

传统企业的敏捷运维之道

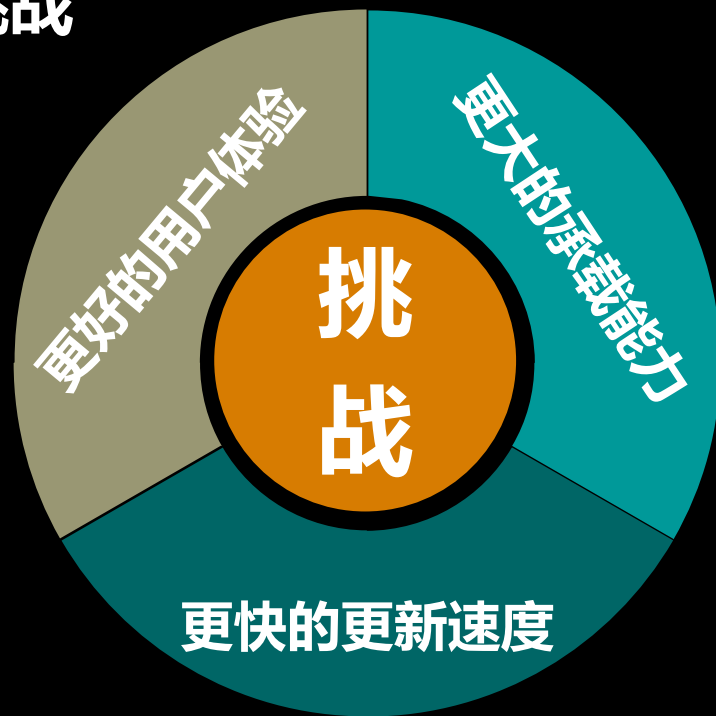
演讲嘉宾 新炬网络架构师 陈能技

敏捷运维之道：

- 传统企业运维面临的挑战
- 传统企业敏捷运维之道
- 总结

互联网+浪潮对传统企业IT能力的挑战

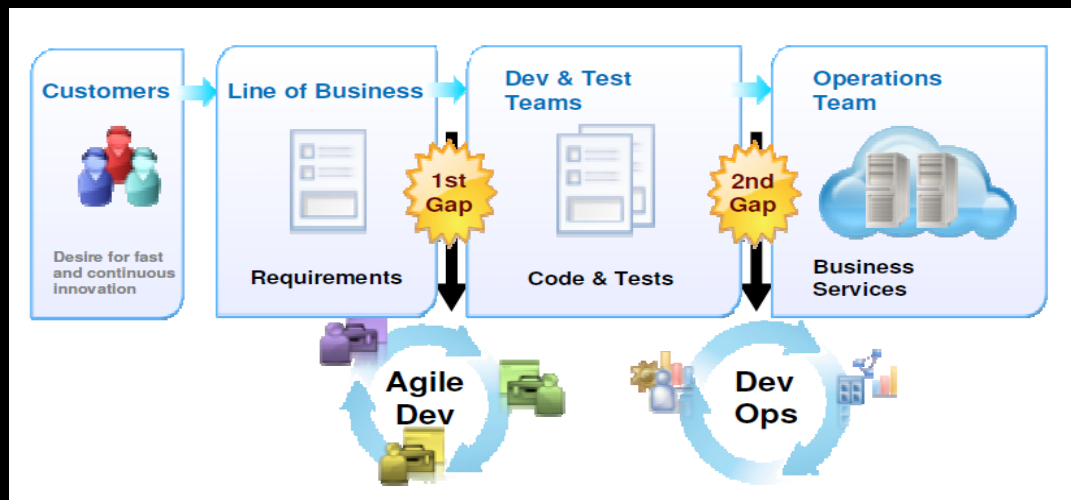
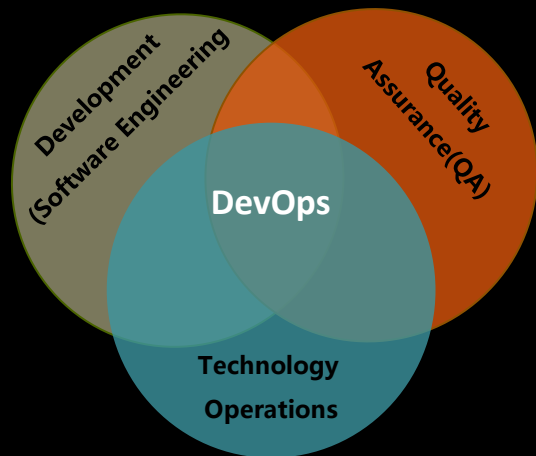
.COM



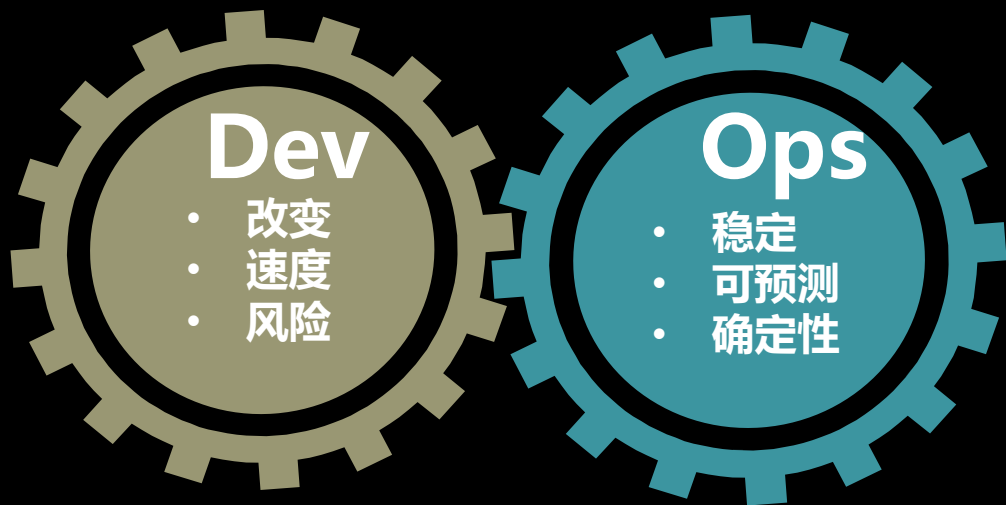
互联网企业的业务正在快速增长，通过强大IT能力正在不断挤占传统企业市场
传统企业互联网+转型对其IT能力提出了空前的挑战

敏捷开发打破第一个Gap，DevOps帮我们打破第二个Gap！

研究显示，在那些引入了DevOps概念的企业中，开发与运营人员在设计、构建、测试工作中共同在内部应用上进行协作之后，交付能力大大提升：交付效率提高30倍、变更失败率降低50%



DevOps旨在通过共同目标，成就协同与效率，提升用户体验



新炬提出DevOps能力矩阵模型

角色视角 X 管理视角

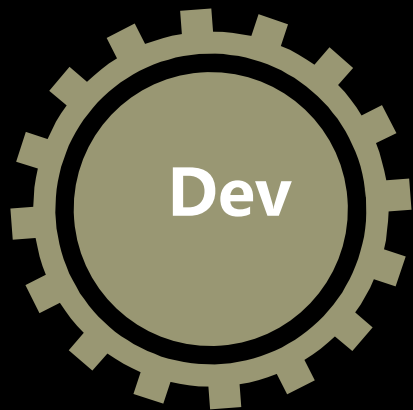
构建企业级DevOps全过程交付能力，提升用户体验！

| | 度量 | 过程 | 工具 |
|----|------------------------------|------------------------------|---|
| 开发 | 代码质量 技术债务 | 小版本迭代开发 持续集成 持续部署 | Docker 微服务 自动部署 |
| QA | 系统测试 测试覆盖率度量 验收测试 | 持续测试 测试环境管理 质量反馈 | 自动化测试 应用性能管理 |
| 运维 | 运营质量 运行质量 用户体验 趋势分析 | 持续发布 性能管理 容量管理 运维管理 | 一体化监控 ITOA 大数据分析 自动化运维 应用性能管理 |

DevOps能力融合4大核心实践

实践1：将开发延伸至生产中（持续集成和交付）

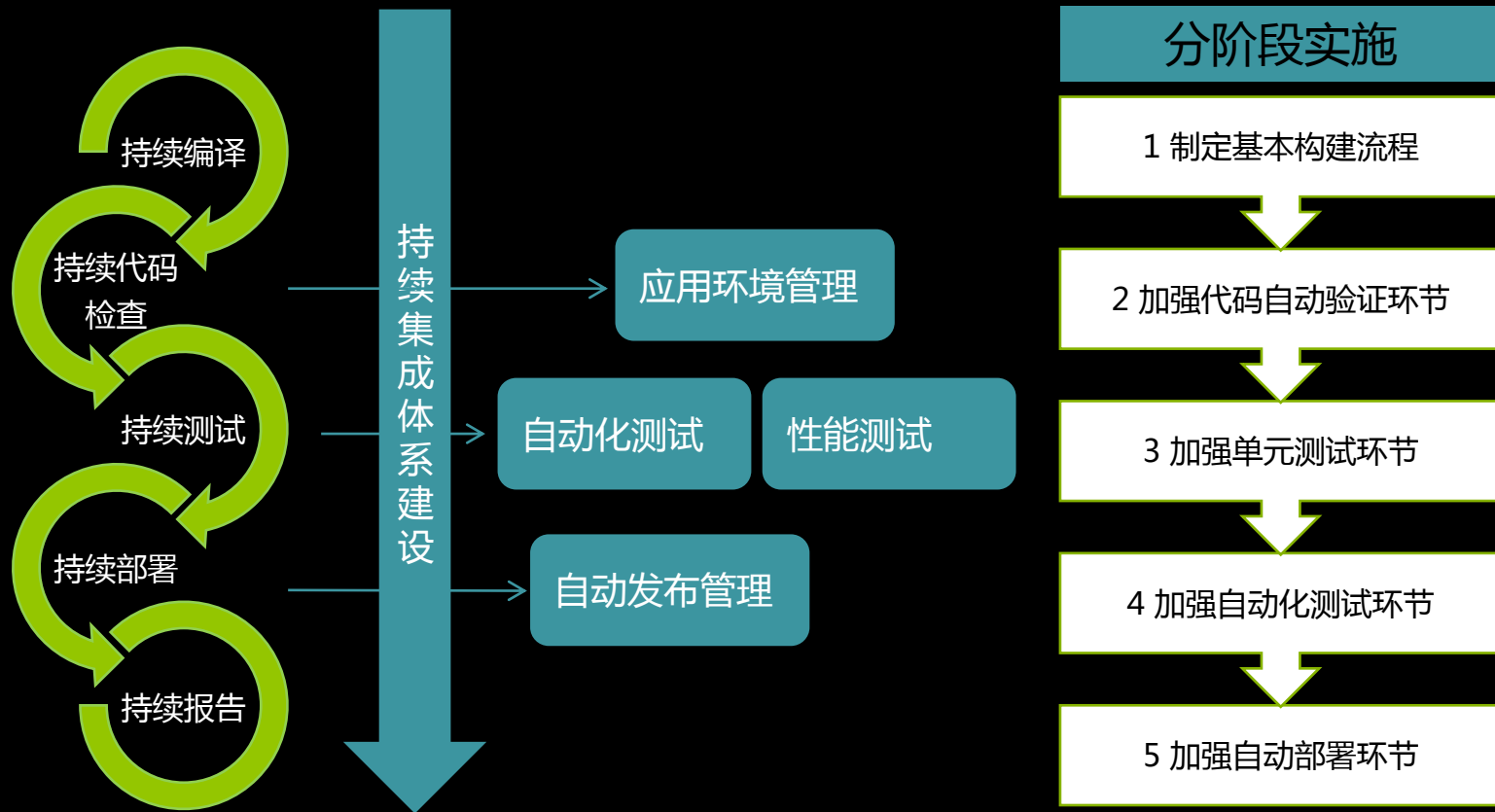
实践2：将开发嵌入到IT运维中（应用端到端管理）



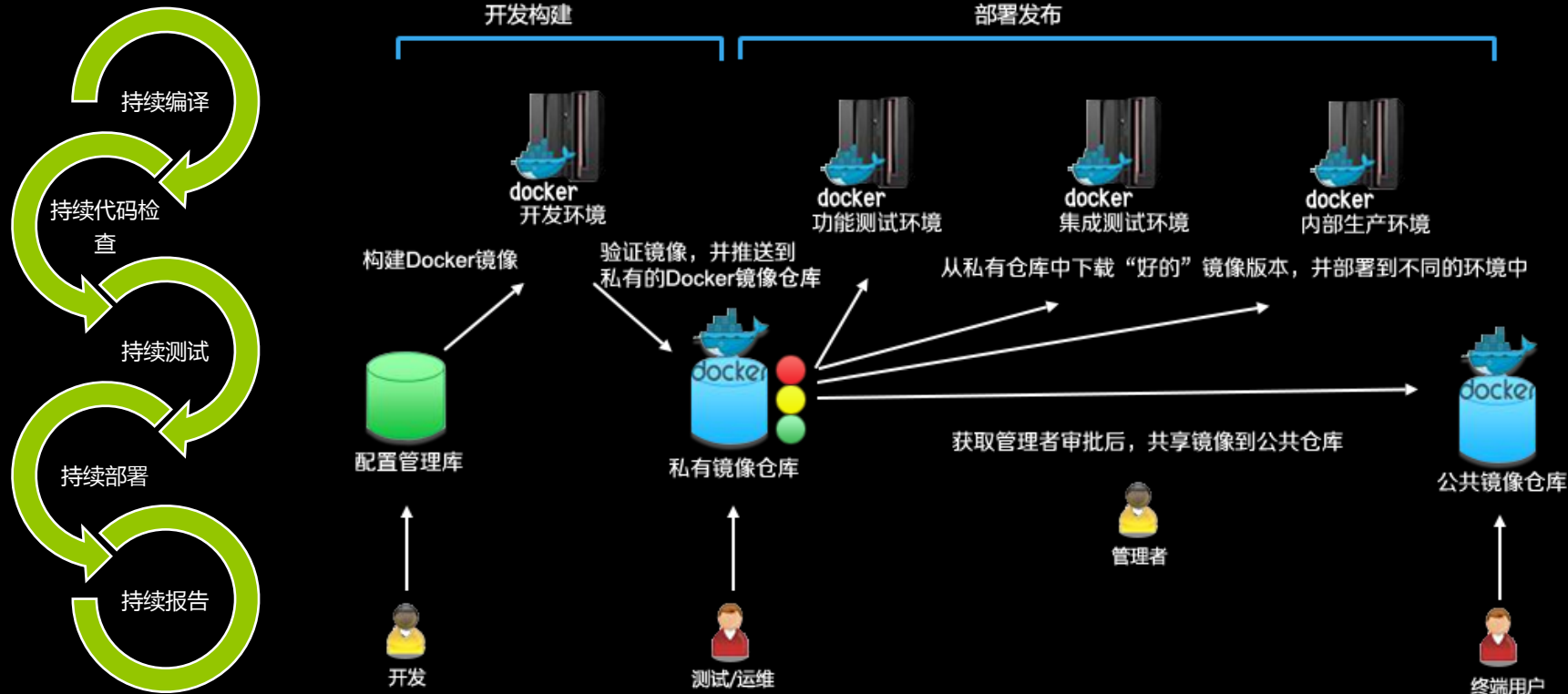
实践3：向开发中加入生产反馈（可视化监控和运维）

实践4：将IT运维嵌入至开发（运维分析及预测）

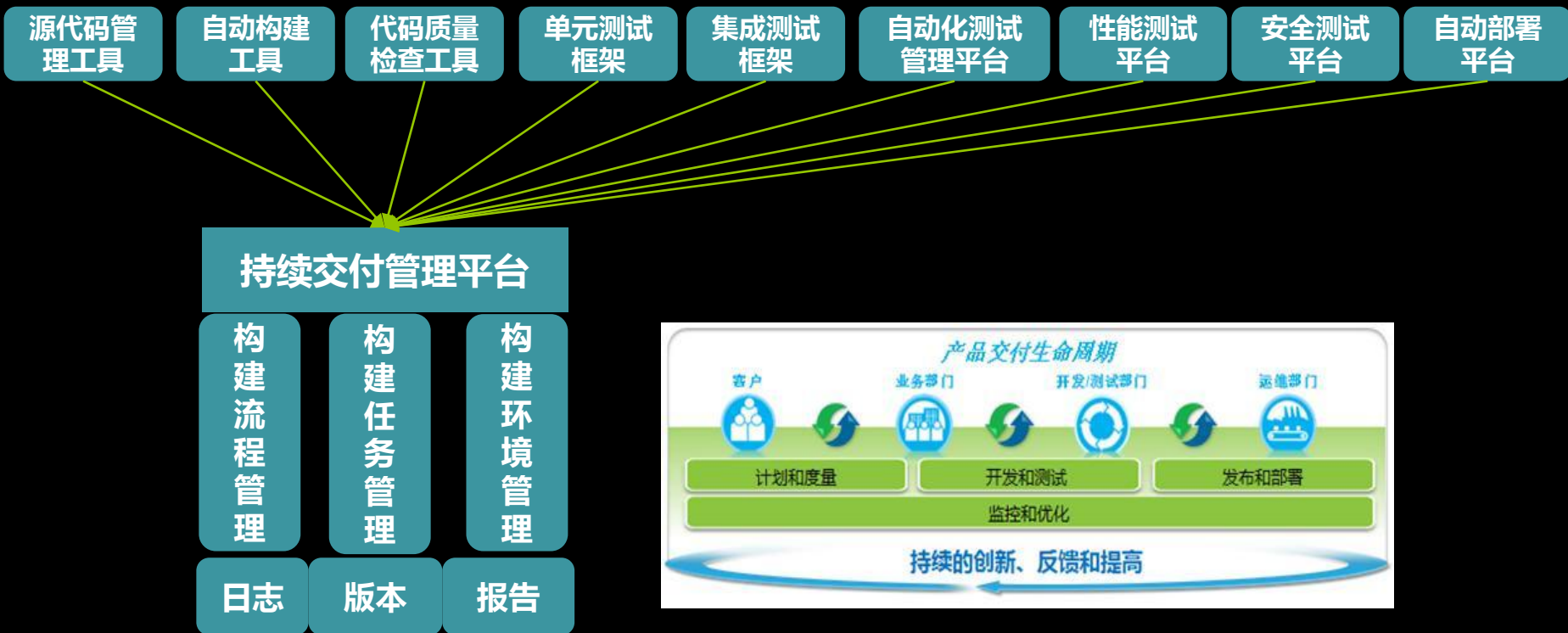
实践1：建立持续集成体系，实现应用集成交付过程标准化与透明化



实践1：基于容器技术的持续交付，确保交付环境的一致性



实践1：构建持续交付管理平台，提升交付过程的效率与质量



实践2：构建用户管理体系，实现用户体验的可视化

- 完整的会话 sessions？
- IP地址？用户名？所属群组？或匿名？
- 哪些是最不满意的用户？

谁 Who

哪里 Where

- 用户主要来自哪里？
- 哪里导致满意度问题？
- 用户在哪里离开业务？

什么 What

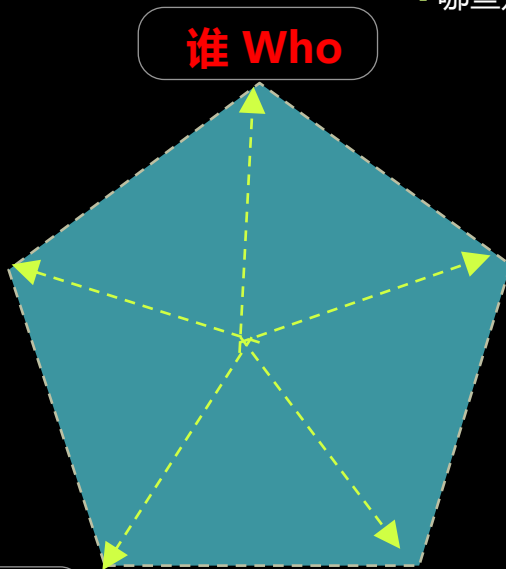
如何 How

何时 When

- 不同时间尺度的访问曲线？
- 什么时间出的错误？
- 什么时间做的动作？

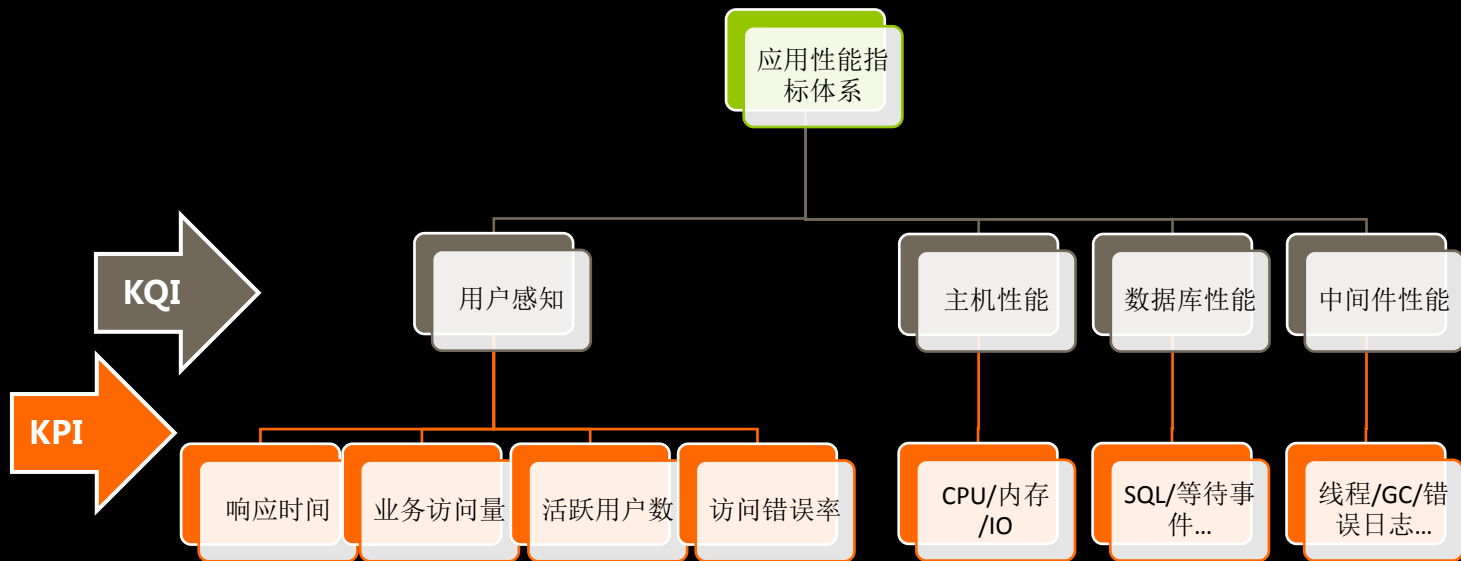
- 访问什么业务？调用什么服务或页面？
- 什么服务访问最多？
- 什么服务/页面最慢？资源开销最大？

- 如何快速准确定位错误？
- 如何知道哪些用户受到不良影响？



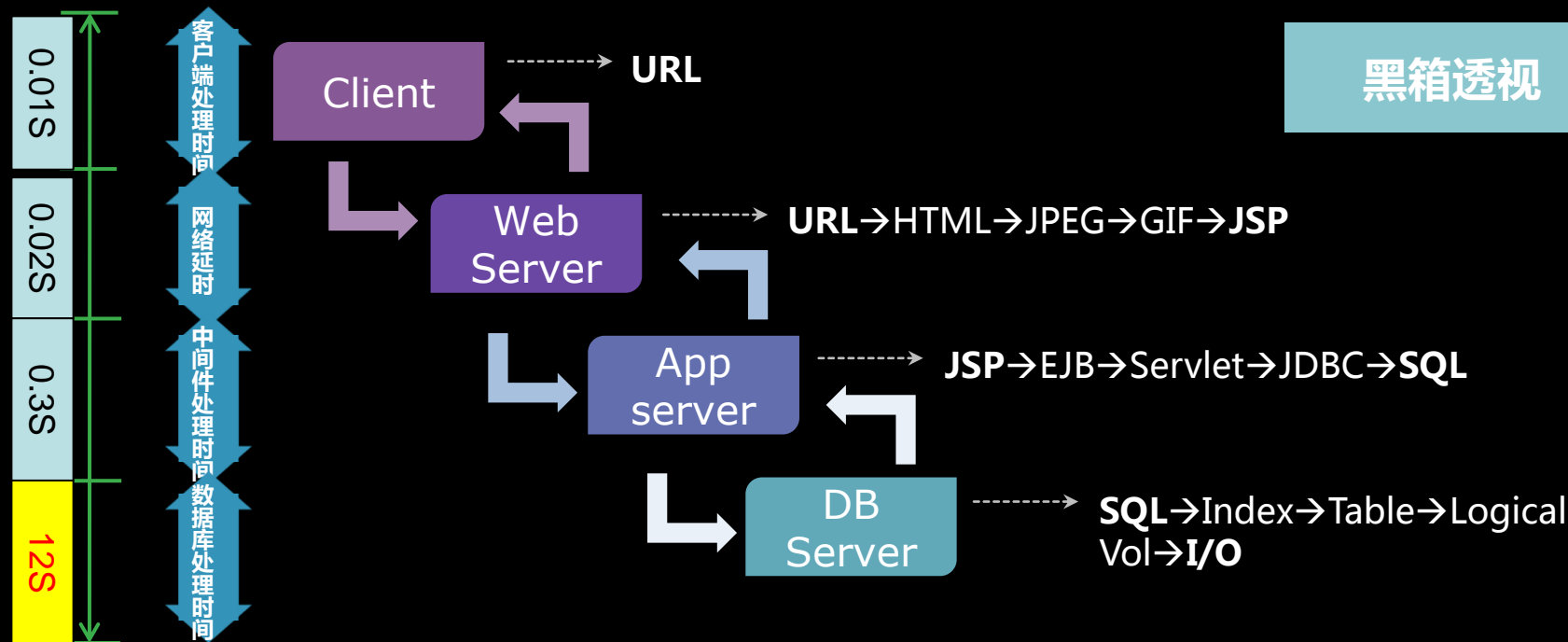
实践2：构建应用指标体系，实现运行效率的可视化

- ◆从业务、客户感知的角度筛选指标，系统指标只用来定义应用生命特征
- ◆简化指标数量，快速判别系统运行效率



实践2：透明化应用交易过程，实现端到端的应用性能管理

通过对中间件交易过程的追踪和记录，实现交易过程的透视，并对相关数据进行分析 and 存储，可实现调用过程分解及性能问题的快速定位。



实践2：透明化应用交易过程，实现各层的调用链关联

- ✓ 通过URL实现业务层与应用层的关联
- ✓ 通过调用链实现应用层之间的关联
- ✓ 通过调用链实现应用层与数据层之间的关联
- ✓ 通过SQLID实现应用层与SQL语句的关联
- ✓ 通过IP地址实现与基础设备层的关联

实现业务层到基础设施的打通

业务地图及钻取分析



实践3：构建立体化监控体系，实现运行状态可视化及深度性能分析

用户

业务

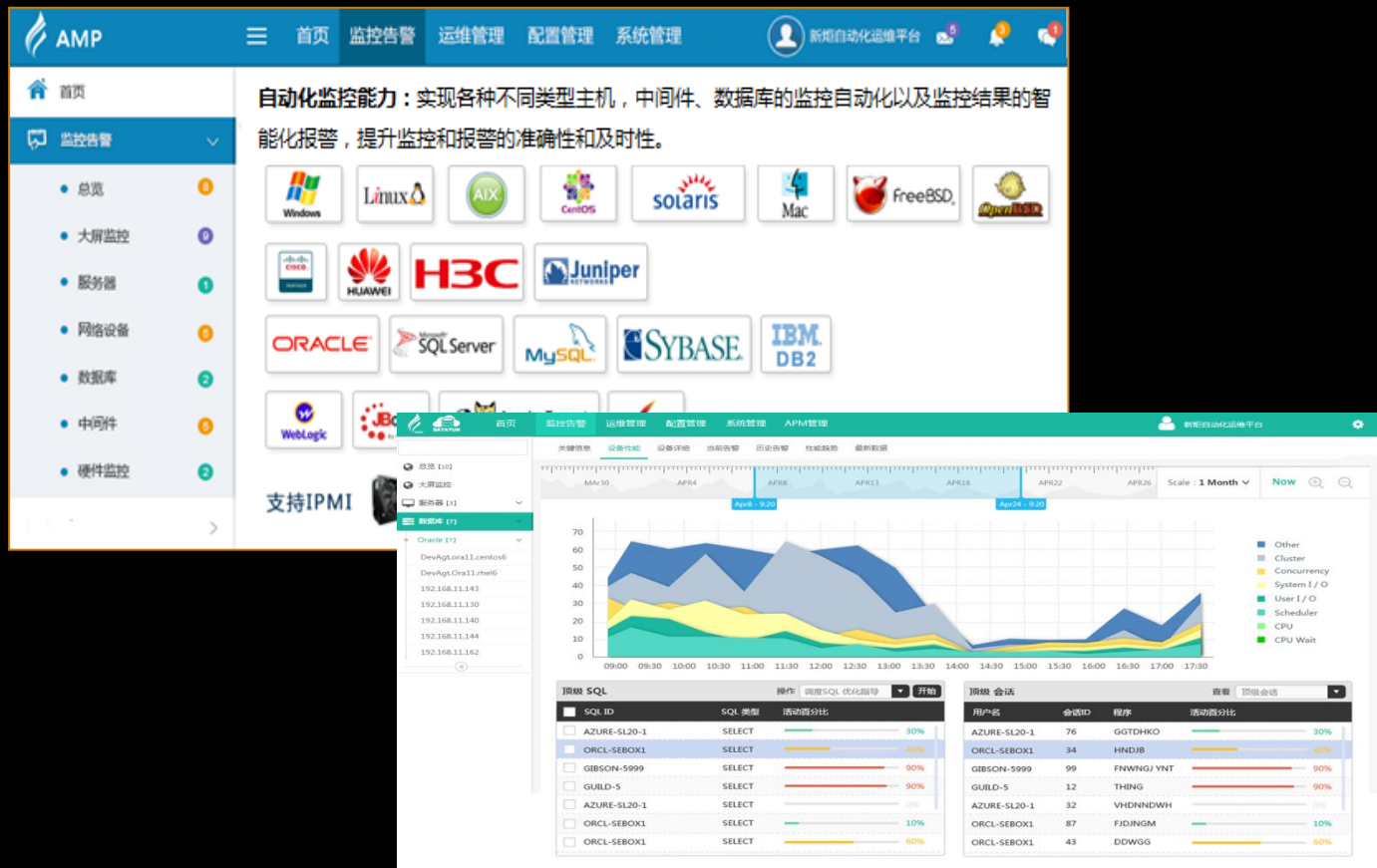
应用

中间件

数据库

主机

存储/网络



实践3：标准化运维操作，实现运维的自动化与可视化



告警自动化处理

例行工作自动化处理

故障自动诊断及分析

自动故障隔离

自动容灾切换

自动伸缩及扩容

实践3：集成原子化操作脚本，实现拖拽式自动化场景构建

上海新炬网络技术有限公司

三

首页

监控告警

运维管理

配置管理

上海新炬网络技术有限公司

三

首页

监控告警

运维管理

配置管理

系统管理

日志管理

Enter keyword

新炬自动化运维平台

监控告警

运维管理

故障诊断

配置管理

系统管理

日志管理

方案信息

方案名称

节点信息

节点样式选择

流程类

开始

结束

执行类

开始

方案流程管理

流程起点

脚本/工具/命令

封装成步骤

脚本1

脚本任务

结束

脚本任务

脚本任务

结束

流程信息

更新时间: 2015-03-02 17:00:00

开始

脚本任务

脚本任务

脚本任务

结束

当前节点执行状态：执行成功

暂停

继续

终止

返回

| 节点名称 | 步骤一 | 脚本文件 | 任务1.sh | 描述 | 任务一 |
|------|-----|------|--------|----|-----|
|------|-----|------|--------|----|-----|

执行日志信息

步骤1 执行过程信息: -- 9 packets transmitted, 9 received, 0% packet loss, time / 999ms

步骤1 执行过程信息: -- rtt min/avg/max/mdev = 0.037/0.042/0.066/0.008 ms

步骤1 执行过程信息: -- executeResultForDTS:0

步骤1 执行成功!

步骤2 开始执行...

步骤2 执行过程信息: -- task1

步骤2 执行过程信息: -- PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.

步骤2 执行过程信息: -- 64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.030 ms

步骤2 执行过程信息: -- 64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.038 ms

切换演练状态：成功

演练总结

演练开始时间: 2015-07-20 18:54:20 结束时间: 2015-07-20 18:55:05 执行人: admin 演练过程正常。

修改

实践3：标准化故障分析路径，实现故障定位的自动化与可视化

故障ROOT分析

告警短信：XX缓存硬件故障，影响XX、XX业务，可能原因为磁盘故障

架构分层原则：越底层的设备可能性越大

路径分析原则：当某个设备出现问题，属于这个调用链上的节点都可能出现告警，按访问顺序，最末端的可能性越大

时间面积原则：结合告警时间先后顺序，告警影响面积等权重分析

用户层

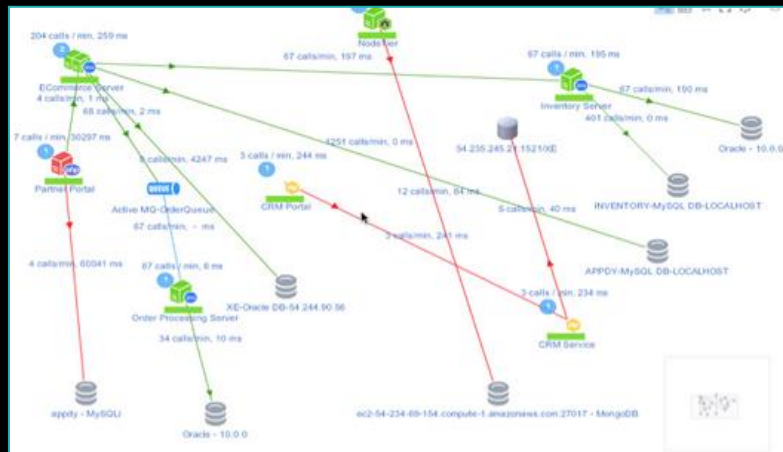
接入层

逻辑层

数据层

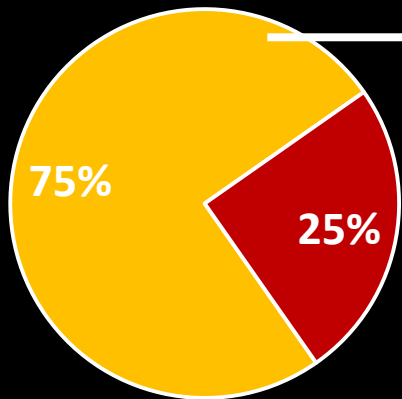
时间相关性

面积权重



实践4：整合利用运维数据，进行运维大数据分析

Gartner



Gartner估计，到2017年全世界将有75%的大企业会积极使用ITOA；利用数据进行如，可用性监控，应用性能监控,故障根源分析,安全审计,容量分析等

分析数据来源

A

机器数据 (Machine Data)

-- IT系统自己产生的数据，包括客户端、服务器、网络设备、安全设备、应用程序、传感器产生的日志

B

通信数据 (Wire Data)

-- 系统之间2~7层网络通信协议的数据，可通过网络端口镜像流量

C

代理数据 (Agent Data)

-- 是在 .NET、PHP、Java 字节码里插入代理程序，从字节码里统计函数调用

D

监控数据 (Monitor Data)

-- 监控采集到的数据库、主机、应用等运行状态及指标数据

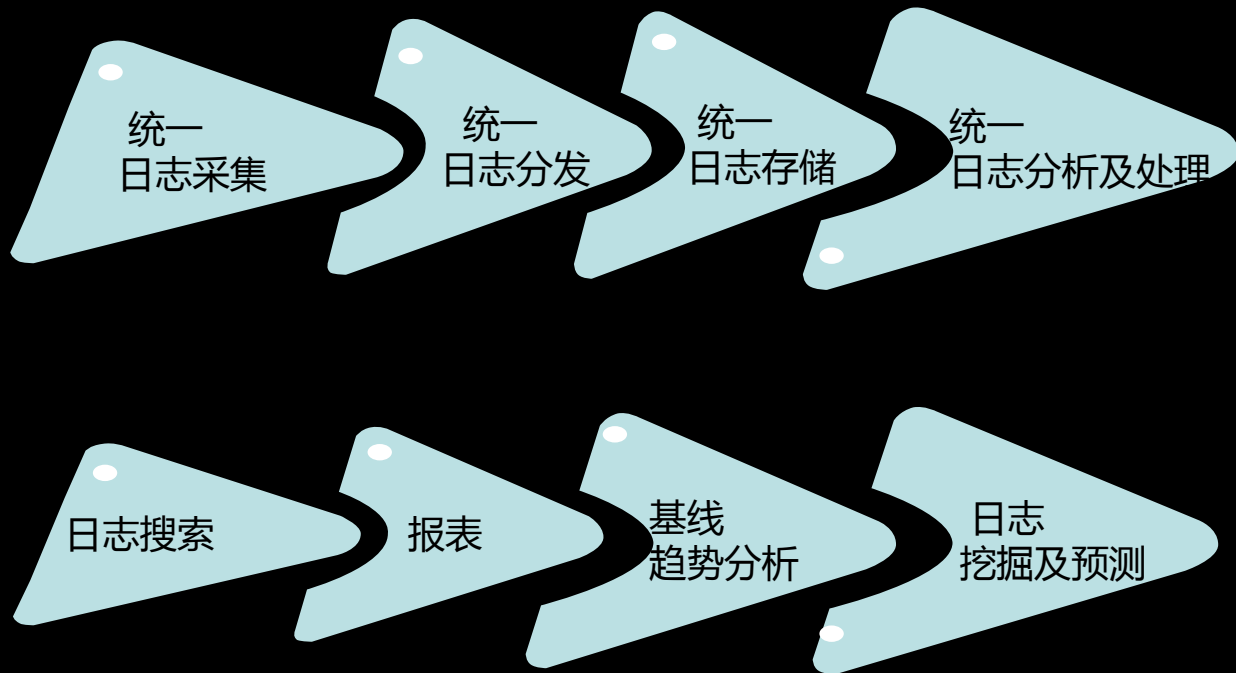
E

探测数据 (Probe Data)

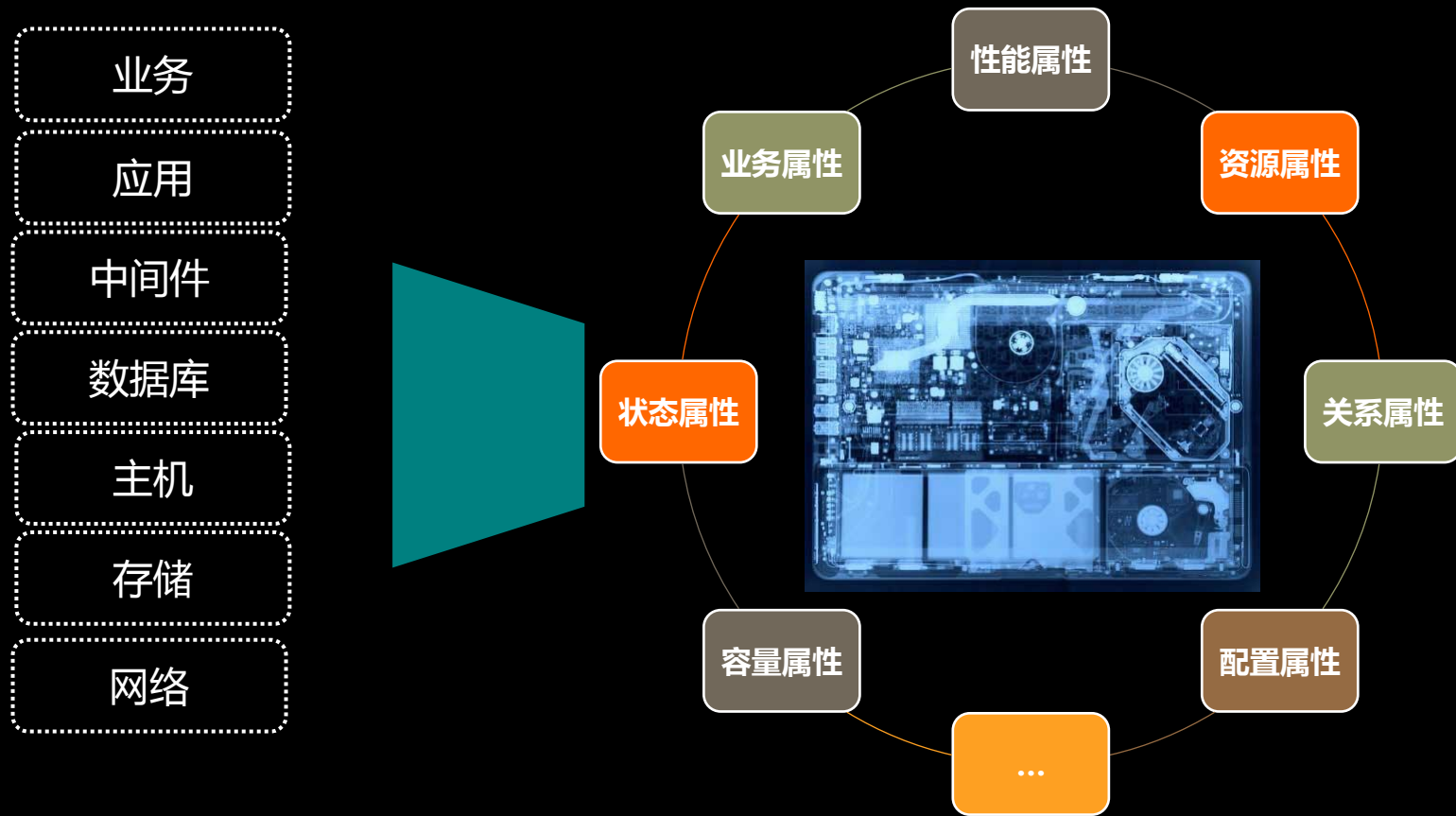
-- 模拟用户请求检测系统，如 ICMP ping、HTTP GET等，能够从不同地点模拟客户端发起

最大力度挖掘数据价值

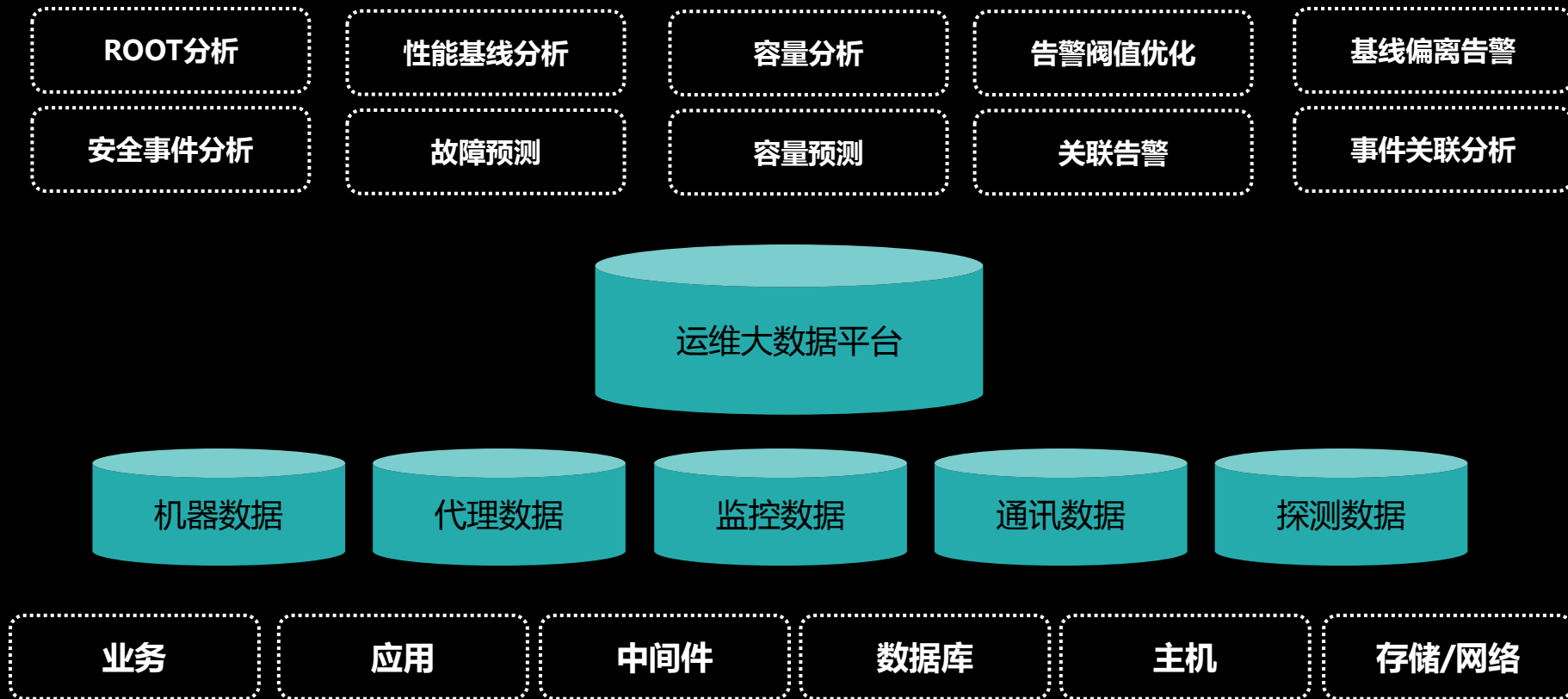
实践4：进行集中式日志管理，实现运行状态的可视化



实践4：构建设备标签体系，进行机器画像与行为预测



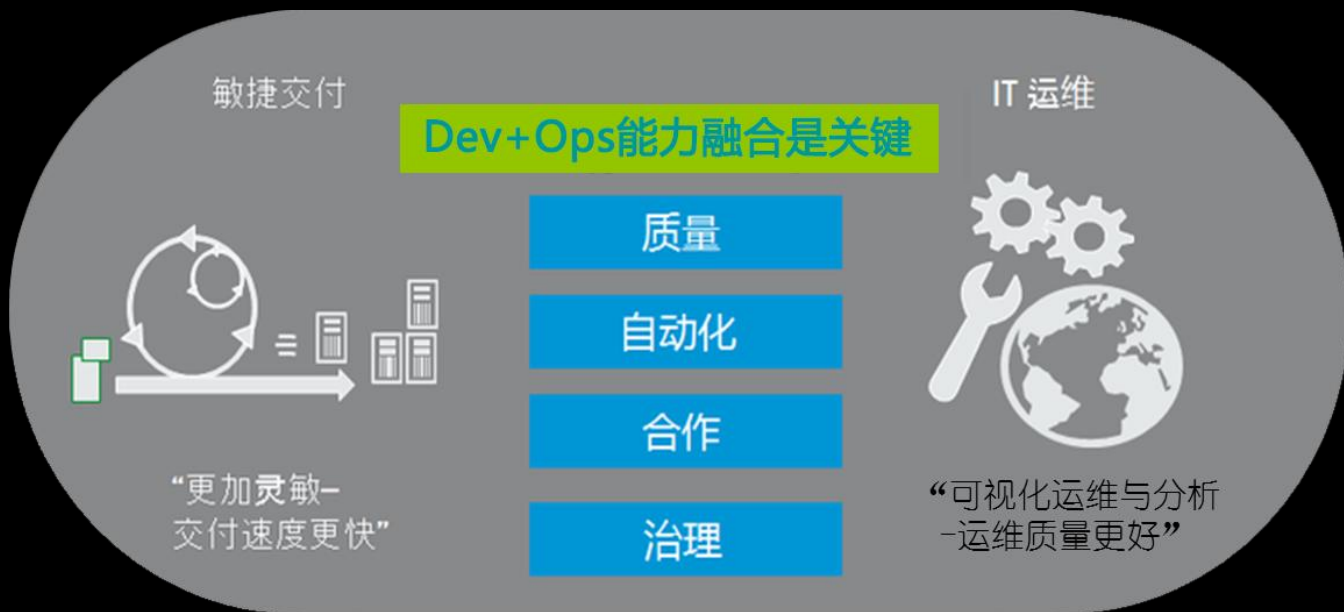
实践4：运行维护预测分析，提升运营敏捷性



基于DevOps能力矩阵模型，实现从敏捷交付到敏捷运维

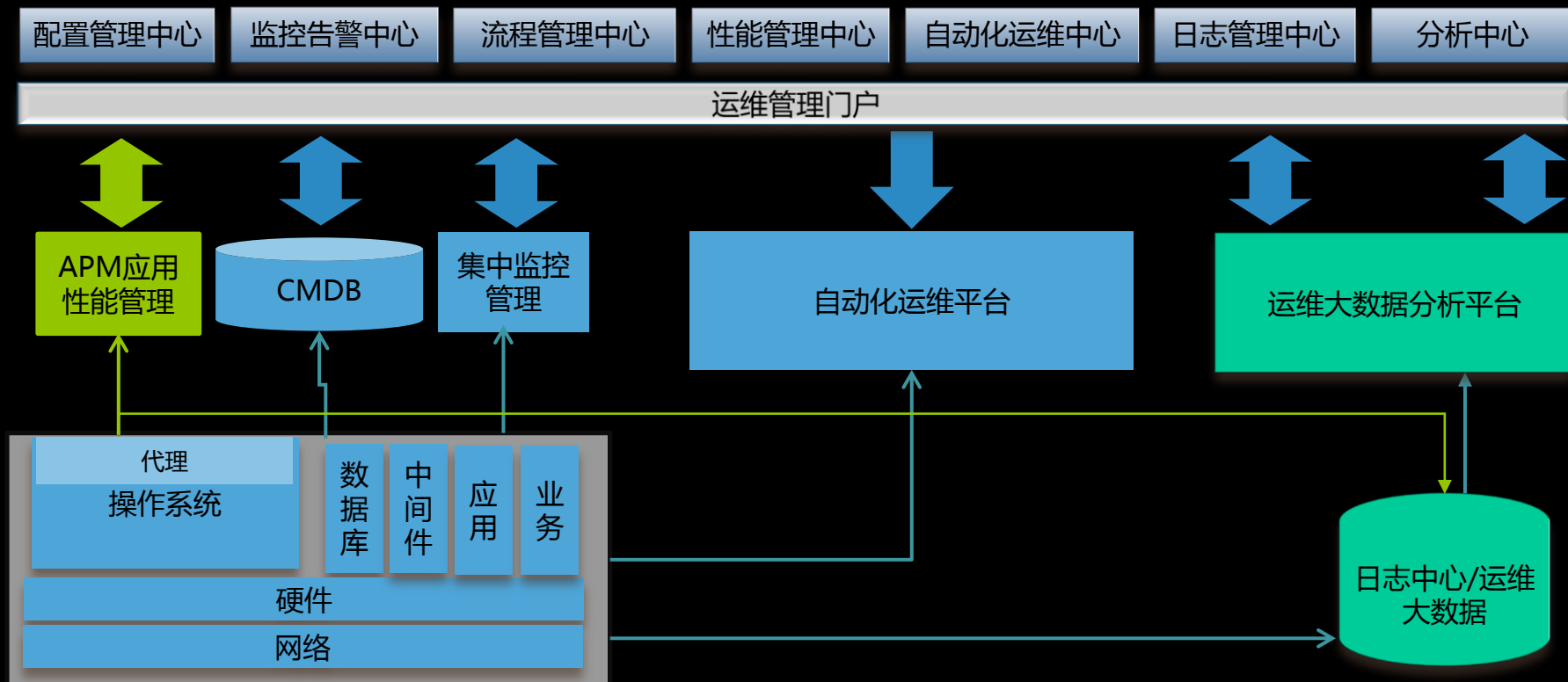
角色视角 X 管理视角

构建企业级DevOps全过程交付能力，提升用户体验！



构建DevOps全过程工具链，打造传统企业的“蓝鲸”体系

DevOps：可视化、数据化、自动化、智能化

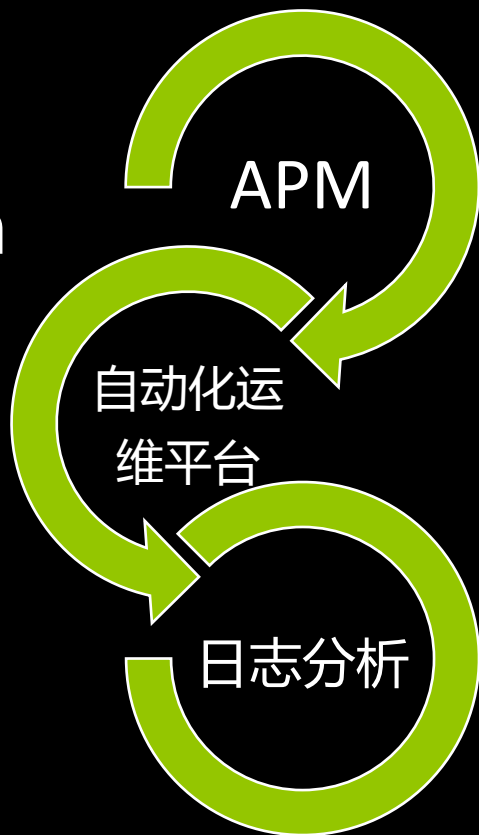


敏捷运维之道，我们一直在努力！

2015

轻维软件

LikeV.cn



2016 *Gdevops.com*

