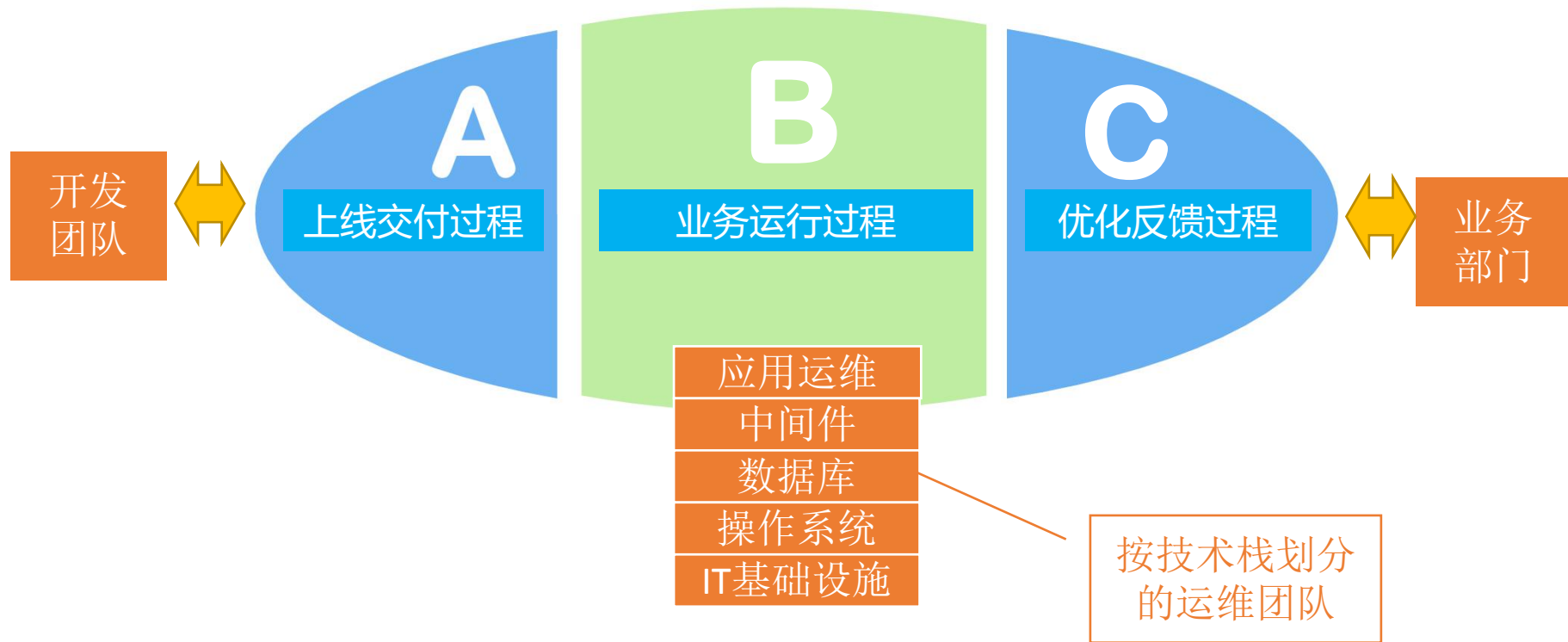


一、将DevOps环展开，一个业务应用的生命周期为：



二、数据中心的技术栈典型分层为：

# BizOps (应用运维) 关注运行态





# BizOps ( 应用运维 ) 的六个环节



# BizOps的2个问题

问题1：

已经拥有开发中心、数据中心、业务运营等部门的企业，是否需要组建专门的应用运维团队？

问题2：

应用运维团队应该归属哪个部门？

# BizOps能解决很多问题

- 理想很丰满，现实很骨感：拍脑袋出不来好需求  
业务人员需要通过应用运维来获取更多生产运行情况的反馈
- 管生不管养，管杀不管埋  
漫天飞舞的需求谁来梳理？多如牛毛的系统谁来管理？
- IT专注于技术，技术服务于业务  
BizOps：不仅关注技术，更关注业务
- “好男人是调教出来的，好系统是运维出来的”

# BizOps在某金融企业中的具体定位





PART

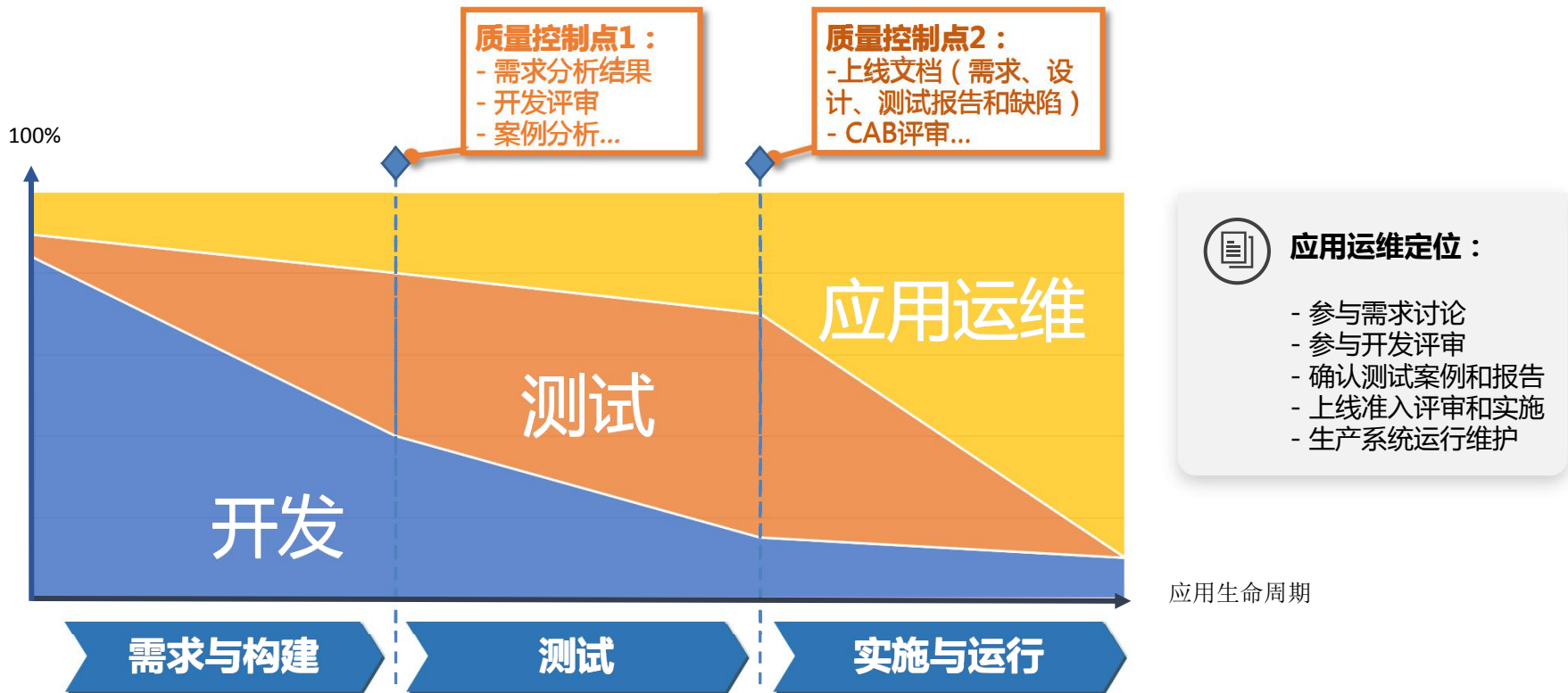
02



## BizOps的关键原则

不是所有的Ops都叫BizOps

# BizOps关键原则一：运维从立项开始，交付从需求开始



## BizOps关键原则二：非功能性需求决定了应用系统的生命力



### ■ 日志管理

日志分级、格式统一、颗粒度合理、连续可读、大小可控、大数据可分析

### ■ 容量管理

数据温度管理、冷热数据分离、历史数据归档、文件自动清理

### ■ 批处理管理

互斥控制、并发控制、断点续跑、异常告警、统一调度

### ■ 流量控制

服务可动态开关、流量可分配权重、负载可灵活调整

### ■ 高可用架构

本地高可用：集群、负载均衡  
同城双活：流量切换、数据同步  
异地灾备：数据复制技术、灾备切换方案

### ■ 一致性机制

交易超时处理、自动化的冲正机制、完善的对账机制

### ■ 异常处理

异常交易应按照业务异常处理流程执行

### ■ 版本管理

随时可回退到稳定版本的能力

### ■ 发布管理

灰度发布、金丝雀发布、不停机热部署、自动化部署

### ■ 业务可视化

实时展示端到端的业务运行情况



## BizOps关键原则三：应用运维要有串联整个运维技术栈的能力

业务流程全局掌控  
业务流程优化建议  
业务需求、请求评估支持

应用架构、技术掌控  
业务逻辑实现  
问题、异常处理.....

业务流程



...

应用

服务接口

业务逻辑

...

批量

数据库

DB2

Oracle

...

MySQL

中间件

Weblogic

Tomcat

...

Tuxedo

操作系统

AIX

Linux

...

Windows

基础设施

服务器

网络

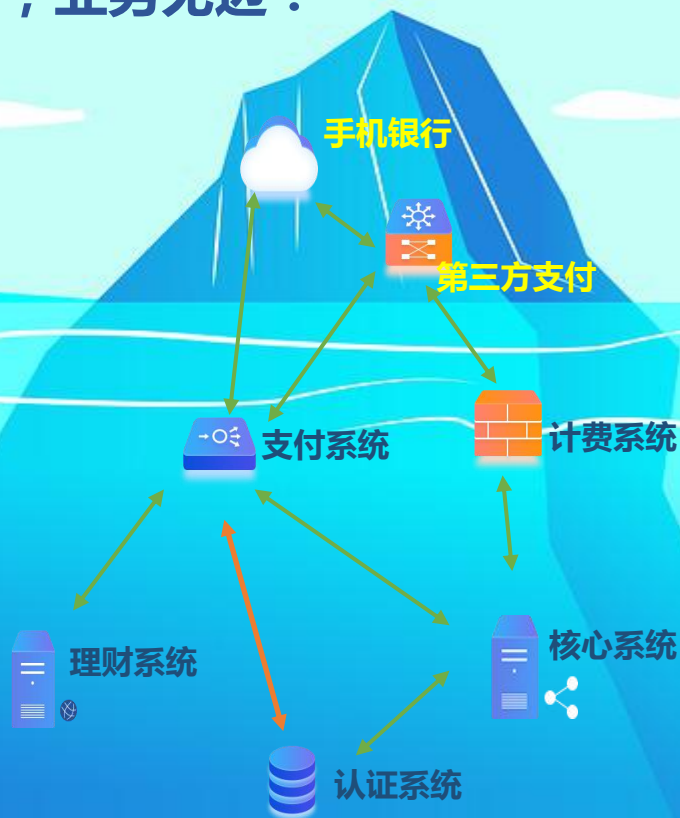
...

安全

不是全栈工程师的人  
要有全栈工程师的心

## BizOps关键原则四：系统有界，业务无边！

应用运维：  
维护的是业务，  
不只是系统！





PART

03



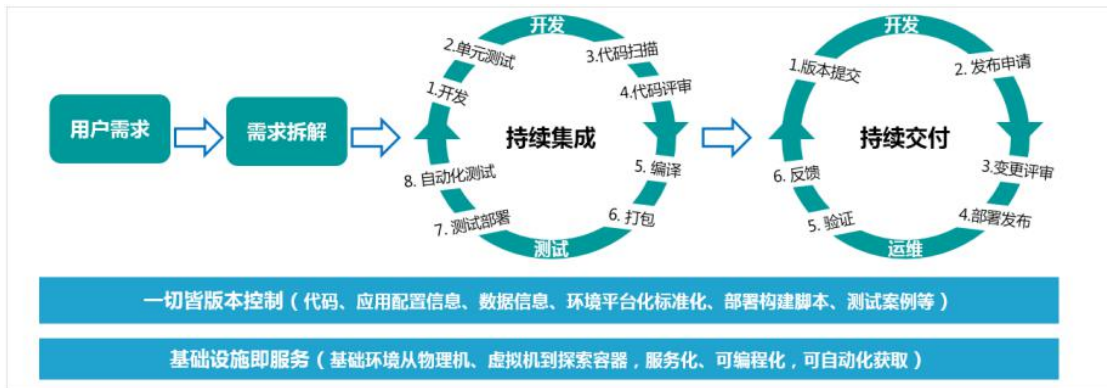
## BizOps的应用实践

# 实施态：BizOps拥抱DevOps

- 从交付能力来看，DevOps能

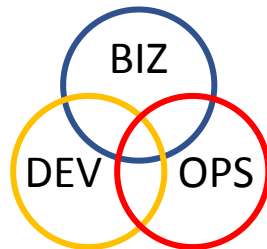
提升交付效率与交付质量

- ① 技术上能持续交付
- ② 业务上能按需而动



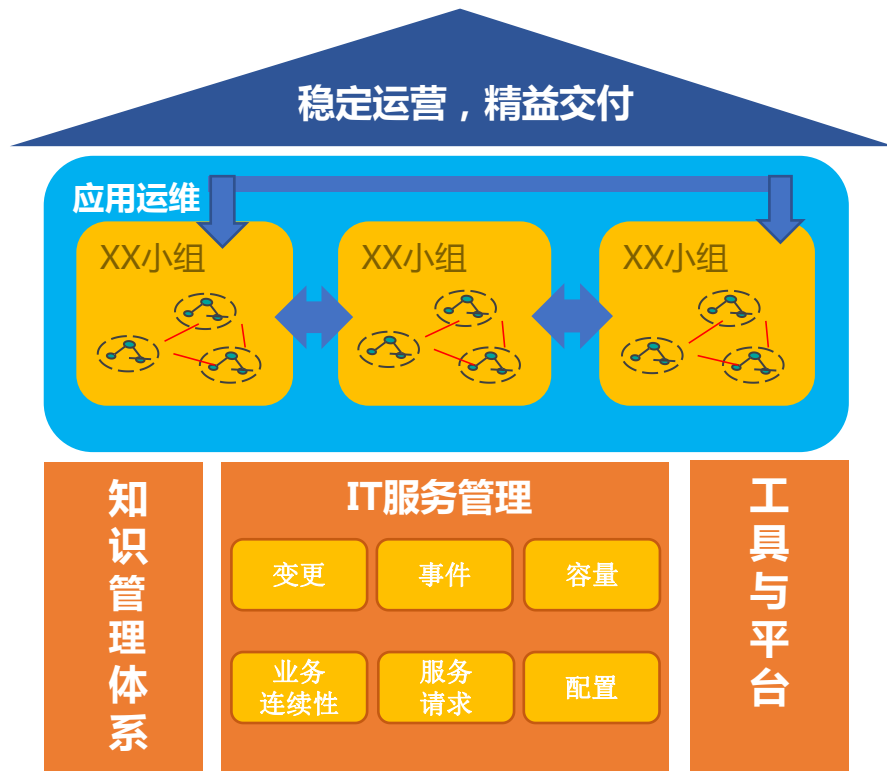
- 从组织架构来看，应用运维是DEV、OPS、BIZ三大部门的连接融合点

- ① System Owner：应用运维人员
- ② DevOps体系中打破部门墙的关键节点



# 运行态：BizOps遵循IT服务管理规范

1. **知识管理体系**，基于生产运营各项工作积累的知识、文章与文档等全生命周期管理，构建嵌入各大IT服务流程与活动
2. **IT服务流程**，专注于**业务连续性**的流程、规范、管理办法和事项。是一种重点化的ITIL，以业务连续性驱动的一系列流程管理活动
3. **工具与平台**，自动化部署发布流程实现，多维度的监控体系工具，业务流程与应用可视化，组件标准化与平台化等，提供技术与工具链支撑



# 业务态：目标是双速的业务，而不是双速的IT



## 实现方式

- 依托于容器适当简化的流程
- 基于开源社区版软件
- 统一全局的工具
- 浮现式设计自主研发

## 实现目标

- 快速高效的交付功能
- 灵活快速的扩展能力
- 端到端一致的流程引擎易于控制
- 自主掌控，产品化

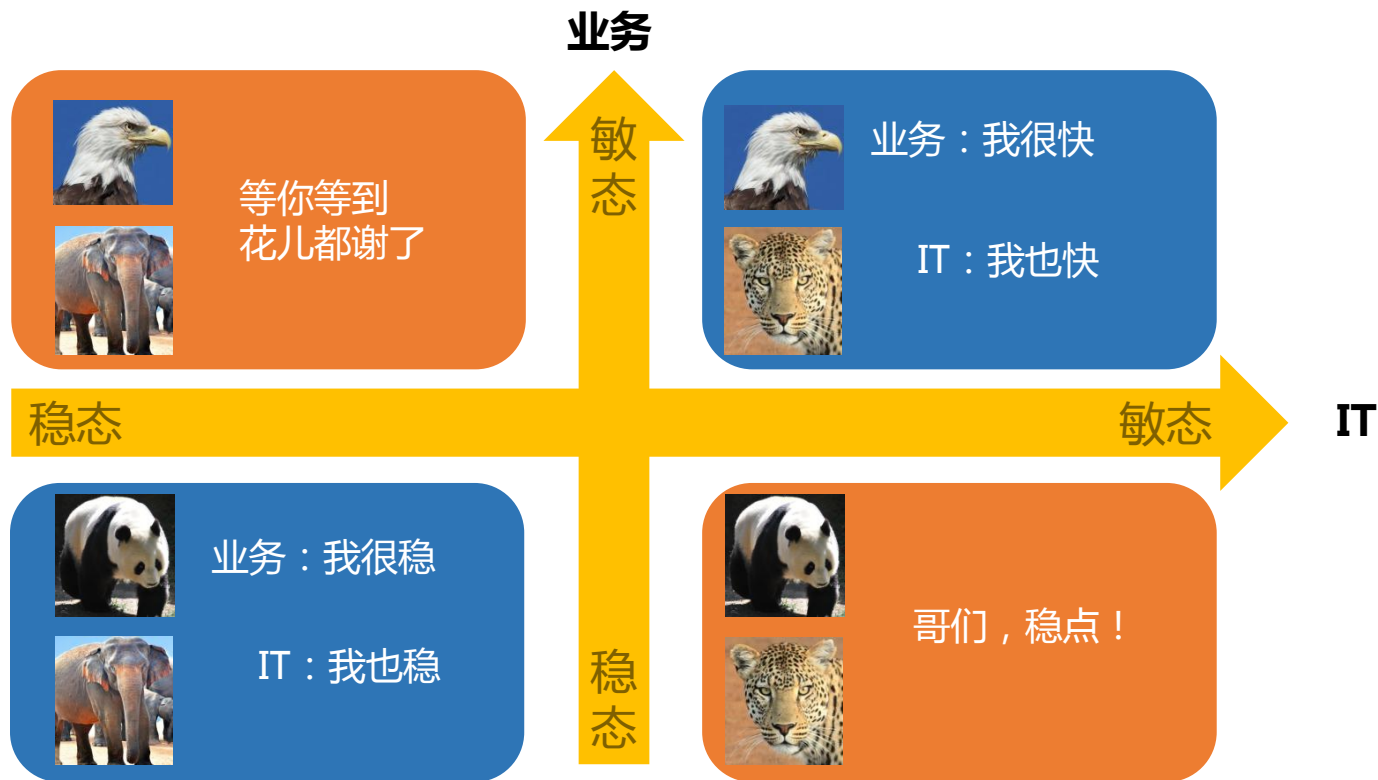


- 现存多种软件并存
- 基于现有流程进行集成
- 成熟工具与流程体系的深度增强
- 按照传统发布管理策略进行部署

- 平台开发熟悉，短期易于实现
- 规避存量软件交付流程的运行风险
- 现有工具的功能挖掘和流程串联
- 确保传统应用部署发布的稳定性

流程融合  
互相驱动  
持续演进

**业务态：目标是双速的业务，而不是双速的IT**







PART

04



## BizOps的典型案列

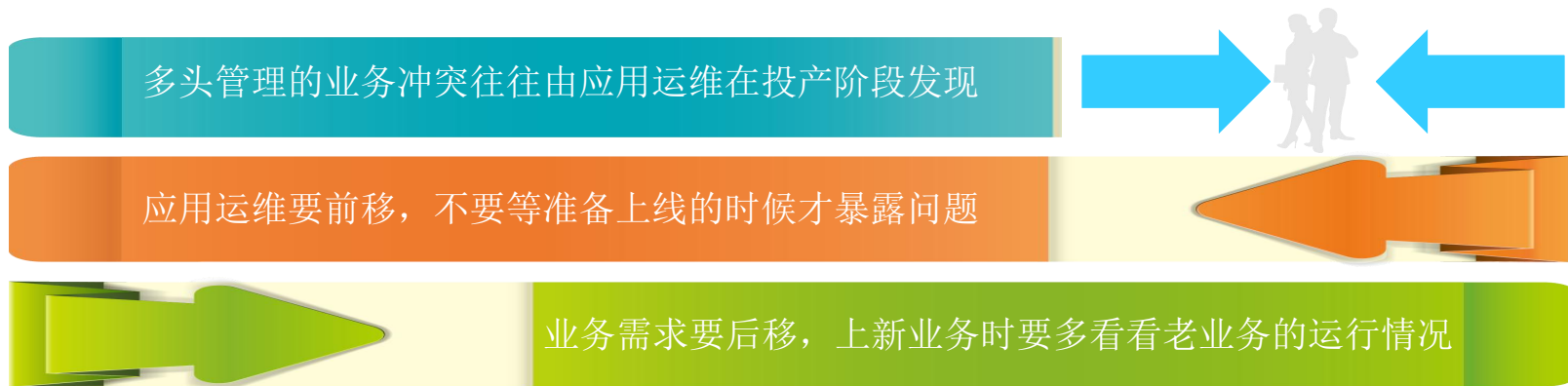
应用运维的用武之地

## 案例一：多头管理的业务冲突（前 VS 后）

场景：

- A部门提出了涉及客群归属的某业务方案，经过几个月的开发测试，在上线投产评审时，应用运维团队提出了异议：经分析生产数据，该方案会将B部门的客户归属到A部门，是否经过B部门确认？

思考：



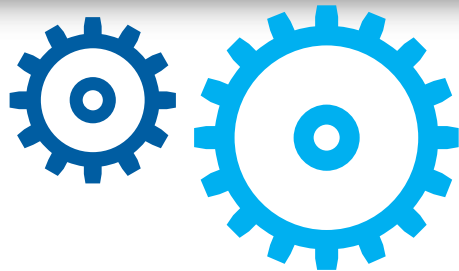
## 案例二：双速IT的差异（慢 VS 快）

### 场景

- 几年前一款复杂的业务产品A上线，深受行业客户欢迎（稳态）
- 今年互联网某交易场景B性能压测，发现签约了A的客户交易响应时间非常慢，很容易超时。（敏态）

### 思考

- 两个不相关的交易场景，由于速度的差异，可能也需要互斥
- 快车道和慢车道，总有并线的时候
- 应用运维做得好，就能成为双速IT之间的变速齿轮！



## 案例三：来自外部的交易高峰（外 VS 内）

### 场景：

- 近年来时常会出现类似双十一、春节红包等来自外部的交易高峰，当出现性能问题时，该如何应对？

### 思考：

- 面对外部冲击，应用运维首当其冲！
- 考验的是整个企业的IT运行体系，应用运维人员只有理解业务，对业务场景深入研究，协调和联合业务部门、开发团队、质量、安全、数据中心技术专家等多方力量，才能分析出交易链条的瓶颈，给出性能优化方案和最佳的资源调配。



## 案例四：端到端的交易分析（点 VS 面）

场景：

- 第三方发起的一笔交易响应时间超时，该交易流转了十几个系统，慢在哪里？

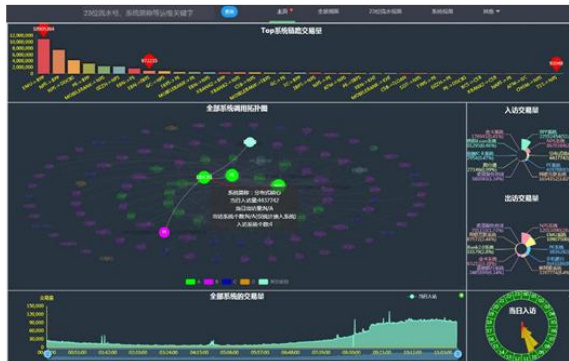
思考：

- 应用运维有**能力**、有**动力**去跨越系统之间的界限，开发运维平台，构建出业务的端到端全景视图，进行交易的链路分析。

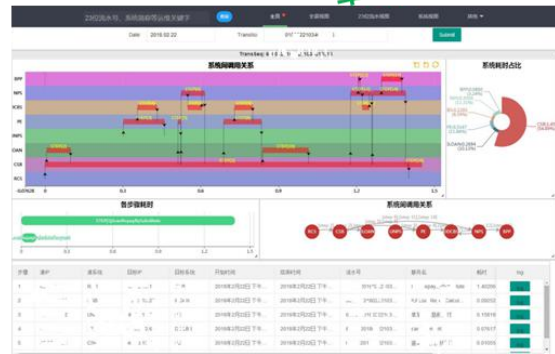
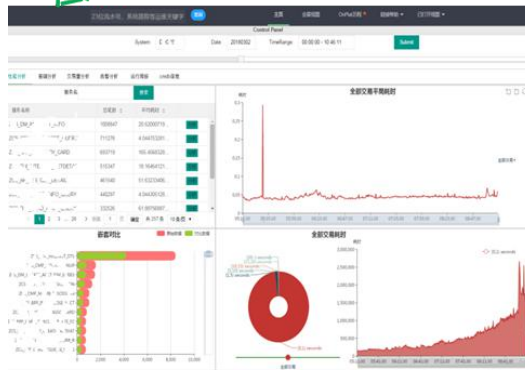
看全景



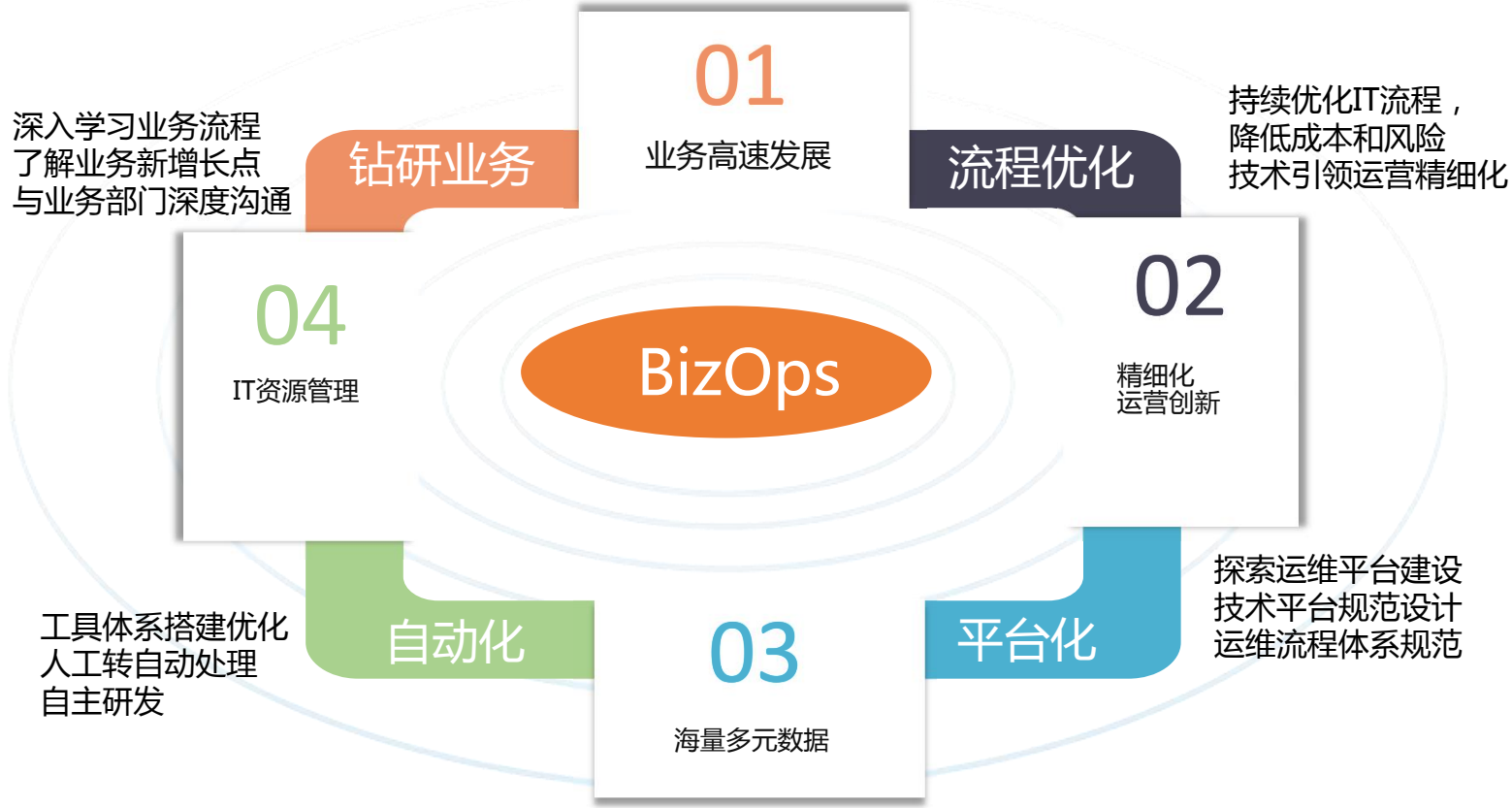
应用下钻



单笔端到端追踪



## 总结：应用运维的未来更有价值！





# Thanks

DevOps 时代社区 荣誉出品



想第一时间看到高效运维社区的  
最新动态吗？

