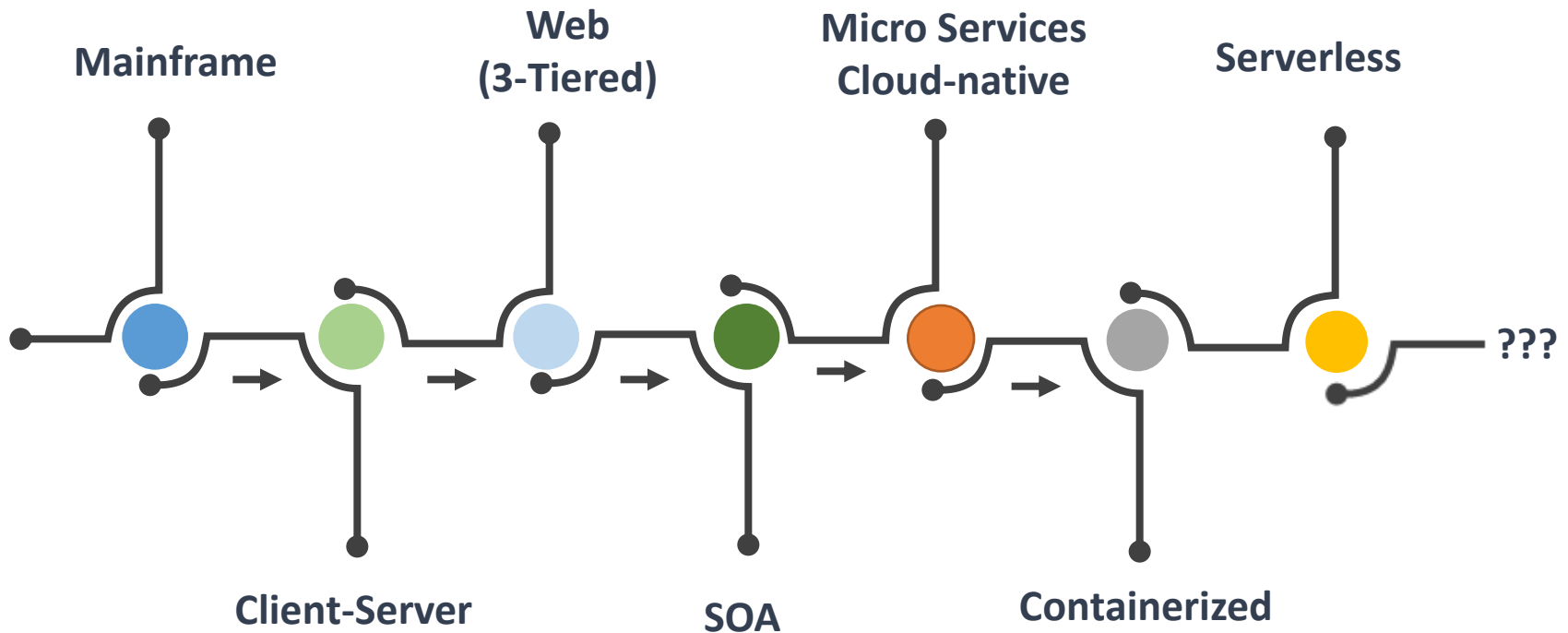


容器化引领IT新常态

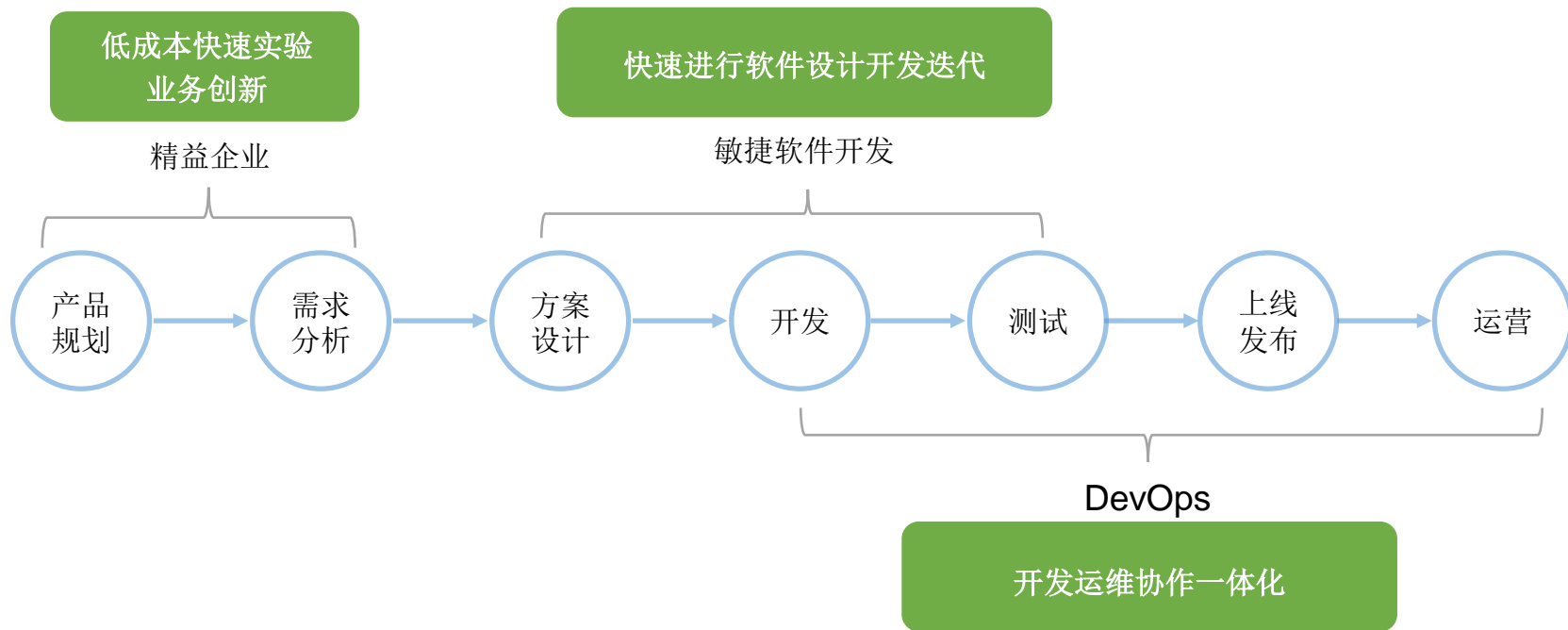
向云迁移，向devops 转型，向新应用架构转变



不停息的应用架构“运动”

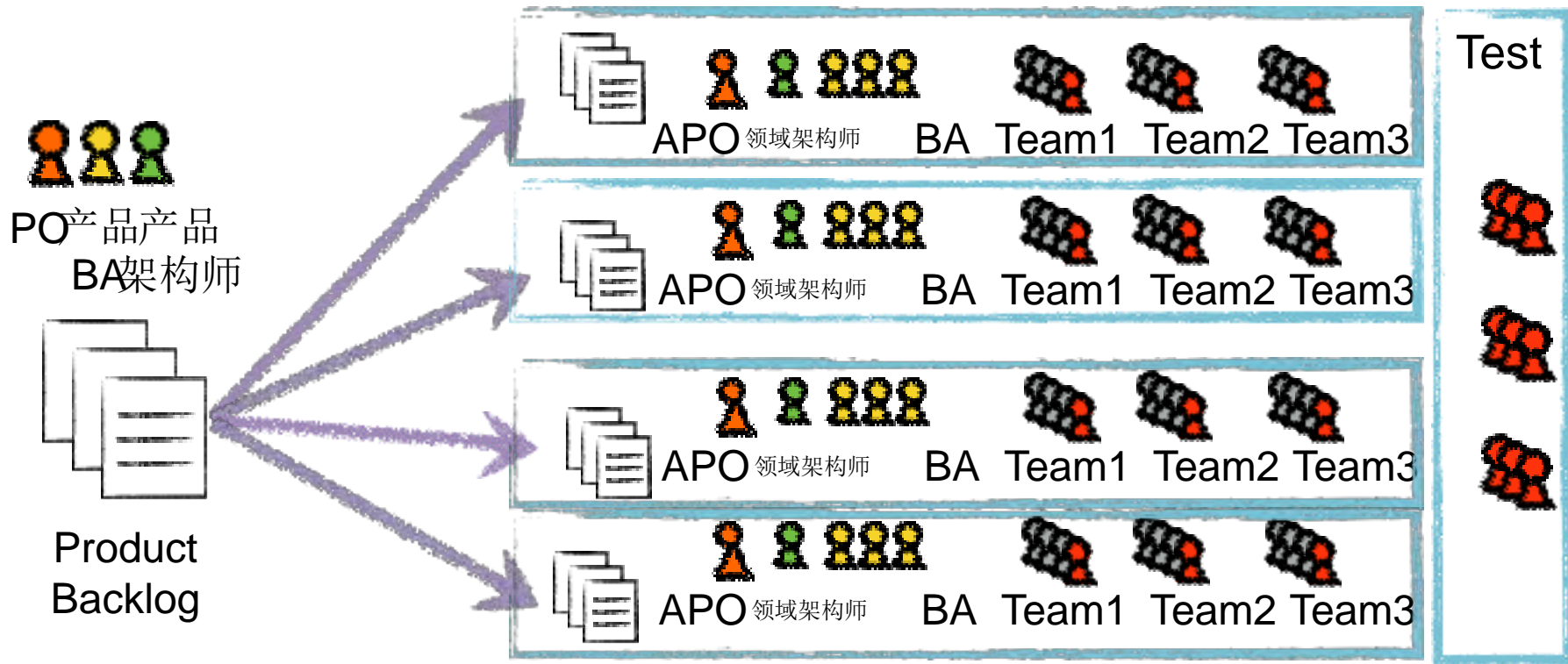


十年软件产品交付变革



互联网思维的软件产品交付理念： 快速的方式交付产品，收集用户反馈，持续对产品进行改进

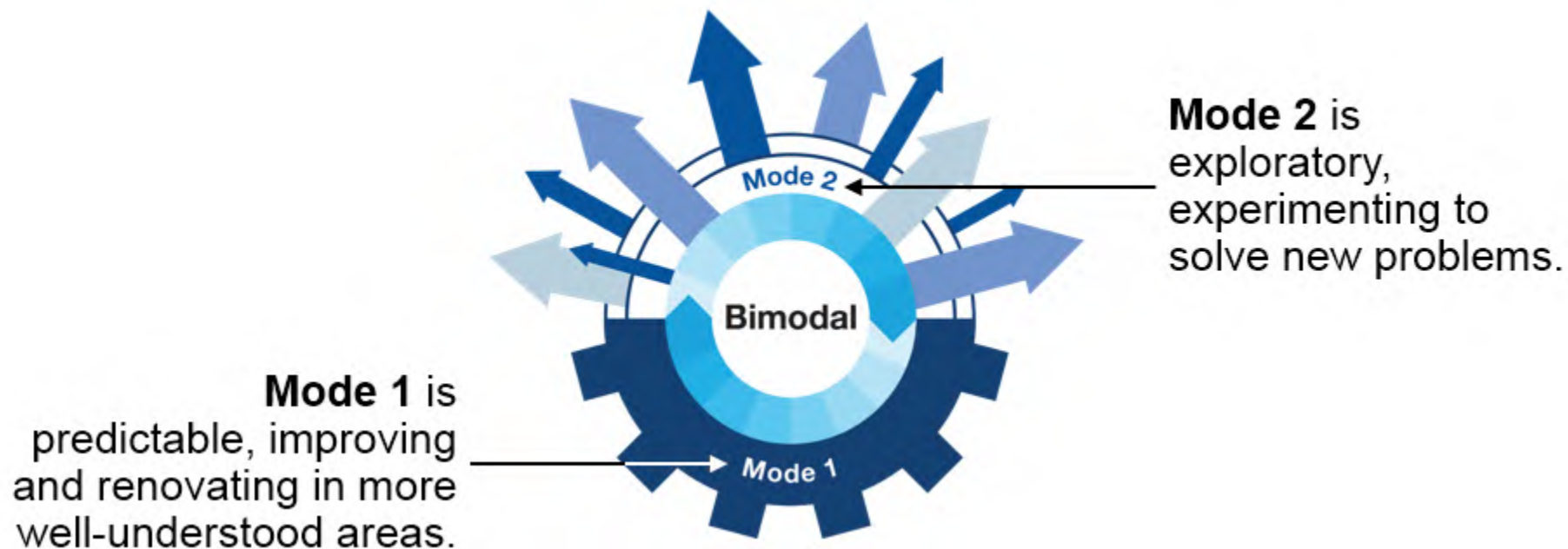
组织结构和协作方式转型



“ 产品必然是其组织内部沟通结构的缩影 ”

—— 康威定律 Conway's laws

Bimodal 正在成为传统企业适应IT新常态而采用的治理模式





**应用容器化会是未来最热门的
“软件运动”**

应用容器化 “运动”

Github 上的Dockerfile

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

nathanxu committed on GitHub Update bookstore.sql | Latest commit be2ce0a 7 days ago

File	Commit Message	Time
agent	add 1.6.1 jar	8 days ago
bookstore	change	11 days ago
Dockerfile	add pinpoint agent	9 days ago
Dockerfile.bak	change database connection	11 days ago
Dockerfile.dbupdate	Update Dockerfile.dbupdate	10 days ago

Build -> Ship - Run

```
+ cd -
+ docker build -t registry.cn-hangzhou.aliyuncs.com/wise2c-uat-
Step 1 : FROM registry.wise2c.com/library/mysql:5.6
+ docker push registry.cn-hangzhou.aliyuncs.com/wise2c-uat-demo
The push refers to a repository [registry.cn-hangzhou.aliyuncs.com]
9b0709177239: Preparing
```

Dockerhub 上的应用

X Databases

TYPE

- Store
- Community (Docker Hub)

DOCKER CERTIFIED

- Docker Certified

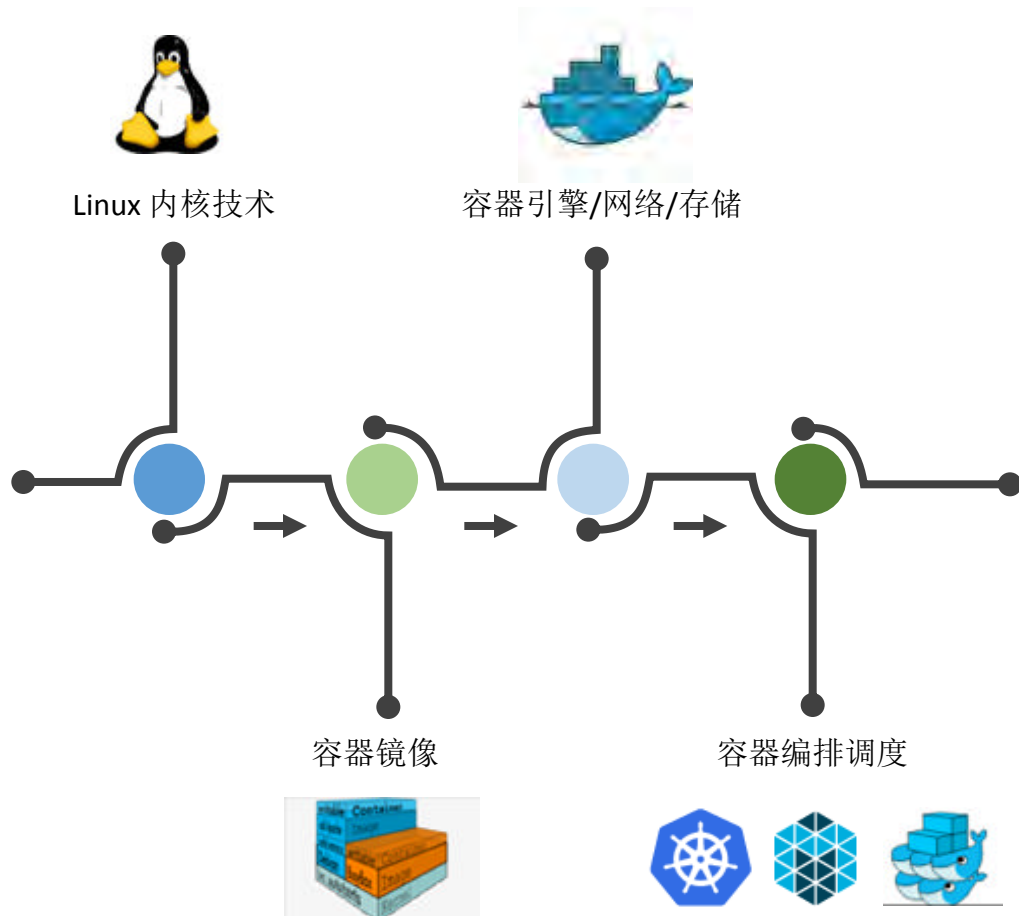
CATEGORIES

- Analytics
- Application Frameworks
- Application Infrastructure
- Application Services

Neo4j Community
Neo Technology, Inc.
Neo4j is a highly scalable, robust, native graph database. It is used in mission-critical apps by thousands of...
Databases

Oracle Database Enterprise Edition
Oracle
Oracle Database 12c Enterprise Edition
Databases

认识容器化技术



容器化技术链

- 容器化相关的技术很多都不是新技术
- 不是拍脑袋的技术试验
- 基于技术界多年积累的技术、工具
- 为应用提供基础设施环境, 研发, 交付到运维完整工具链



思想/理念

- Build once, run everywhere
- Build -> ship -> Run
- Immutable infrastructure
- Immutable Deployment

面向应用的规范

- 应用封装
- 应用交付
- 应用编排

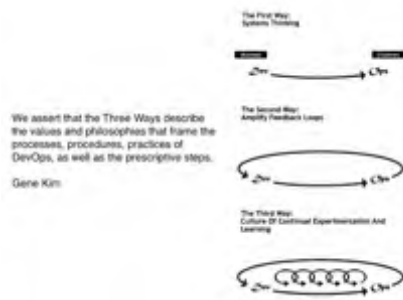
容器技术的出现，也在促进其它技术发展和落地



Container OS

- THE TWELVE FACTORS
 - I. Codebase
 - II. Dependencies
 - III. Config
 - IV. Backing Services
 - V. Build, release, run
 - VI. Processes
 - VII. Port binding
 - VIII. Concurrency
 - IX. Disposability
 - X. Dev/prod parity
 - XI. Logs
 - XII. Admin processes

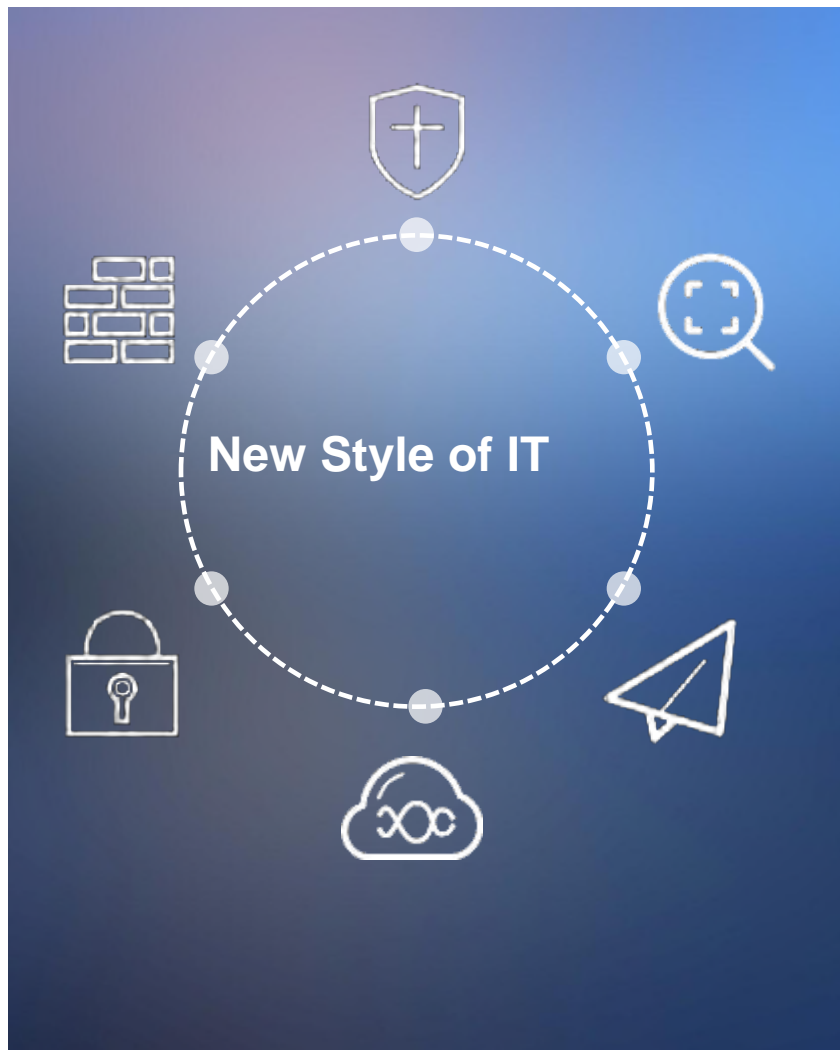
Cloud native 应用架构



devops 工具和理念









微服务



跨越两种IT模式

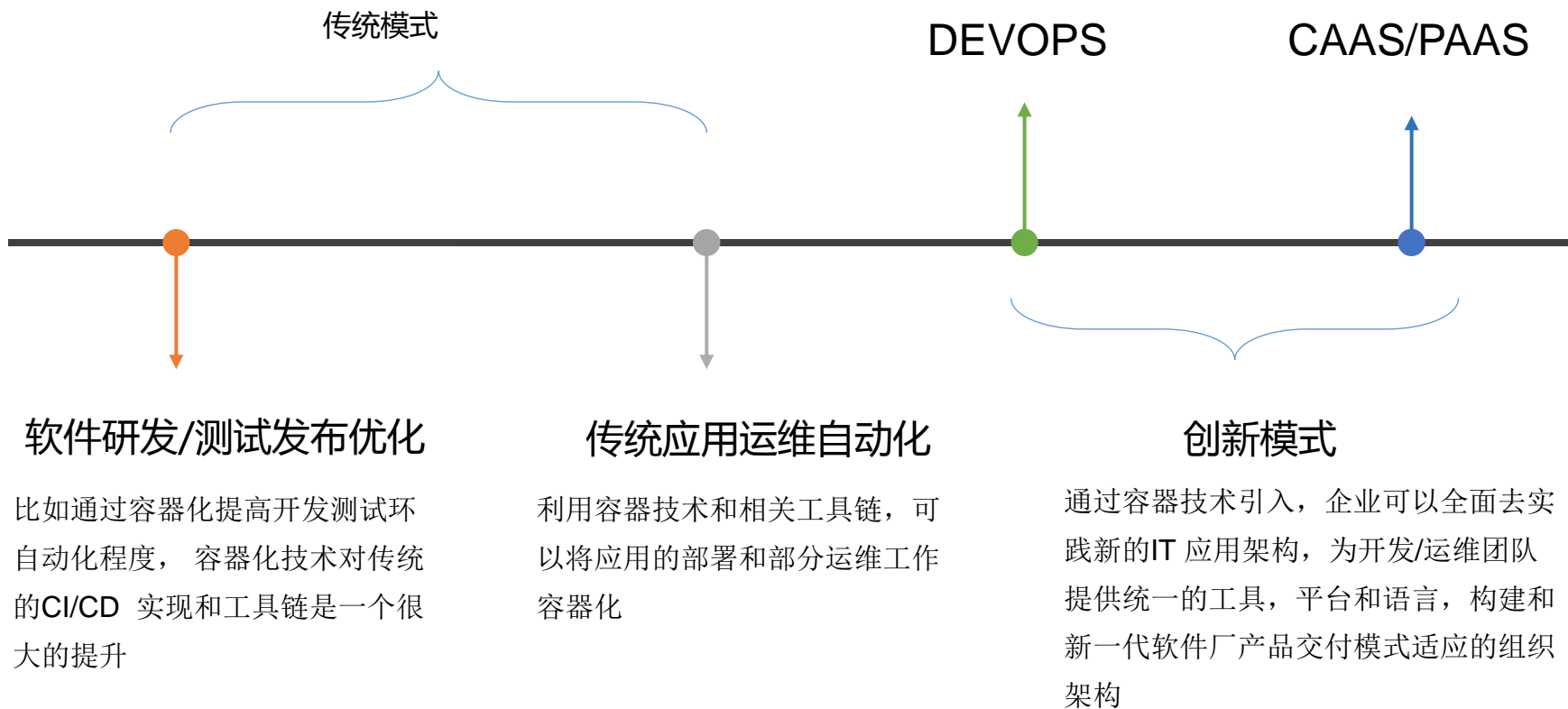
容器技术既是从过去传统IT技术和思想的升华，也包含从互联网IT实践中出来新思想和理念。

传统企业IT的容器实践和导入，是一个从旧IT架构逐渐过渡到新IT架构的过程

- | | |
|---|---|
|  安全 |  弹性 |
|  稳定 |  敏捷 |
|  可靠 |  Lean |

传统企业容器化实践路径

传统企业容器化实践路径

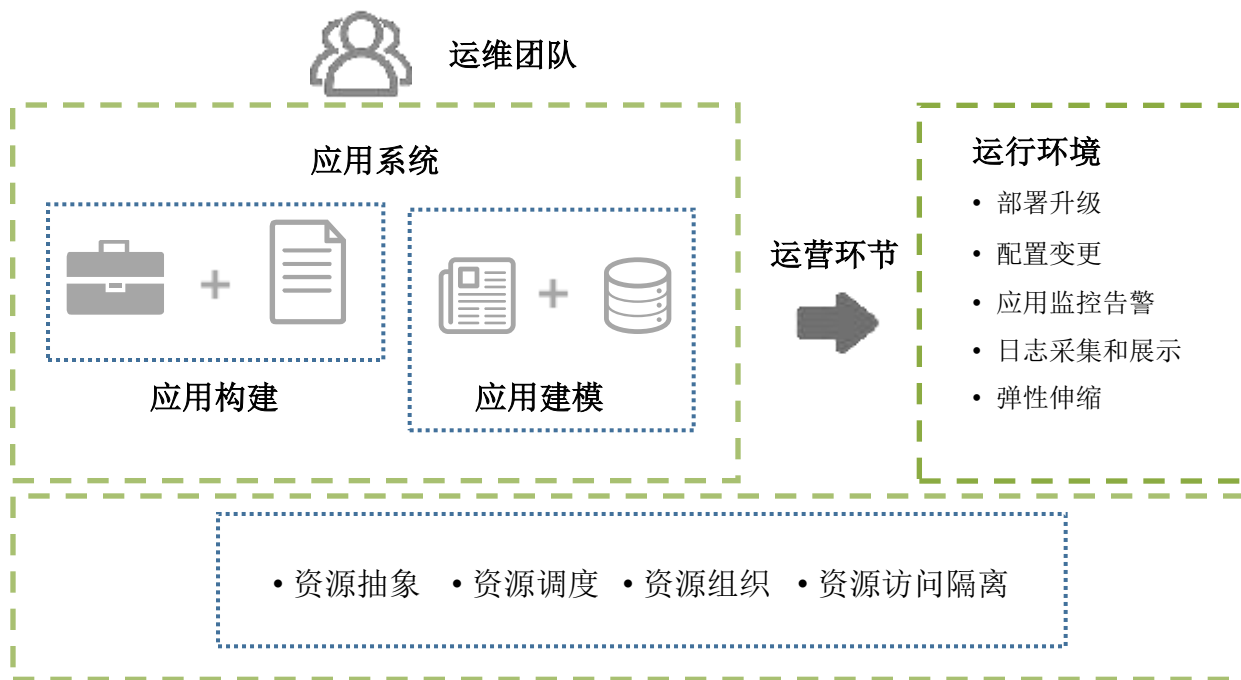


先谈“恋爱” 再“结婚”

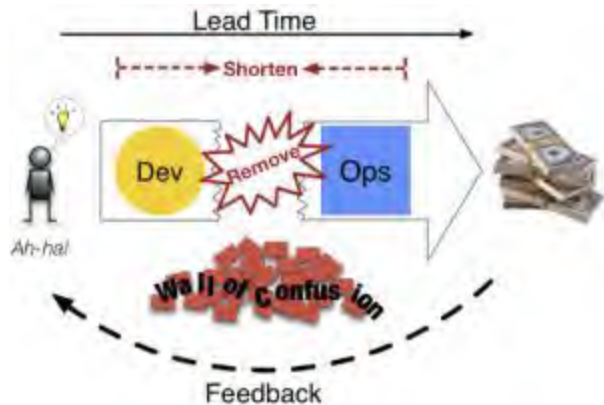


容器化技术还是一个处在“moving fast”阶段，容器化过程是一个由点及面，由可靠到创新的过程，传统企业可以尝试先作从封装，部署到最后全面容器化平台托管应用这样一个逐渐导入的过程

用容器工具链建设自动化运维平台



- 取代过去传统的用脚本进行配置环境,应用部署的方法
- 在部署平台上实现一整个开发/测试环境的一键部署
- 在实现自动部署的同时, 在平台提供各种高级部署功能
- 滚动更新, 蓝绿发布



容器技术和DevOps

- 使用容器技术来打造CI/CD 流水线
- 快速构建dev/test 环境
- 建立 Build – ship – run 流程

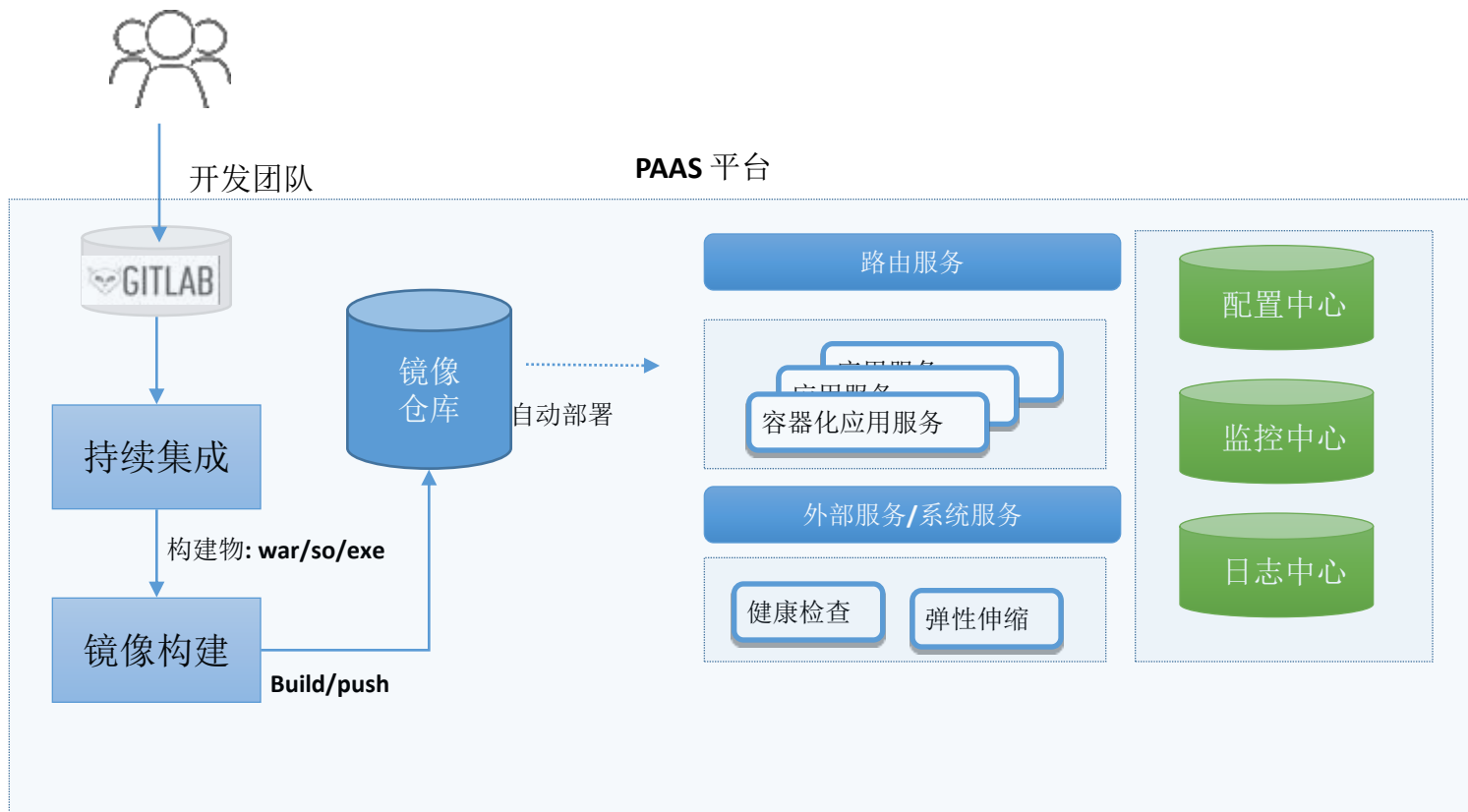
让开发和运维用同一种共语言沟通

- 应用打包 – docker file
- 应用配置 - compose file
- 应用编排 - compose file/secrets/configmap



利用容器技术构建PAAS 平台

在双模IT 中 为cloud native 应用提供开发平台和弹性/敏捷的运行环境



容器化挑战和观察



- 具体技术问题，比如docker engine 的平滑升级，镜像垃圾回收， 存储启动, 安全
- Moving fast. 相关技术都处于快速迭代周期，路线变化快等等
- 分化和割裂的容器生态圈，对容器管理 技术选型带来一定困难
- 从基础设施层到上层编排和调度， 各种各样样的“标准”,给上层应用带来困难
- 对容器化技术的本质认识，包括对容器化技术的过度定位
- 创新技术在企业导入的通用性问题，思维认识理念，团队能力等

VM vs. Docker



Size		
Startup		
Integration		

容器真的不是虚拟机

- 应用为中心
- Immutable Infrastructure



容器是可以很好的支持传统应用！

- Host network/pid , volume, K8s stateful controller
- 最保守的应用场景：把应用放到一个“沙箱”而已



是否一定要使用overlay network !

⑧ 其实企业应用场景，用户更喜欢扁平网络

Thanks
