

DOIS | 2018 · 深圳站
DevOps 落地，从这里开始

DevOps 国际峰会

暨 DevOps 金融峰会

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：2018年11月2日-3日

地址：深圳市南山区圣淘沙大酒店（翡翠店）



中小银行的DevOps实践

@李丙洋

2018.11.02

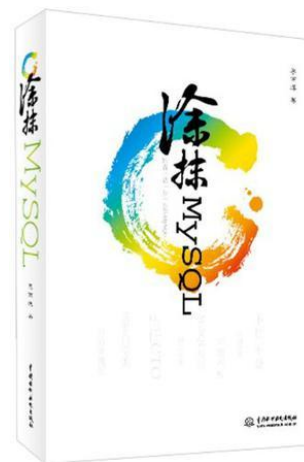
我是谁

DOIS

- » 我是 李丙洋
- » 网名 君三思
- » 来自 富民银行



著有:



[涂抹MySQL](#)



[涂抹Oracle](#)

目录

- ➔ **1** 为什么要做运维自动化
- 2** 我们做了什么
- 3** 我们在做以及将要做什么

传统运维支撑工作的痛点

交付的困境

系统环境复杂, 不同操作系统(如 Unix/Linux/Windows等), 不同主机服务器(如应用服务器、数据库服务器、前置服务器等), 不同设备(如交换机、防火墙、负载均衡、IDS、WAF等)

应用架构愈发复杂的挑战

系统关联关系复杂, 业务处理环节众多, 业务关联性强, 并且随着业务的增长, 应用架构的变化导致运维架构的复杂度急剧上升, 传统运维模式很难支撑现有的运维架构

人为手工操作的风险

版本发布、日常巡检、故障排查、补丁处理等手工操作, 标准化程度低, 操作过程中应用发布时间较长且出现误操作的风险较大

问题/故障定位的效率

应用发布过程中, 由于操作过程的复杂性和人为原因, 容易发生误操作, 而且难以及时发现。问题出现后重现操作过程困难, 不能快速准确地定位问题根源, 最终影响业务系统的正常运行。



日常运维工作



审计与报表支持

- 支持多数据中心的硬件/软件/操作的全面报表;
- 自动生成通用的合规报表;
- 创建可交互的图表, 提供多层次数据, 能深入挖掘;
- 可以导出报表数据与其它报表工具进行集成;

软件/补丁自动化安装

- 操作系统、数据库、中间件、应用服务等补丁批量安装/回滚;
- 工作流方式, 支持统一的调度和协同工作;
- 实现IT流程完整的生命周期, 流程闭环管理;
- 提供可视化工具定制流程;

帐户集中管理

- 提供集中、统一的用户认证和权限管理;
- 实现基于角色、细粒度、灵活的用户权限控制;
- 提供单点登录功能, 避免用户记忆很多口令;
- 支持多种外部的用户认证方式, 包括LDAP、AD等。

日常巡检

规范性质的执行情况检查;
系统状态检查及容量管理检查, 如: CPU/内存/磁盘/表空间/补丁/网络策略/安全设置等;
支持手动、触发式或设定的时间窗口自动周期性执行;

探针服务优化

- 支持数据中心大多数的操作系统;
- 对目标机器的磁盘和内存要求不高;
- 运行时占用资源少, 运行稳定;

IT资产管理

自动采集各种IT资产的配置信息;
对设备进行批量操作, 提高管理效率;
IT资产统一管理;

运维需求 Requirements

运维自动化的优势



自动化

- 改变传统的运维模式中手工应用发布的流程，建立运维自动化操作平台，实现应用发布和巡检的自动化；
- 降低手工操作的潜在风险，减少人为因素带来的影响
- 减轻运维人员负担，提升工作效率，降低运维成本；

- 系统的设计满足操作简单，便于理解和良好的用户体验。减少和屏蔽目前版本升级上线过程中，复杂的命令操作方式，实现升级流程配置化。
- 用户与系统有良好的交互性，每步的运行结果都有实时的展现，同时也方便事后查询。

可视化

运维自动化的优势

DOIS



效率高

系统变更、发布、应急处置自动化，降低对操作人员的需求，提高操作效率



可靠性强

通过运维系统的封装后，将日常的运维操作图形化、标准化，有效避免人为误操作风险



服务标准化

运行成本可进行计量，事件处置或运行分析等均有统一流程步骤，运维服务标准化



易管理

平台化模式运作，自动化程度高，操作流程可控，易于管理，运维操作的数据全面真实

目录

1 为什么要做运维自动化

➔ **2** 我们做了什么

3 我们在做以及将要做什么

先说说怎么做



DEVOPS不是自动化发布系统，不是上个工具就万事大吉
要围绕着企业开发的实际情况，开发流程甚至组织模式同步变更
建立统一的标准、团队协作的文化、完善的基础设施、组织架构的调整适应。

系统建设的基础 - 标准化



系统基线



硬件配置、操作系统版本;
操作系统用户、环境变量;
基础监控、探针组件;

应用服务



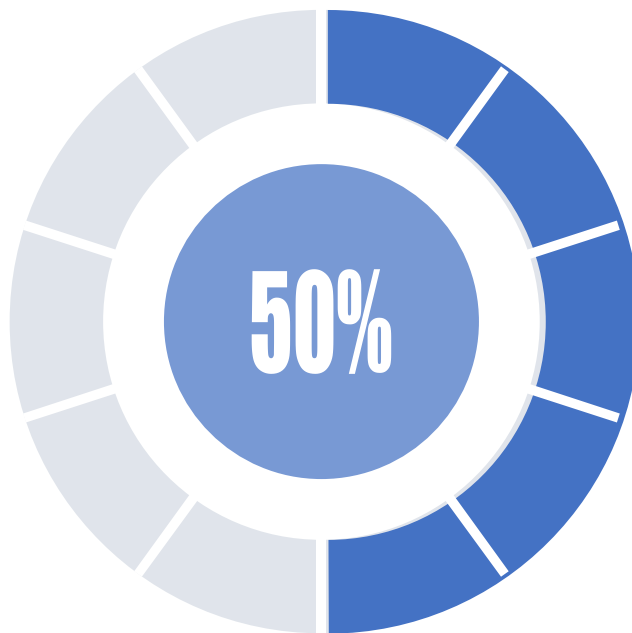
源码管理、分支/版本管理、基线管理;
"统一"JDK版本、Tomcat版本;
"统一"应用路径、包命名规范;

配置与日志



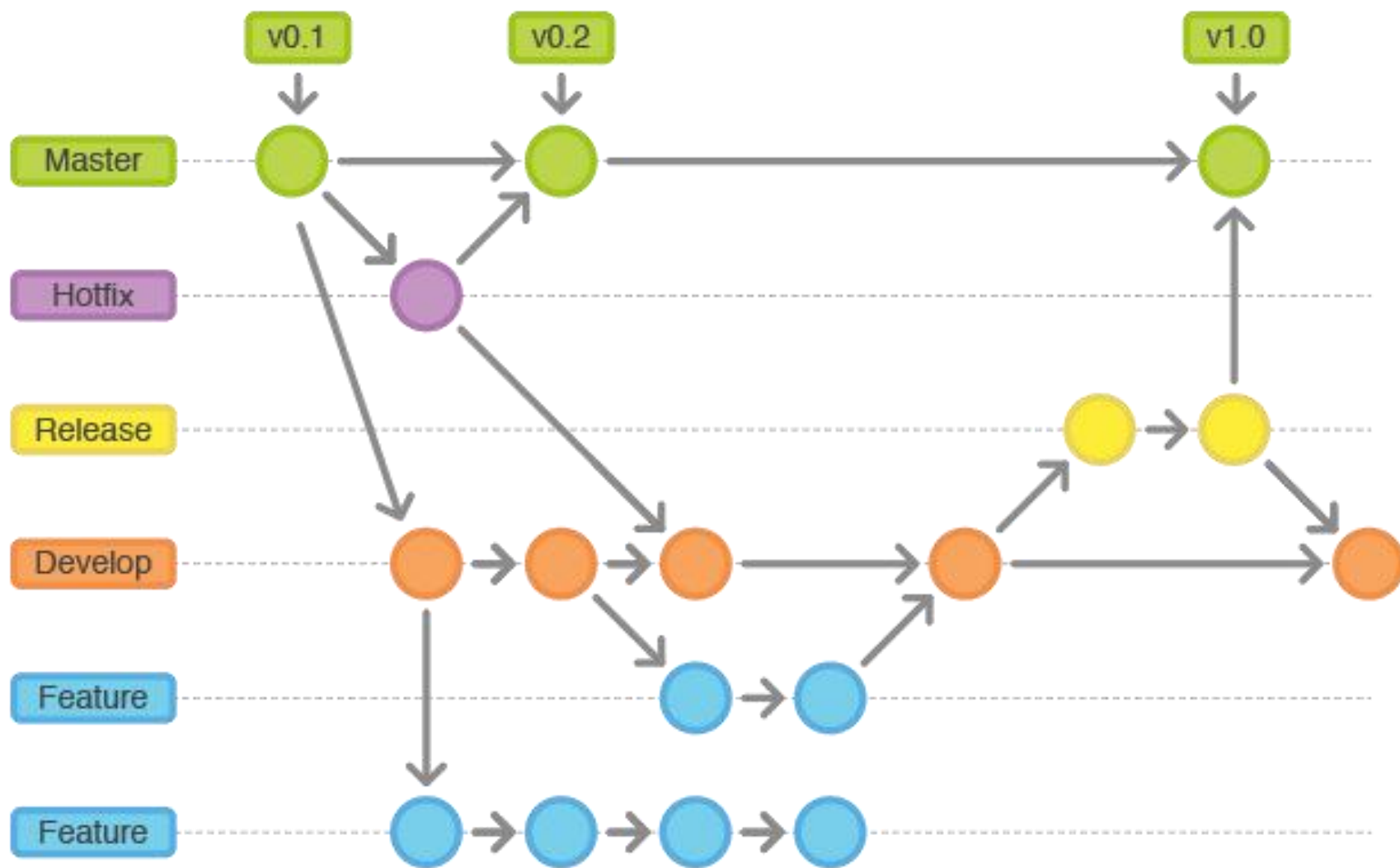
引入配置中心;
强制域名化;
日志输出格式、路径"统一";

系统建设的基础 - 准生产环境



发布到生产环境前的预演

系统建设的基础 - 版本管理策略

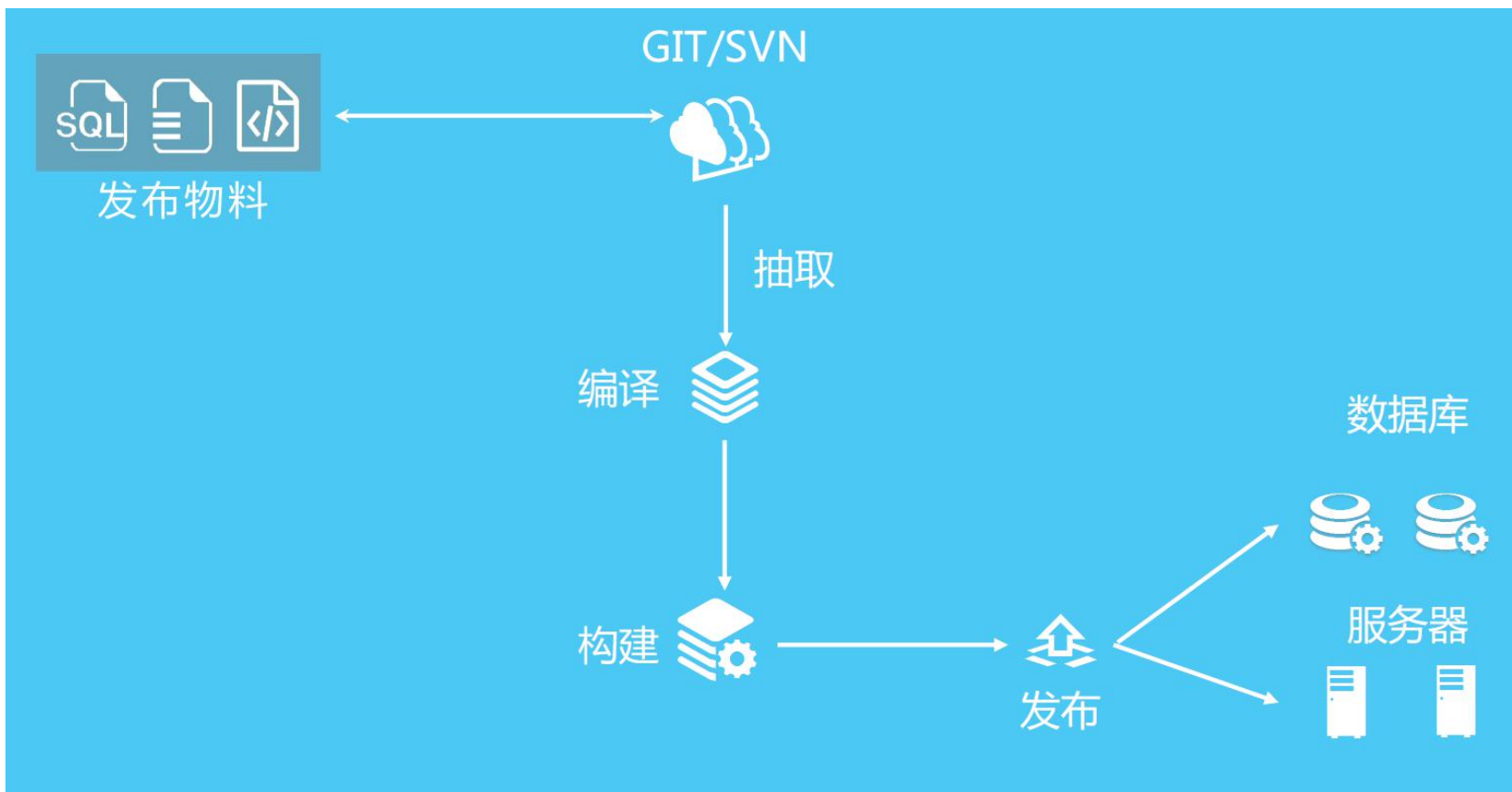


制度要跟上



规范、流程、管理办法

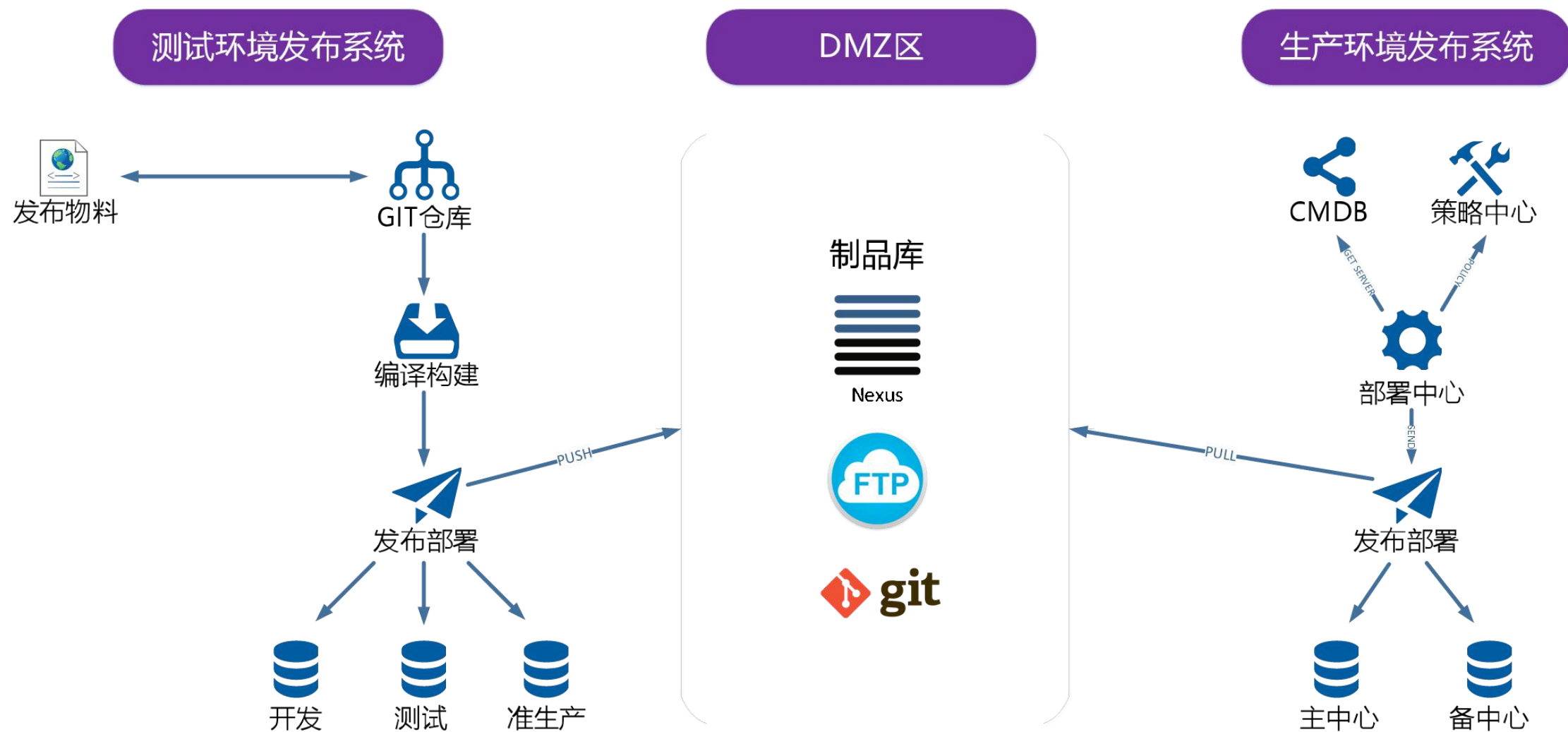
传统发布流程



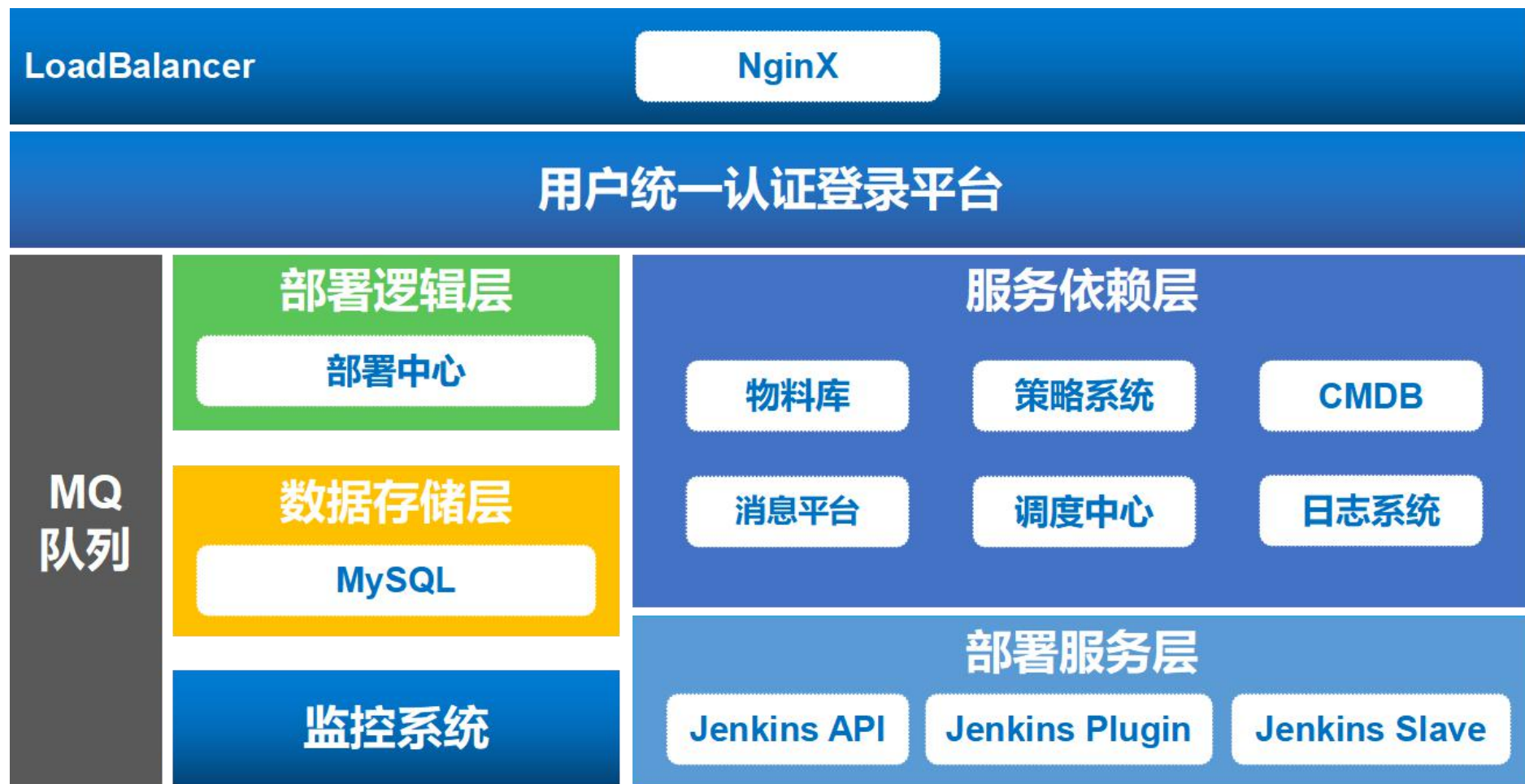
实现从代码编译到应用发布的全程可视化持续部署
固化发布流程，服务启停，应用备份，发布及回滚，风险可见

自动化发布 - 发布流程

DOIS



自动化发布 - 系统架构



自动化发布 - 功能概述



功能	功能简介
发布	实现应用发布过程的一系列过程编排。主要包括：数据字典维护(含应用、服务、服务器等)、发布模板、发布请求、作业、应用war包、脚本、发布过程、监控路径、发布日志、应用日志等
巡检	包括：巡检模板，巡检作业，巡检目标，对象管理，系统巡检，应用巡检，服务巡检，日志分析等
系统管理	包括用户管理、权限管理、角色管理、资源管理、操作日志、系统配置等

自动化发布 - 原型设计

父编号	数据编号	数据名	备注	操作
123	123001	XXXX	XXXX	编辑 删除
123	123002	XXX	XXX	编辑 删除

数据字典

发布计划名称	发布计划时间	应用列表	完成状态	操作
XXX	20180628	应用列表	●	编辑
XXX	20180528	应用列表	●	编辑
XXX	20180428	应用列表	●	编辑
XXX	20180328	应用列表	●	编辑

发布计划

计划名称: []

计划时间: []

添加应用

选择应用: [aaa-app] 输入版本: [master] 优先级: [1] 删除

选择应用: [bbb-app] 输入版本: [v10.0-reg.2] 优先级: [2] 删除

选择应用: [ccc-app] 输入版本: [v10.0-reg.2] 优先级: [3] 删除

提交 重置

管理发布计划

应用列表	版本号	优先级	服务器	进度显示	完成状态	日志查看	操作
xxx-app	v10.0-reg	1	app-1	○	发布成功	日志	发布 回滚 启动 停止
xxx-app	master	2	app-2	○	回滚失败	日志	发布 回滚 启动 停止
xxx-app	v2.0.0-reg	3	app-3	○	启动成功	日志	发布 回滚 启动 停止
xxx-app	v3.0-reg	14	app-4	○	停止失败	日志	发布 回滚 启动 停止

应用列表

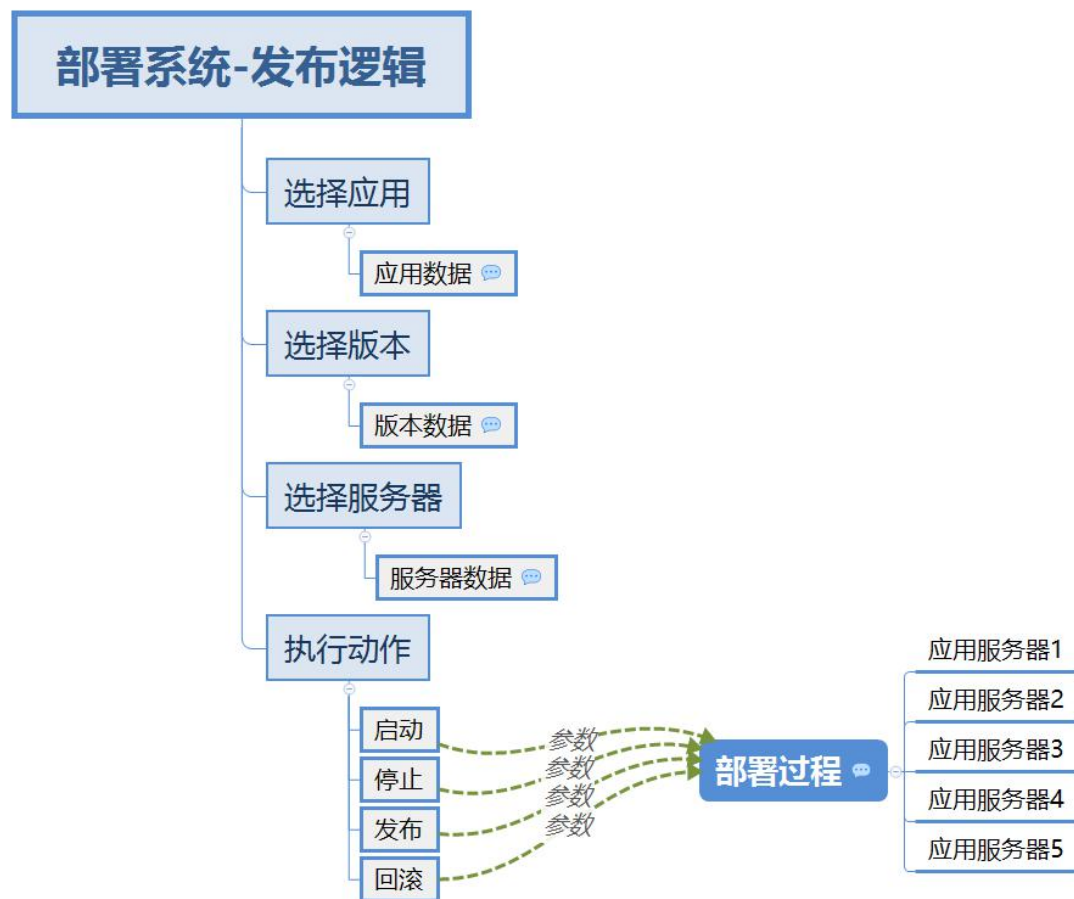
应用名	版本号	状态	执行编号	日志查看	创建时间
123	123001	XXXX	XXXX	日志查看	2018-5-28
123	123002	XXX	XXX	日志查看	2018-5-28

发布日志

项目组名	组标识	应用名	负责人	端口号	构建类型	包类型	构建url	校验url	云效url	备注	操作
xxx战队	xxx	xxx-app	aaa	8080	mvn	war	/job/xxx	/app	http://xxxx	123	编辑 删除
xxx战队	xxx	app	aaa	8080	gradle	jar	/job/xxx	/app	http://xxxx	123	编辑 删除

应用配置

自动化发布 - 发布逻辑



自动化发布



DOIS

Dashboard / 发布计划 / 发布计划

Dashboard × 应用配置 × 服务器配置 × 发布计划 ×

shishuocms 实例列表, 待发布版本 2018090912

服务器	当前版本	应用状态	显示进度	状态描述	jenkins日志	tomcat日志	应用日志	操作
			<div><div></div></div> 100%	发布成功	查看	查看	查看	冷备 发布 回滚 启动 停止
			<div><div></div></div> 0%	SFTP拉取应用成功	查看	查看	查看	冷备 发布 回滚 启动 停止
			<div><div></div></div> 0%	SFTP拉取应用成功	查看	查看	查看	冷备 发布 回滚 启动 停止

发布计划	状态	评审	发布	操作
发布计划测试026		评审成功	待发布	计划编辑 应用变更
发布计划测试025		评审成功	待发布	计划编辑 应用变更
发布计划测试024		评审成功	待发布	计划编辑 应用变更
发布计划测试023		评审成功	执行中	计划编辑 应用变更
发布计划测试022		评审未处理	待发布	计划编辑 应用变更

Total 31 10/page < 1 2 3 4 > Go to 1

自动化运维 - 装机



- » 统一管理不同环境中的主机资源
- » 支持按需(不同类型)快速初始化服务器
- » 虚拟化平台的容器管理中心
- » 具备权限管理、日志管理、操作统计

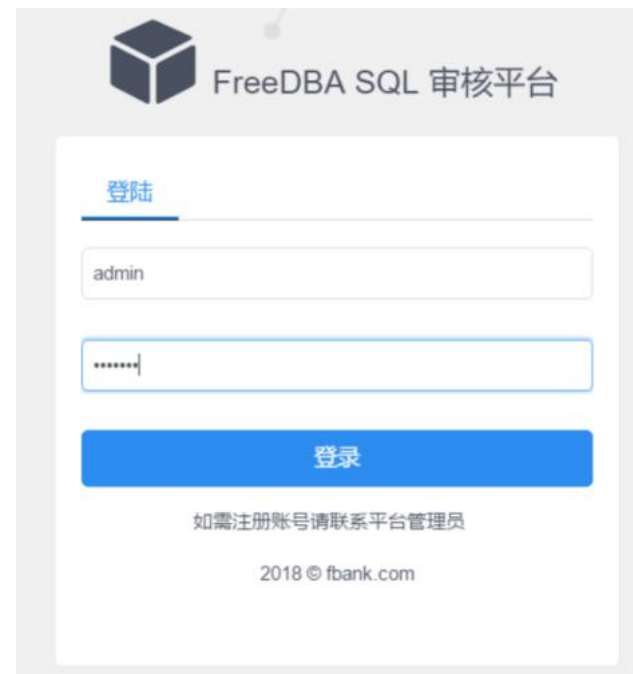
虚拟机(remote)

id	名字	状态	ip	cpu	内存(GB)	操作
vm-590	10.1.10.244	poweredOn	10.1.10.244	1	2	关机 启动 控制台 编辑 删除
vm-1169	DNS服务器	poweredOn	10.1.40.11	2	4	关机 启动 控制台 编辑 删除
vm-1548	DR-mysql01	poweredOn	10.1.0.66	2	4	关机 启动 控制台 编辑 删除
vm-1549	DR-mysql02	poweredOn	10.1.0.67	2	4	关机 启动 控制台 编辑 删除
vm-1550	DR-mysql03	poweredOn	10.1.0.68	2	4	关机 启动 控制台 编辑 删除
vm-505	ELK01	poweredOn	10.1.10.221	2	8	关机 启动 控制台 编辑 删除
vm-507	ELK02	poweredOn	10.1.10.222	2	8	关机 启动 控制台 编辑 删除
vm-508	ELK03	poweredOn	10.1.10.223	2	8	关机 启动 控制台 编辑 删除
vm-560	FastDFS	poweredOn	10.1.10.15	2	4	关机 启动 控制台 编辑 删除
vm-841	FastDFS-cloneforYuYang	poweredOn	10.1.11.231	2	4	关机 启动 控制台 编辑 删除
vm-620	FastDFS group02	poweredOn	10.1.10.22	2	4	关机 启动 控制台 编辑 删除

自动化运维 - 数据库订正

DOIS

- » SQL自动补全
- » DDL语句标准化审核
- » SQL语句检测、执行、回滚
- » 支持工作流
- » 用户权限管理



感谢: *mysql-inception*



自动化运维 - 配置中心



- » 统一管理不同环境中的配置
- » 配置变更可实时更新，无须应用端重启
- » 支持对配置的版本管理
- » 具备权限管理、发布审核、操作审计

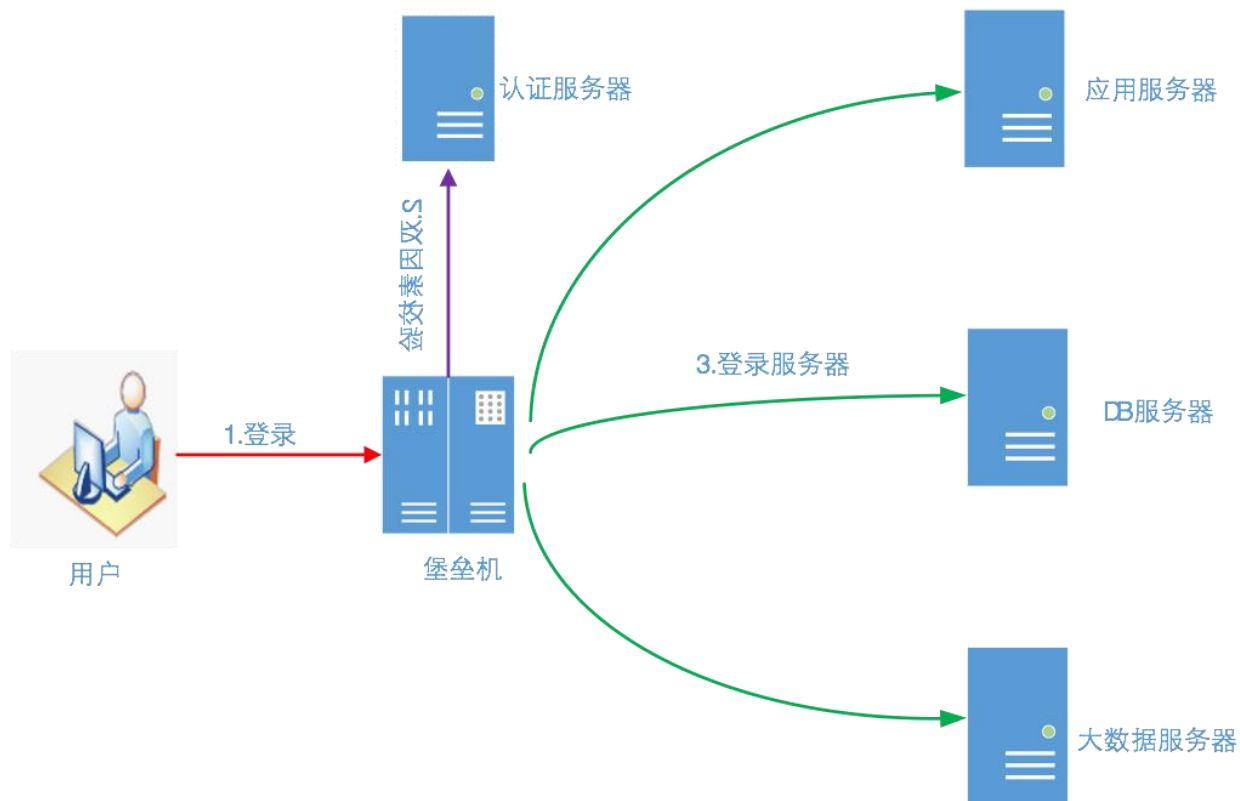


Apollo

A reliable configuration management system

自动化运维 - 操作审计

- » 采用堡垒机做为运维安全审计系统
- » 操作人员使用自己专用账户
- » 采用双因素校验，免密登录服务器，防止服务器密码泄露
- » 接入生产环境所有设备
- » 安全团队定期对堡垒机记录事件进行审计



DevOps是一场持久战



DevOps最重要的目标是为交付过程赋能，加速研发

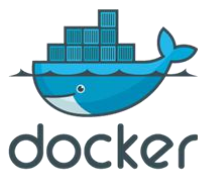
DevOps是个持久战，特别是面对存量

DevOps需要强力的保障团队

自动化发布系统 - 工具链

DOIS

环境&资源



工具链

源码



配置



缺陷



编译



Jenkins



PM



REDMINE

flexible project management



Zen&Tao
禅道

测试



监控



目录

1 为什么要做运维自动化

2 我们做了什么

→ 3 我们在做以及将要做什么

我们又遇到痛点了



发布系统功能仍然很弱，并不完善
灰度发布需求强烈，但对现有系统改造较大

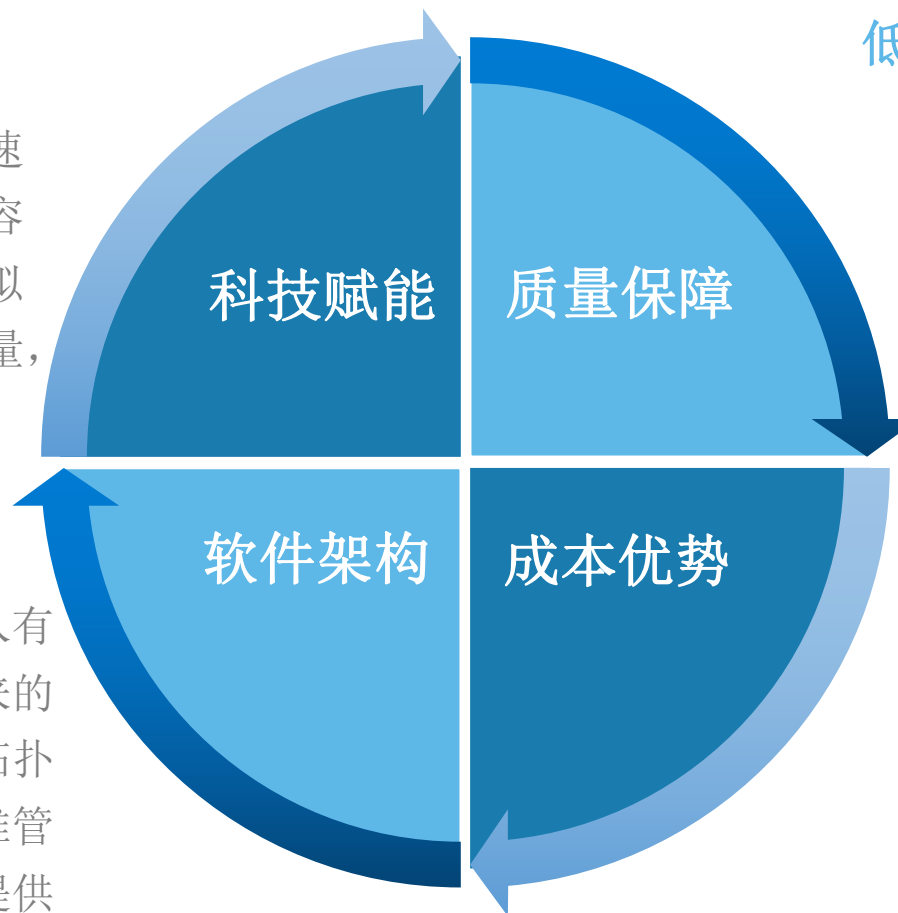
为什么要容器化

支撑业务的基础设施

容器化可做为支撑业务系统快速落地的最佳技术手段；运行于容器上的应用健壮程度远高于虚拟化；容器化的应用系统非常轻量，也将为平台快速复用打下基础。

软件架构的先进性

我行互联网业务微服务化的引入有其优势，但其自身特点同步带来的服务拆分，服务器数据增多，拓扑结构复杂化，这些又都会使运维管理工作变的更有挑战。容器云提供解决这类问题的基础平台。



低风险的快速部署提升质量保障

一次构建，多次交付，随着交付物的标准化，打包至镜像的应用模板不管在开发、测试或是生产环境，都是使用相同的一套程序代码和运行环境，从而有效的降低我行因环境差异而导致的软件故障。

成本优势

容器可以显著减少启动和管理的虚拟机数量，改变每个应用都需要一个(至少)虚拟机的需求，减少大量的存储和内存资源的浪费，进而转化为巨大的成本节省，保守估计能够节约50%的服务器成本。

容器的优势



更低的使用成本

相比VMWare和KVM虚拟化密度提升数倍

企业级容器云平台

采用企业级商用解决方案，达到4个9以上
稳定性

效率

更高的效率

提升开发、测试和运维效率

成本

devops

完善的支持devops

微服务、容器化和CI/CD一体化，产品上
张时间显著缩短，支持全自动化运维，减
轻运维人员工作量

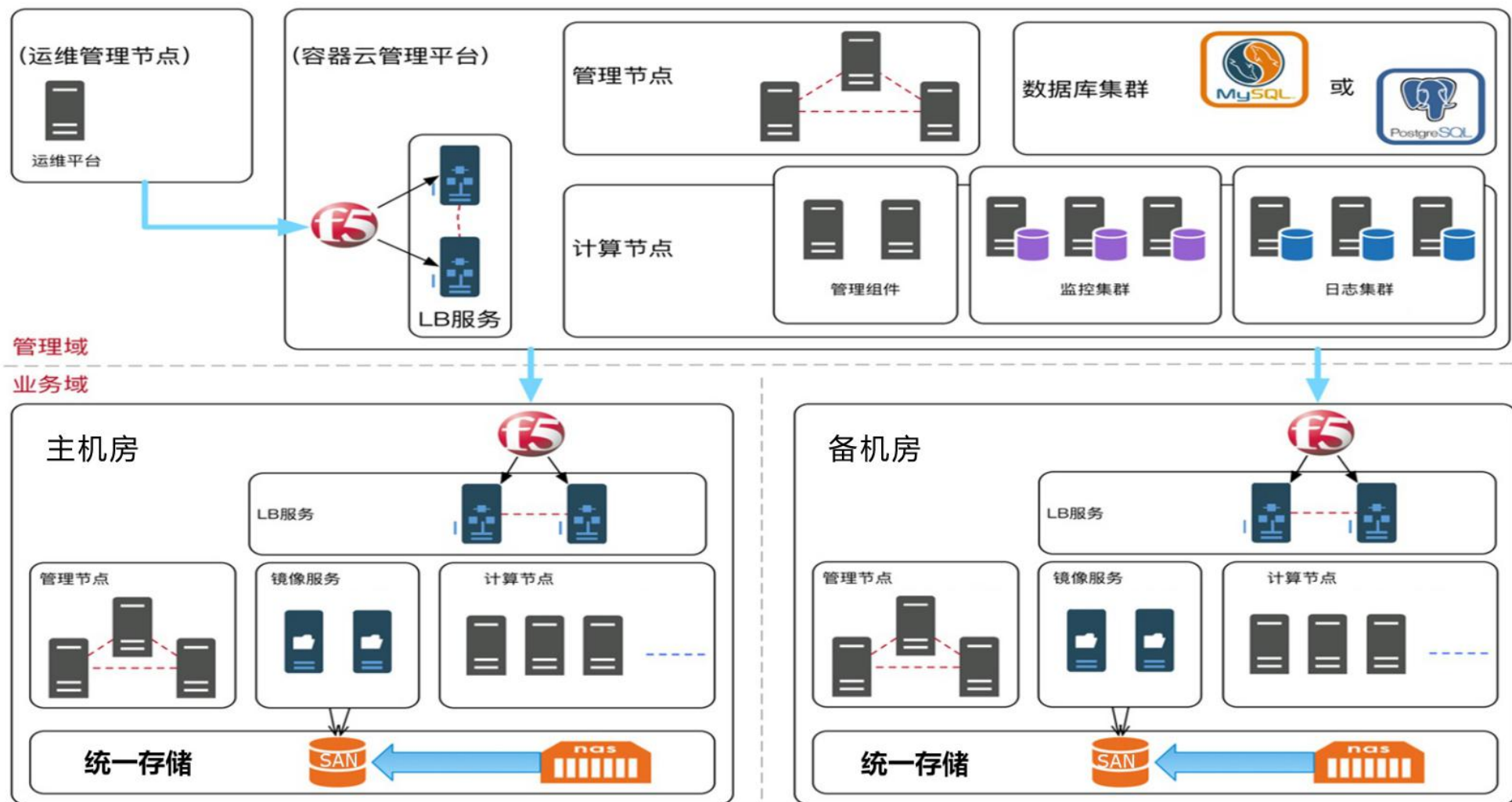
稳定

多元化支撑

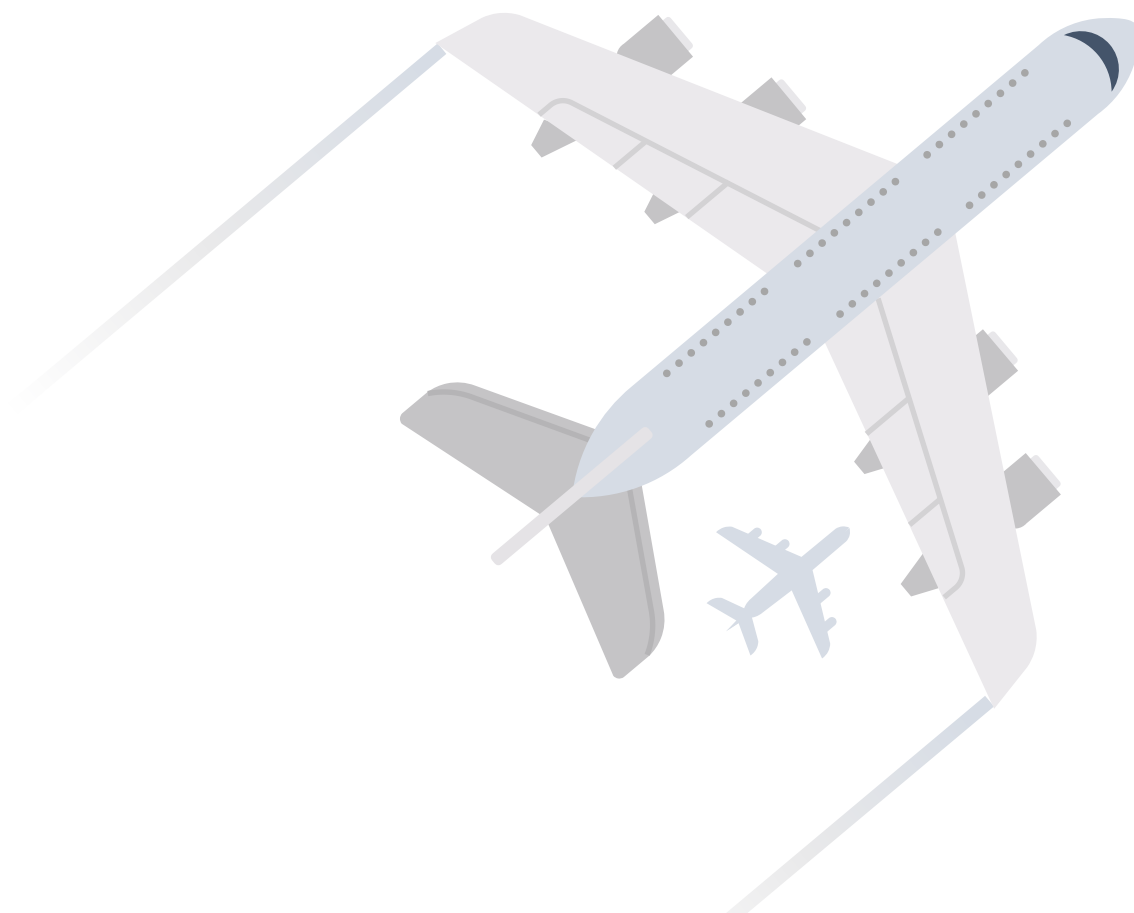
除了支持传统应用外，也支持大数据、人工
智能和高性能计算集群平台

多元化

部署架构



ServiceMesh



给传统应用，插上微服务的翅膀



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？

