DevOps软件开发模式

（下面内容节选自华为、微软等公司培训资料，版权归公司所有）

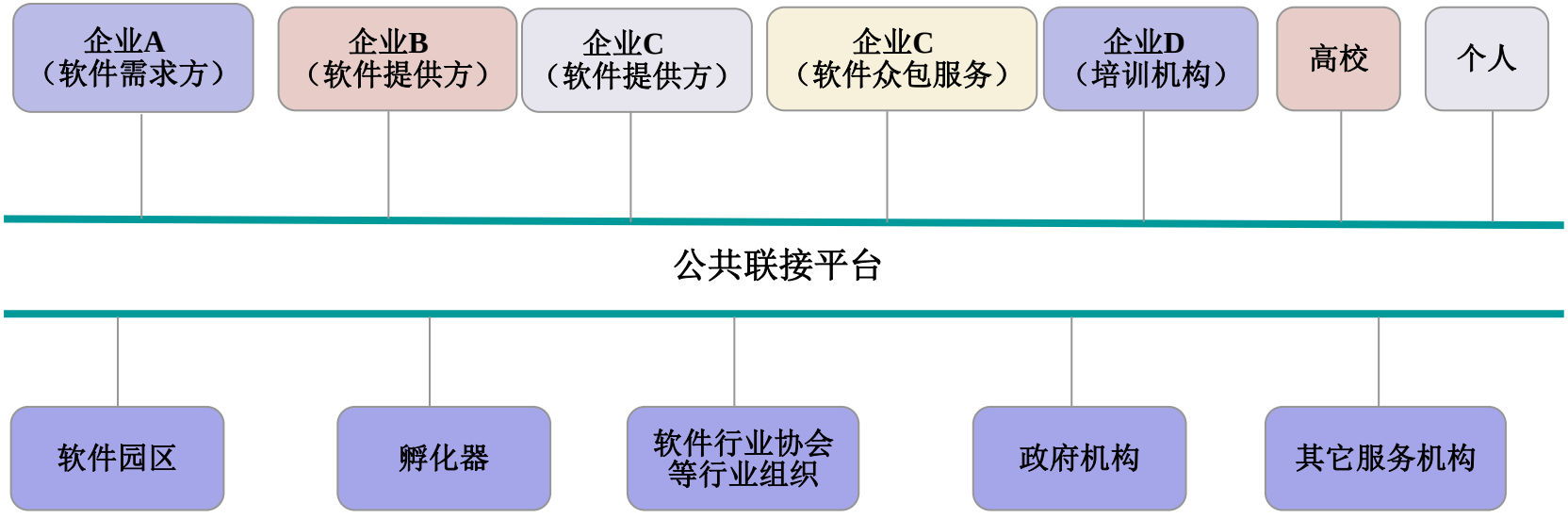
洪玫 教授

四川大学软件学院

E-Mail：hongmei@scu.edu.cn

软件开发、运行、维护的生态环境变化

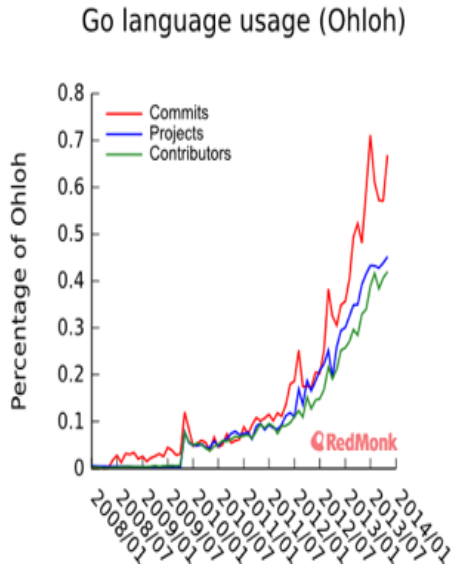
- 软件需求方与软件提供方、软件提供方之间更好地协作，以共同把握需求、进度、质量以及风险。
- 软件园区、孵化器向软服务转型，整合在园企业优势，并为企业提供研发架构、质量、工具等方面的服务与经验。
- 高校、培训机构等与产业需求密切衔接，培养符合产业需求并具有工程化实战能力的人才。



软件开发特点——多语言混合编程

云计算、大数据开源项目中呈现多语言混合编程状态

Go语言发展迅速，Cloud Foundry，Docker，CoreOS

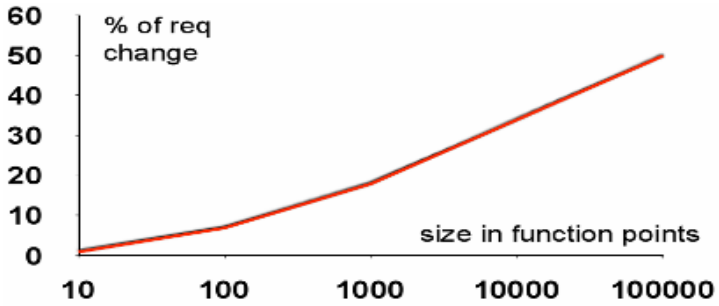


GitHub排名前5000项目语言使用情况：

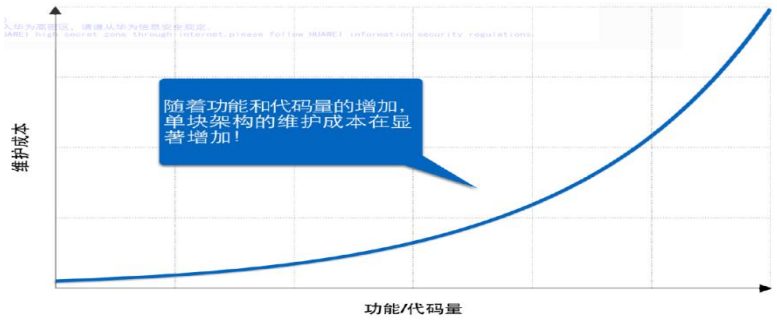
编程语言	1~10	1~100	1~1000	1~5000
JavaScript	7	54	385	1605
CSS	2	8	41	174
Ruby	1	9	153	786
Python		5	64	420
Unknown		5	30	138
C++		4	22	108
PHP		3	38	248
Shell		3	19	89
Objective-C		2	89	495
C		2	31	185
Go		2	13	61
Java		1	32	255

软件规模和复杂度增加

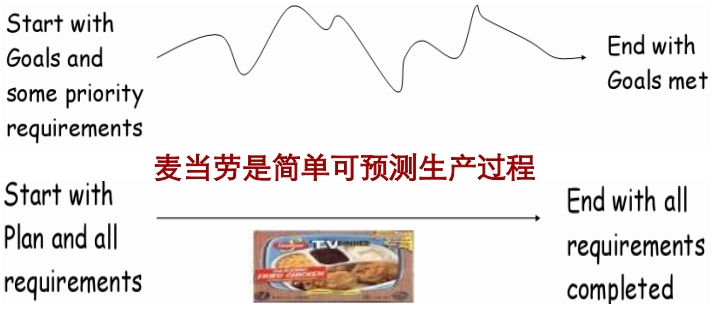
随软件规模增长，需求变化呈非线性增长



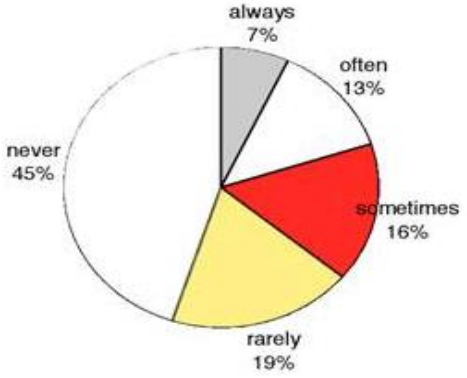
随软件规模增长，维护成本呈非线性增长



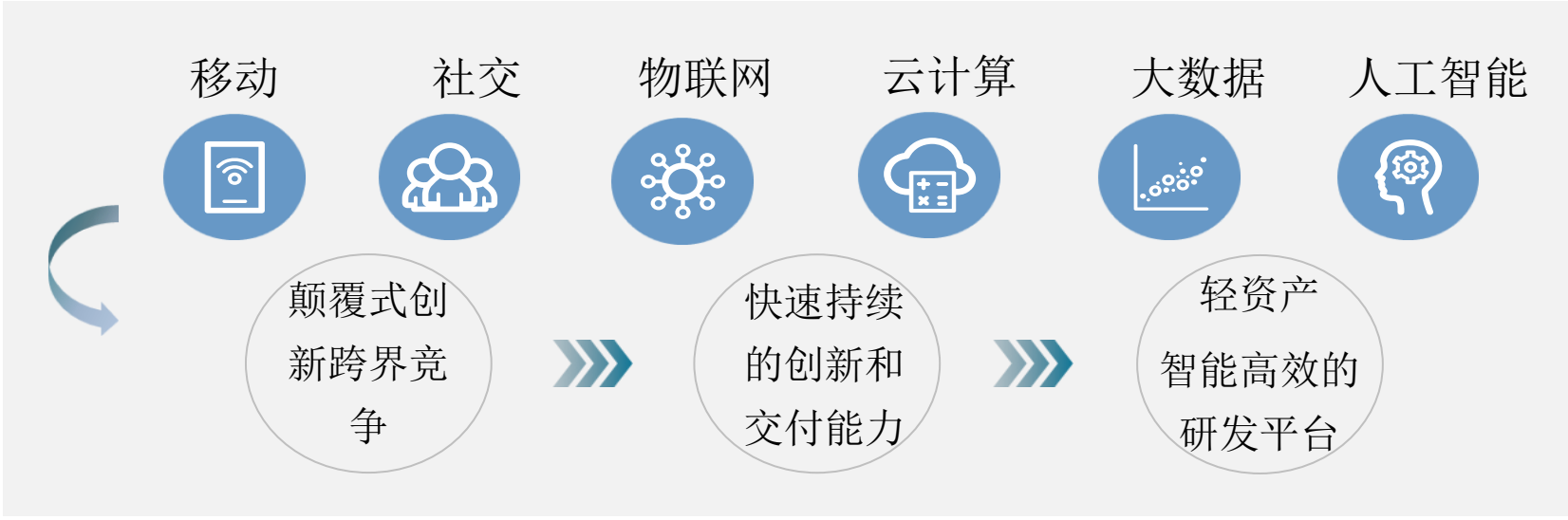
软件开发是复杂不可预测的经验控制过程



45%特性从未被使用，80%很少使用



软件跨领域融合

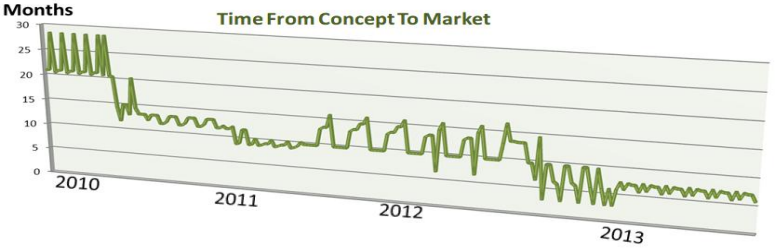

移动
社交
物联网
云计算
大数据
人工智能
颠覆式创新跨界竞争
快速持续的创新和交付能力
轻资产智能高效的研发平台

用户需求变化快、交付周期短

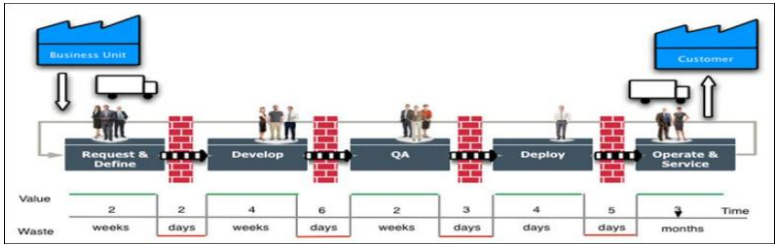
技术趋势、不断演进的客户期望、持续变化的商业环境驱动对敏捷化的需求。

从软件交付到最终客户上线时间长，效率低，开发测试环境、类生产环境、生产环境难以保证一致性。

软件交付周期越来越短，快速试错，快速反馈，一天交付多次，按需发布，随时把idea转变为软件。



从软件开发到软件交付业务流中有太多的浪费与等待，效率低下。



软件开发面临多重挑战

交付频率高，研发周期短

- 市场变化快，产品盈利窗口窄
- 按需发布，一天交付多次，快速试错，快速反馈
- 持续快速创新，快速将idea转变为产品

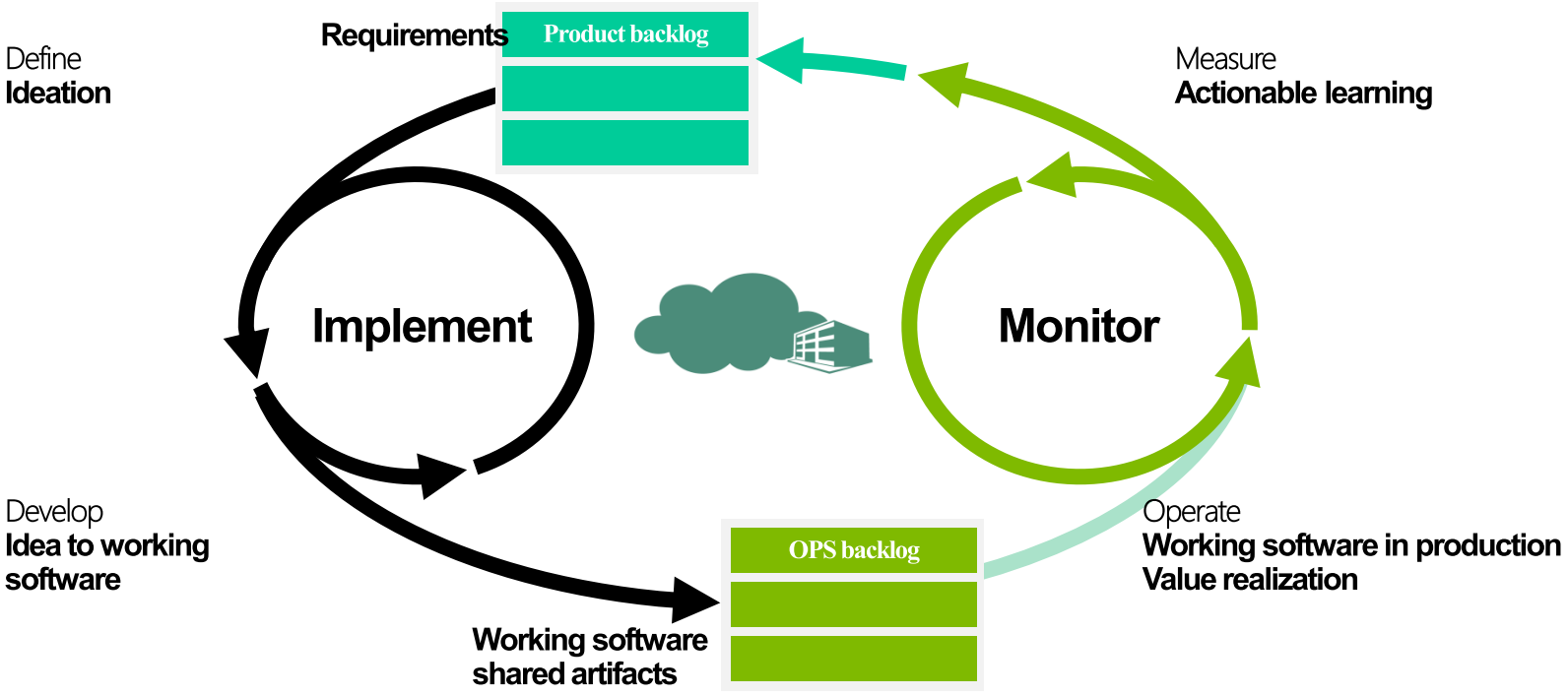
跨地域协作多，研发平台复杂

- 国际化、跨地域团队沟通协作多
- 新技术、新语言学习曲线长
- 工具部署和维护低效、复杂

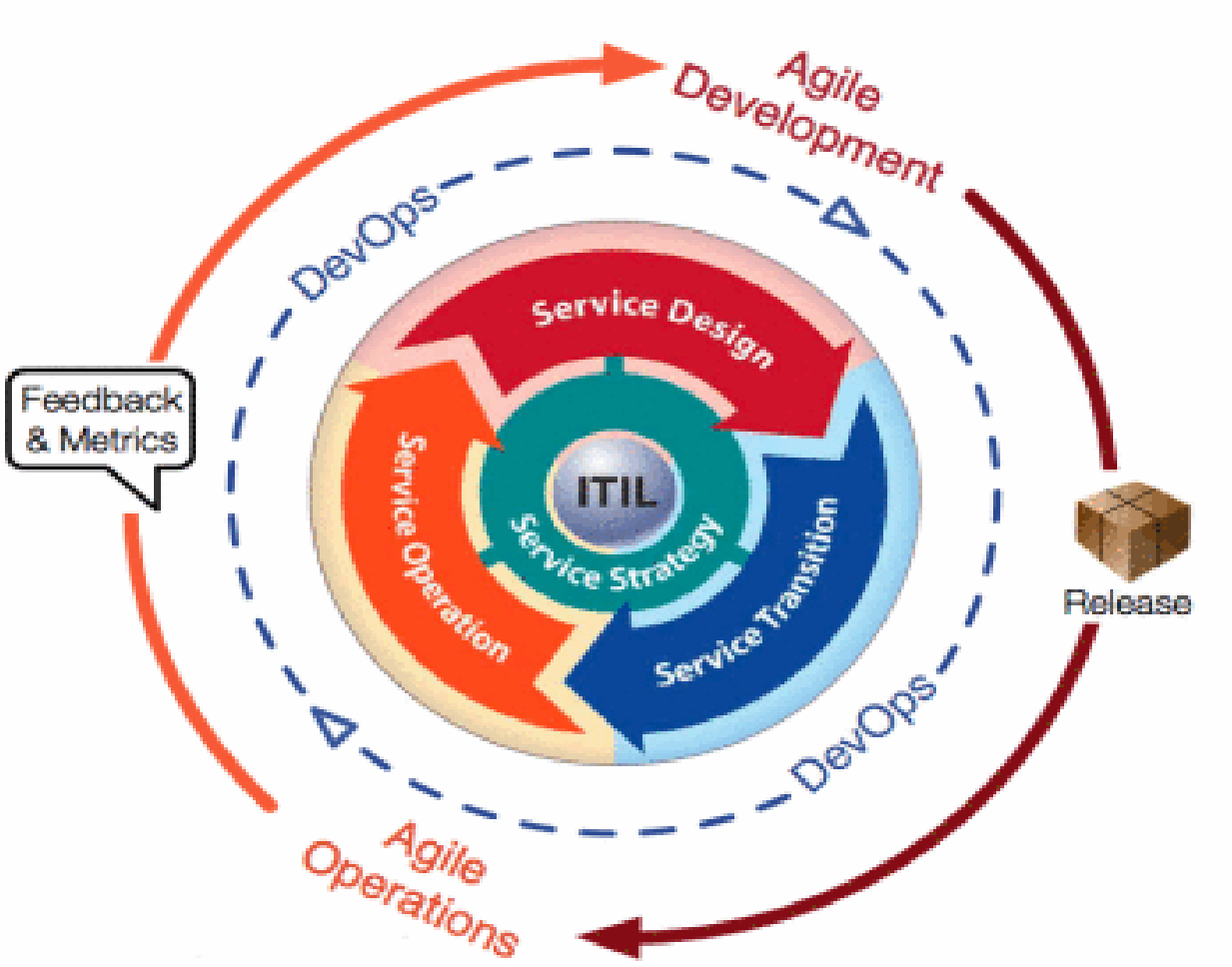
开放与安全要求高

- 平台开放，能与其它平台进行对接或数据交互
- 数据在传输与存储上要求安全并且可靠

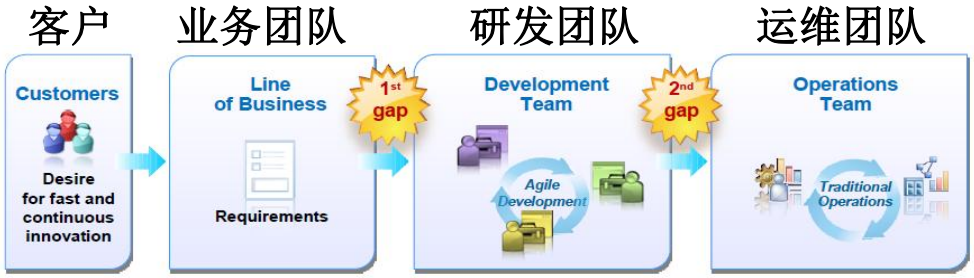
软件生命周期-1

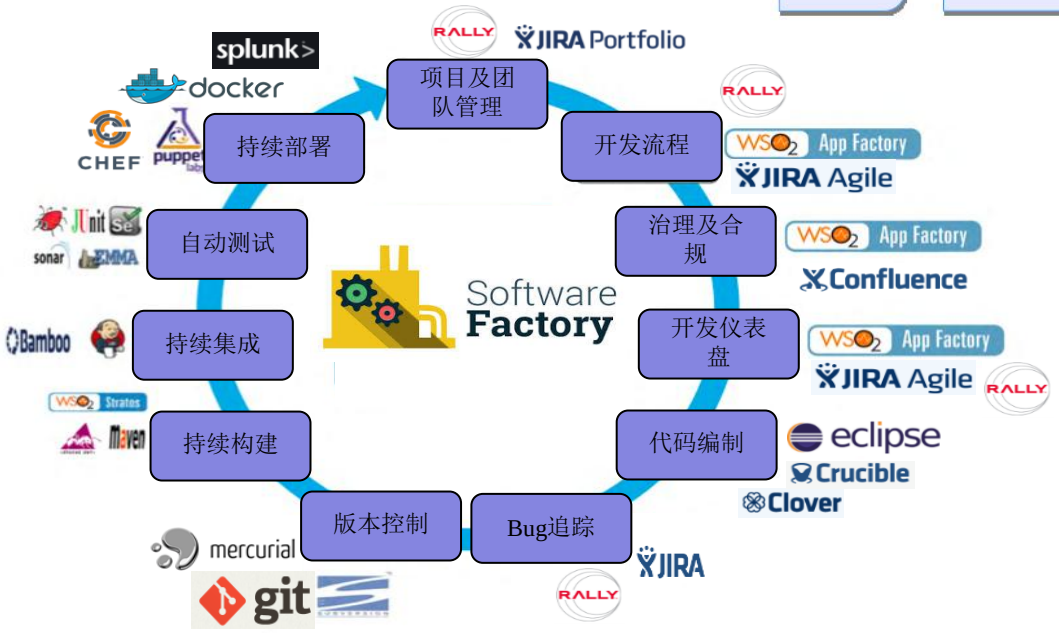

Define
Ideation
Requirements
Product backlog
Measure
Actionable learning
Implement
Monitor
Develop
Idea to working software
OPS backlog
Operate
Working software in production
Value realization
Working software
shared artifacts

软件生命周期-2


Agile
Development
DevOps
Service Design
Feedback
& Metrics
Service Operation
ITIL
Service Strategy
Service Transition
Release
DevOps
Agile
Operations

DevOps理念





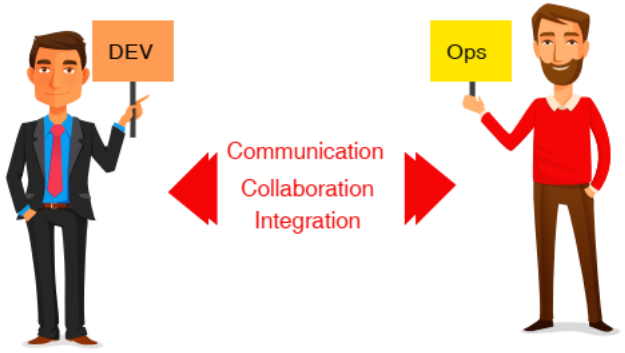
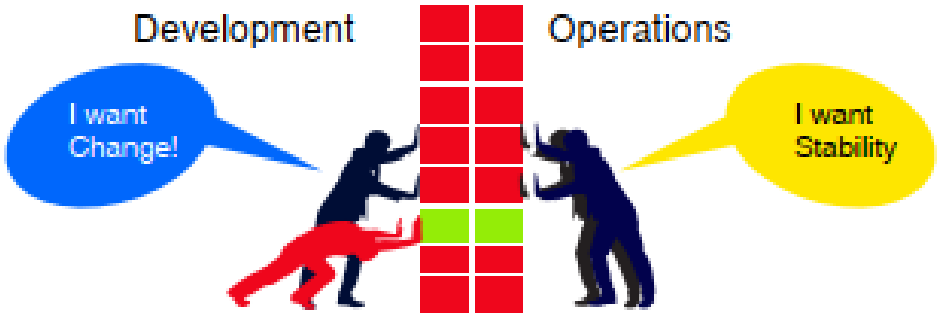


Figure 3: Open Source DevOps, Agile, and Cloud Tooling

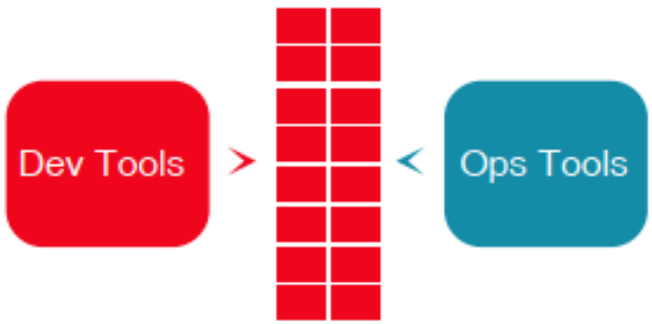
为什么真的需要DevOps？

Developers always want to deliver changes as soon as possible.

Operations want reliability and stability.



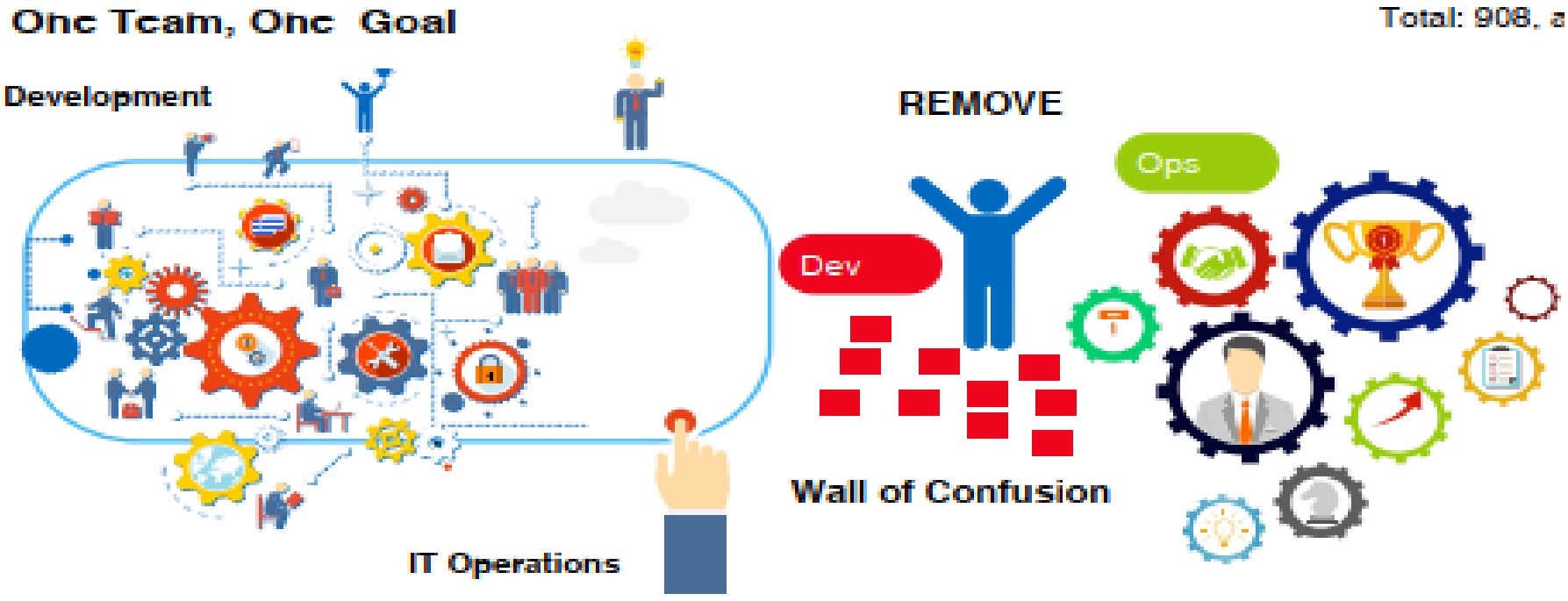
Wall of Confusion



Wall of Confusion

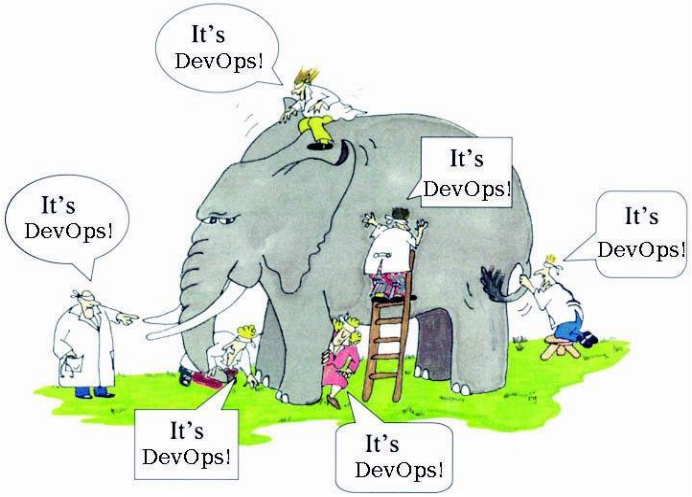
DevOps break down the walls between development and operations team, unifying development to operations for better, faster outcomes.

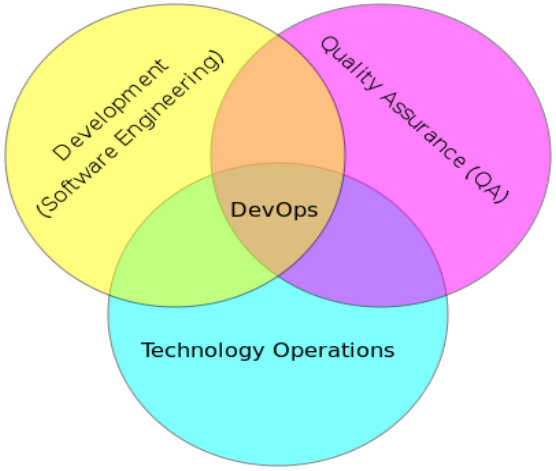
为什么真的需要DevOps？



Dev Ops Focuses both the Apps team's drive for agillity responsiveness and the NOC's concern with quality and stability on the ultimate goal of providing business value

DevOps的不同视角





研发工程师：DevOps是一组技术和工具(Jenkins/Puppet..)

咨询师：DevOps是一种新的软件开发模式

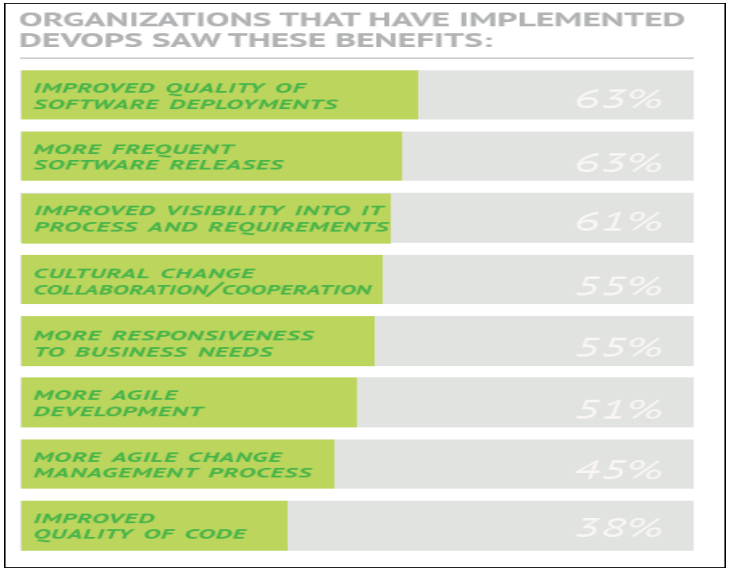
HR：DevOps是一个组织和职位

大师：DevOps是一种文化和理念

企业高管：DevOps是一种全新的商业模式和管理体系

"DevOps是软件开发、运维和质量保证三个部门之间的沟通、协作和集成所采用的流程、方法和体系的一个集合，它是人们为了及时生产软件产品或服务，以满足某个业务目标，对开发与运维之间相互依存关系的一种新的理解。——-WikiPedia

DevOps收益



【实施DevOps的收益调查（排序）】：

- 提升软件部署质量（占63%）
- 更频繁的软件发布（63%）
- 提升IT过程和需求的可视化程度（61%）
- 改变合作协同文化（55%）
- 响应更多的业务需求（55%）
- 开发更敏捷（51%）
- 管理过程更敏捷（45%）
- 提升代码质量（38%）

《State of DevOps Report》 – 2013年，90个国家4000人调查报告

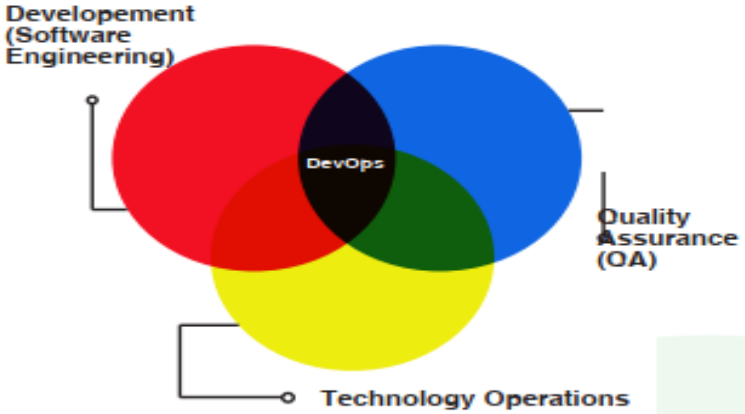
截止2014年，采用DevOps的IT组织比例为66%， IT组织的效率要高出5到7倍，变更多出14倍，变更故障率下降50%。

DevOps是如何工作的？

DevOps is a way of thinking.

Culture	• Hearts & Minds • Embrace Change
Automation	• CI/CD • "Infrastucture as Code"
Lean	• Focus on producing for the end-user • Small batch sizes
Metrics	• Measure everything • Show the Improvement
Sharing	• Open information sharing • Collaboration

CALMS Model



Five Basic Principles of DevOps :

- Eliminate the blame game, Open post-mortems, Feedback, Rewarding failures
- Continous Delivery, Monitoring, Configuration Management
- Business value for end user
- Performance Metrics, Logs, Business goals Metrics, People Integration Metrics, KPI
- Ideas, Plans, Goals, Metrics, Complications, Tools

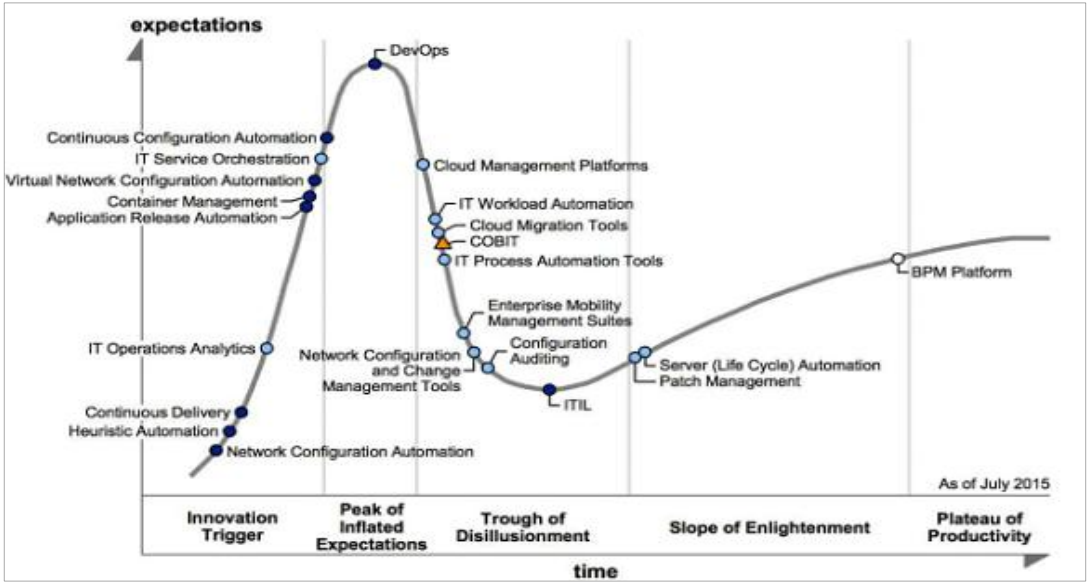
DevOps combines the best of all teams providing the following:

- Develops and verifies against production-like systems
- Reduces cost/time to deliver - Deploy often, deploy faster with repeatable, reliable process
- Increases Quality - Automated testing, Reduce cost/time to test
- Reduces Defect cycle time - Increase the ability to reproduce and fix defects
- Increases Virtualize Environments utilization
- Reduces Deployment related downtime
- Minimizes rollbacks

DevOps的五个要素

- **文化**：建立一体化的全功能团队，打破开发(Dev)与技术运营(Ops)隔阂
- **自动化**：自动化一切可以自动化的
- **精益**：以精益的方式小步快跑，持续改善
- **度量**：建立有效的监控与度量手段快速获得反馈，推动产品和团队的持续改进
- **分享**：不同职能、不同产品之间分享经验

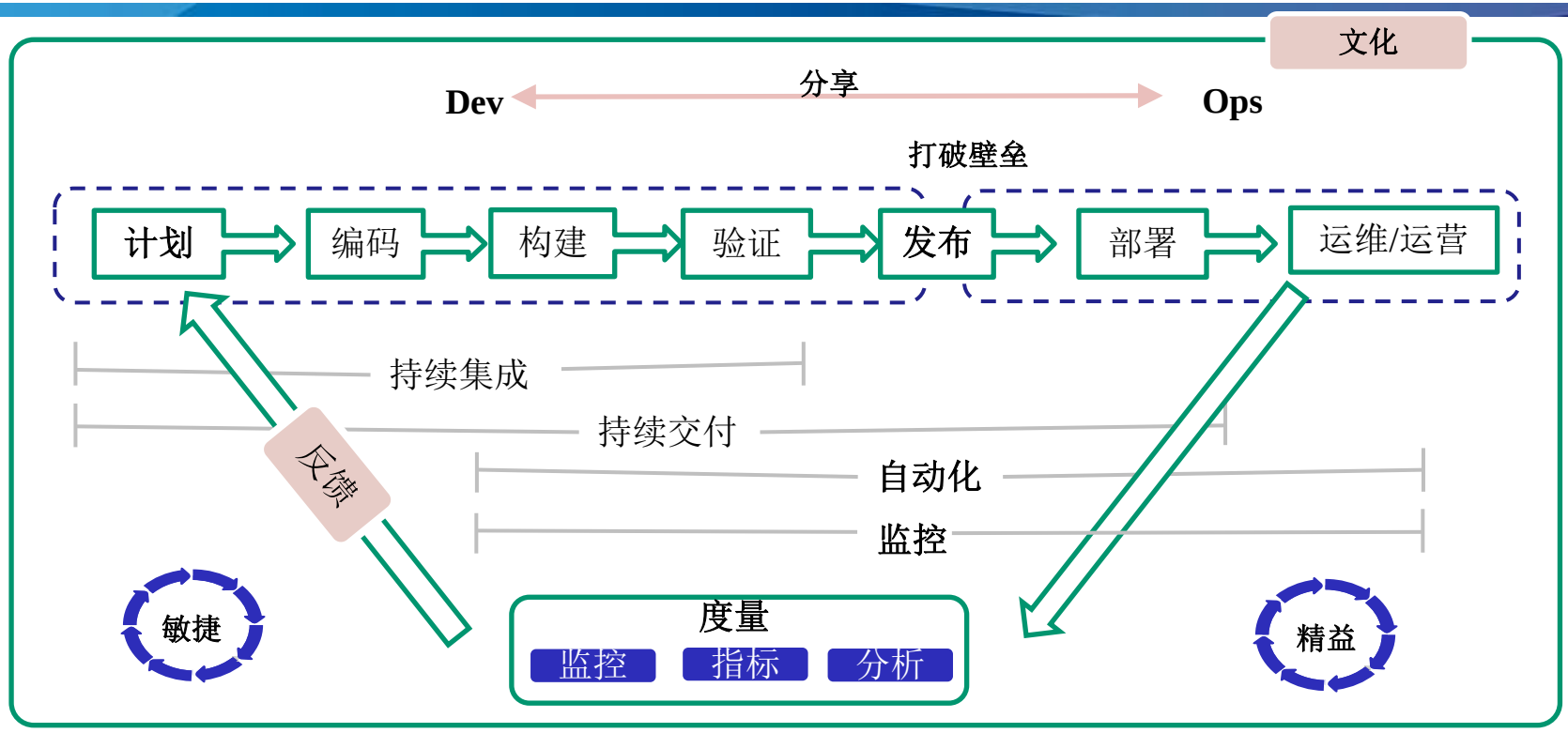
DevOps软件开发模式

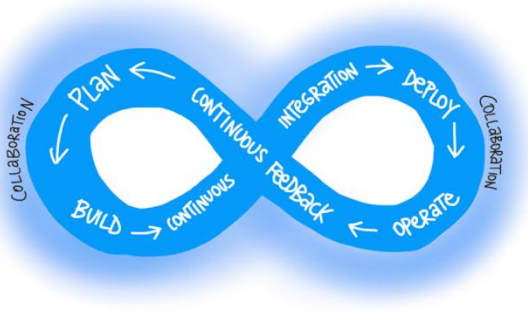


来源：**Gartner 2015 I&O Automation 报告**

- 根据Gartner 2015 I&O Automation 报告，DevOps处于技术发展的最高点，实践受到高度关注。
- 根据Puppet Labs领导的年度DevOps发展报告，企业的IT服务绩效和DevOps推崇的普遍实践有明显正相关，部署效率快30倍，变更失败率低50%。

DevOps生命周期过程

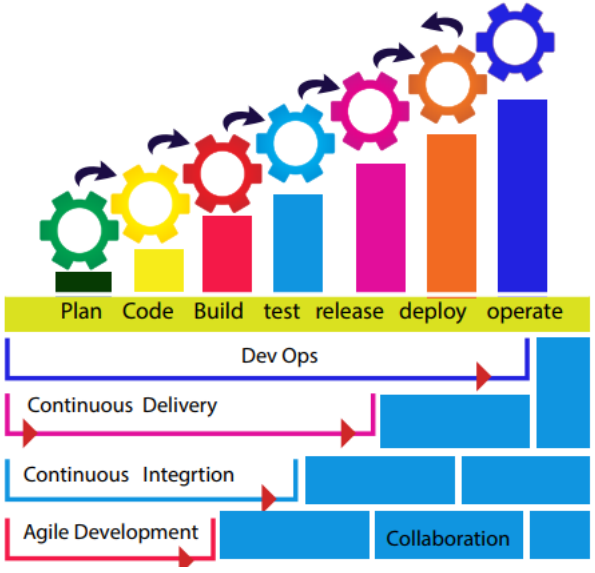




计划-构建-持续集成-部署-运行-持续反馈-计划

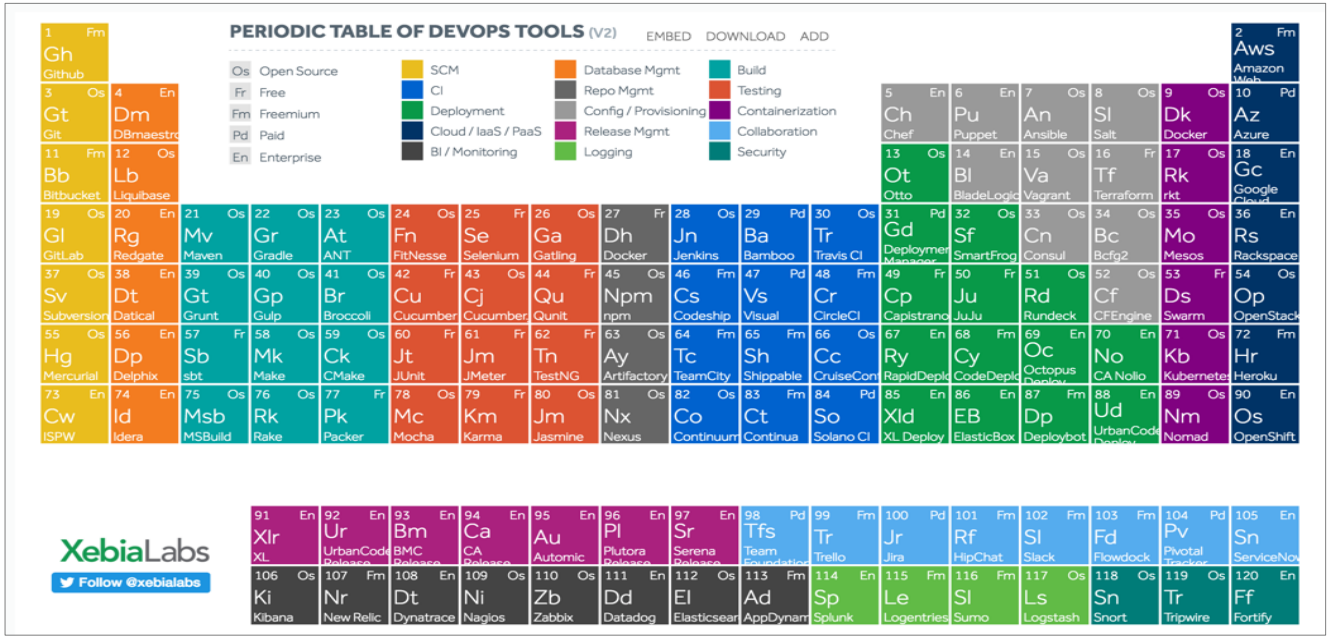
没有自动化就有DevOps

- Automate Provisioning - Infrastructure as Code
- Automate Builds – Continuous Integration
- Automate Deployments – Defined Deployment Pipeline and Continuous Deployments with appropriate configurations for the environments
- Automate Testing – Continuous Testing, Automated tests after each deployment
- Automate Monitoring – Proper monitors in place sending alerts
- Automate Metrics – Performance Metrics, Logs



DevOps生命周期过程相关工具

- Project项目管理
- SCM源代码配置管理
- CI持续集成
- Deployment部署
- Repo Mgmt组件库管理
- Build构建
- Testing测试
- Monitoring监控
- Release发布
- Collaboration协作
- Cloud云计算
- ……

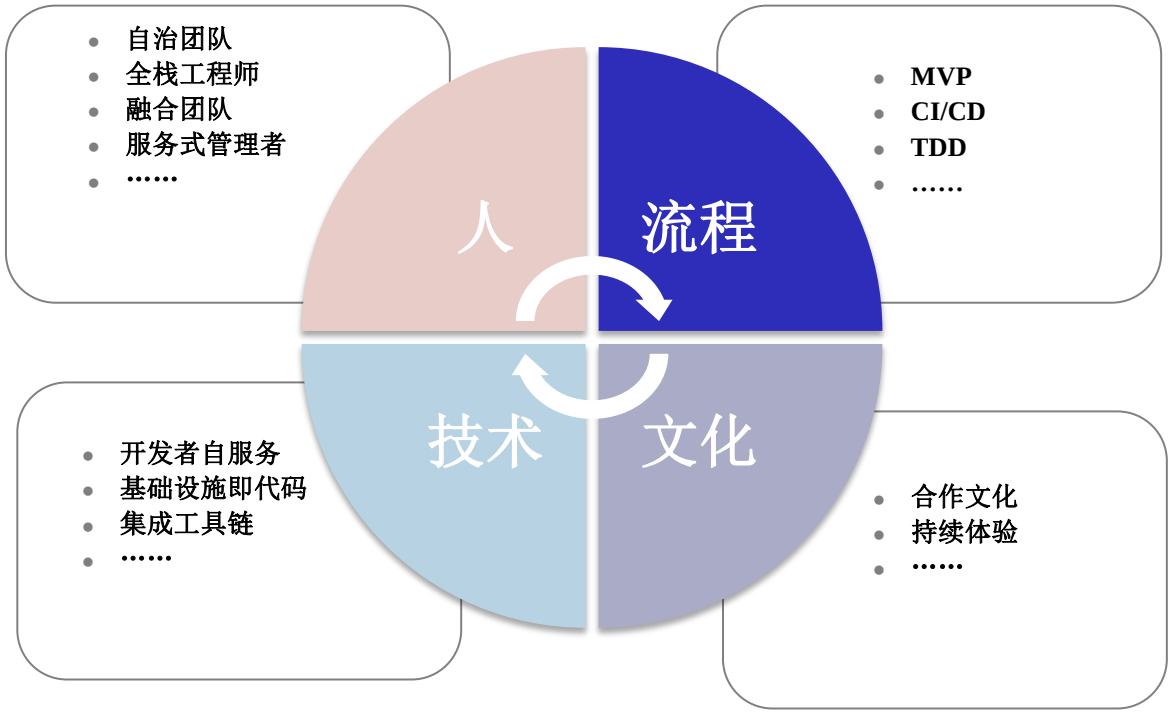


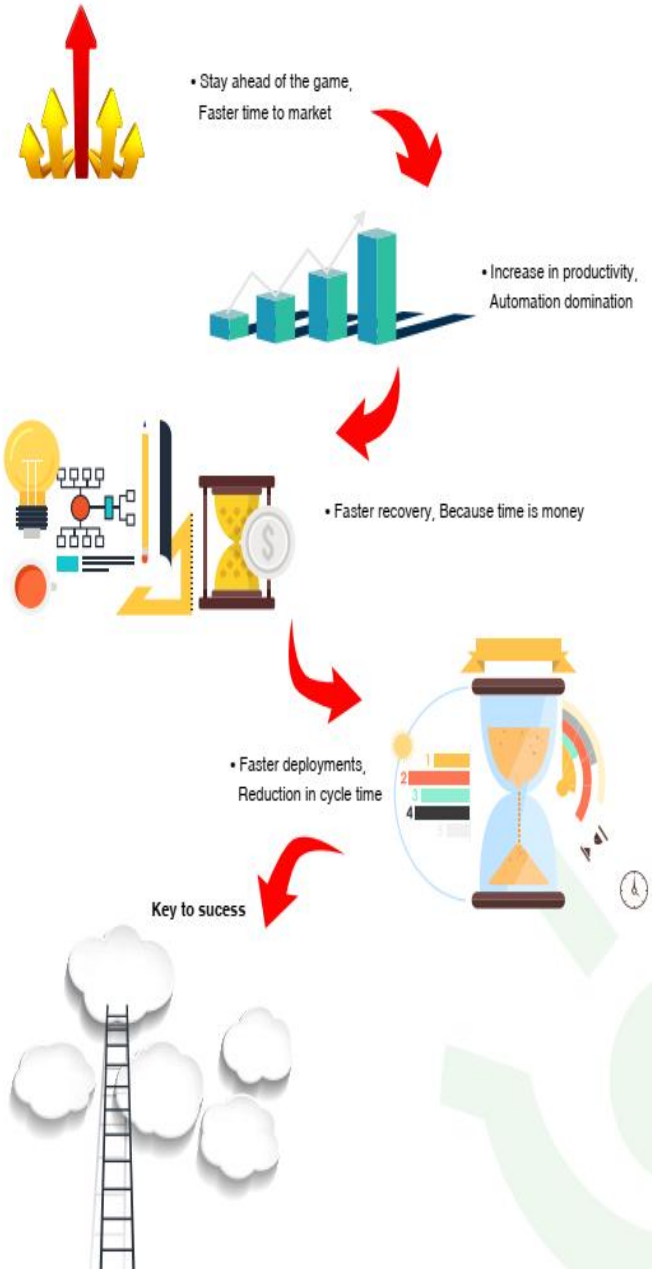
来源：XebiaLabs，15类，120种工具https://xebialabs.com/periodic-table-of-devops-tools

DevOps Tool Chain

- Code — code development and review, source code management tools, code merging
- Build — continuous integration tools, build status
- Test — continuous testing tools that provide feedback on business risks
- Package — artifact repository, application pre-deployment staging
- Release — change management, release approvals, release automation
- Configure — infrastructure configuration and management, Infrastructure as Code tools
- Monitor — applications performance monitoring, end – user experience

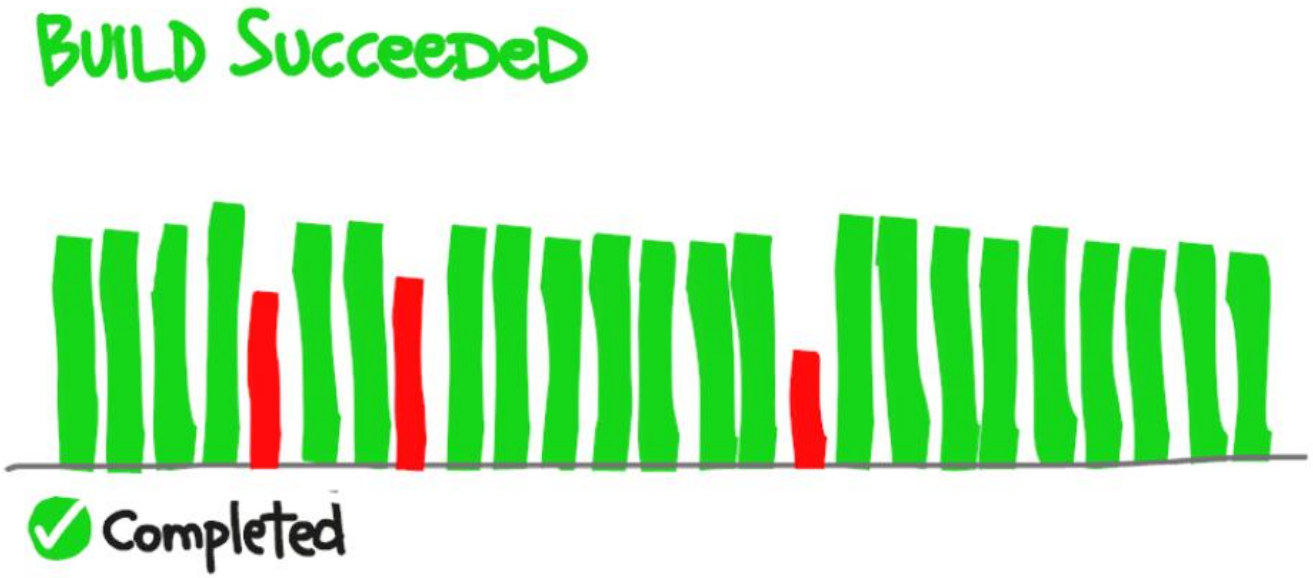
如何实施DevOps过程？





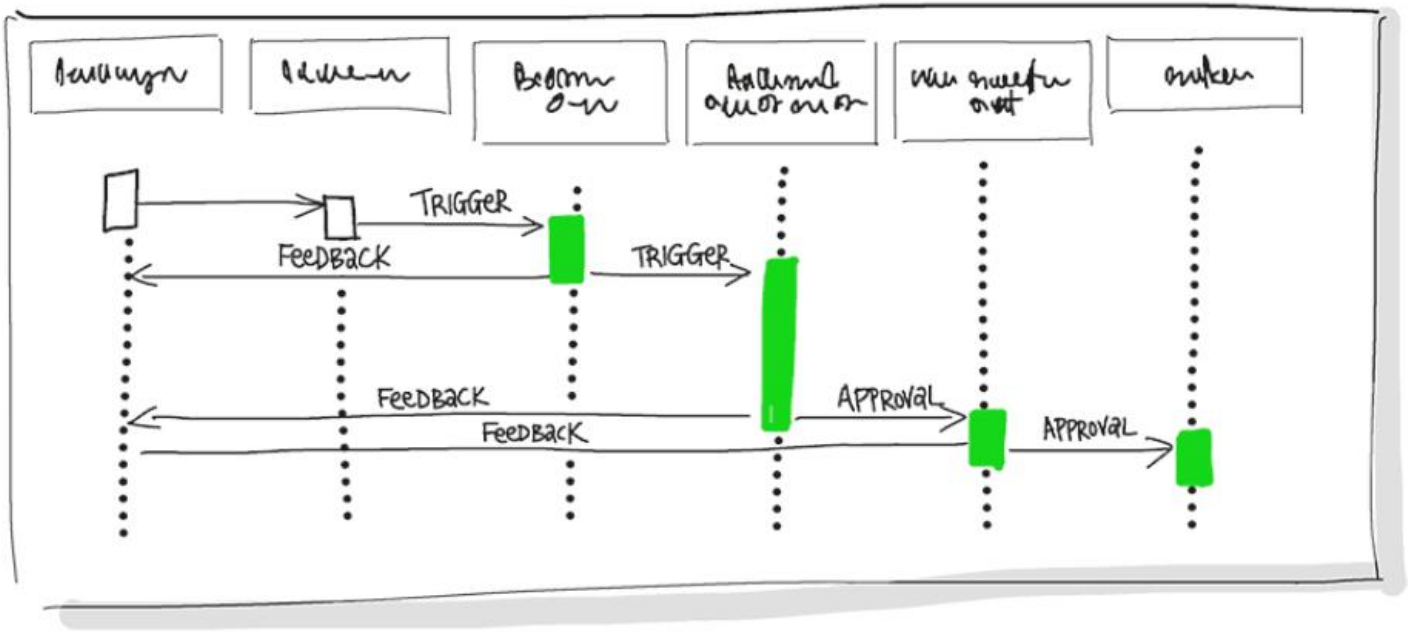
1. 持续集成

- 推动了代码的持续合并和测试，早期发现缺陷，减少合并问题的时间浪费，为开发团队的快速反馈。



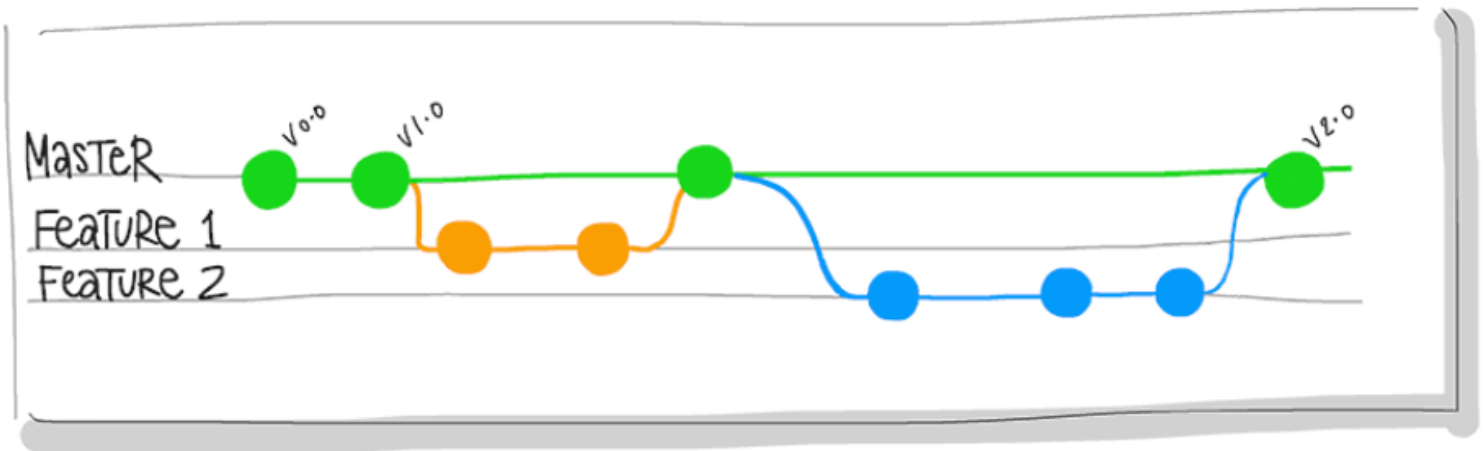
2. 持续发布

- 持续向生产和测试环境提供软件解决方案，可帮助组织快速修复错误，并响应不断变化的业务需求。



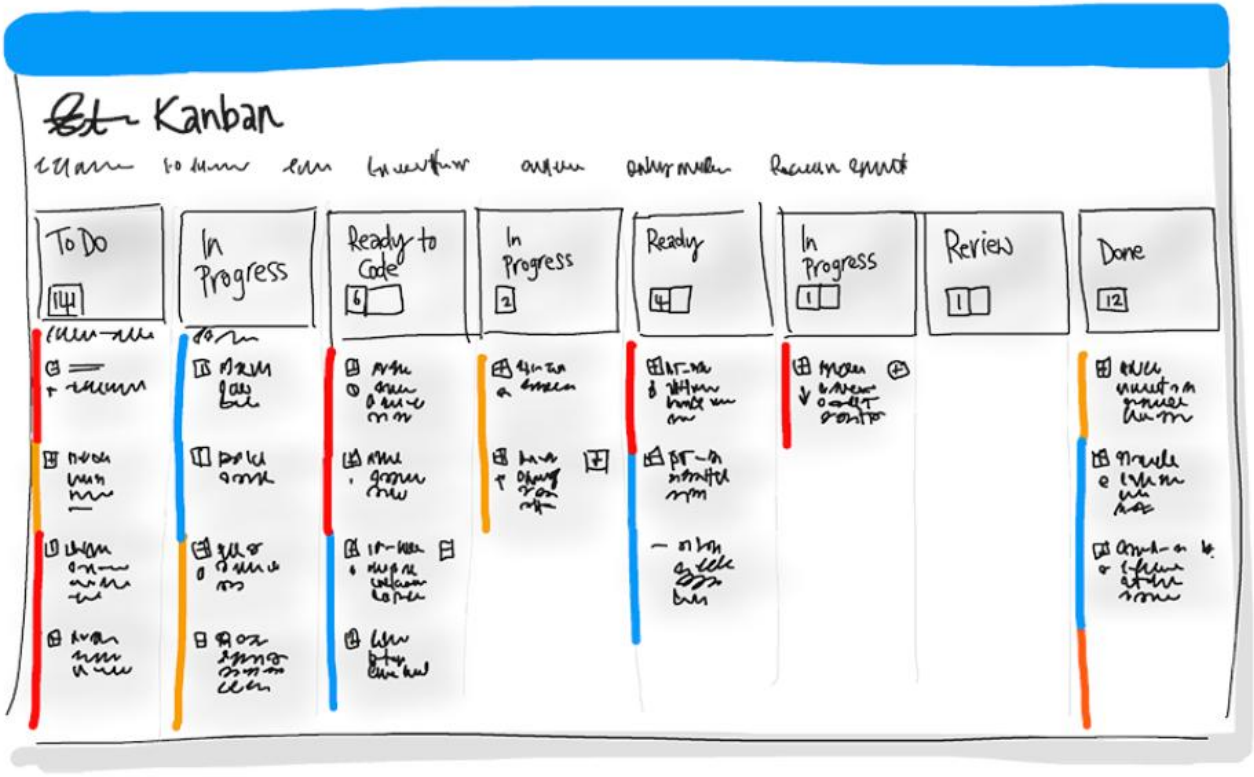
3. 版本控制

- 版本控制，通常使用Git，使位于世界任何地方的团队能够在日常开发活动中有效地进行通信，并与用于监视、部署等活动的软件开发工具集成。



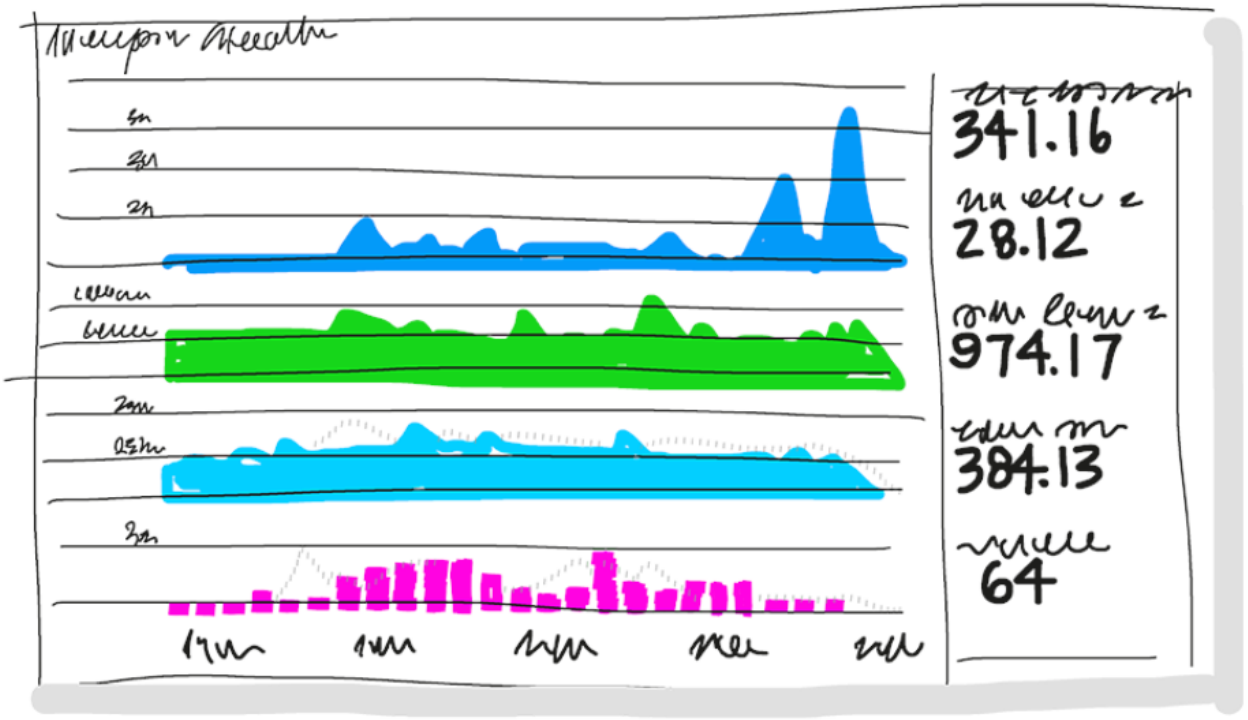
4. 敏捷开发和精益项目管理

- 敏捷规划和精益项目管理技术用于规划，并将工作划分为冲刺、管理团队能力，并帮助团队快速适应不断变化的业务需求。



5. 监控和日志

- 监控和记录正在运行的应用程序，包括应用程序运行和客户使用状况，可帮助组织形成假设，并快速验证或改变策略。捕获丰富数据，并以各种日志记录格式存储。

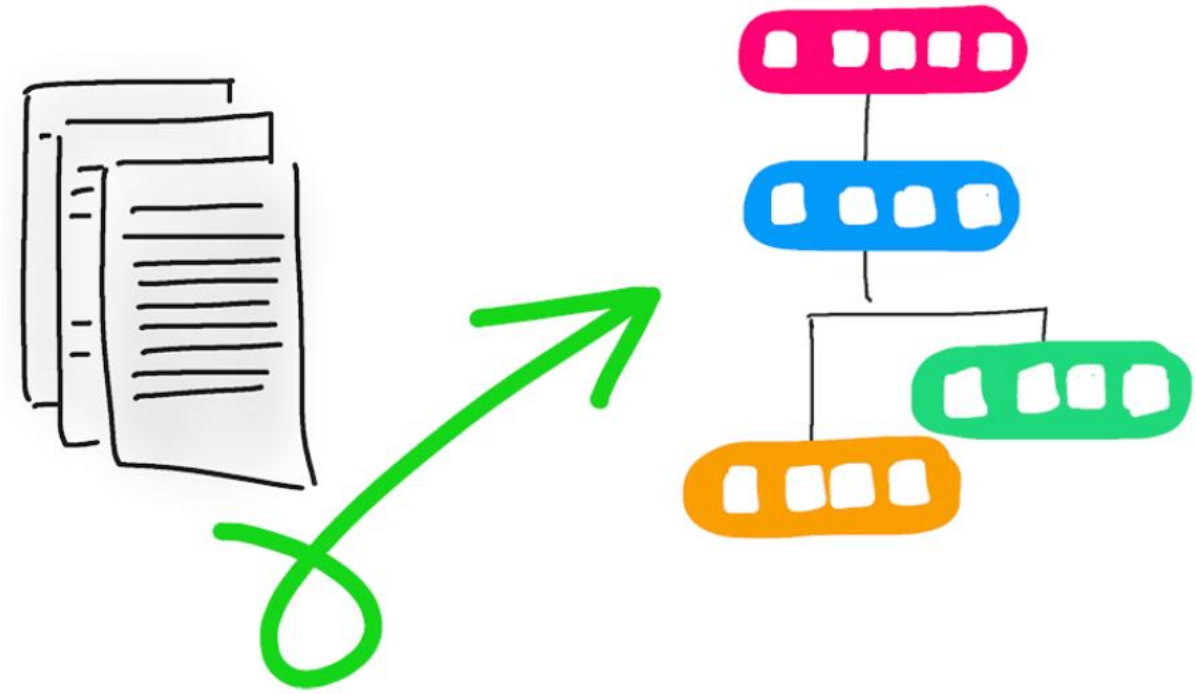


6. 采用公有云和混合云

- 公共云和混合云使不可能变得容易。云已经消除了传统的瓶颈，并帮助将基础设施商品化。无论您使用基础架构即服务（IaaS）来提升和转移现有应用程序，还是使用平台即服务（PaaS）来获得前所未有的生产力，云都可以为您提供无限制的数据中心。

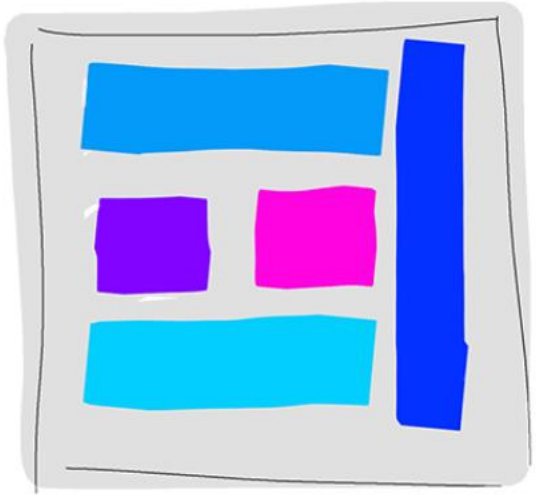
7.作为代码的基础设施

- 基础设施即是代码 （IaC）是一种实践，能自动化、确认环境的创建和拆除，以帮助发布安全和稳定的应用程序托管平台。

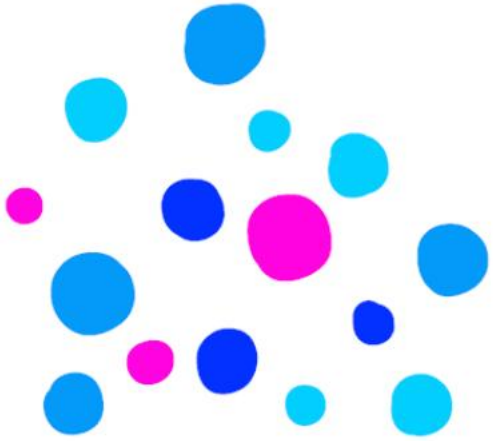


8. 微服务

- 微服务架构，将业务用例划分成小型可重用的服务，通过接口的合同进行通信，该架构实现了可扩展性和效率。



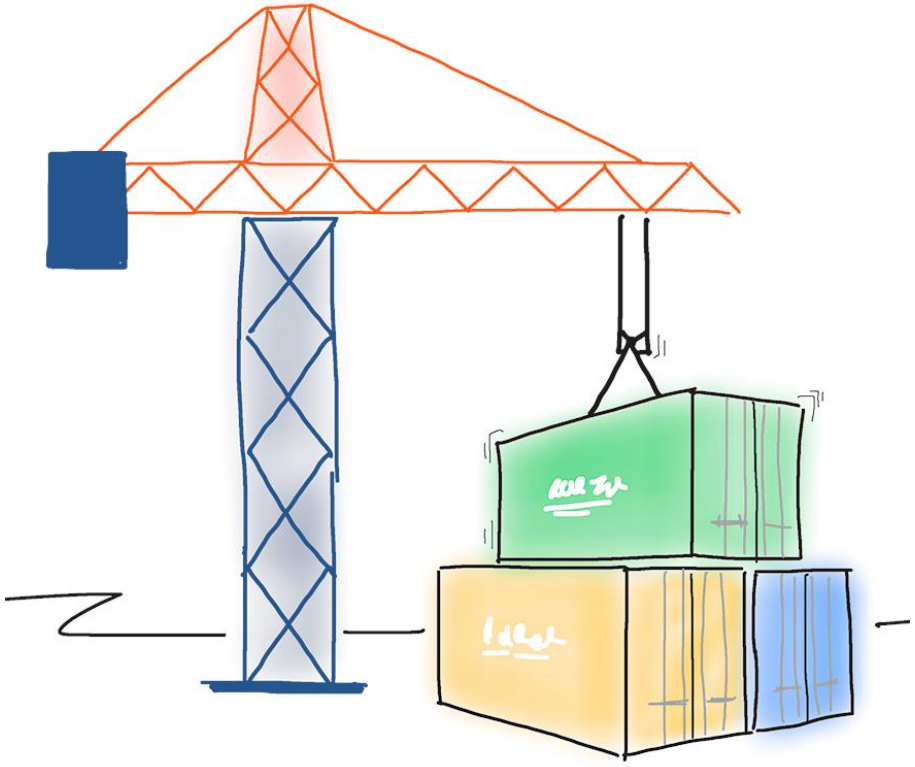
MONOLITHIC/LAYERED



MICROSERVICES

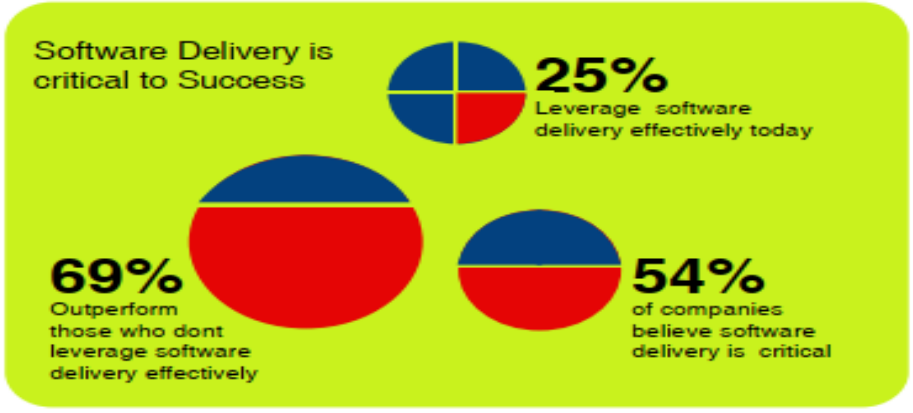
9. 使用容器

- 容器是虚拟化的下一个发展，它们比虚拟机更轻量级，可以更快地发挥作用，并且可以通过文件轻松配置。



DevOps的挑战和问题

It doesn't matter whether you are in Cloud, Enterprise or Mobile. For each one of you, **stable software delivery on time** is the key to your business success.



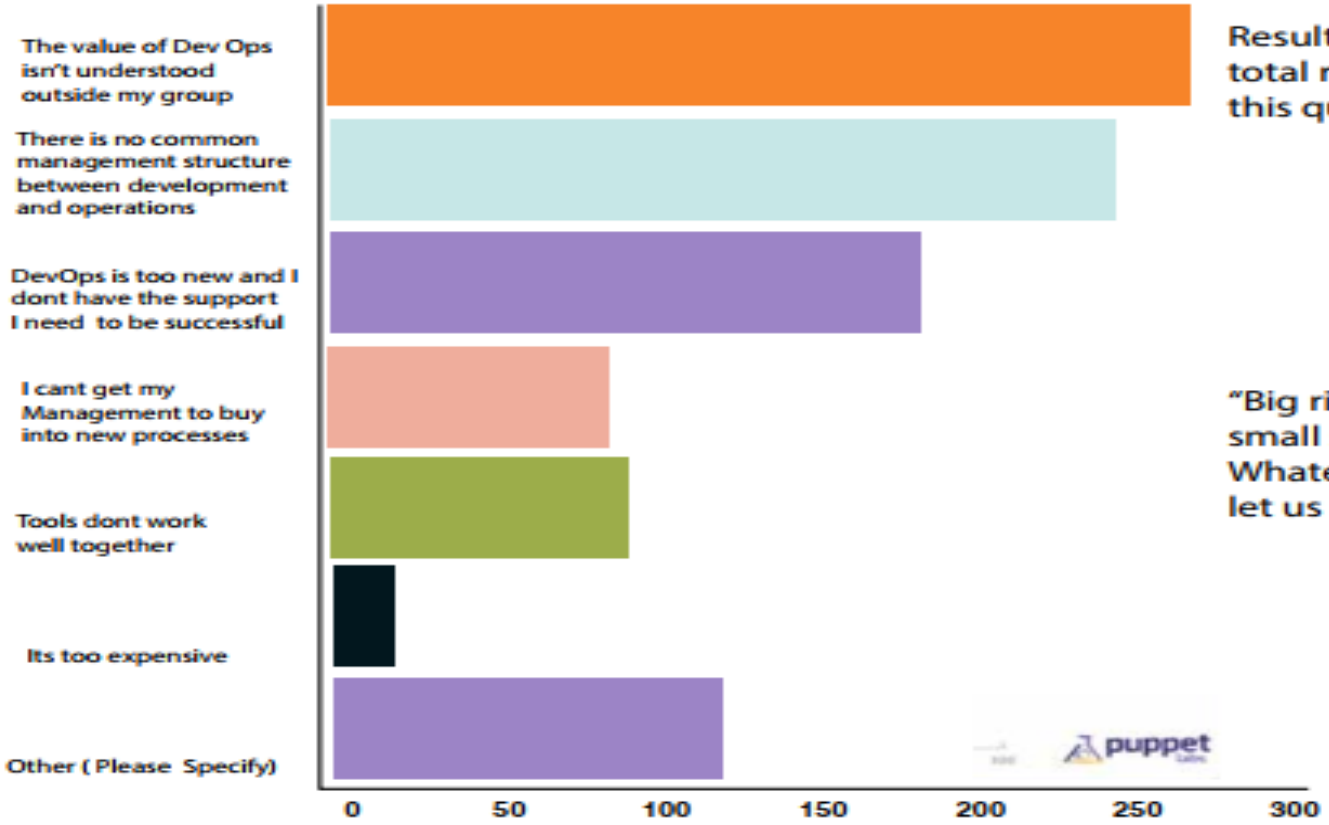
Some of the serious issues blocking your software delivery are:

- Building and maintaining servers – Time consuming and unproductive
- No environment management - Differences in development and production environments
- Slow deployments – Costly error prone manual process and efforts
- No shared ownership - Lack of feedback and proper metric leads
- No proper configuration management - Discrepancies in managing configurations
- Deployments are a blocker - Upgrade risk due to manual management of multiple application configuration and versions - Dependency on specific deployment engineer
- Production downtime - Due to lack of improper deployment instructions / checklist
- Hacking – Fixing directly in production (instead of a proper hotfix process) and forgets to check-in into source control

Realize your entire business gets impacted if you do not have Continuous Delivery. To enable that, you must adopt DevOps

实施DevOps的障碍

What are the biggest obstacles in implementing DevOps in your organiztion?
(Select all that apply)



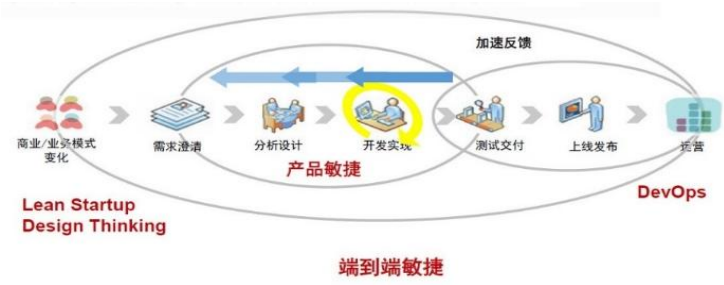
Results out of 579 total responses to this question

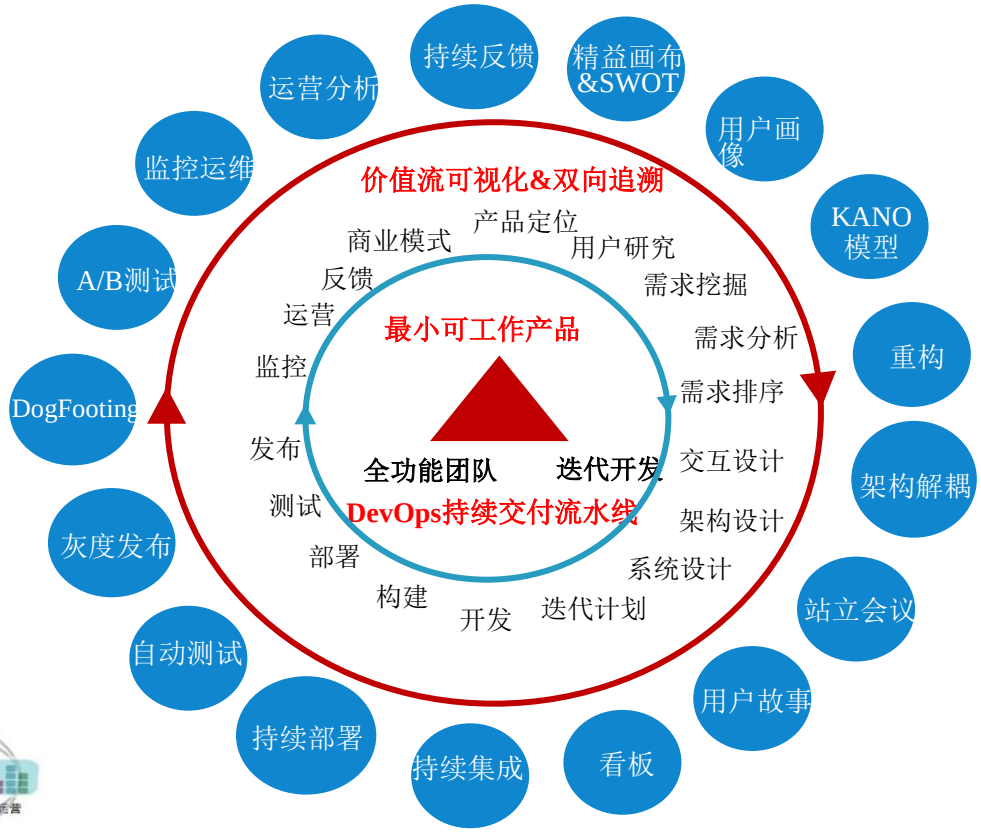
"Big rivers often begin as small streams"
Whatever your barriers are, let us help you to overcome it.

华为敏捷开发过程：端到端

- 华为敏捷项目管理，融合了敏捷、精益、DevOps理念，**不只是开发阶段的敏捷**，而是从市场，到开发、运维、运营的端到端敏捷。

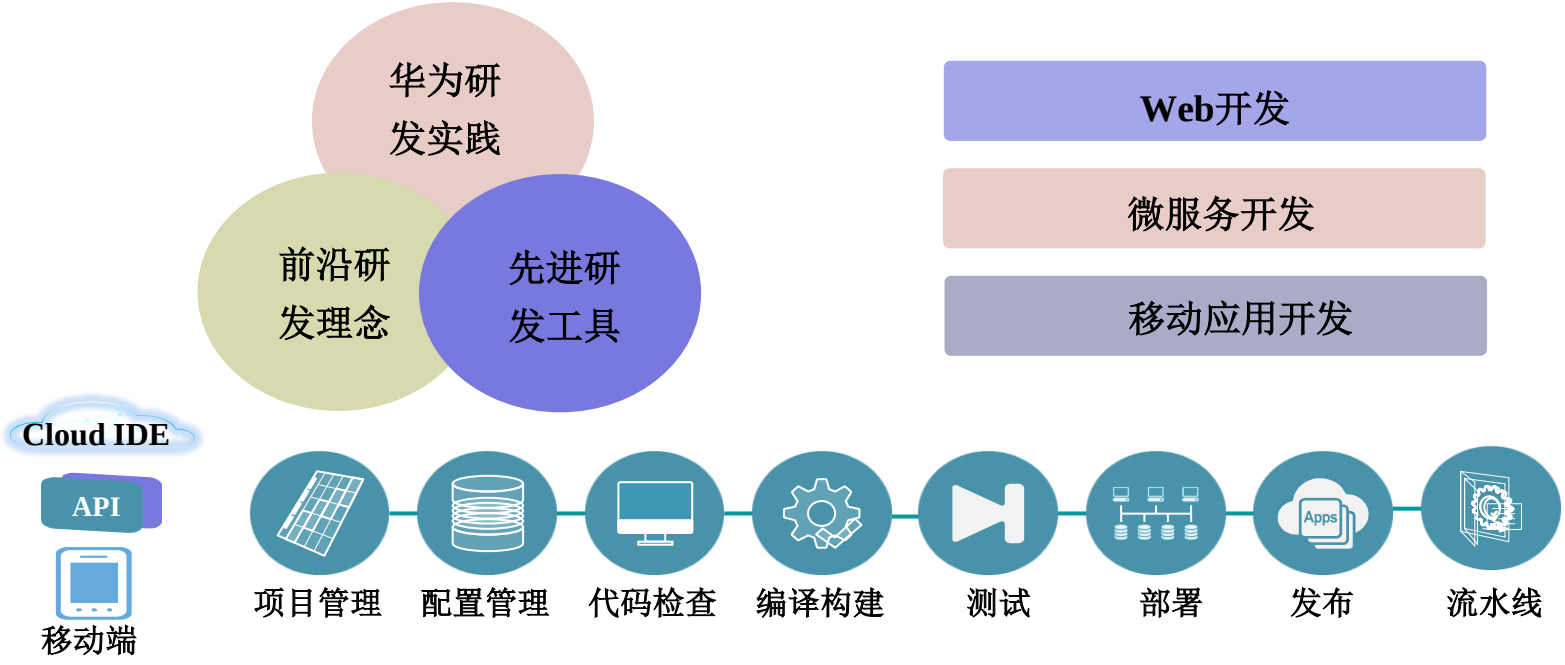
- 华为1217敏捷项目管理模型包含1个铁三角、2个端到端交付环、17个实践。





华为软件开发云：一站式云上DevOps平台

软件开发云（DevCloud ）是集华为研发实践、前沿研发理念、先进研发工具为一体的研发云平台；面向开发者提供研发工具服务，让软件开发简单高效。



全自动化的DevOps持续交付

敏捷团队可以将精力聚焦于业务的分析、设计、开发，后续的构建、测试、部署、发布、运维均由自动化工具实现持续交付与反馈。