

朝夕教育

DevOps 分享

讲师：王震寰

讲师简介

讲师：王震寰

多年开发及项目管理经验。曾就职于远洋地产、银铃候鸟养老基金会（中国老龄事业发展基金会下属）、中海友成等公司，主要负责研发团队管理。现为朝夕教育项目管理方向讲师。

拥有PMP、PMI-ACP、PRINCE2 Practitioner、CSM (Certification Scrum Master)、CSPO (Certification Product Owner)、A-CSM (Advanced - Certification Scrum Master)、A-CSPO (Advanced - Certification Product Owner)、CSP-SM (Certification Scrum Professional — Scrum Master)、CAL1 (Certification Agile Leadership 1)、SAFe-SA (Scaled Agile Framework - Agilist)、Scrum @ Scale Practitioner 以及 EXIN DevOps Master 认证。

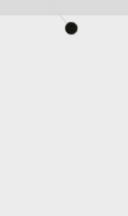
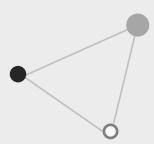
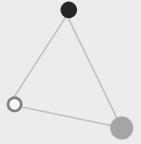


Link In

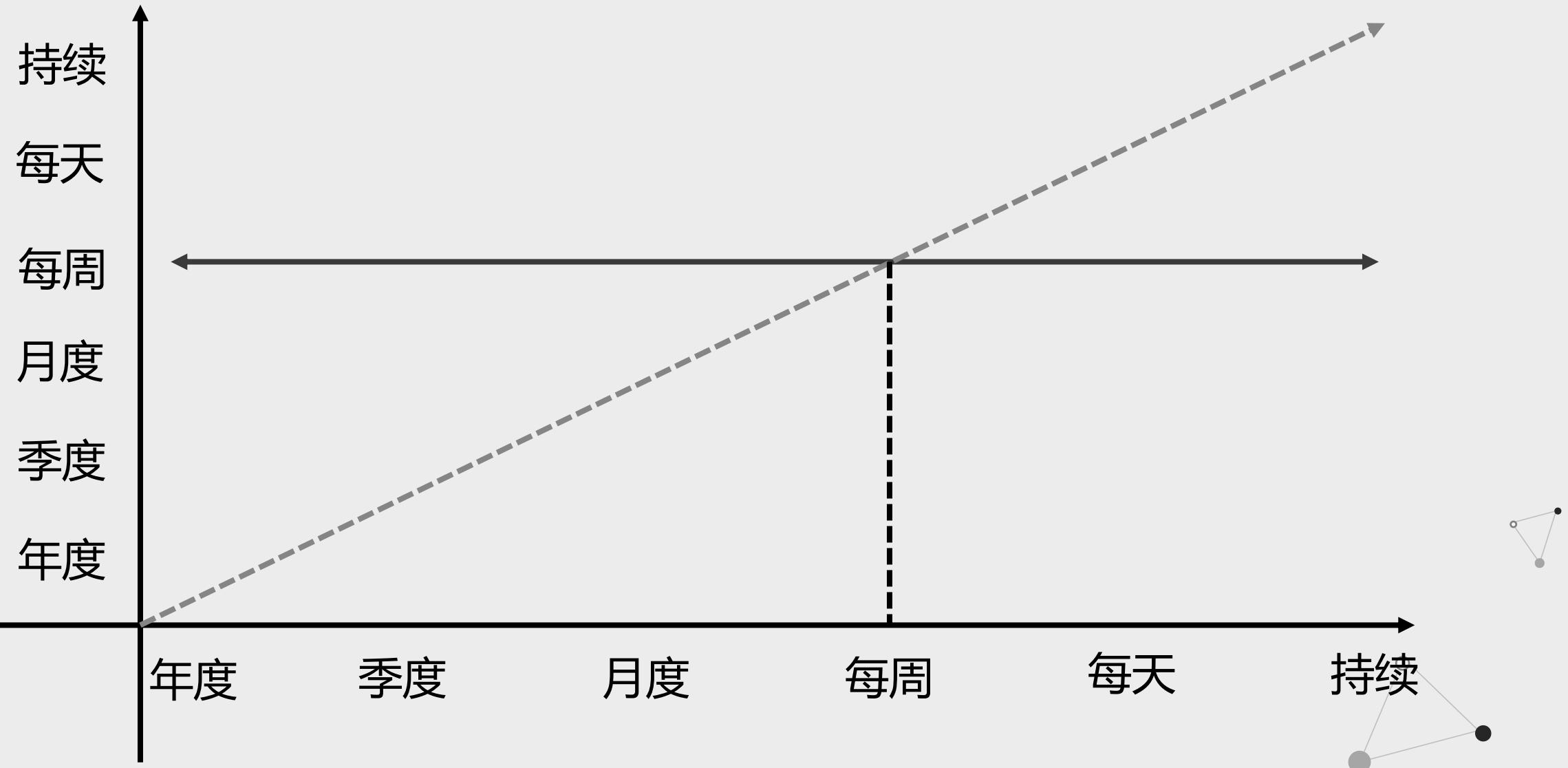


00

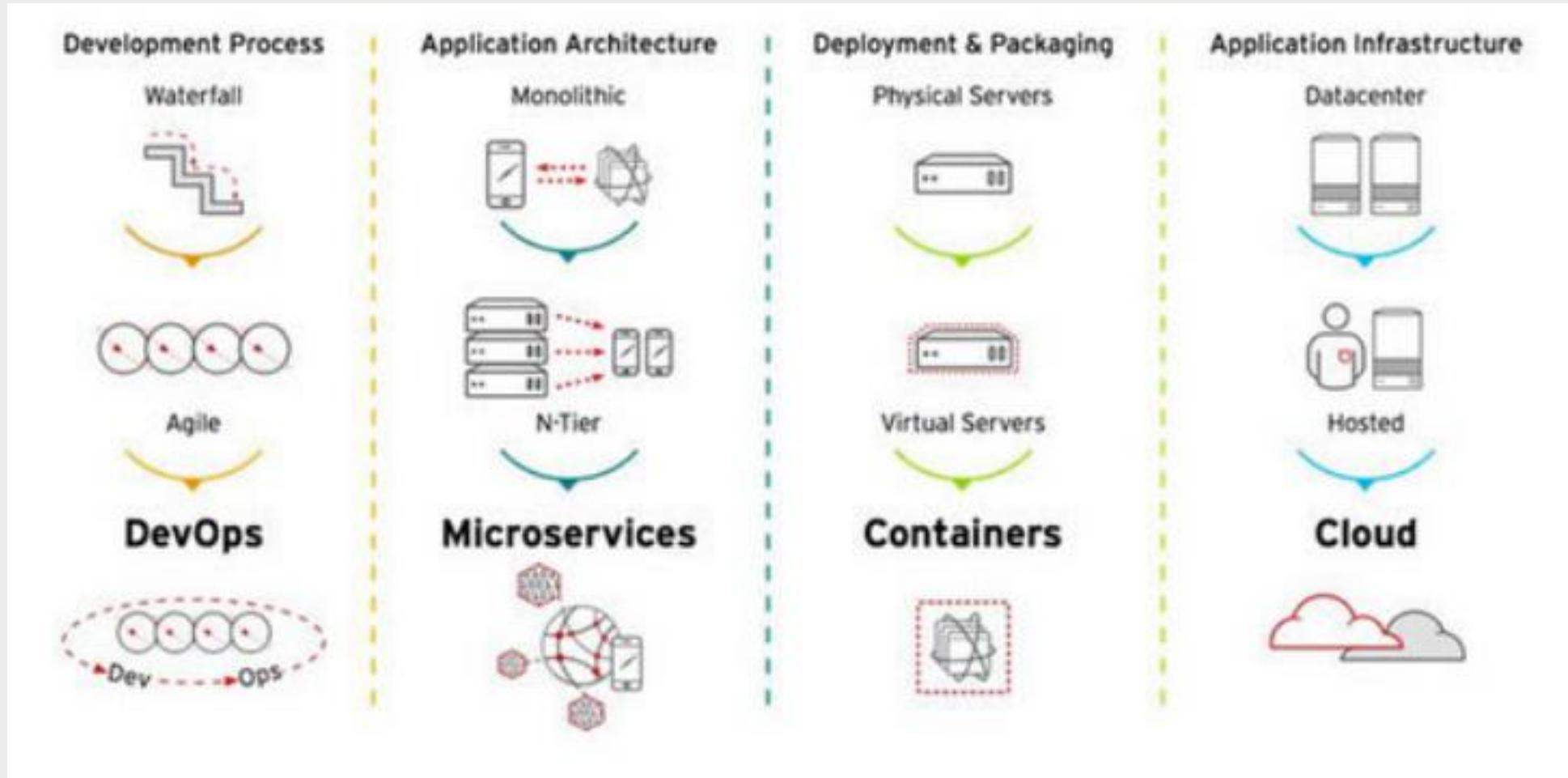
开篇的话



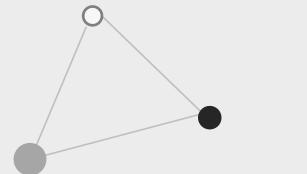
VUCA (volatility、uncertainty、complexity、ambiguity) 的时代



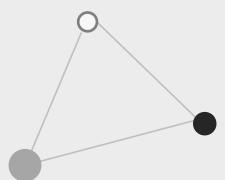
IT的技术革命



图片来自红帽副总裁 Ashesh Badani 发布会。



我们为什么要学DevOps ?



数字化时代的到来



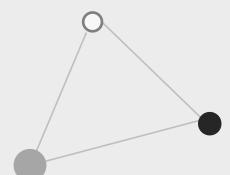
Owns no Taxis



Owns no Inventory

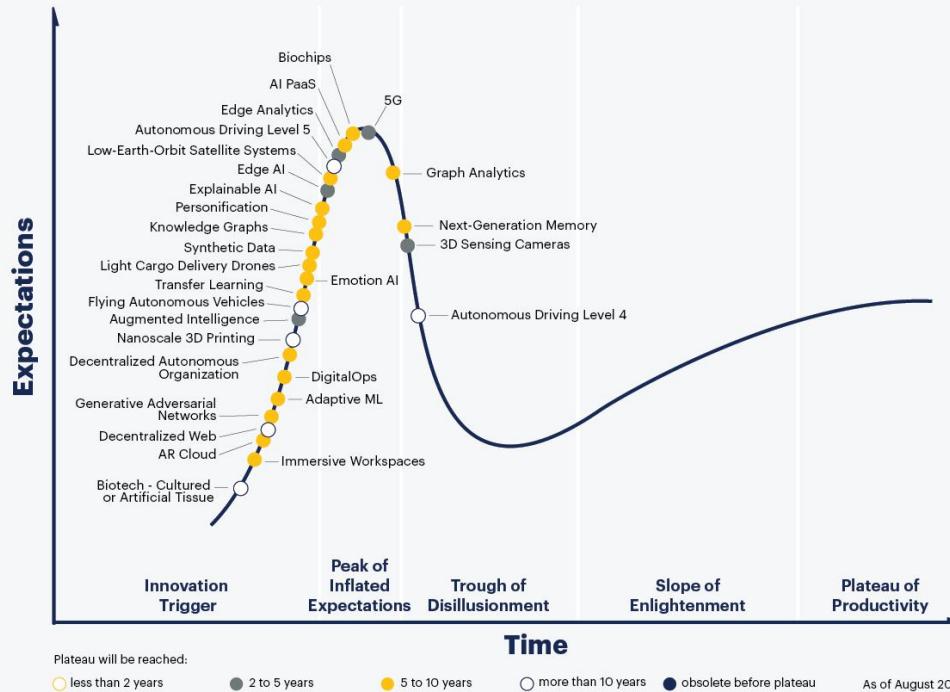


Owns no Real estate



加德纳技术成熟曲线 (10 of the Most Hyped Technologies of 2019)

Gartner Hype Cycle for Emerging Technologies, 2019

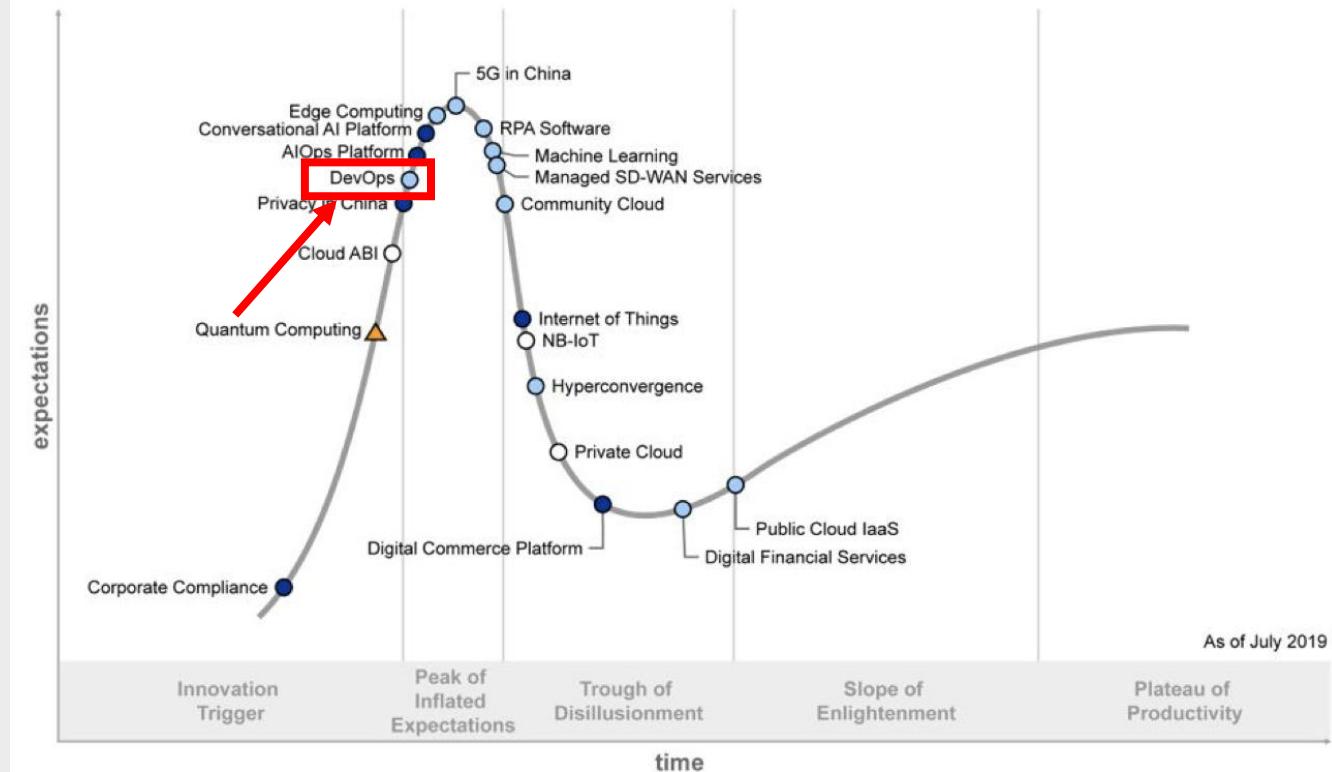


gartner.com/SmarterWithGartner

Source: Gartner
© 2019 Gartner, Inc. and/or its affiliates. All rights reserved.

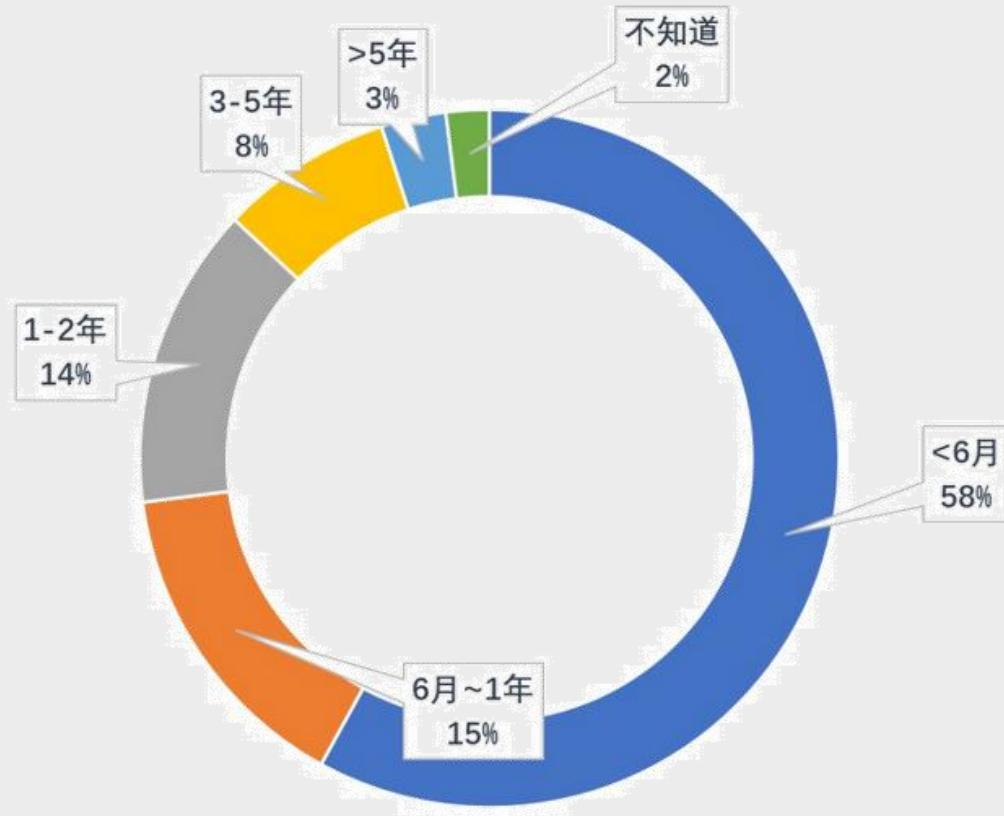
Gartner

Hype Cycle for ICT in China, 2019

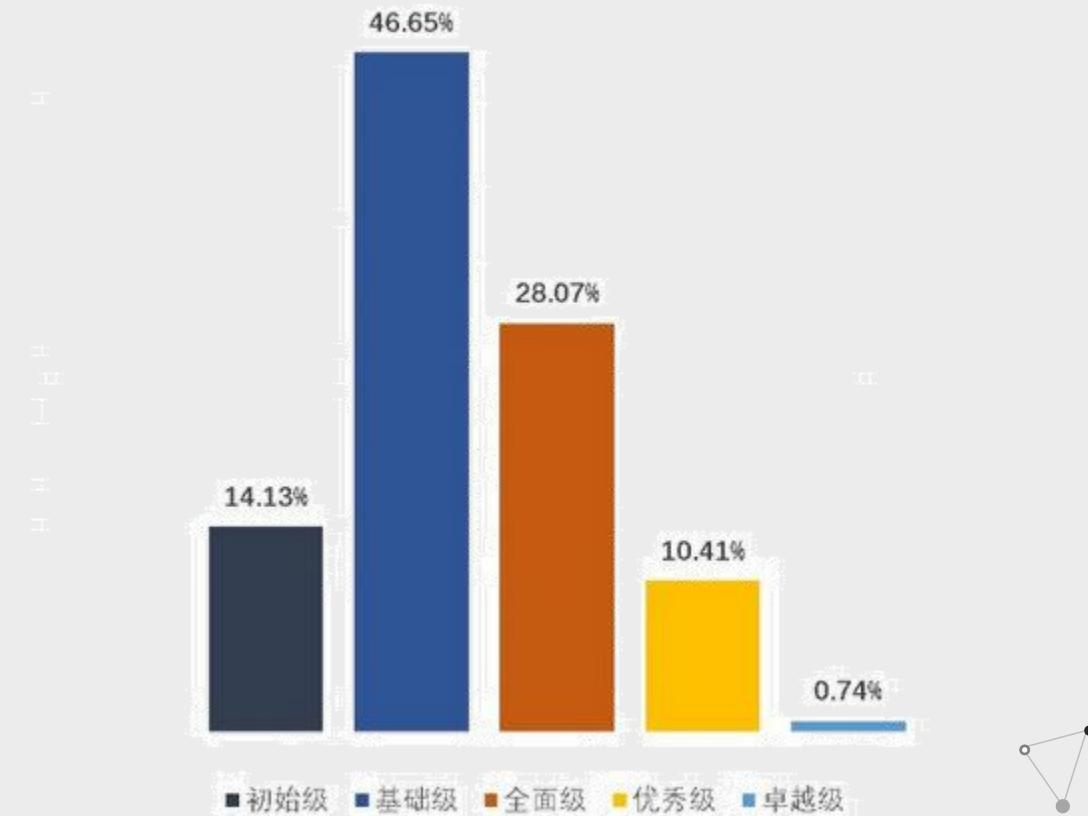


Source: Gartner
ID: 368738

DevOps的现状

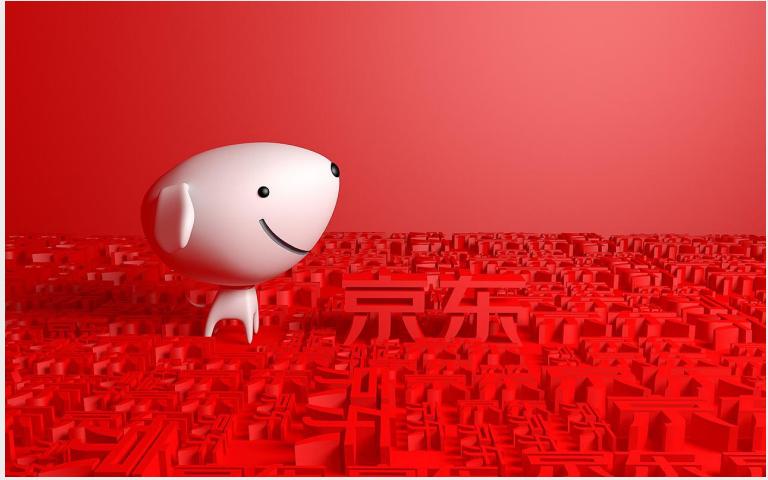
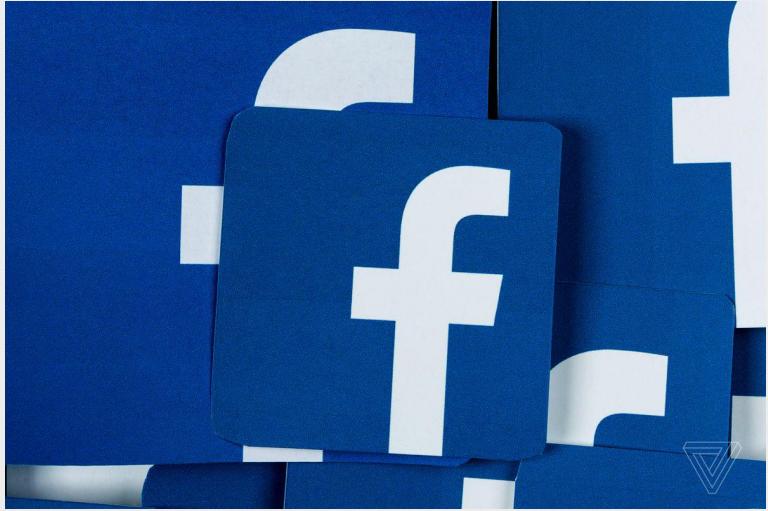


超7成被调研者DevOps经验少于1年。58%被调研者DevOps经验小于半年；15%的被调研者DevOps经验不到1年；25%被调研者DevOps经验大于1年。

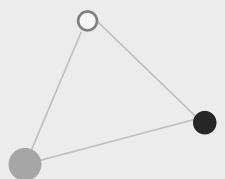


DevOps已经在国内逐步落地实践。有14.13%的企业开始在组织内局部范围尝试DevOps；46.65%的企业在组织内较大范围推广DevOps实践并且初见成效；28.07%的企业已经在组织内全面推行DevOps实践，并将其贯穿于软件开发的全生命周期中，整体交付效率得到显著提升。

哪些公司在做DevOps ?



我们对DevOps的理解



DevOps 道法术器

价值观，对目标
价值的定位

实现价值观的
战略、方法

战术、技术、
具体的手段

用工具提高效率
复杂问题简单化

道 法 术 器

VALUE

快速交付价值，灵活响应变化

WHY

全局打通敏捷开发 & 高效运维

HOW

系统应用指导原则、最佳实践

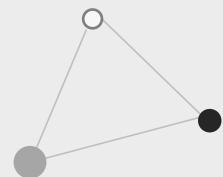
WHAT

端到端工具链相互通联和整合

系统思考的层次

解决问题的层次

快速交付价值和灵活响应变化



全局打通敏捷开发和高效运维

流程 (Process)

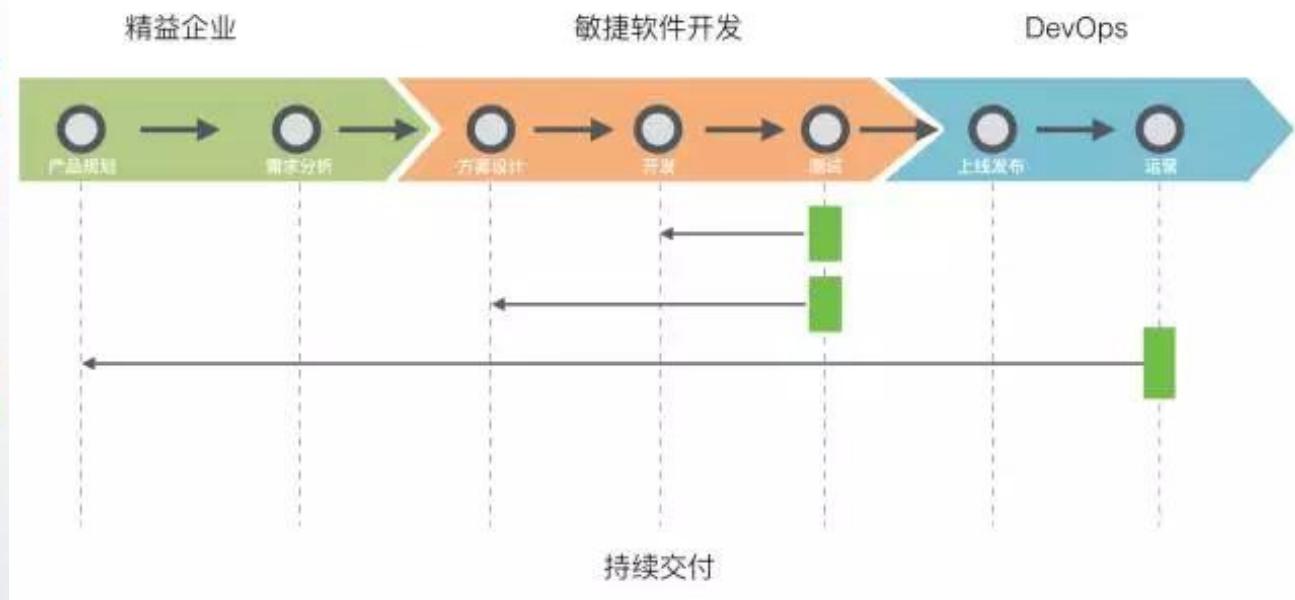


注：以精益管理为基石，以敏捷开发、持续交付和轻量级ITSM为支撑，其考试知识点广、注重实操、实践作业落地性强，
搭配以基于DevOps的实践沙盘，实乃广大互联网从业者提升及证明DevOps能力的有效途径！

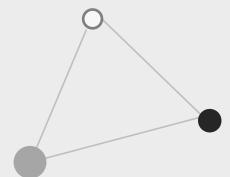
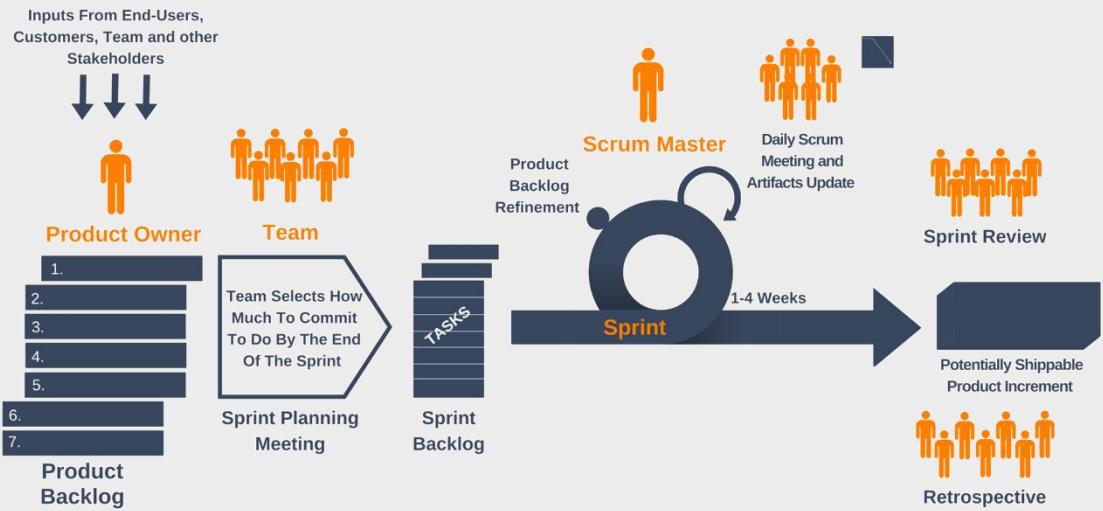
精益企业

敏捷软件开发

DevOps



管理维度



DevOps工具

PERIODIC TABLE OF DEVOPS TOOLS (V1)

XebiaLabs
Deliver Faster

This periodic table illustrates the classification of various DevOps tools across different categories and properties. The table is organized into groups and periods, similar to the periodic table of elements.

Legend:

- Os: Open Source
- Fr: Free
- Fm: Freemium
- Pd: Paid
- En: Enterprise

Category	Sub-Category	Tool	Properties
Database	Relational	MySQL	Os, En
	NoSQL	MongoDB	Os, Fr
	Big Data	HDFS	Os, Fr
	Cloud	AWS Lambda	Os, Fm
	Graph	Neo4j	Os, Fr
	Time Series	InfluxDB	Os, Fr
SCM	Version Control	Git	Os, En
	Code Review	GitHub	Os, Fr
	Continuous Integration	Jenkins	Os, En
	Build Automation	Apache Maven	Os, En
	Dependency Management	Gradle	Os, En
	Deployment Automation	Ansible	Os, En
Build	Containerization	Docker	Os, En
	Virtualization	Kubernetes	Os, En
	Cloud Native	Amazon Lambda	Os, Fm
	CI/CD Pipelines	GitLab CI	Os, En
	Deployment Scripts	Ansible	Os, En
	Monitoring	Splunk	Os, En
Testing	Unit Testing	JUnit	Os, En
	Integration Testing	Cucumber	Os, En
	System Testing	Nexus	Os, En
	Performance Testing	Apache JMeter	Os, En
	Security Testing	OWASP ZAP	Os, En
	Code Quality	SonarQube	Os, En
Deployment	Infrastructure as Code	Terraform	Os, En
	Configuration Management	Puppet	Os, En
	Orchestration	Docker Compose	Os, En
	Automation	Ansible	Os, En
	Monitoring	CloudWatch Metrics	Os, En
	Logs	Logstash	Os, En
Containerization	Containers	Docker	Os, En
	Orchestrators	Kubernetes	Os, En
	Runtimes	RocketMQ	Os, En
	Storage	Amazon S3	Os, Fm
	Networking	Amazon VPC	Os, Fm
	Security	AWS Lambda	Os, Fm
Cloud / IaaS / PaaS	Compute	Amazon Lambda	Os, Fm
	Storage	Amazon S3	Os, Fm
	Networking	Amazon VPC	Os, Fm
	Database	Amazon RDS	Os, Fm
	Serverless	Amazon Serverless Framework	Os, Fm
	Serverless Functions	Amazon Step Functions	Os, Fm
Monitoring	Metrics	CloudWatch Metrics	Os, En
	Logs	CloudWatch Logs	Os, En
	Metrics	CloudWatch Metrics	Os, En
	Logs	CloudWatch Logs	Os, En
	Metrics	CloudWatch Metrics	Os, En
	Logs	CloudWatch Logs	Os, En
Collaboration	Code Review	GitHub	Os, Fr
	Project Management	Asana	Os, En
	Task Management	Trello	Os, En
	Issue Tracking	Jira	Os, En
	Code Review	GitLab CI	Os, En
	Deployment Automation	Ansible	Os, En
Logging	Logs	Logstash	Os, En
	Metrics	CloudWatch Metrics	Os, En
	Metrics	CloudWatch Metrics	Os, En
	Logs	CloudWatch Logs	Os, En
	Metrics	CloudWatch Metrics	Os, En
	Logs	CloudWatch Logs	Os, En
Security	Encryption	TLS	Os, En
	Identity Management	AWS IAM	Os, En
	Compliance	PCI DSS	Os, En
	Encryption	TLS	Os, En
	Identity Management	AWS IAM	Os, En
	Compliance	PCI DSS	Os, En

Share

[Twitter](#) [LinkedIn](#) [Google+](#) [Email](#) [Print](#)

Embed

[HTML](#) [JSON](#) [XML](#)

Become Excellent!

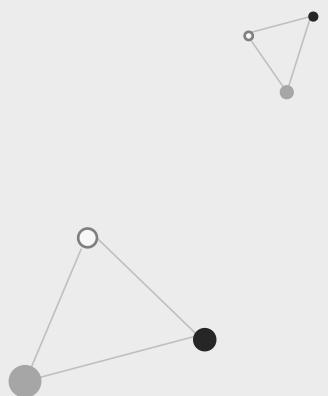
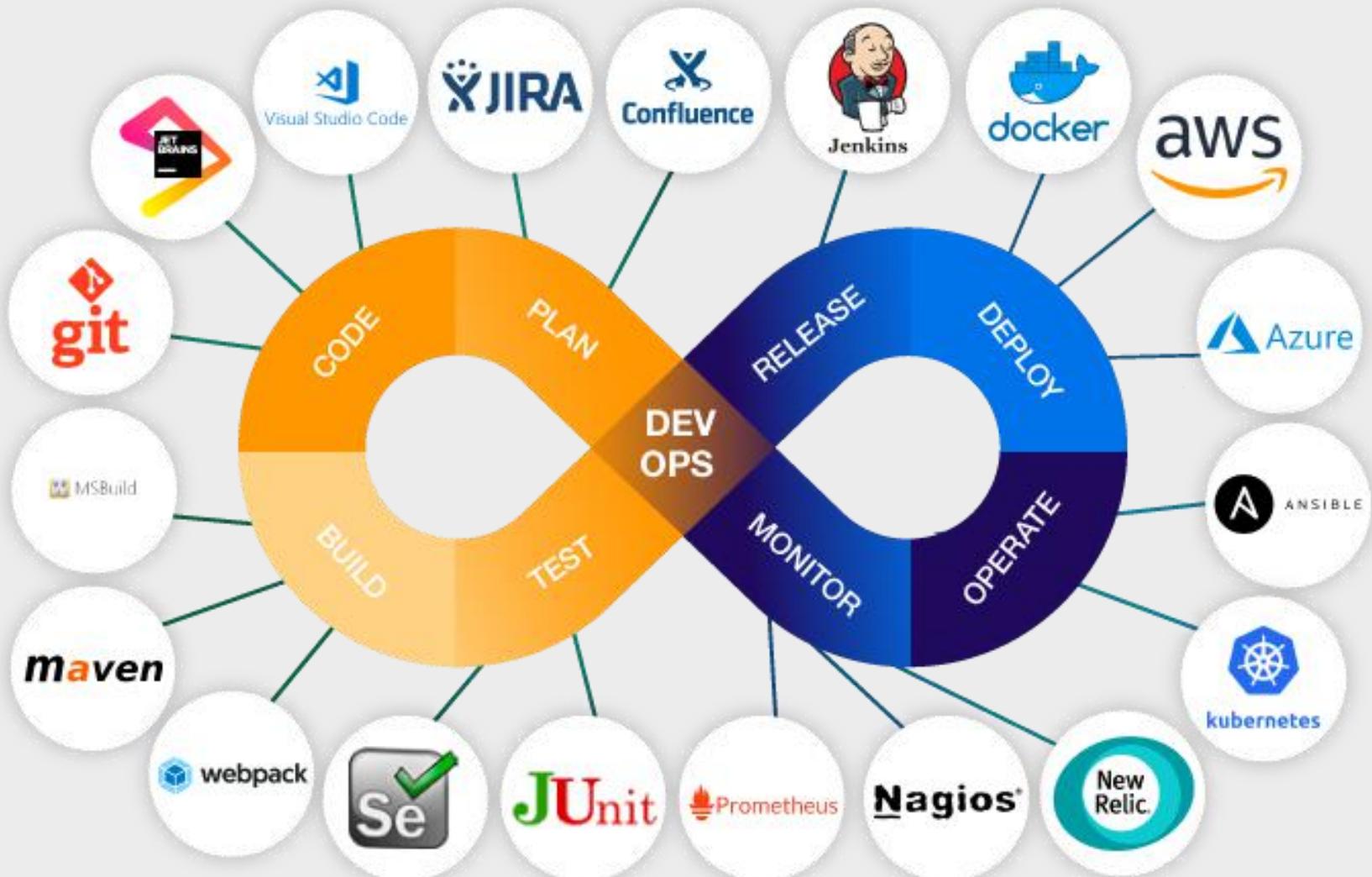
[Subscribe here!](#)

© 2018 XebiaLabs, Inc. All rights reserved. XebiaLabs is a registered trademark of XebiaLabs, Inc. Other products and services may be trademarks of their respective owners.

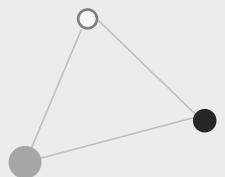
91. En	92. En	93. En	94. En	95. En	96. Pd	97. En	98. En	99. Fm	100. Pd	101. Fm	102. Fm	103. Fm	104. Pd	105. En
Xlr	Ur	Ls	Bm	Hp	Ex	Pl	Sr	Tr	Jr	Rf	Sl	Fd	Pv	Sn
XL Release	UrbanCode Release	CA Service Virtualization	BMC Release Process	HP Cedar	Excel	Plutora Release	Serenity Release	Trello	Jira	HipChat	Slack	Flowdock	Pivotal Tracker	ServiceNow
106. En	107. Os	108. Fm	109. Os	110. Os	111. Os	112. Os	113. Os	114. Fm	115. Os	116. Fm	117. Os	118. Os	119. Os	120. En
Sp	Ki	Nr	Ni	Gg	Ct	Gr	Ic	Sl	Ls	Lg	Gr	Sn	Tr	Cy
Splunk	Kibana	New Relic	Nagios	Ganglia	Cacti	Graphite	Icinga	Sumo Logic	Logstash	Loggly	Graylog	Snort	Tripwire	CyberArk



DevOps工具



我们现在“该不该”落地DevOps ?



该优化哪里？



5 件

0 (s) 等待

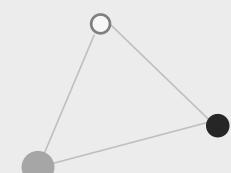
500 件 

500 件

0 (s) 等待

50 件 

要的是没有瓶颈



目 录

01. 凤凰项目

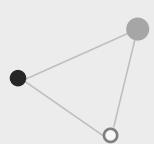
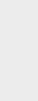
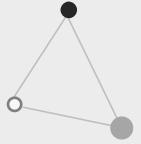
03. “项目生命周期”

02. 什么是DevOps ?

04. 关于认证

01

凤凰项目



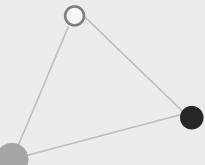
凤凰项目 (The Phoenix Project)

沙盘简介

凤凰项目DevOps沙盘是欧洲著名沙盘研发机构Gaming Works的创始人Jan Schilt先生联手《凤凰项目》一书作者Gene Kim先生联手开发的同名沙盘推演教程。

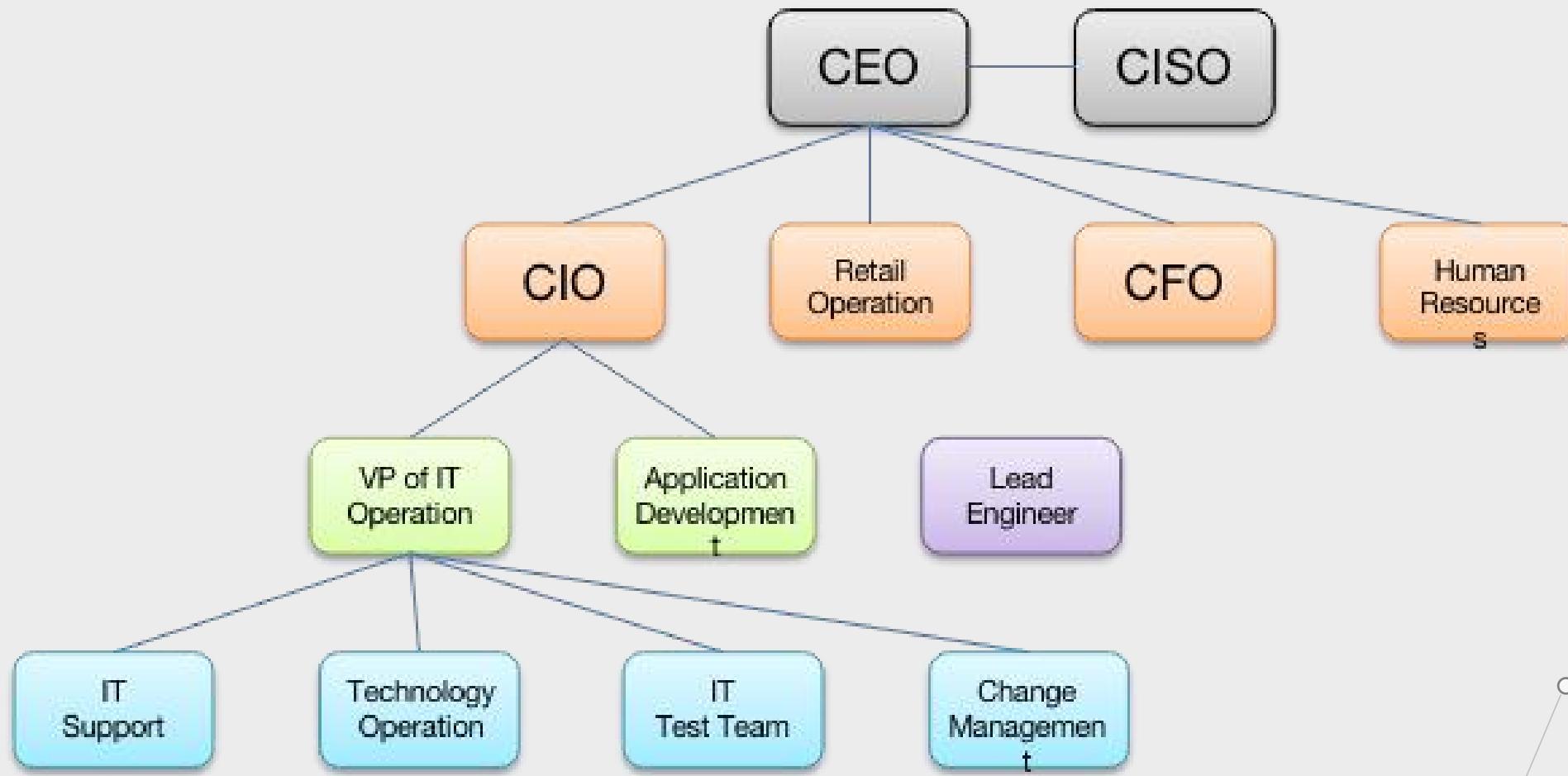
《凤凰项目》沙盘演练覆盖了人们实际业务场景中几乎所有的核心角色，适用范围广，特别适用于：

- 希望通过精益（LEAN），敏捷（AGILE）和IT服务管理如（ITIL）等最佳实践为业务创造价值的IT专业人士。
- 希望运用敏控项目管理，通过DevOps转化IT项目成果，实现业务收益的项目经理、IT管理人员和业务人员。



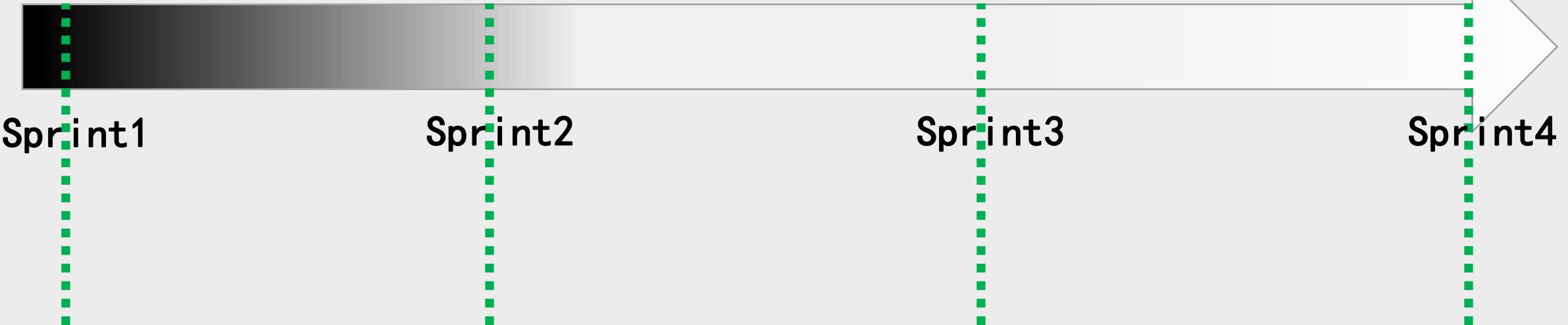
凤凰项目 (The Phoenix Project)

无极限零件公司的组织关系



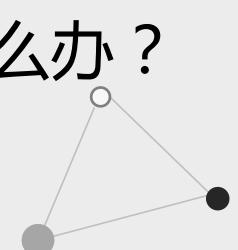
凤凰项目 (The Phoenix Project)

思考项目如果我们来做，会遇到什么问题？



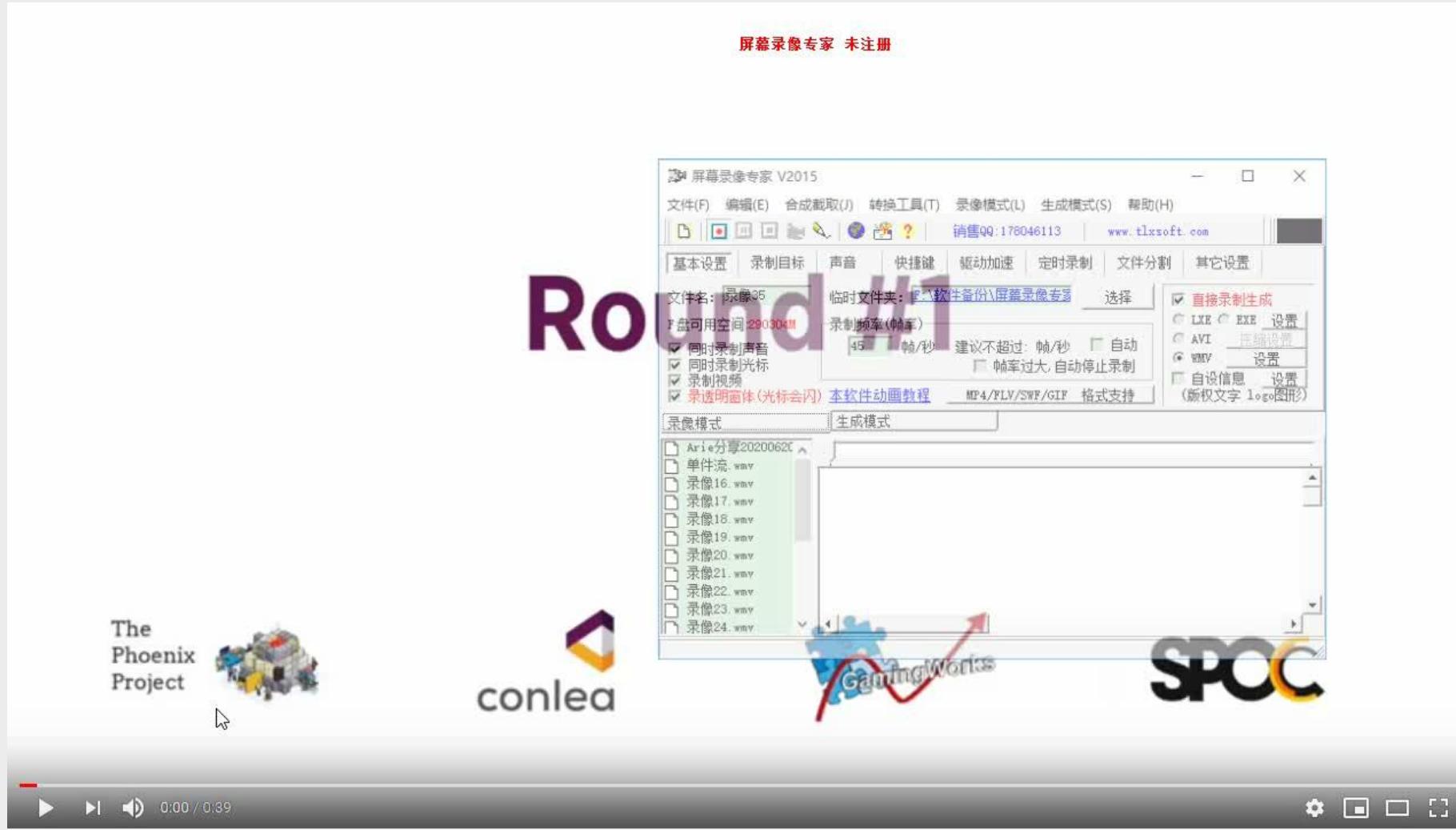
如果一个项目：

1. 没有项目经理，我们找谁？
2. 需求多的做不完怎么办？需求如何进行排序？
3. 任务如何跟进？各个角色的瓶颈怎么识别？
4. 任务经常有“失败”（迭代结束时测试人员发现“BUG”）的，怎么办？
5.



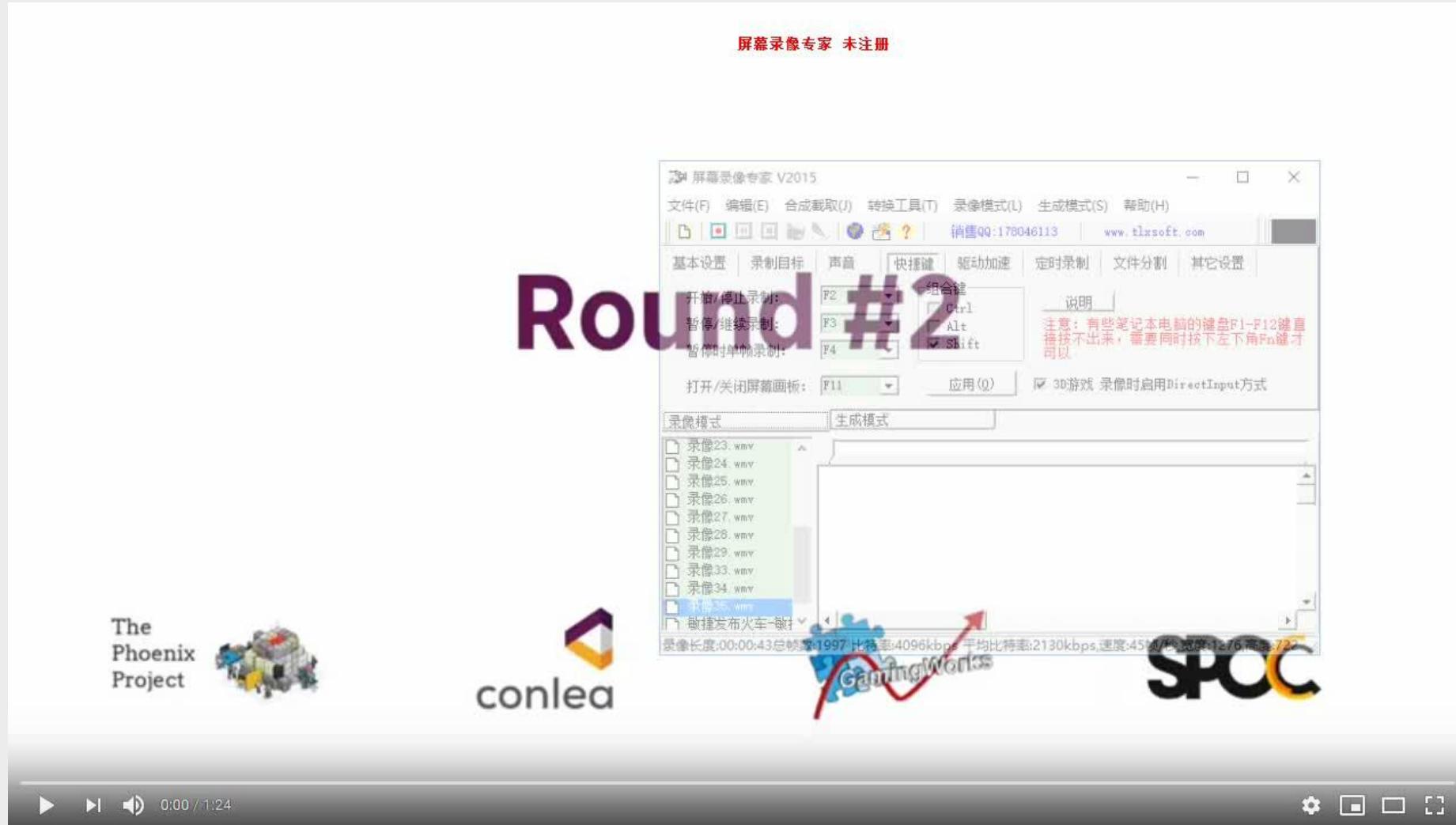
凤凰项目 (The Phoenix Project)

凤凰项目沙盘 —— Round 01



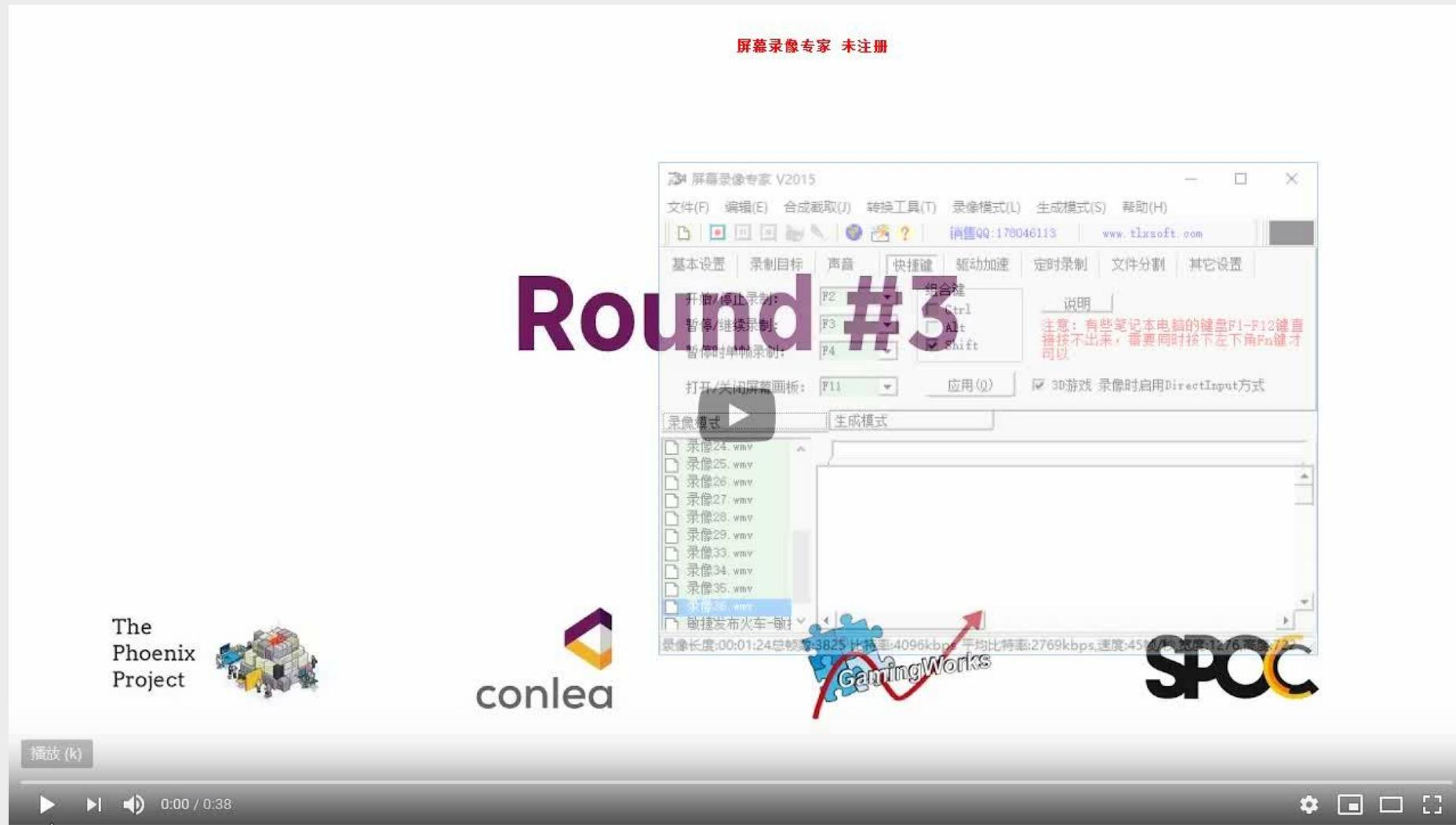
凤凰项目 (The Phoenix Project)

凤凰项目沙盘 —— Round 02



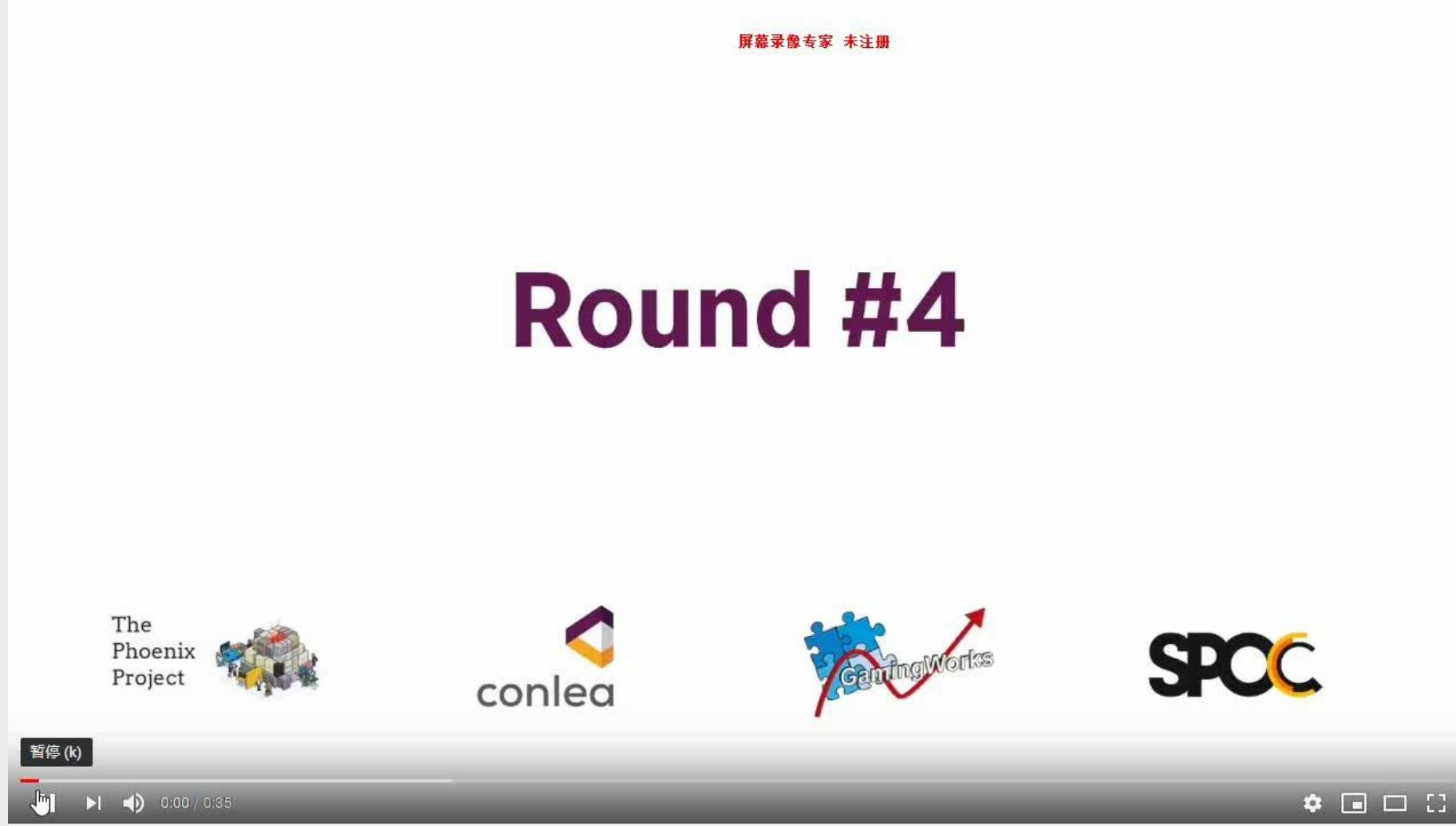
凤凰项目 (The Phoenix Project)

凤凰项目沙盘 —— Round 03



凤凰项目 (The Phoenix Project)

凤凰项目沙盘 —— Round 04



凤凰项目 (The Phoenix Project)

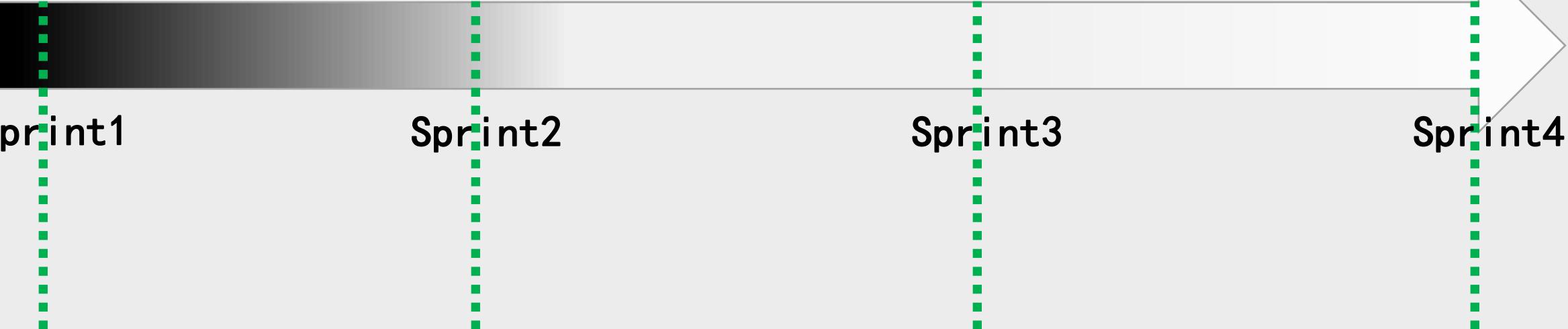
凤凰项目沙盘的进展

Sprint1

Sprint2

Sprint3

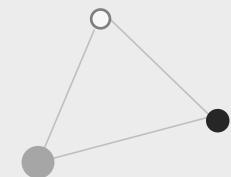
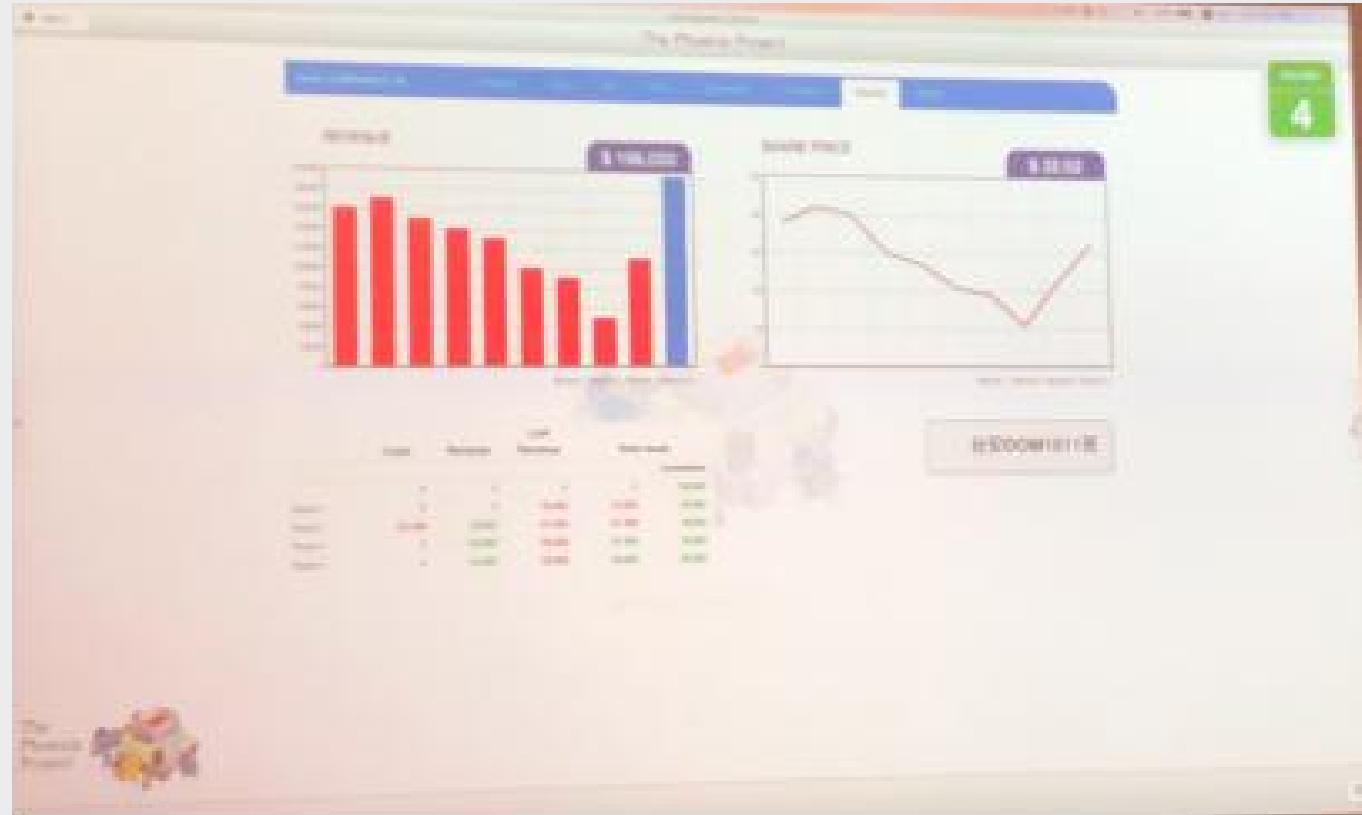
Sprint4



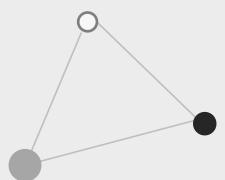
乱

凤凰项目 (The Phoenix Project)

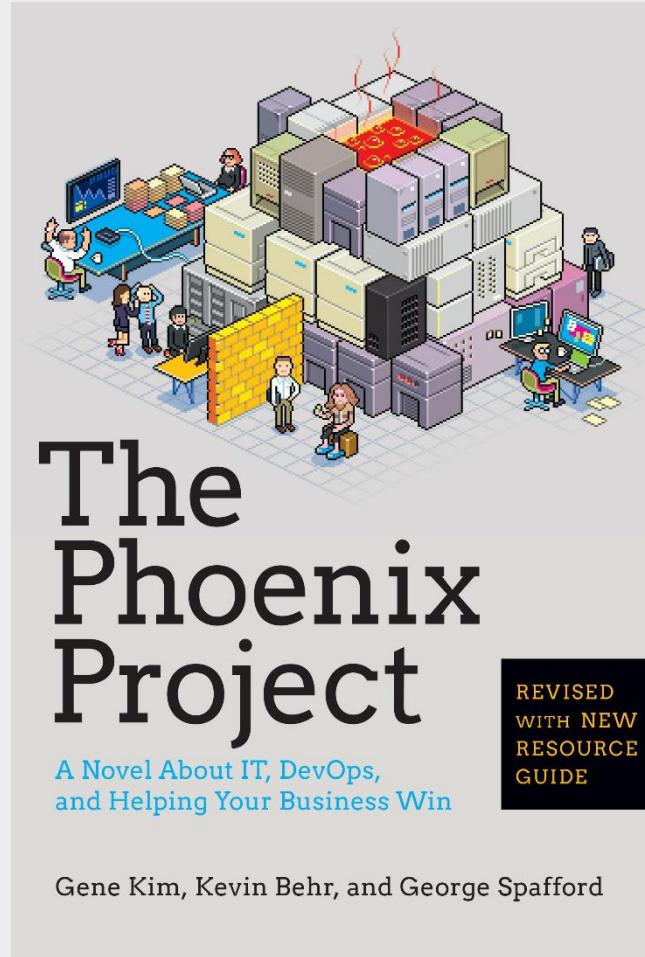
凤凰项目沙盘的成果展示



沙盘的回顾



凤凰项目 (The Phoenix Project)



1、团队的发展阶段

2、项目成功的定义

3、敏捷基础

4、部署流水线

5、价值流

6、可视化

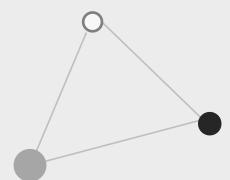
7、单件流

8、测试驱动

9、管理依赖

10、规模化

1、团队的发展阶段

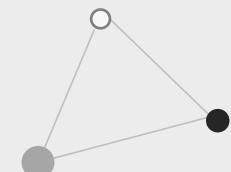


凤凰项目 (The Phoenix Project) —— 团队的发展阶段

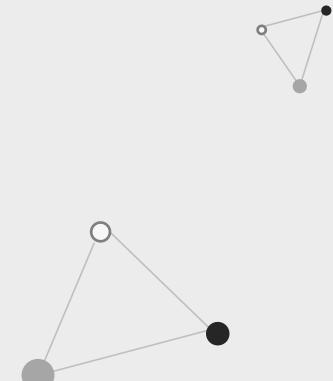
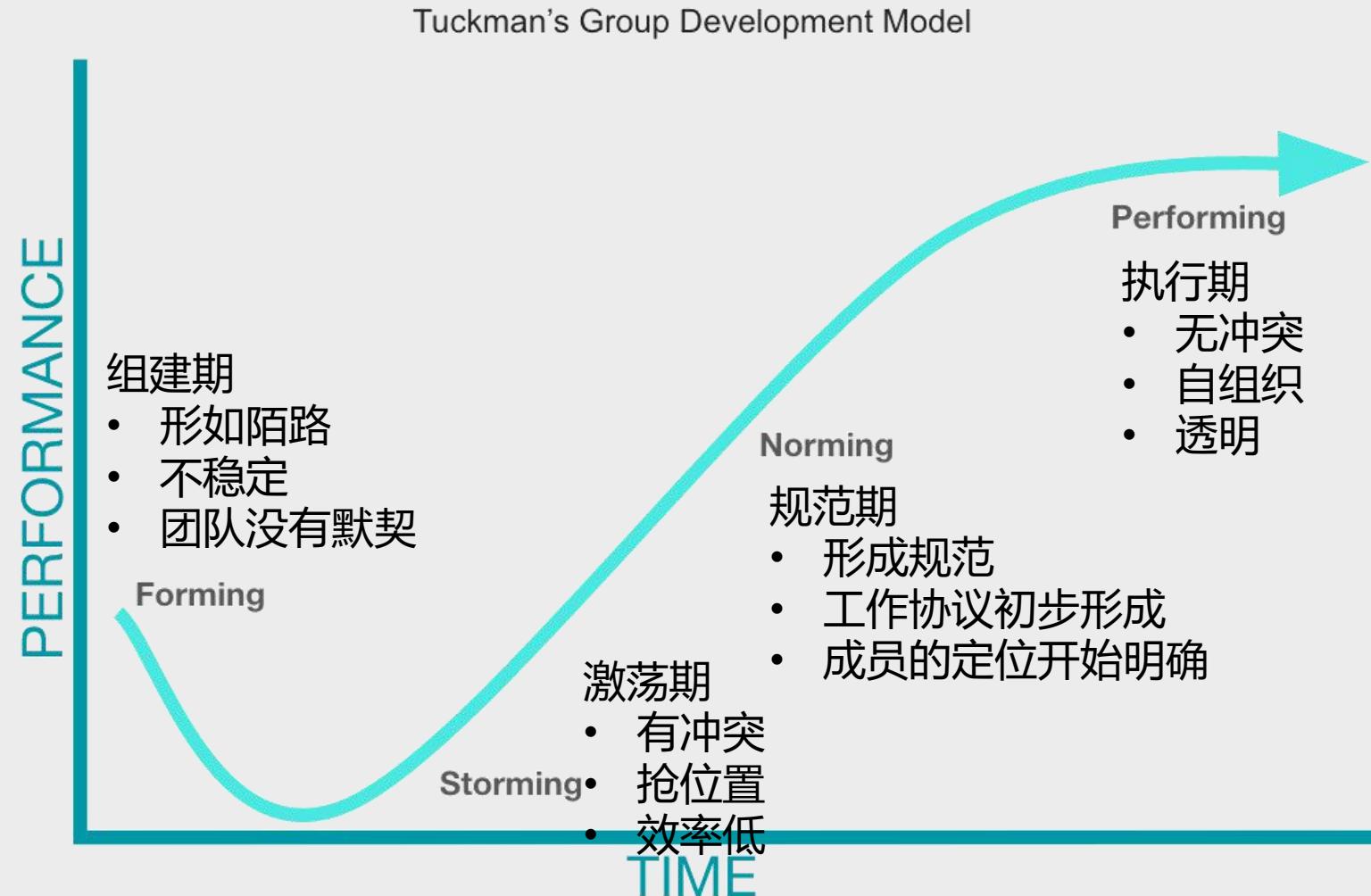
什么是团队？

“团队是朝着一个**共同目标努力**的群体，人们相互之间彼此依赖，成员之间有一定的熟悉度。团队会有自己的一组信念，反映了大多数成员的想法。另外，有内部和外部作用力促使团队建立和维护其团队的同一性。”

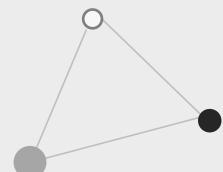
——《Effective DevOps》



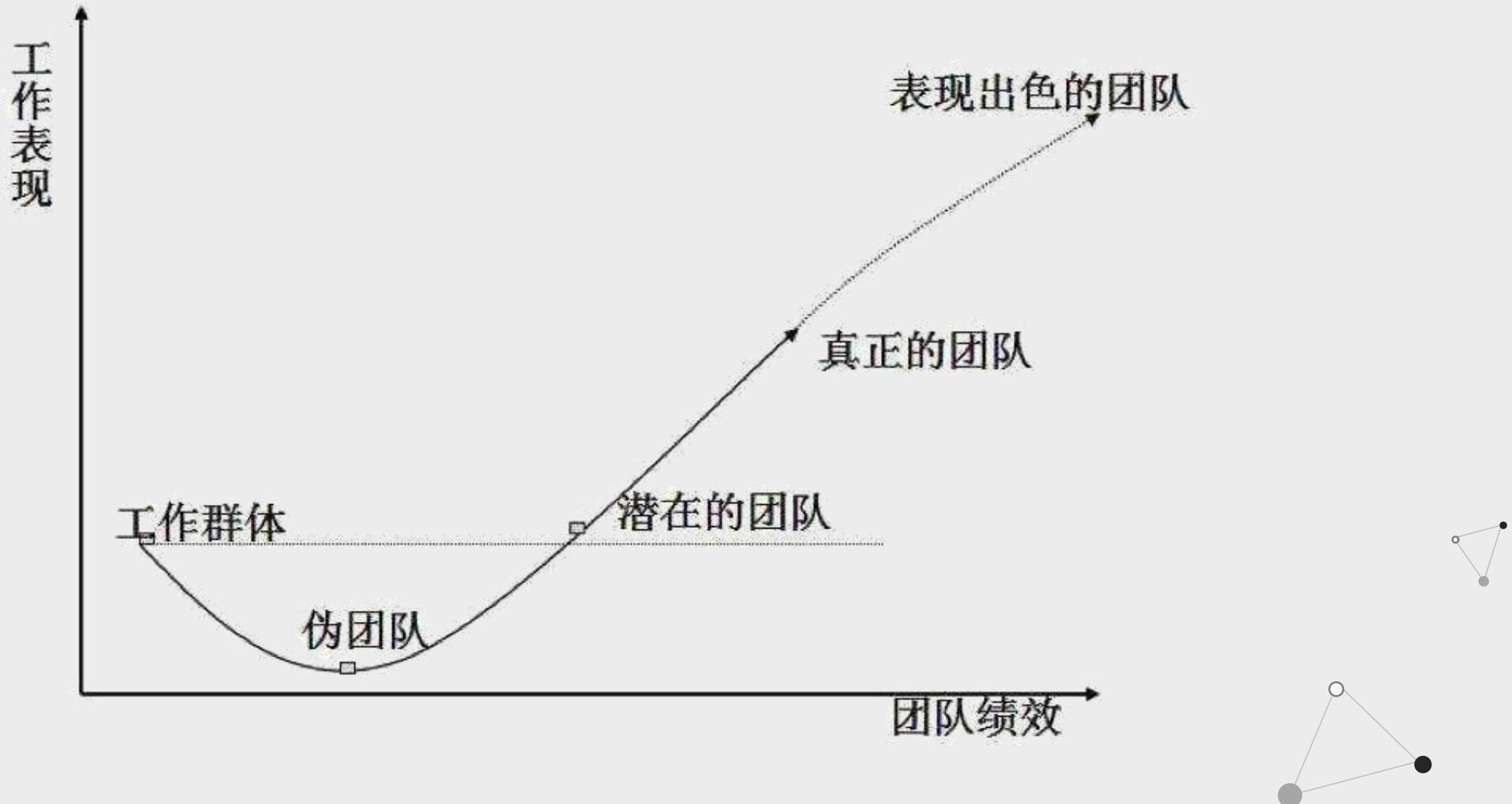
凤凰项目 (The Phoenix Project) —— 团队的发展阶段



**了解了塔克曼团队发展阶段模型，
那么我们实际团队的情况是什么样的？**

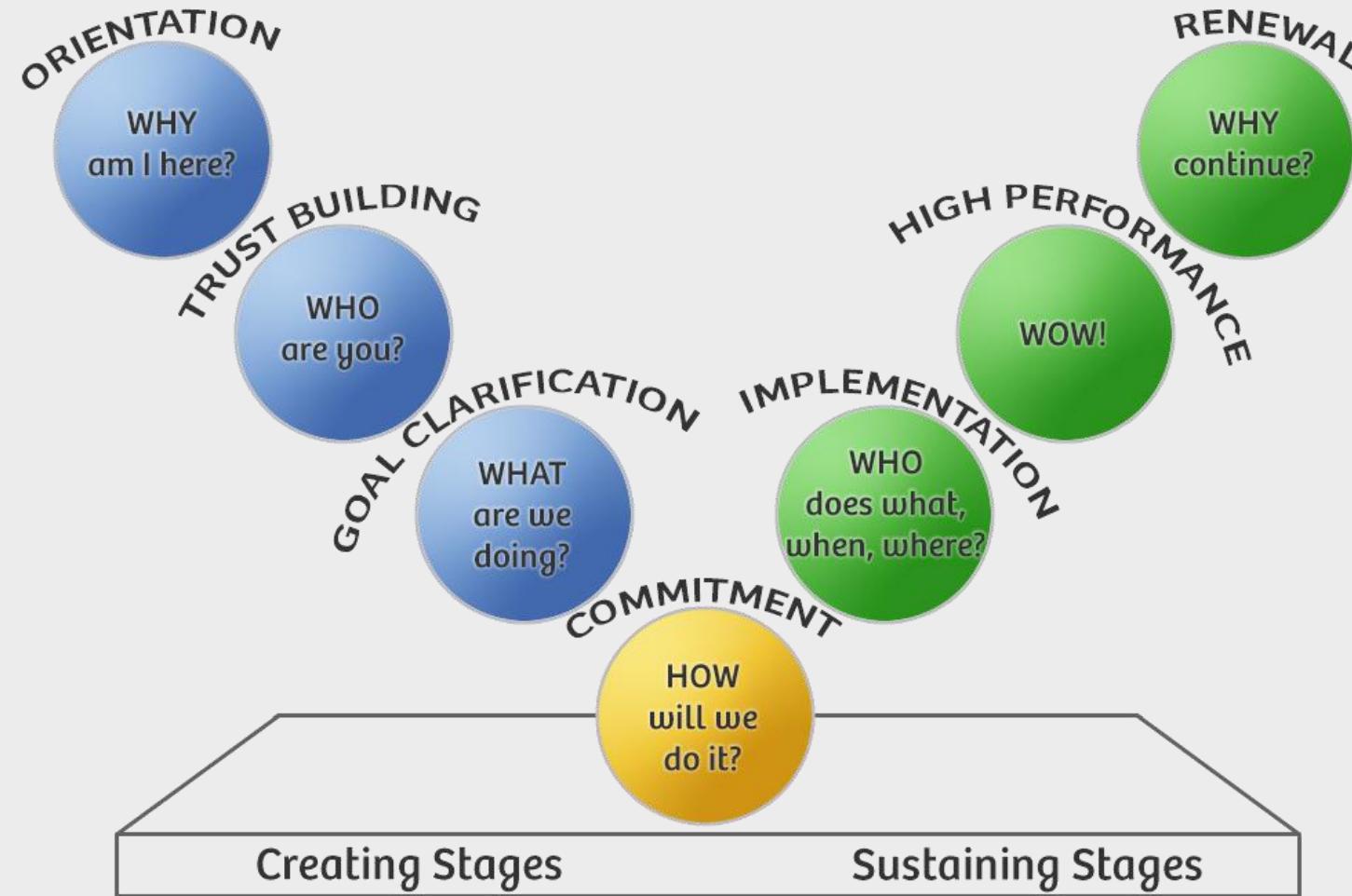


凤凰项目 (The Phoenix Project) —— 团队的发展阶段

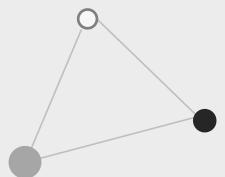


凤凰项目 (The Phoenix Project) —— 团队的发展阶段

Drexler – Sibbet Model (选修)

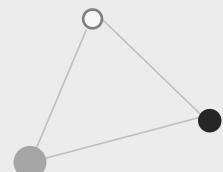


2. 项目成功的定义 (含项目管理基础)



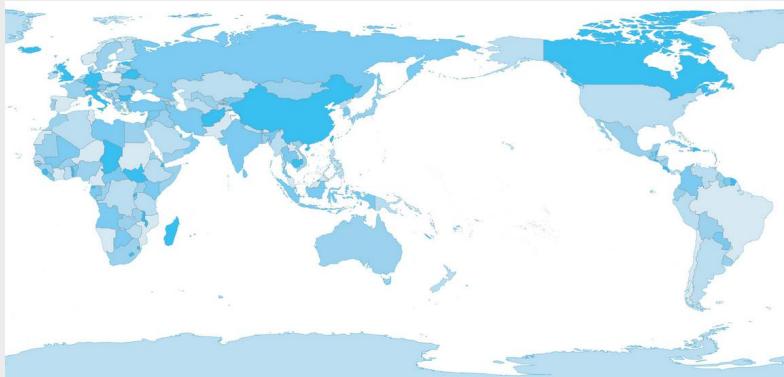
凤凰项目 (The Phoenix Project) —— 项目成功的定义

说说大家心中的项目经理？



凤凰项目 (The Phoenix Project) —— 项目成功的定义

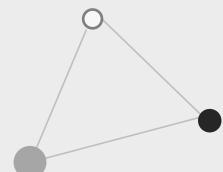
项目经理 (Project Manager)



架构设计、测试、文档编写等，全能人才，无处不在？



联络的中心，核心枢纽？



凤凰项目 (The Phoenix Project) —— 项目成功的定义

Coding们为什么总和PM（产品经理&项目经理）打架？

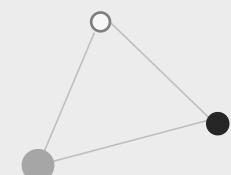


凤凰项目 (The Phoenix Project) —— 项目管理基础

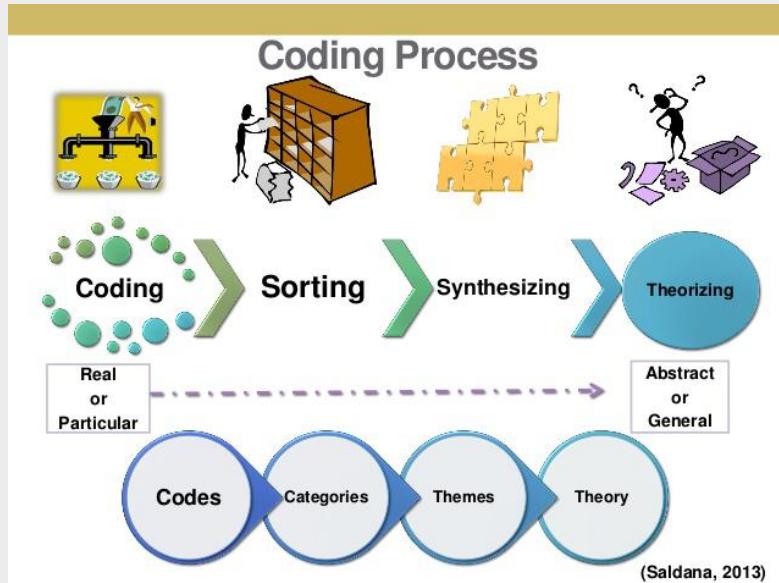


技术思维

管理思维



技术思维



关注每一项任务的本身，而不是整体目标。不重视计划，对任务缺乏控制。

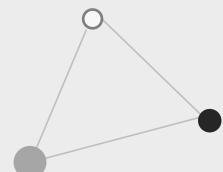
以过程为中心

管理思维



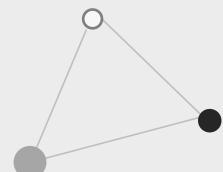
关注整体目标，实现技术路线，影响目标的实现因素、各种事件对目标的影响，区分重点。

以目标为中心

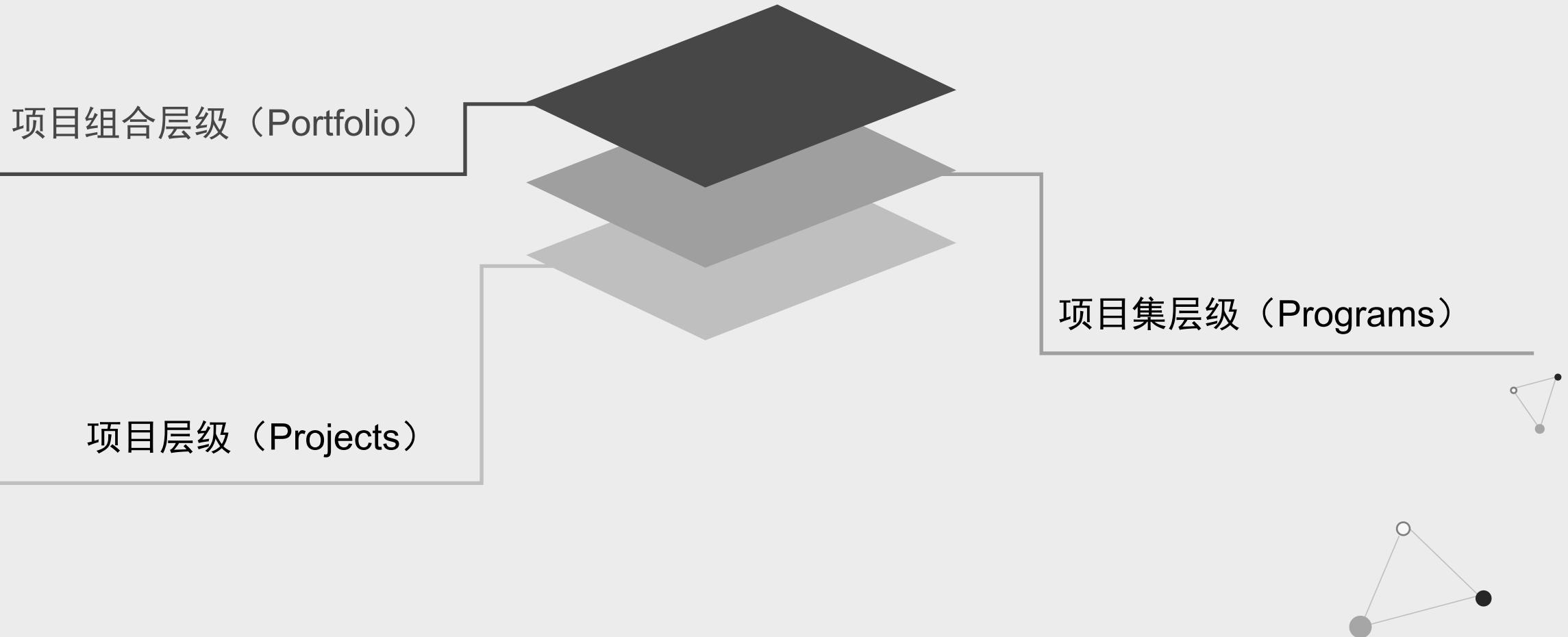


凤凰项目 (The Phoenix Project) —— 项目管理基础

从开发人员到项目经理，重要的是
得实现自己角色的转化，其本质是学会思维的转化。



凤凰项目 (The Phoenix Project) —— 项目成功的定义

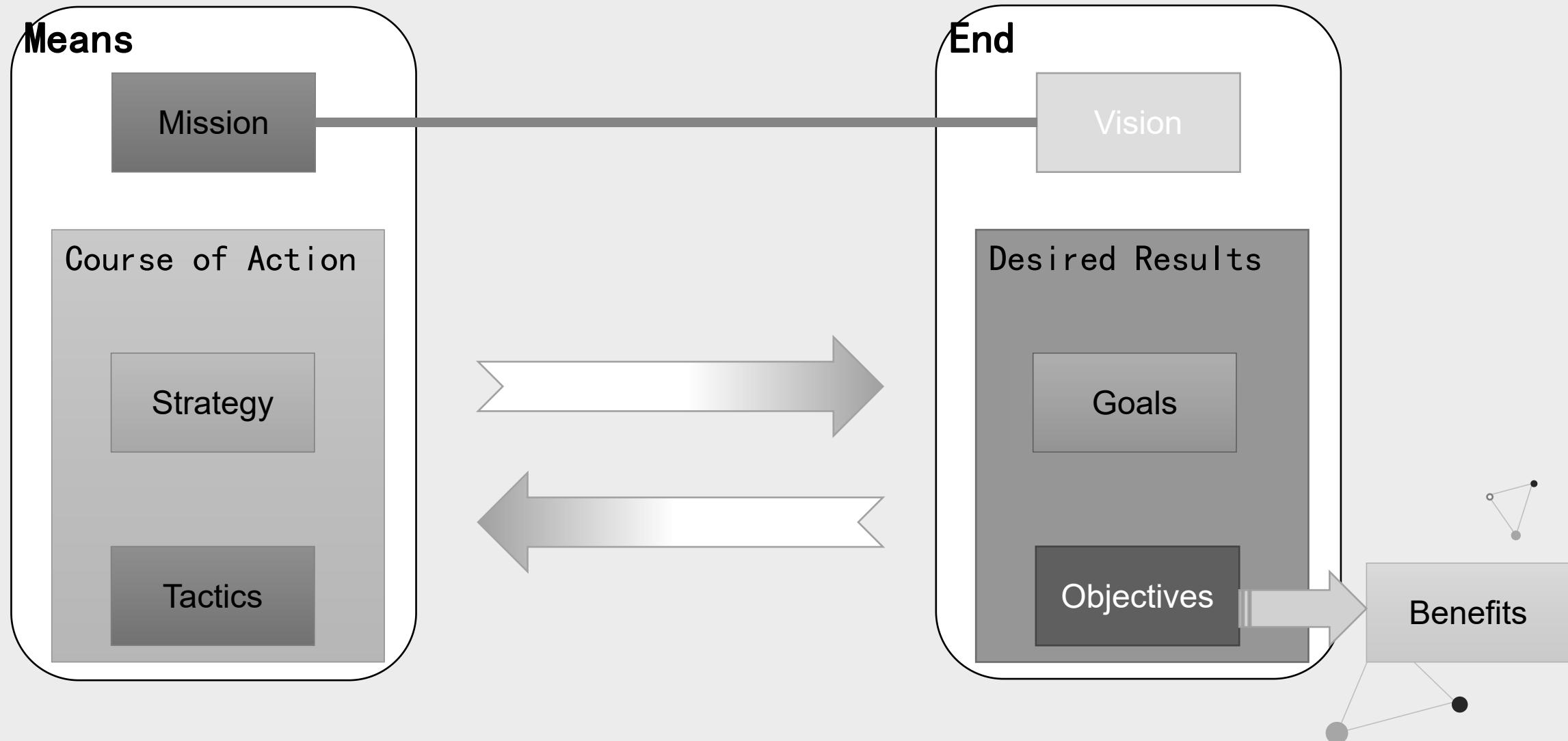


凤凰项目 (The Phoenix Project) —— 项目成功的定义

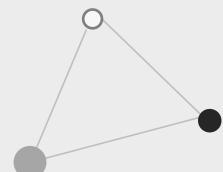
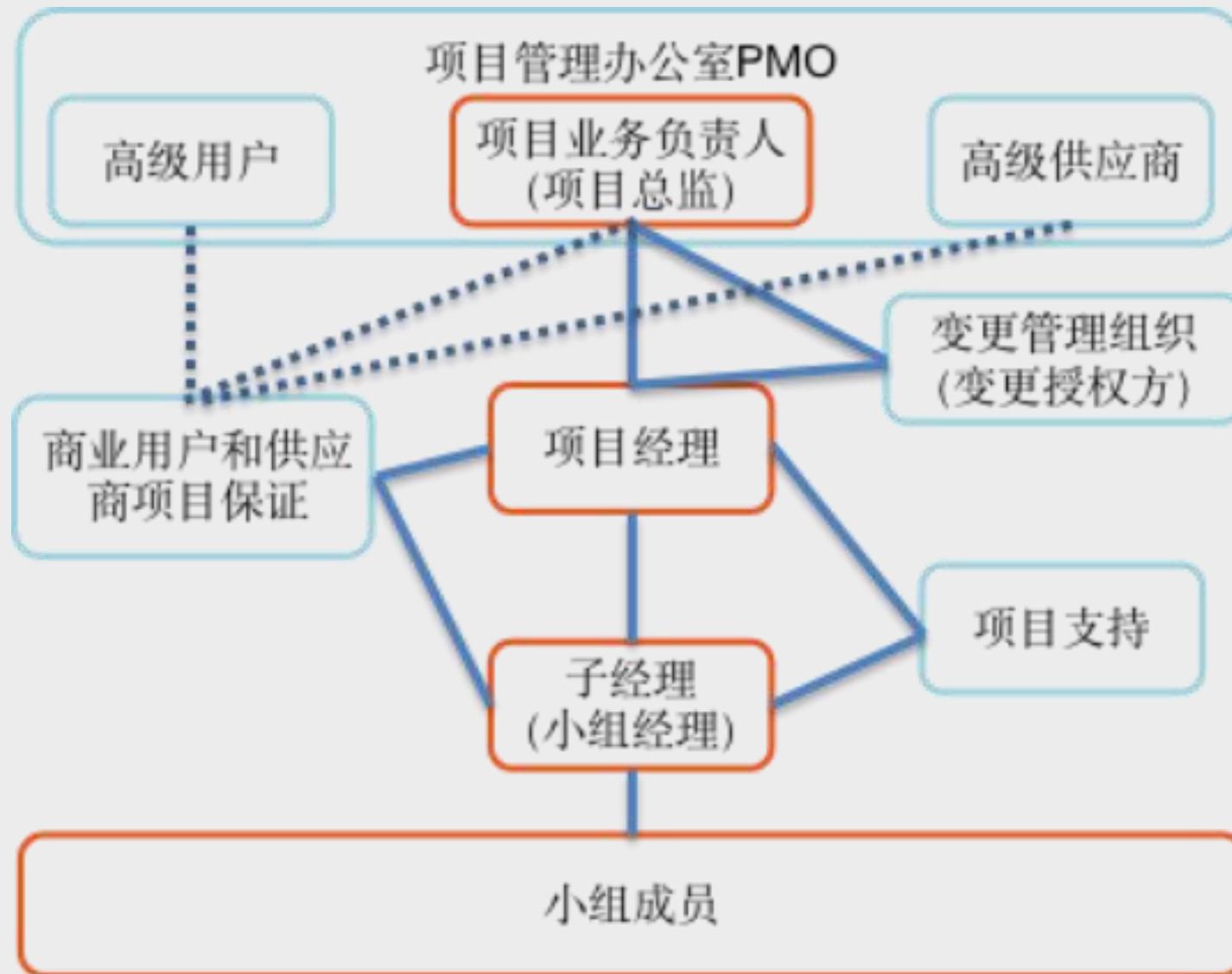


凤凰项目 (The Phoenix Project) —— 项目成功的定义

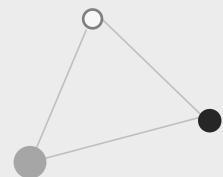
Vision–Mission–Strategy–Tactics–Goals–Objectives–Benefits



凤凰项目 (The Phoenix Project) —— 项目成功的定义

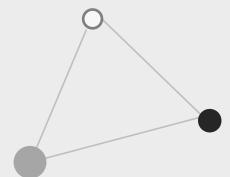


凤凰项目 (The Phoenix Project) —— 项目成功的定义



凤凰项目 (The Phoenix Project) —— 项目成功的定义

十大知识领域 (源自 PMBOK)



凤凰项目 (The Phoenix Project) —— 项目成功的定义

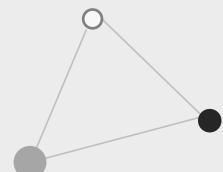
传统的项目经理



以技术专长作为管理
下属强有力的基础

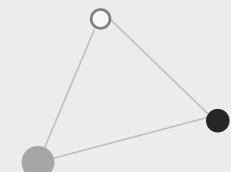
忙于开会并且
忙于解决一个又一个
的问题

处理紧急问题



凤凰项目 (The Phoenix Project) —— 项目成功的定义

敏捷领导力层次



凤凰项目 (The Phoenix Project) —— 项目成功的定义

专家型领导 (Expert)



特点：

1. 增量性改善，与人（干系人）交互少
2. 忙于工作细节
3. 强烈主张坚持意见，几乎没有反馈
4.



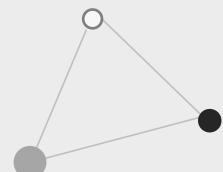
凤凰项目 (The Phoenix Project) —— 项目成功的定义

成就型领导 (Achiever)



特点：

1. 分析现状，策略寻求干系人的支持
2. 和员工开会商讨以获取支持
3. 主张适当的调整，获得支持
4. 完成目标←----→有限的反馈
5.



凤凰项目 (The Phoenix Project) —— 项目成功的定义

催化型领导 (Catalyst)



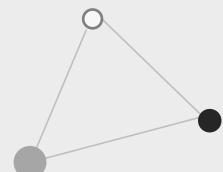
特点：

1. 创建团队工作文化
2. 参与制团队；引导者；公开交换想法、观点
3. 在主张和适应中灵活调整，寻求和应用反馈
4.



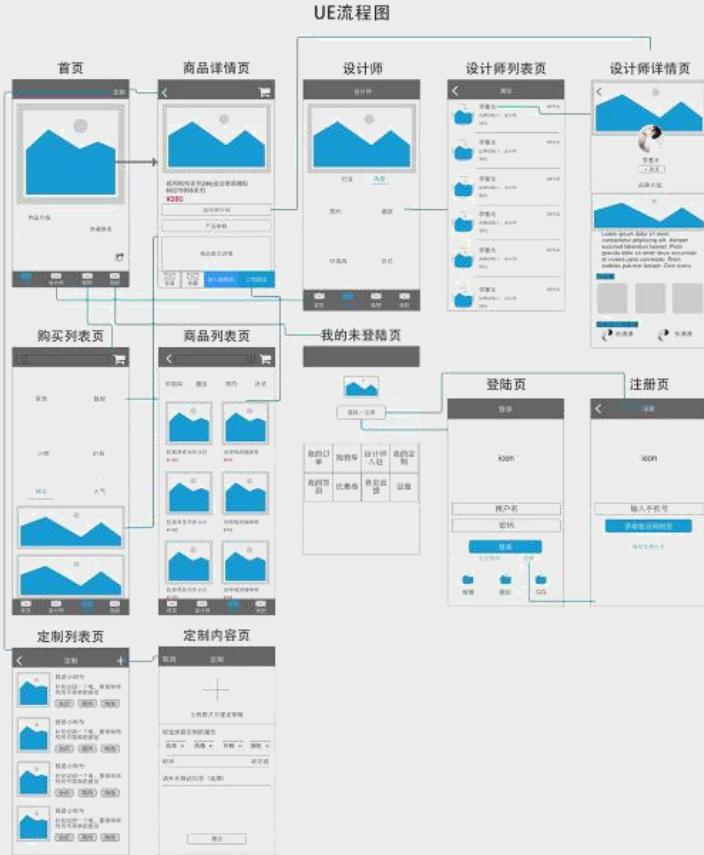
凤凰项目 (The Phoenix Project) —— 项目成功的定义

说完了项目经理，那么产品经理呢？



凤凰项目 (The Phoenix Project) —— 项目管理基础

产品经理 (Product Manager)



画界面原型的

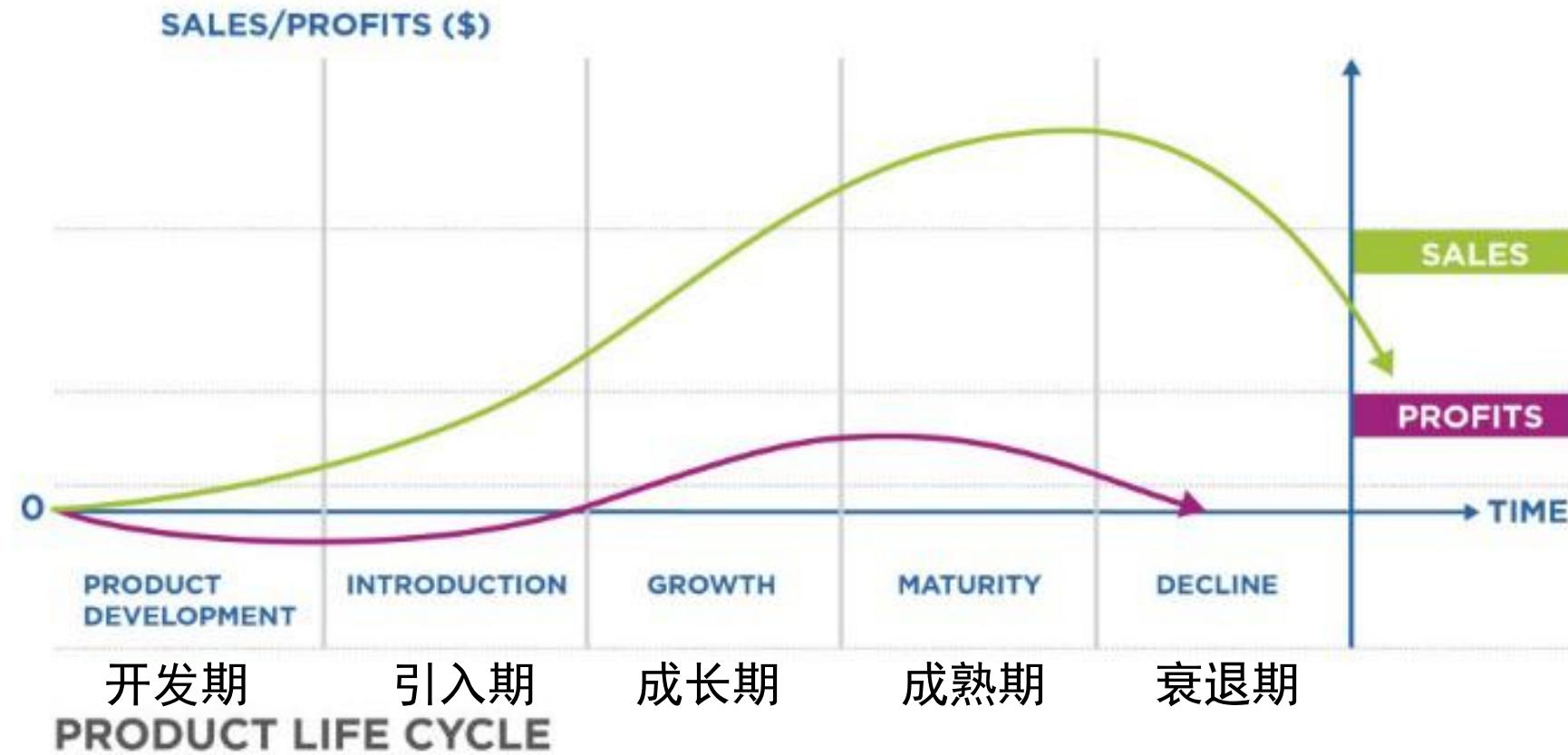
和客户沟通的
(类似销售)



添加“没用”的需求,
然后……

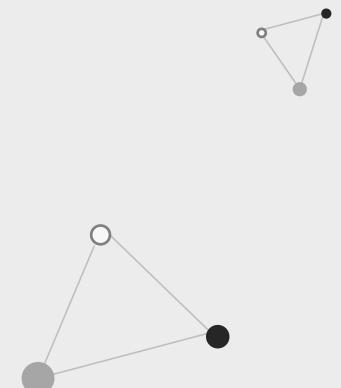
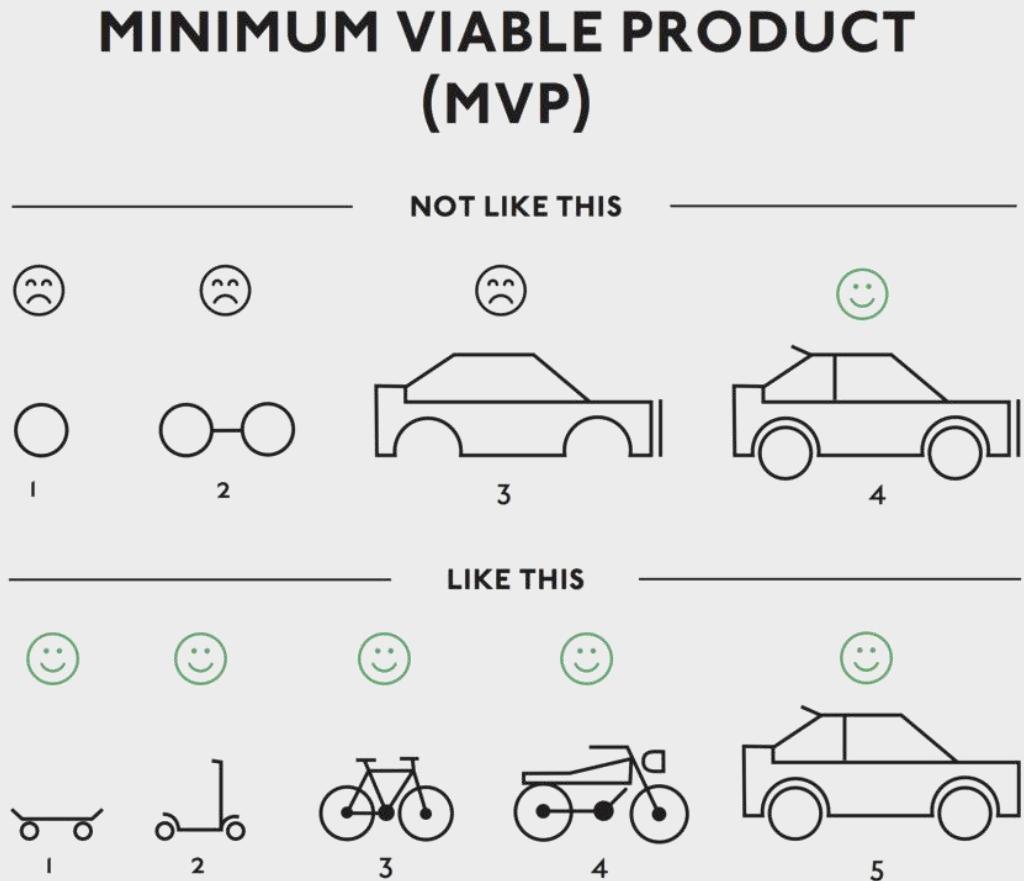
凤凰项目 (The Phoenix Project) —— 项目管理基础

产品生命周期



凤凰项目 (The Phoenix Project) —— 项目管理基础

产品生命周期



凤凰项目 (The Phoenix Project) —— 项目管理基础

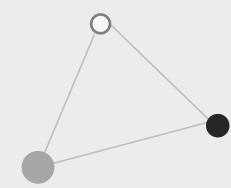
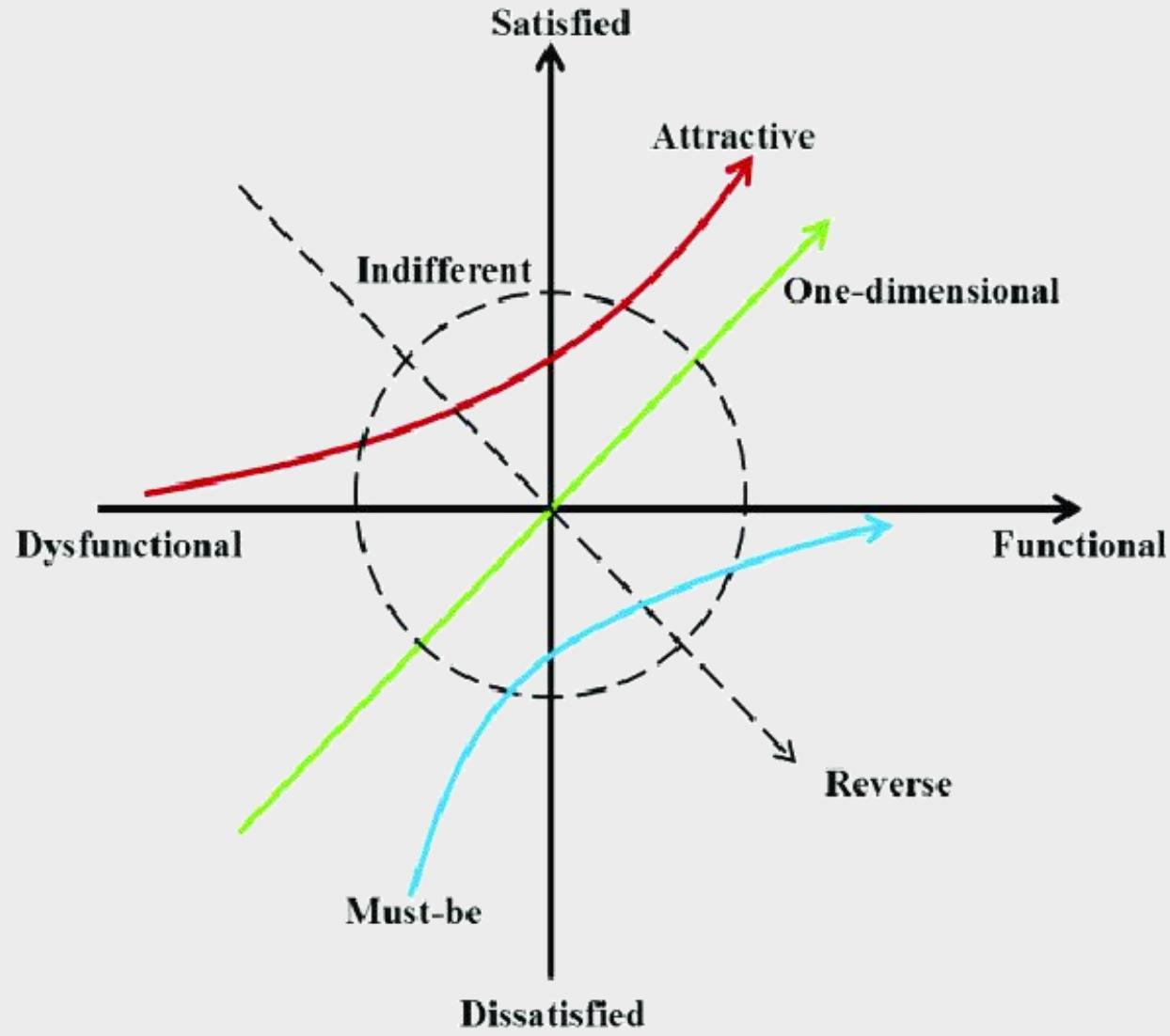
产品生命周期

Hits	A metric from the early, foolish days of the Web. Count people instead.
Page views	Marginally better than hits. Unless you're displaying ad inventory, count people.
Visits	Is this one person visiting a hundred times, or are a hundred people visiting once? Fail.
Unique visitors	This tells you nothing about what they did, why they stuck around, or if they left.
Followers/friends/likes	Count actions instead. Find out how many followers will do your bidding.
Time on site, or pages/visit	Poor version of engagement. Lots of time spent on support pages is actually a bad sign.
Emails collected	How many recipients will act on what's <i>in</i> them?
Number of downloads	Outside app stores, downloads alone don't lead to lifetime value. Measure activations/active accounts.

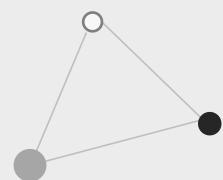


凤凰项目 (The Phoenix Project) —— 项目管理基础

Kano Model



3、敏捷基础



凤凰项目 (The Phoenix Project) —— 敏捷基础

The background image shows a group of people, including men and women, gathered around a large, detailed map or blueprint spread out on a table. They appear to be discussing and pointing at specific parts of the map, suggesting a collaborative planning or design process.

敏捷软件开发宣言

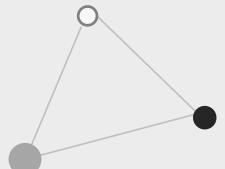
我们一直在实践中探寻更好的软件开发方法，
身体力行的同时也帮助他人。由此我们建立了如下价值观：

个体和互动 高于 流程和工具
工作的软件 高于 详尽的文档
客户合作 高于 合同谈判
响应变化 高于 遵循计划

也就是说，尽管右项有其价值，
我们更重视左项的价值。

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

殃殃的书屋



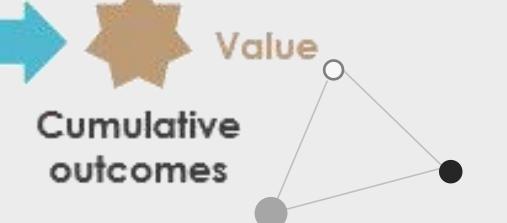
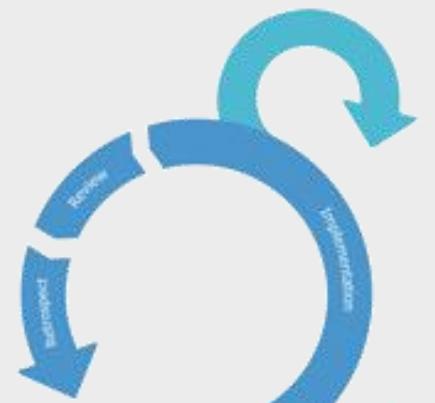
凤凰项目 (The Phoenix Project) — 敏捷基础



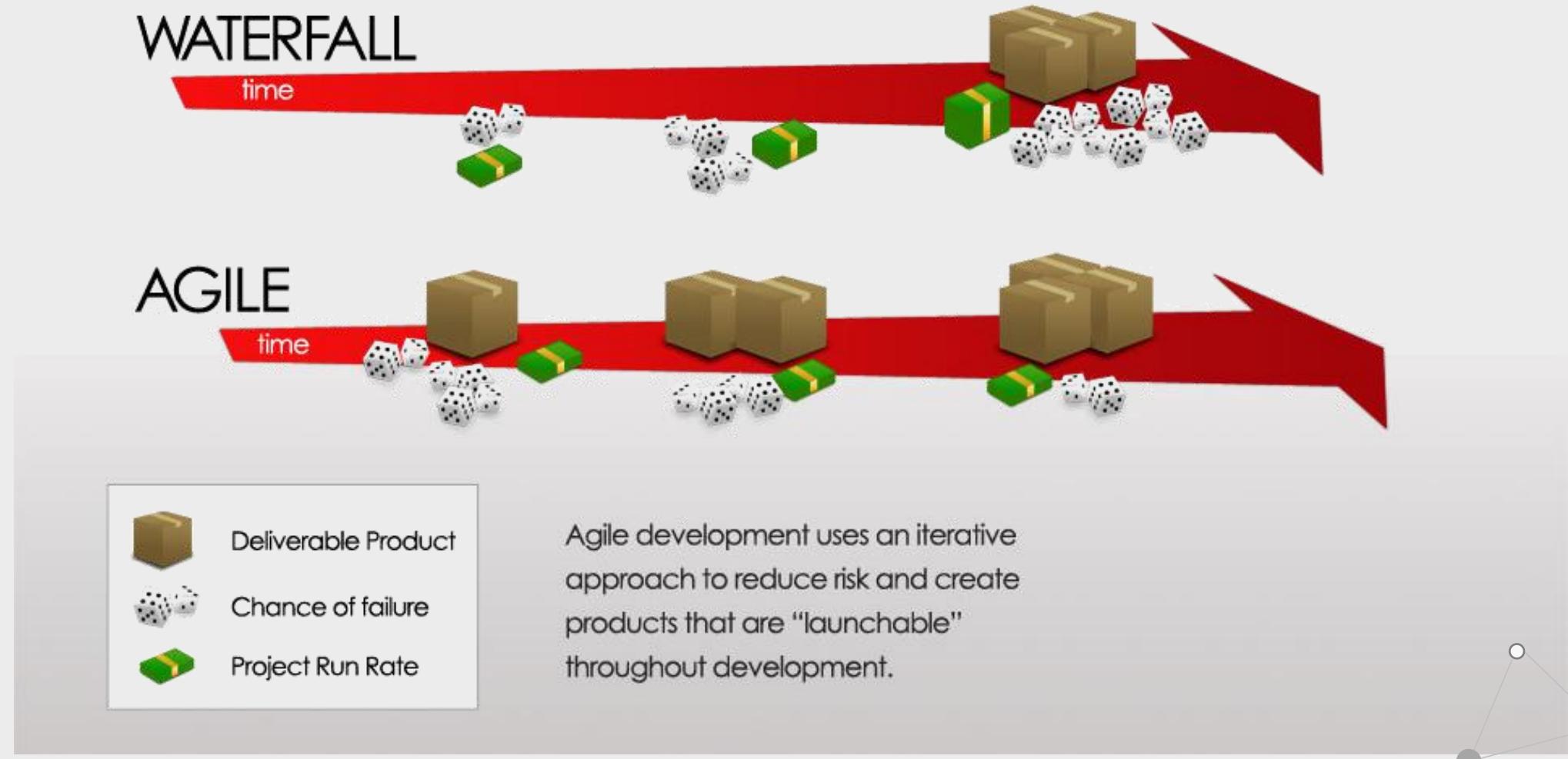
Waterfall Project ►

Project Timeline

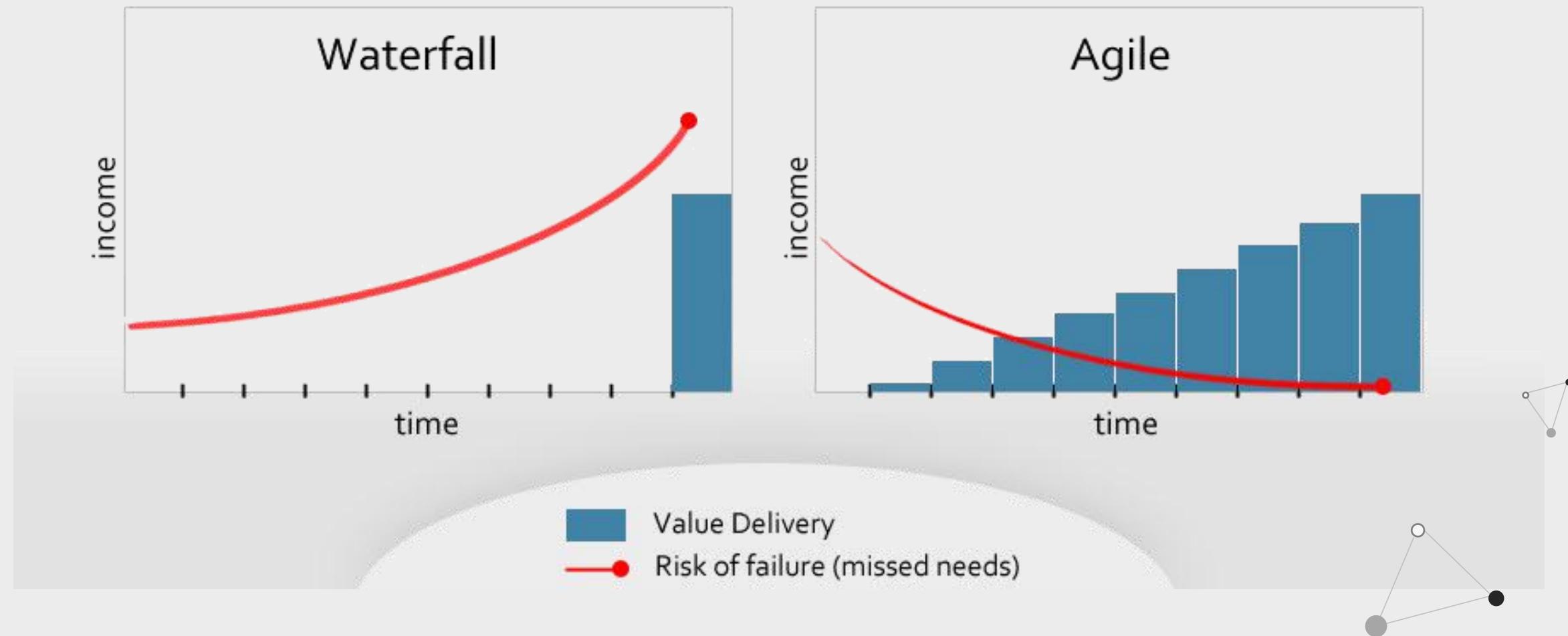
Agile Project ►



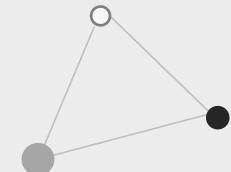
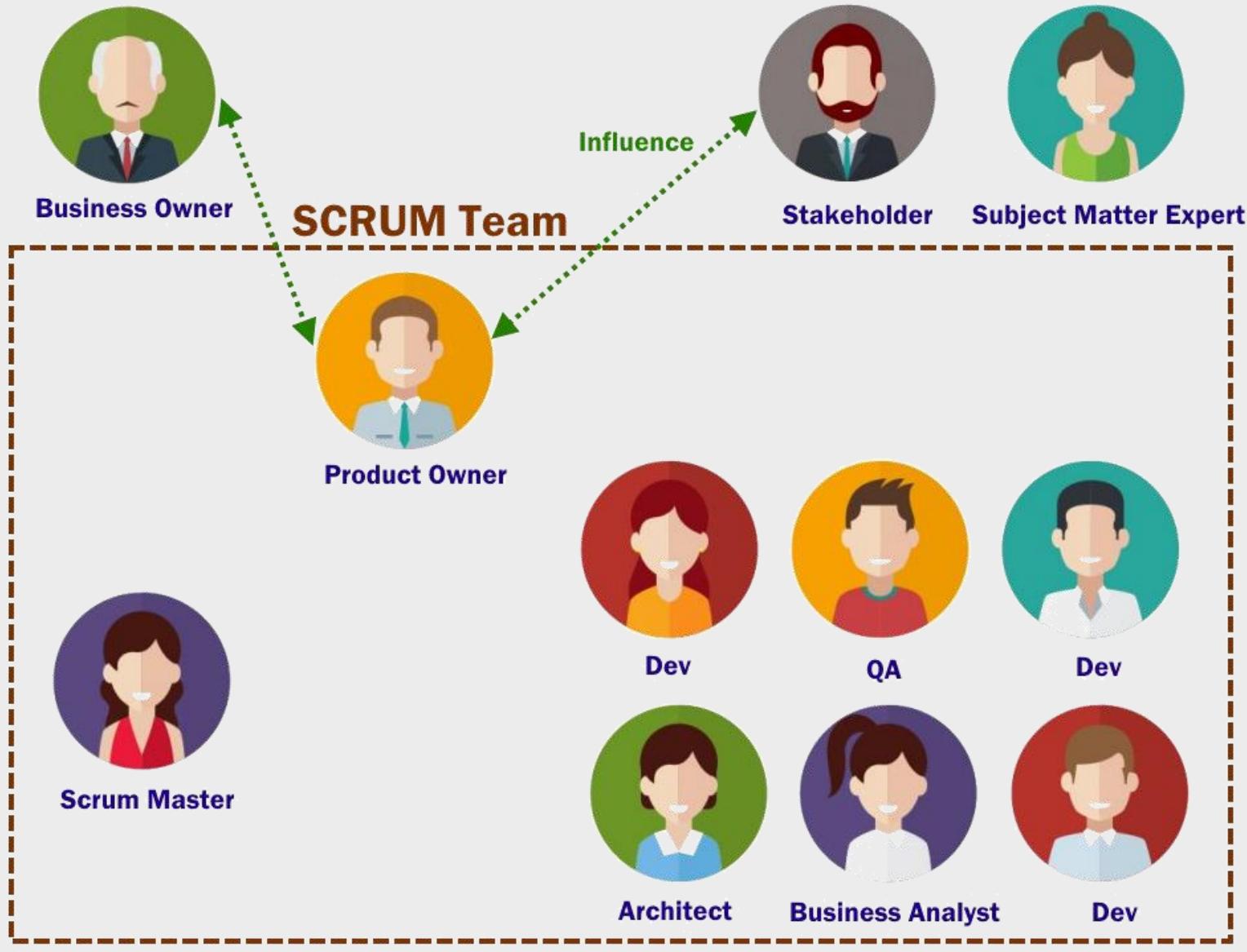
凤凰项目 (The Phoenix Project) —— 敏捷基础



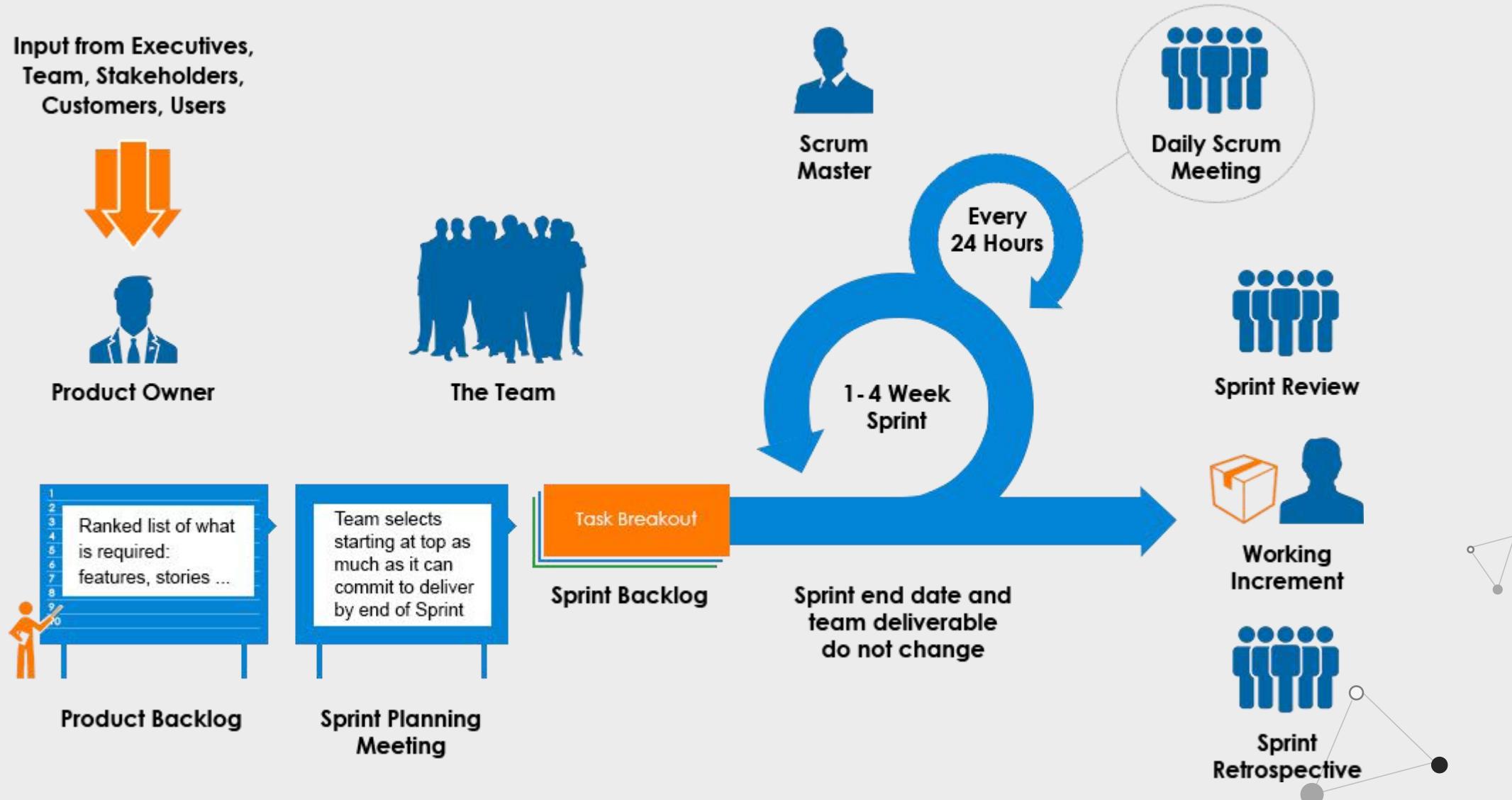
凤凰项目 (The Phoenix Project) —— 敏捷基础



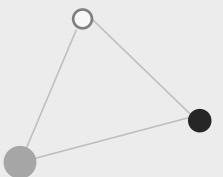
凤凰项目 (The Phoenix Project) — 敏捷基础



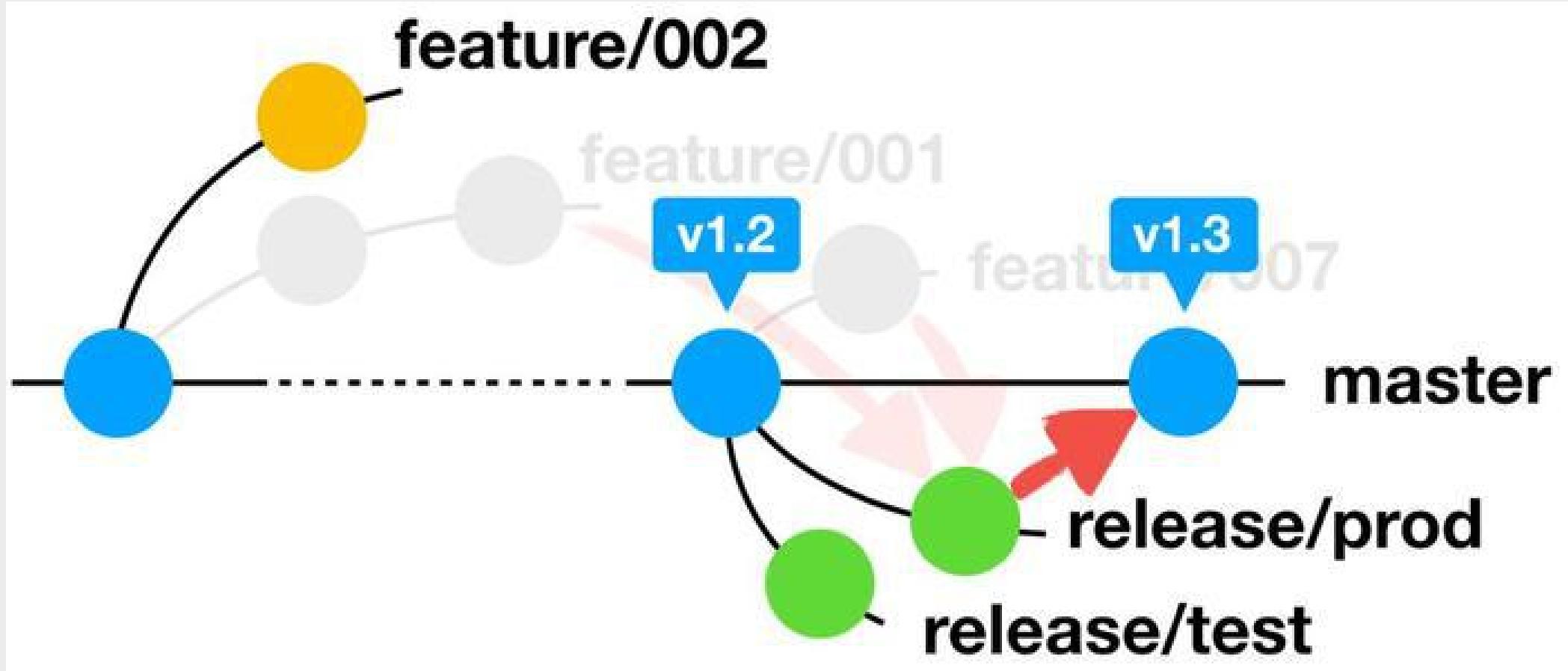
凤凰项目 (The Phoenix Project) — 敏捷基础



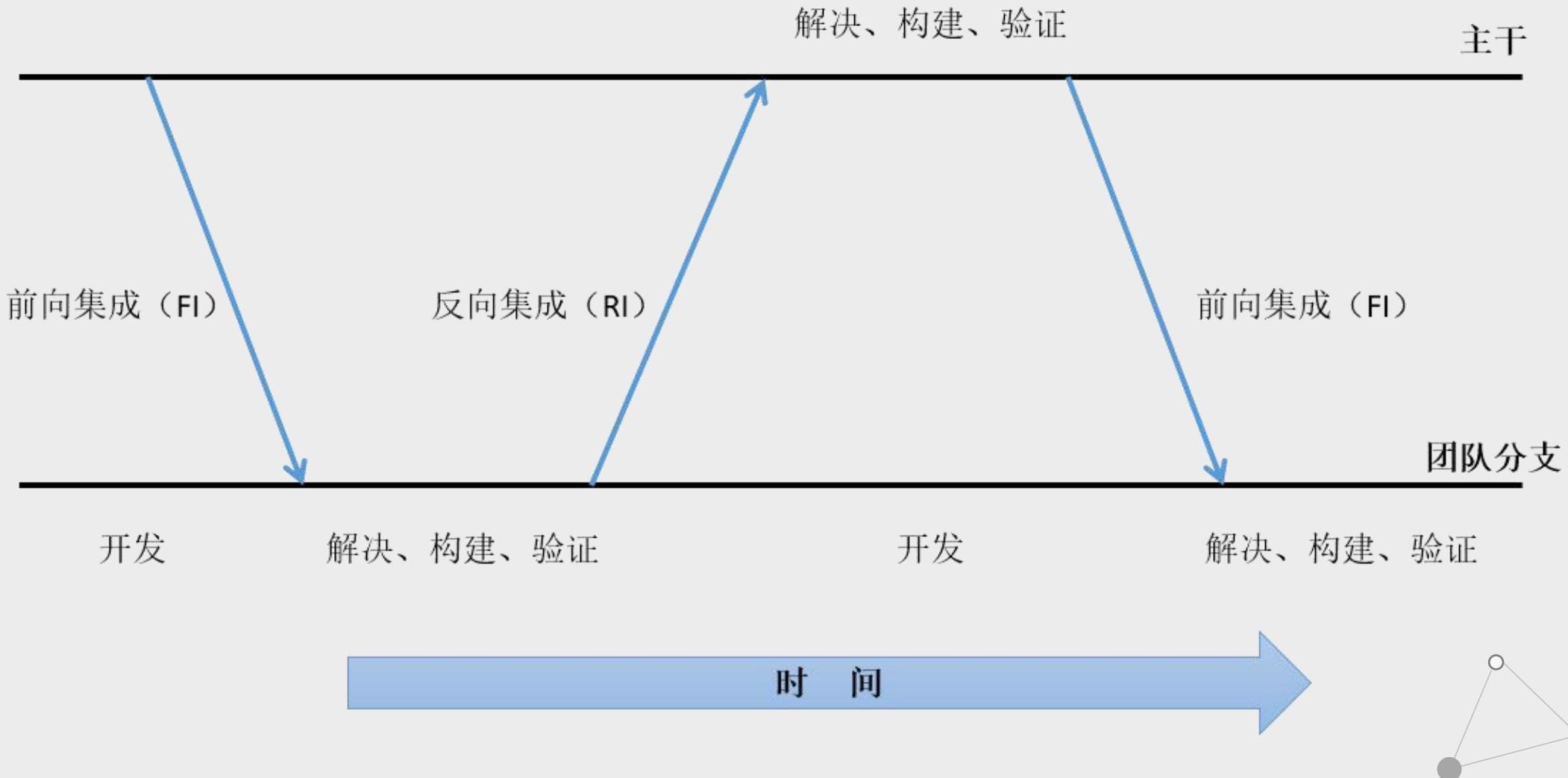
4、部署流水线



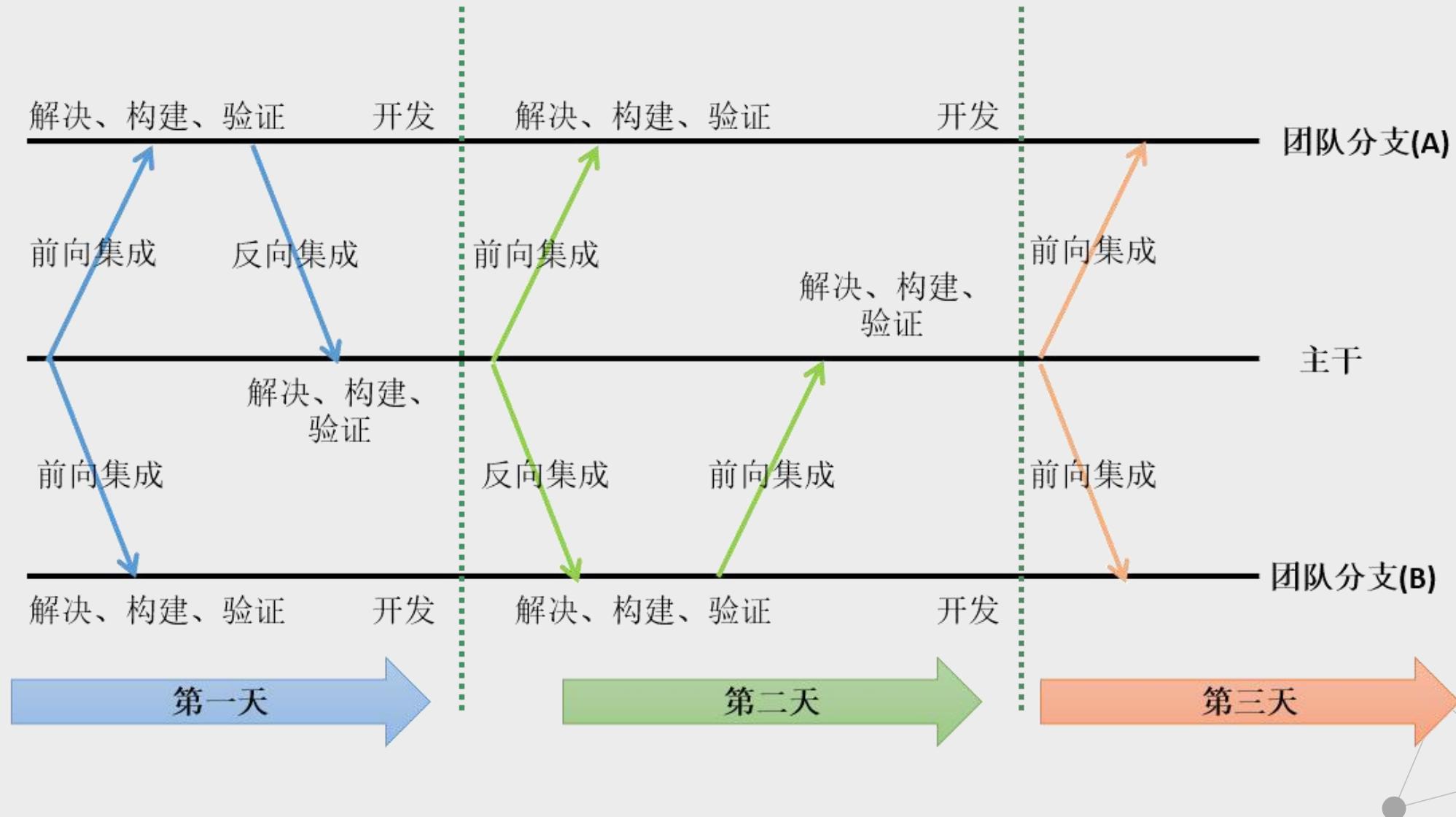
凤凰项目 (The Phoenix Project) —— 部署流水线



凤凰项目 (The Phoenix Project) —— 部署流水线

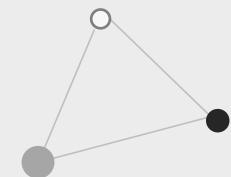
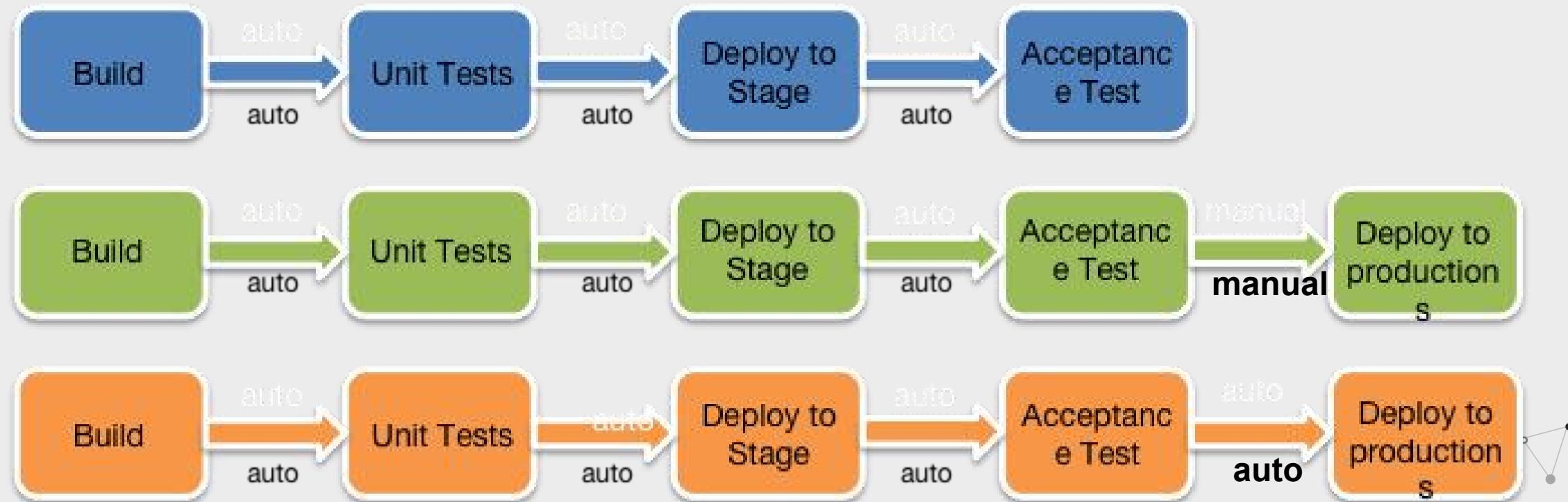


凤凰项目 (The Phoenix Project) —— 部署流水线



凤凰项目 (The Phoenix Project) —— 部署流水线

持续集成、持续交付、持续部署 三者的区别



凤凰项目 (The Phoenix Project) —— 部署流水线

持续集成 (CI) 需要具备的必要条件



全面自动化测试



灵活的基础设施



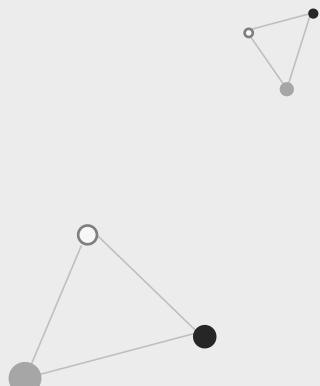
版本控制工具



自动化的构建和
软件发布流程的
工具

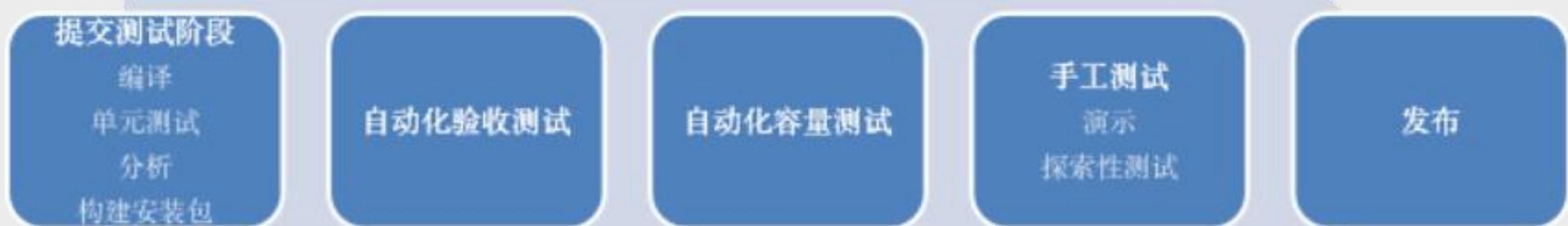


反馈机制

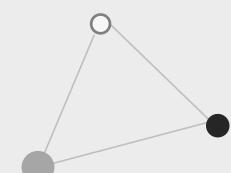
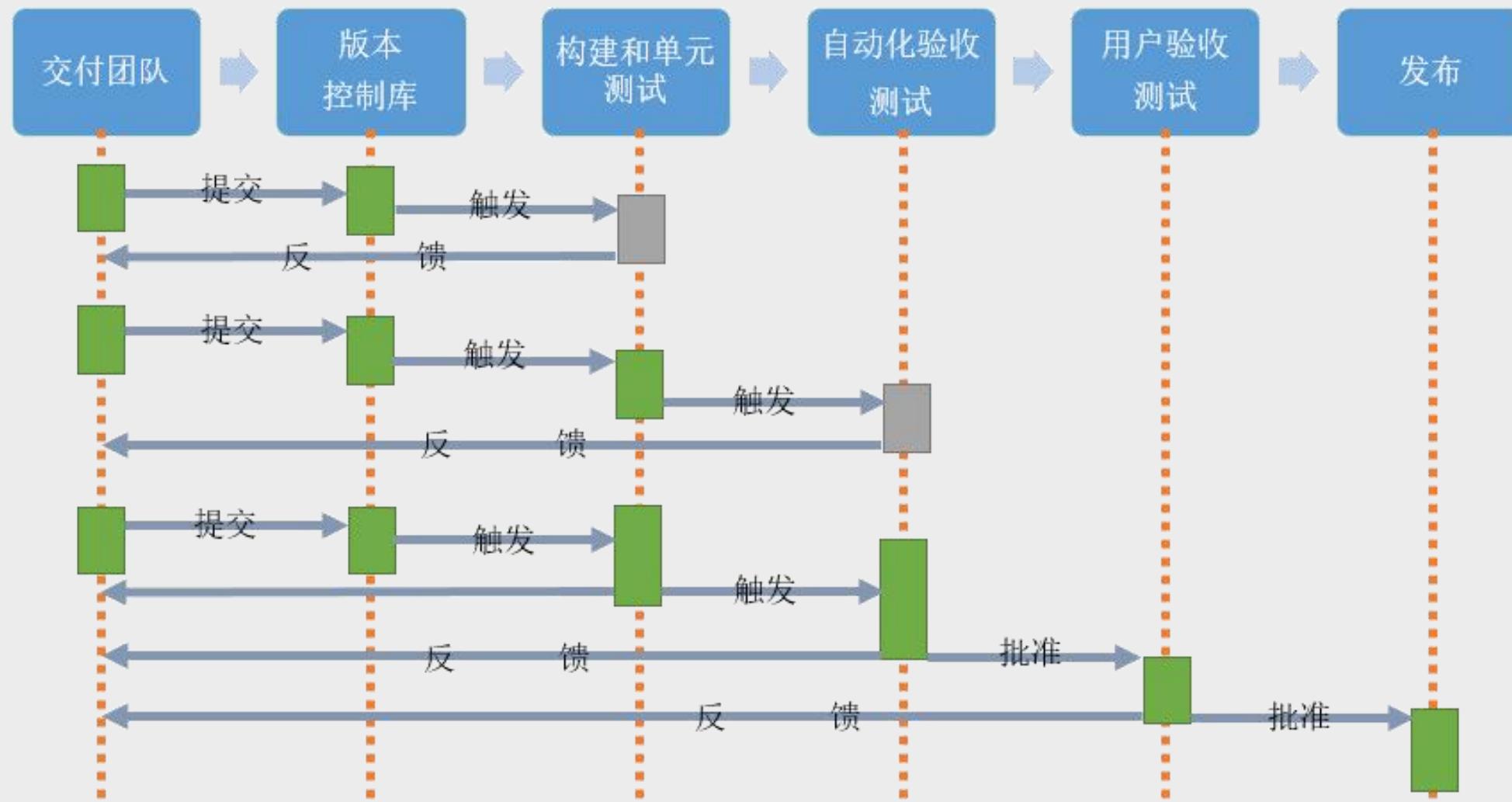


凤凰项目 (The Phoenix Project) —— 部署流水线

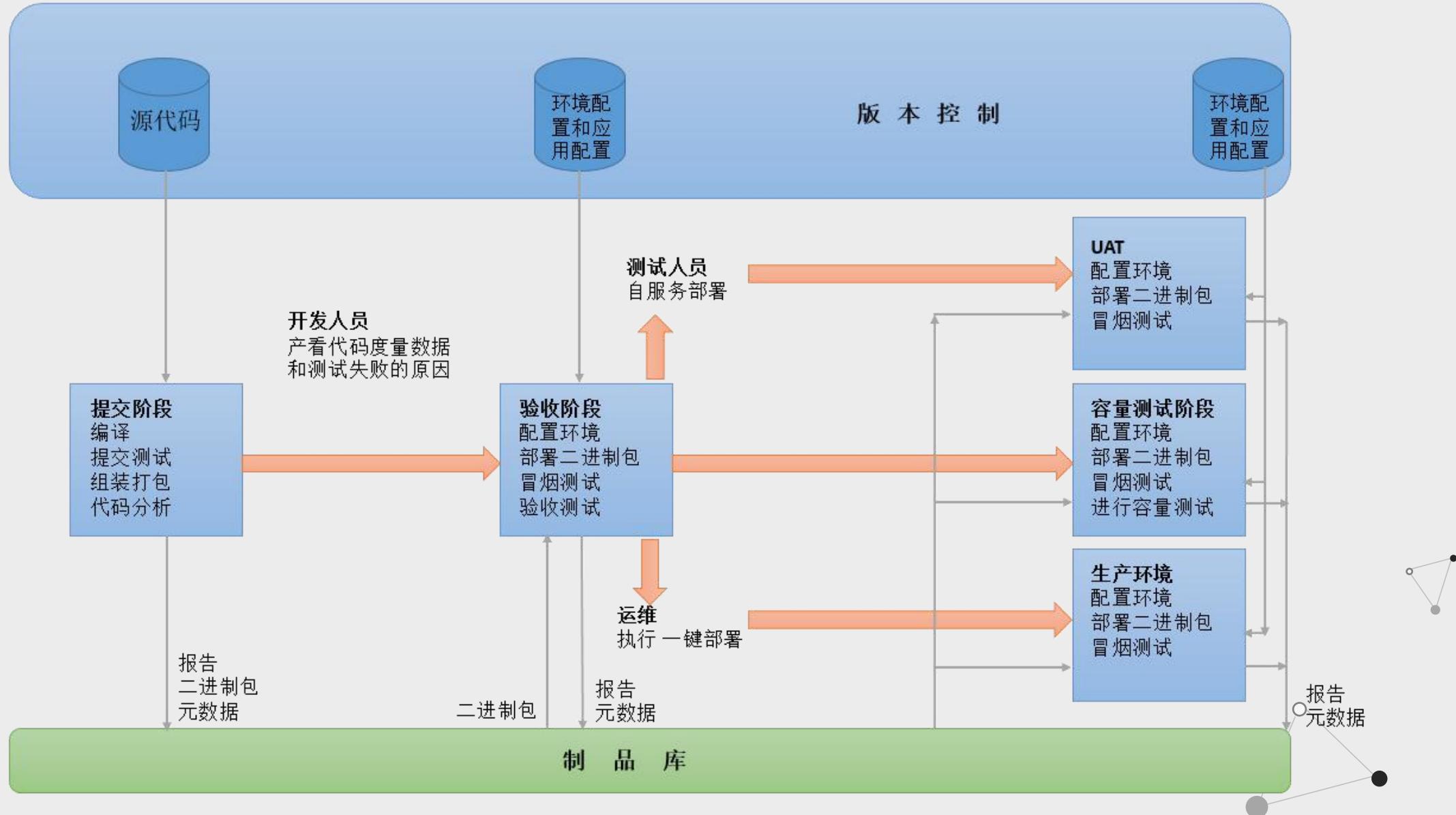
DevOps部署流水线



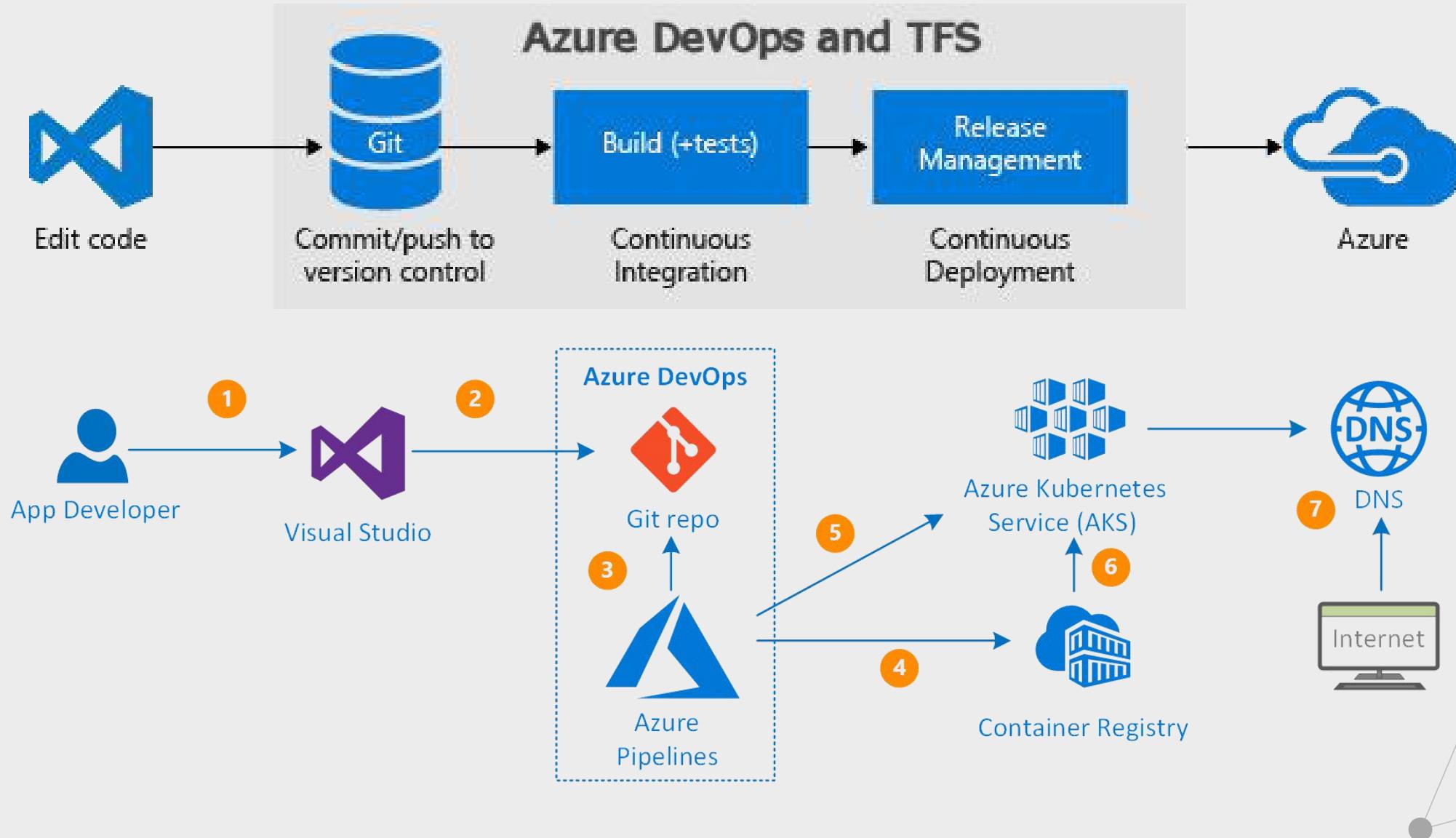
凤凰项目 (The Phoenix Project) —— 部署流水线



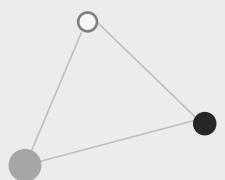
凤凰项目 (The Phoenix Project) —— 部署流水线



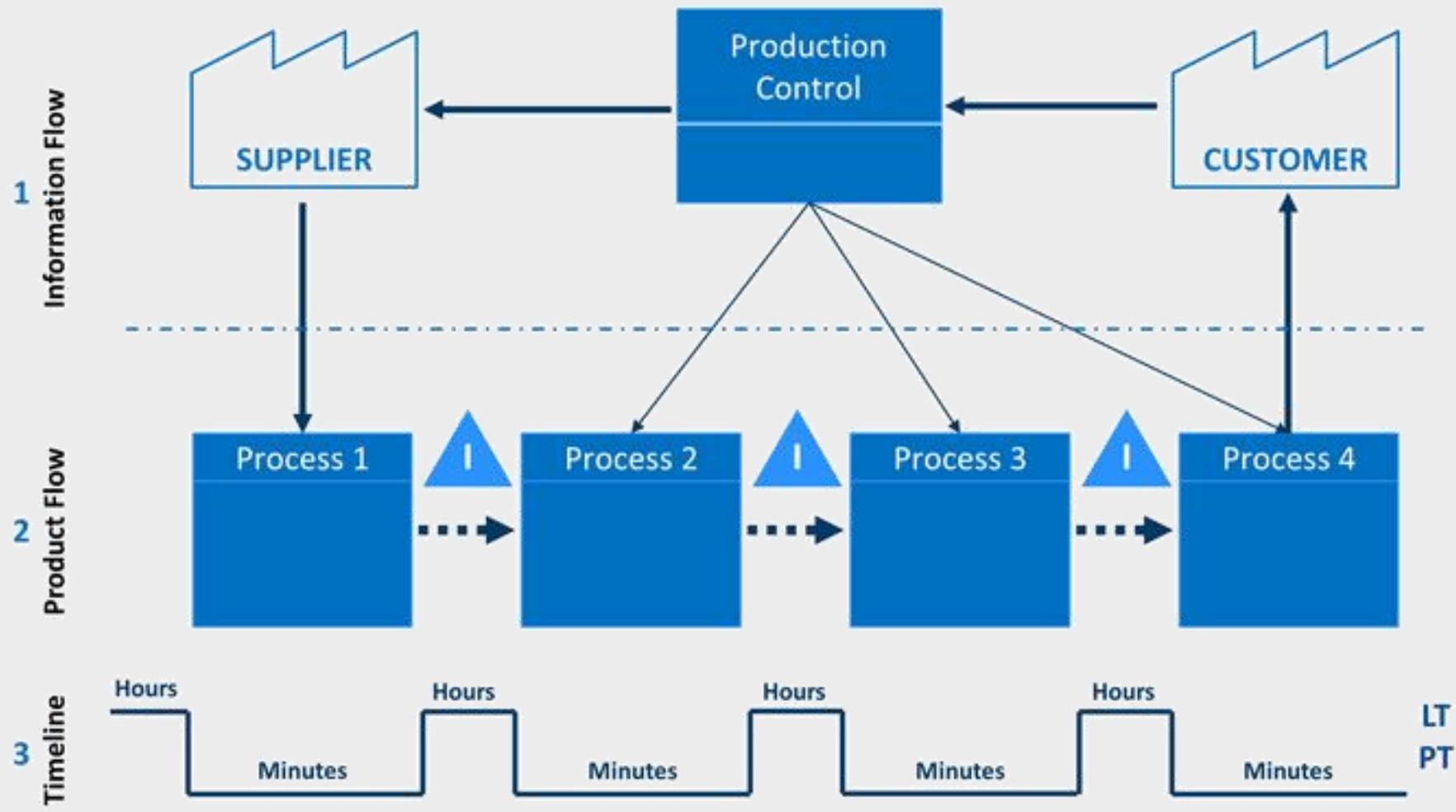
凤凰项目 (The Phoenix Project) — 部署流水线



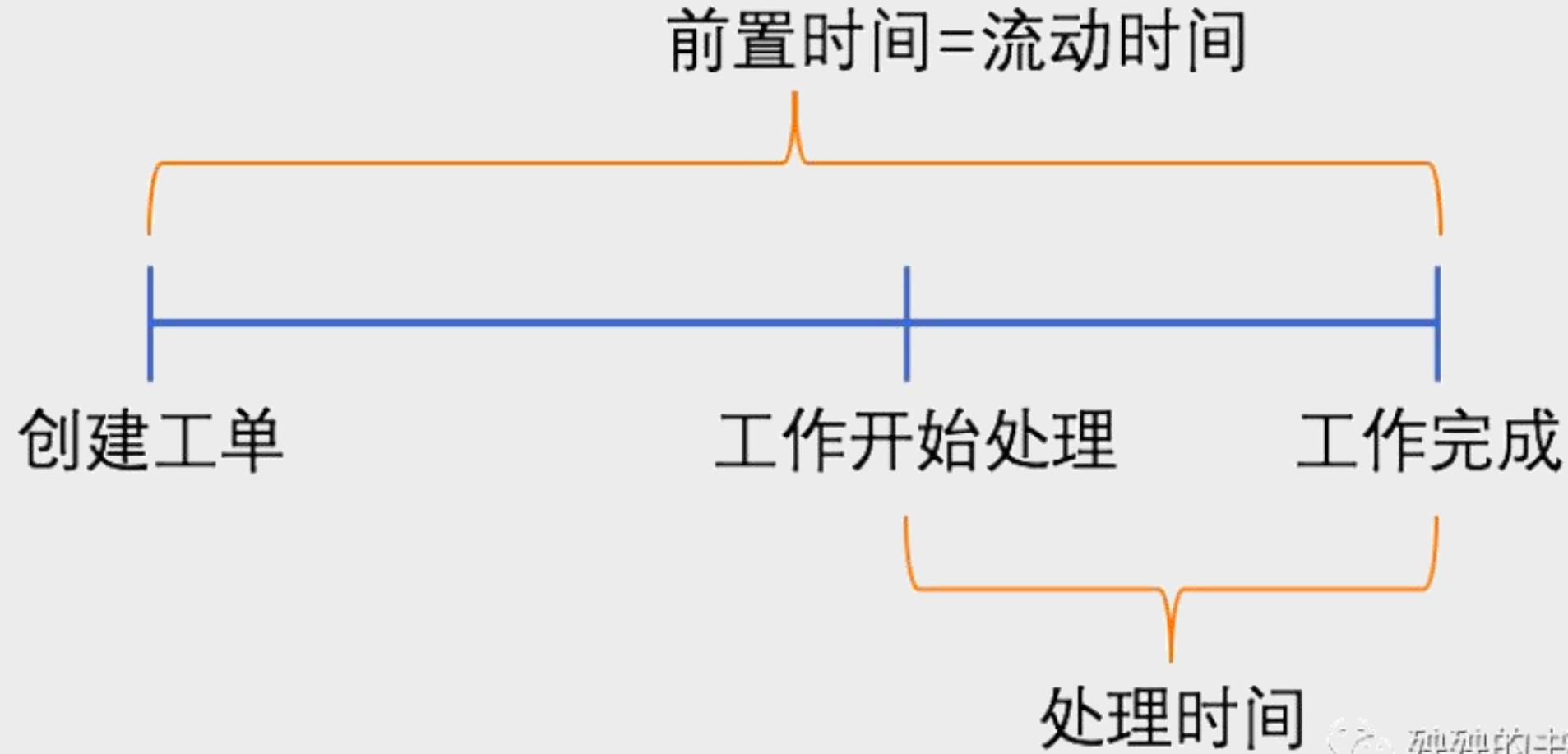
5、价值流



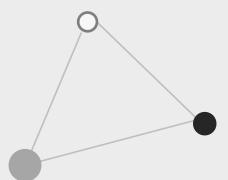
凤凰项目 (The Phoenix Project) —— 价值流



凤凰项目 (The Phoenix Project) —— 价值流

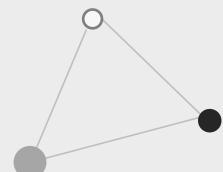


6、可视化



凤凰项目 (The Phoenix Project) —— 可视化

你是否也受困于不停地写“周报” & “月报”？
你觉得团队上报给你的“周报” & “月报”反映的是真实的项目的进展？



凤凰项目 (The Phoenix Project) —— 可视化

敏捷板 | 非公开 | 邀请 | 凤凰 A B BL +22 | 显示菜单

要做的事情

- 英文字幕SRT校对-已完成
- 翻译规范制定
- 百度云链接 - 视频、字幕档、讲稿
- Aegisub 使用心得
- 英文字幕校对
- 翻译公约
- 翻译校对

+ 添加另一张卡片

中文SRT翻译-进行中

- 1.0.2 - The Impact of Professional Coaching
- 1.0.4 - Sample Coaching Session
- 1.1.1 - Qualities of Transformational Questions
- 1.1.2 - The Value of Non-Leading Questions
- 1.2.4 - How Sharing Can Promote Forward Momentum
- 1.2.5 - Sharing Without Being Attached
- 1.2.6 - Mirroring the Client's Way of Speaking
- 1.3.5 - Working Towards the Client's Stated Outcomes

+ 添加另一张卡片

中文SRT翻译-已完成

- 1.3.6 - Developing Relationship and Session Coaching Agreements
- 1.4.5 - Noticing and Reflecting on Client Progress
- 1.4.6 - Seeing the Client as Fully Competent

+ 添加另一张卡片

中文翻译初审-进行中

- 1.2.2-中英文双语文字版
- 1.2.1-中英文双语文字版
- 1.3.1 - 中英文双语文字版
- 1.3.2 - Defining Measures of Success with the Client
- 1.3.3 - Exploring What is Meaningful to the Client
- 1.3.4 - Defining What to Address or Resolve
- 1.4.4 - Challenging the Client

+ 添加另一张卡片

中文翻译初审-完成

- 1.1.1-中英文双语文字版
- 1.1.2-中英文双语文字版
- 1.1.3-中英文双语文字版
- 1.4.1 - Acknowledging the Client's Work in the Coaching Process
- 1.4.2 - Respecting the Client's Work
- 1.4.3 - Expressing Support for the Client

+ 添加另一张卡片

中文翻译终审-进行中

- + 添加卡片

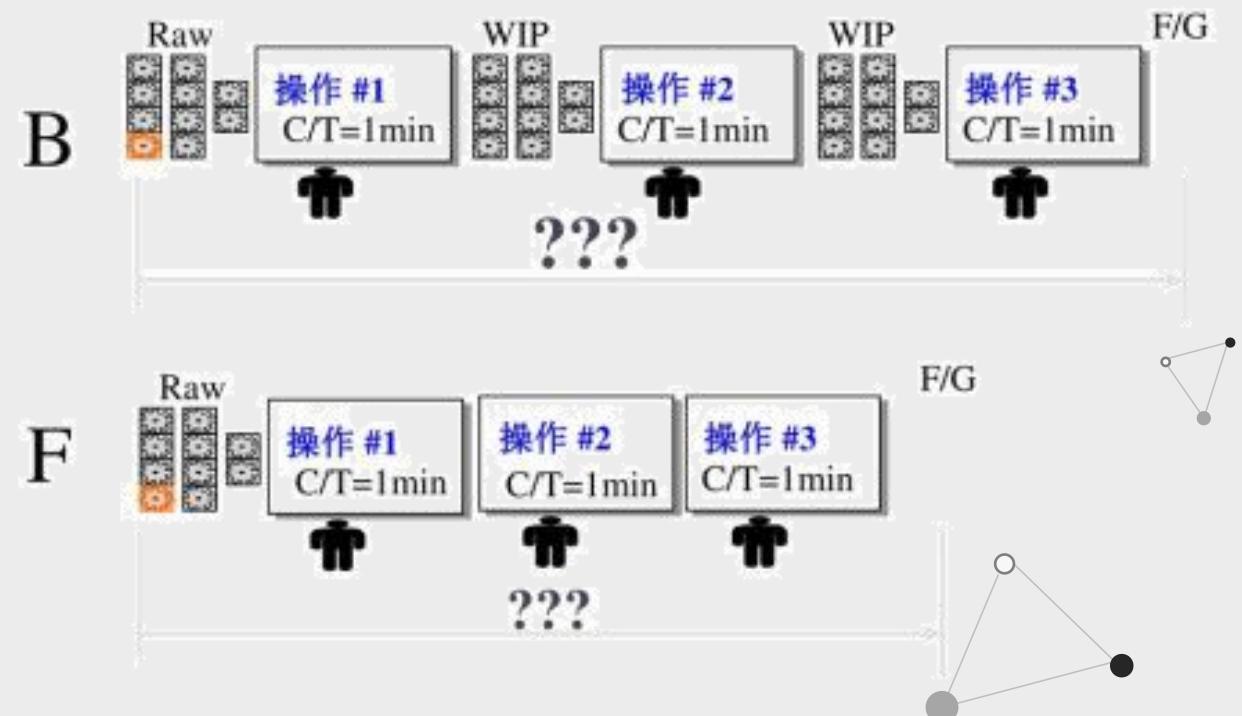
Butler

凤凰项目 (The Phoenix Project) —— 可视化

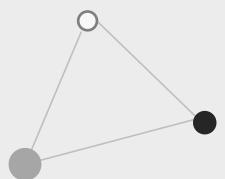
看板 (kanban) 系统的三要素之 在制品限制 (WIP) : 利特尔法则 (Little's Law)

Lead Time (产出时间) = 存货数量 × 生产节拍

(解释: 吞吐量一定时, 任务越多, 速率就越慢 → → → 等待时间就越长)

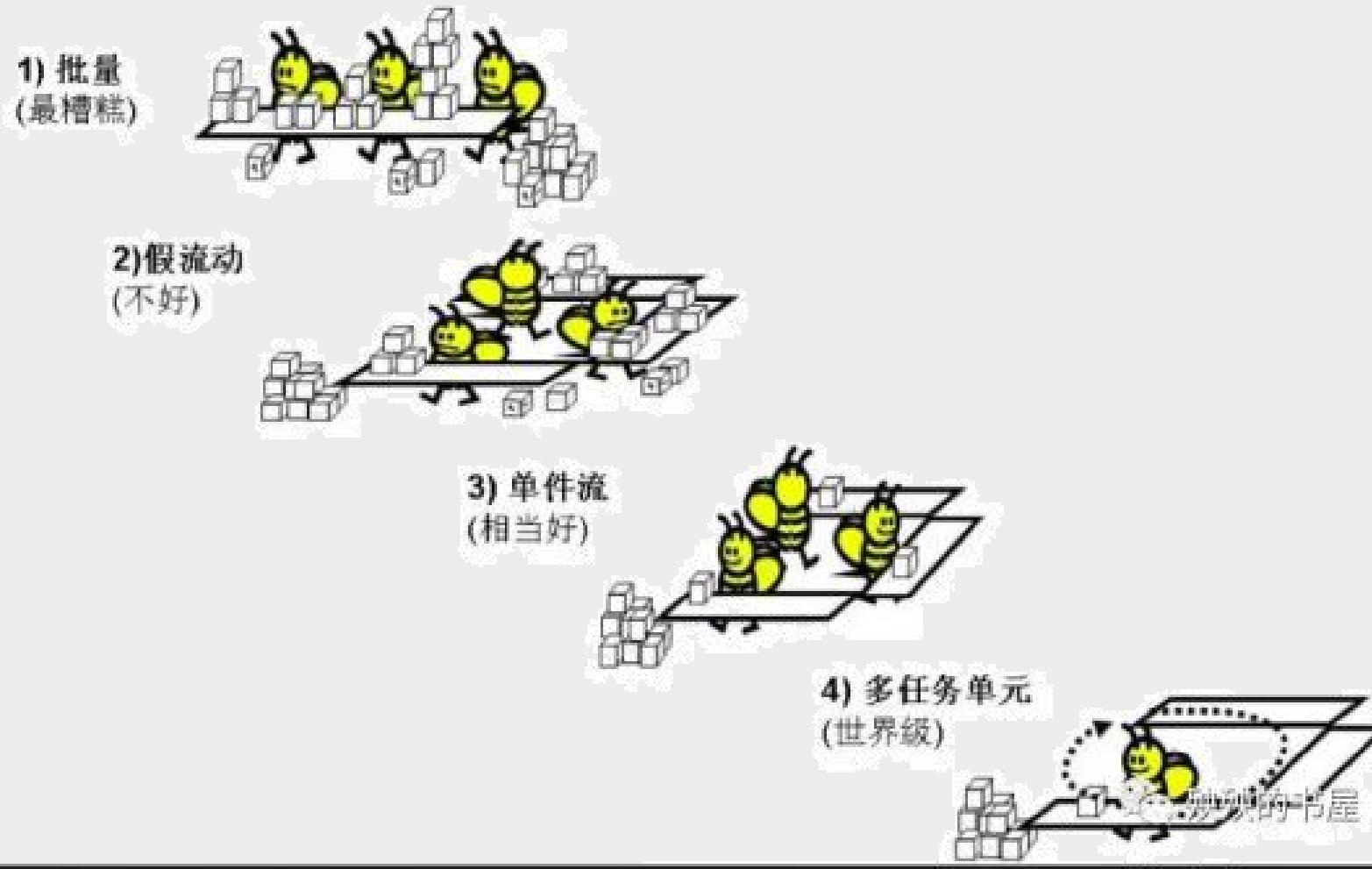


7、单件流



凤凰项目 (The Phoenix Project) —— 单件流

单件流 (one-piece-flow)



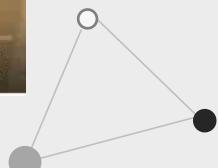
凤凰项目 (The Phoenix Project) —— 可视化

单件流 (one-piece-flow)

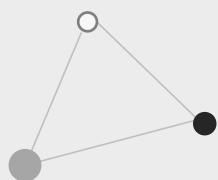
- 个体或团队一次性的完成一个事项，保证快速、流畅的流动。
- 单件连续流可以紧密合作的单个团队中发生。在一个工作团队（或Scrum团队）中，对在制品进行多次转换（每次只局限于单件制品）。团队在非常短的周期做了一点分析，一个小的设计、一个小的构建和一个小的测试。个人是全栈的。



凤凰项目 (The Phoenix Project) —— 单件流



8、测试驱动

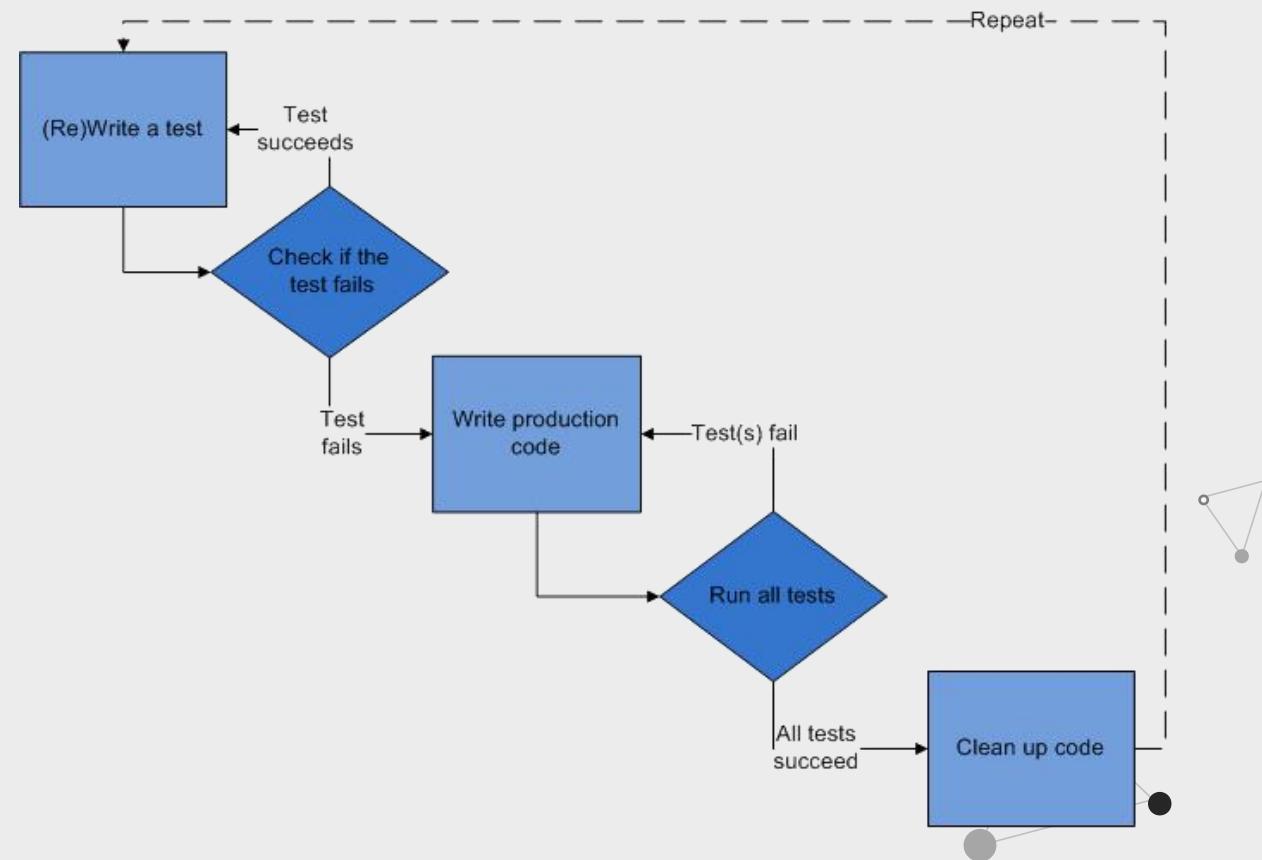


凤凰项目 (The Phoenix Project) —— 测试驱动

测试驱动开发倡导在开发之前通过一个测试来获得反馈。测试驱动开发倡导的理念是要在开发之前编写测试代码。因为开发人员实际开发之前要先清楚功能是如何被测试的。

测试驱动开发流程：

- 编写测试用例
- 运行测试用例
- 编写项目代码
- 运行测试用例
- 重构代码



凤凰项目 (The Phoenix Project) —— 测试驱动

测试驱动开发的优点：

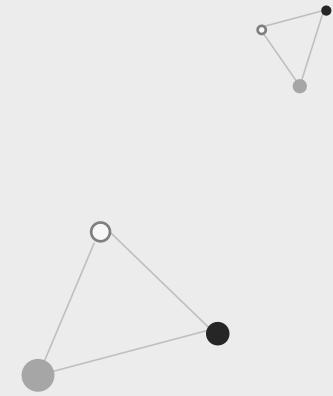
- 有助于设计简单、清晰而易用的接口
- 模块切分的足够小，但是模块之间保持极低的耦合度
- 肆无忌惮的重构
- 测试代码是“活”的软件文档，它硬性规定了实现代码必须满足需求，达不到就报错

测试驱动力所不及的地方：

- 测试驱动开发不可能让人立即具有设计出优美解决方案的能力，或者说是优秀的分析与解决问题的能力
- 测试驱动开发不能接受研发投入，也能很少能解决开发周期
- 测试驱动开发不能杜绝所有软件缺陷



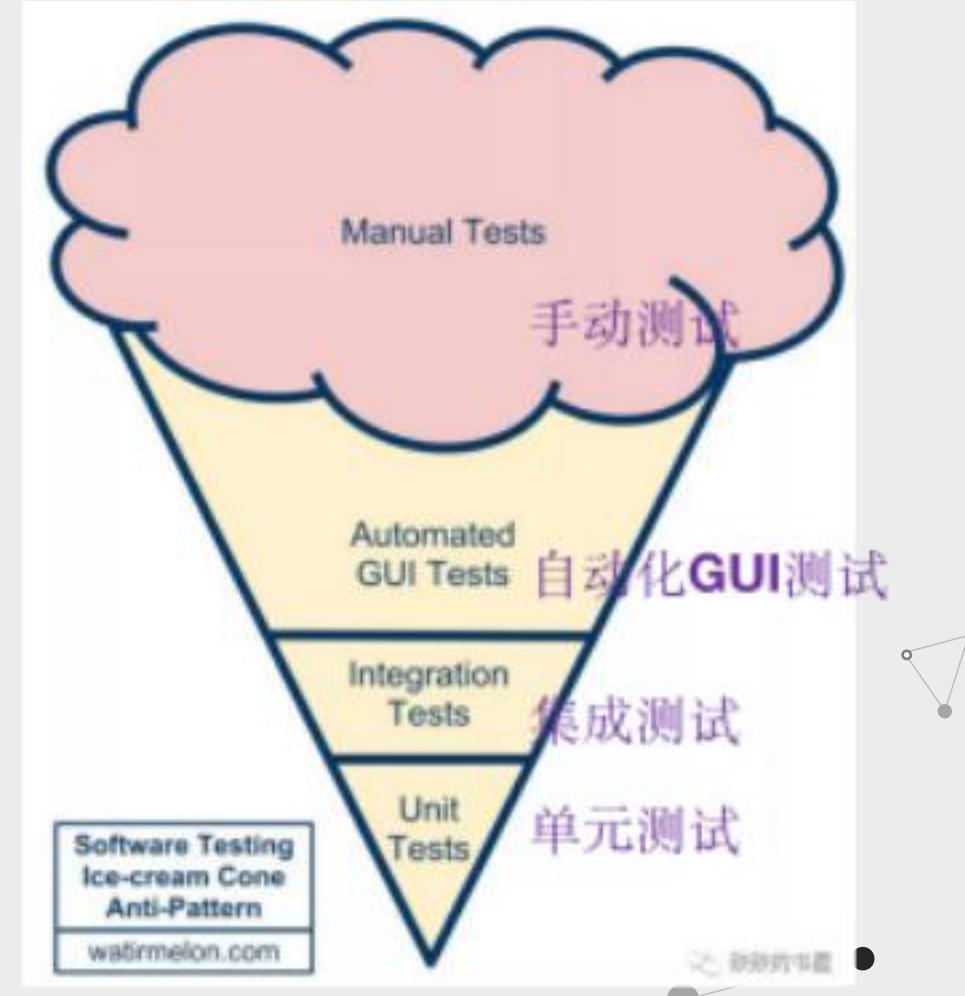
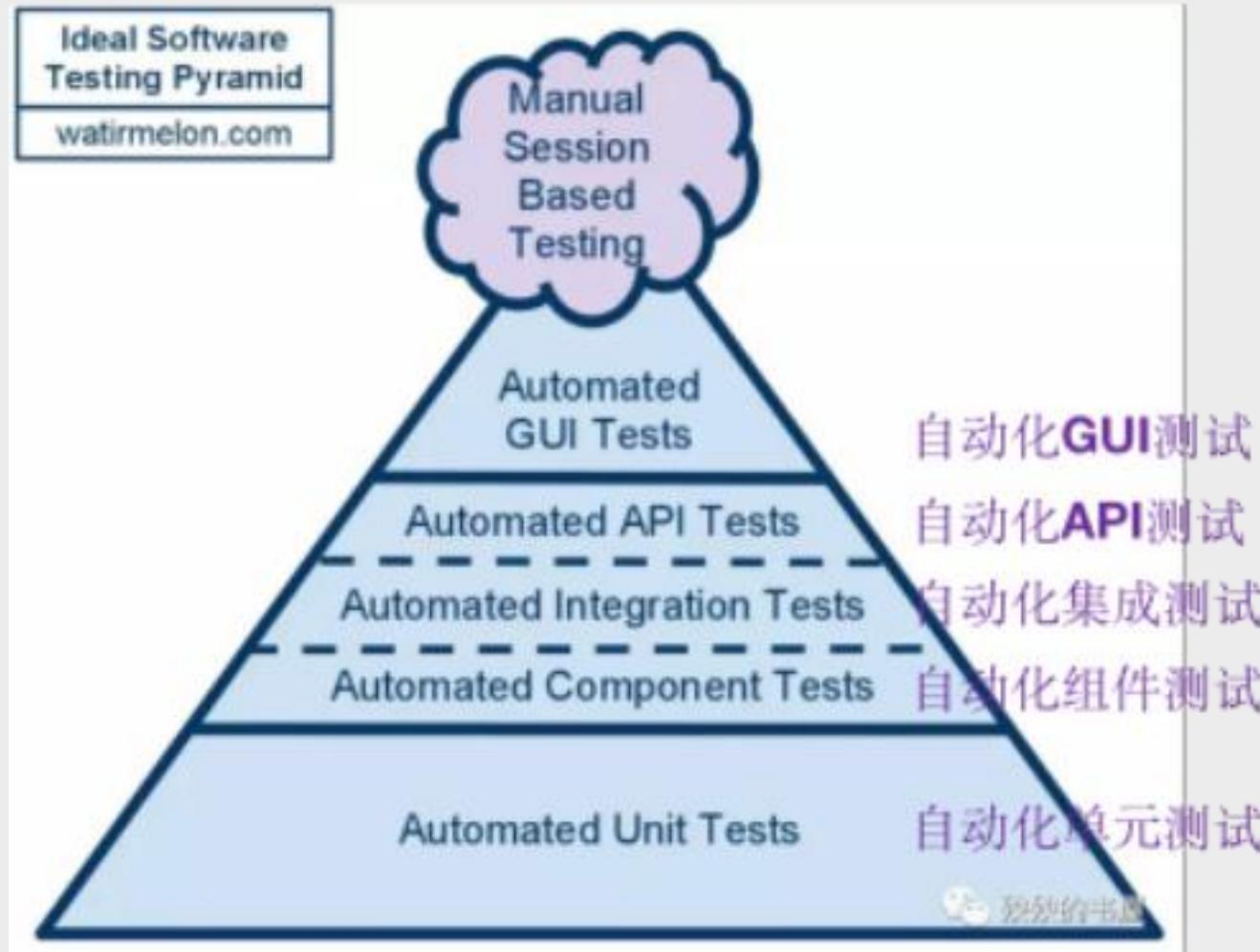
凤凰项目 (The Phoenix Project) —— 测试驱动



凤凰项目 (The Phoenix Project) —— 测试驱动



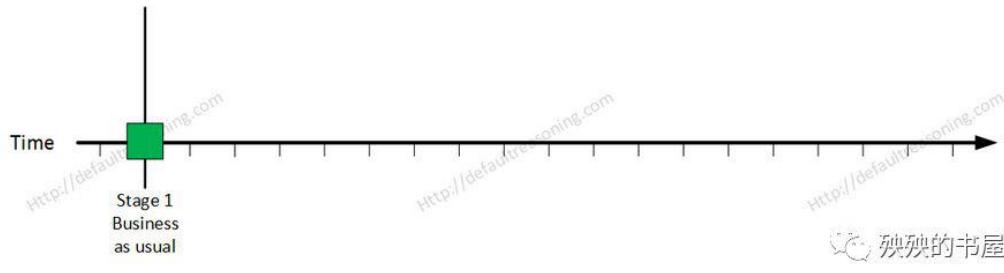
凤凰项目 (The Phoenix Project) —— 测试驱动



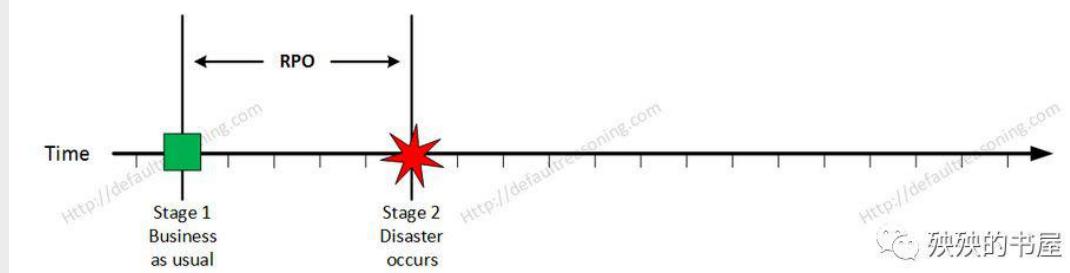
凤凰项目 (The Phoenix Project) —— 测试驱动

RPO & RTO

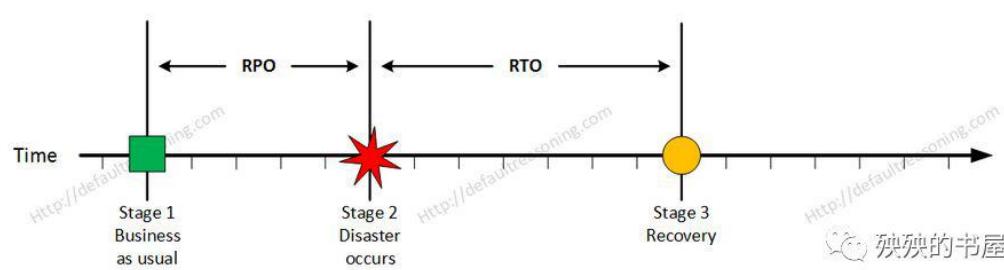
Stage 1: Business as usual



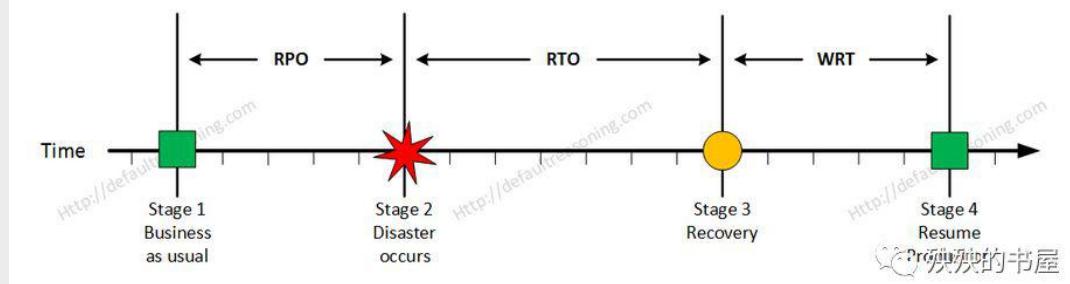
Stage 2: Disaster occurs



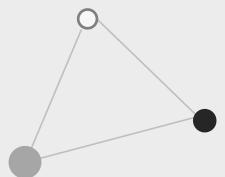
Stage 3: Recovery



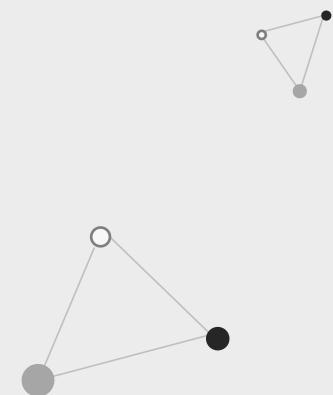
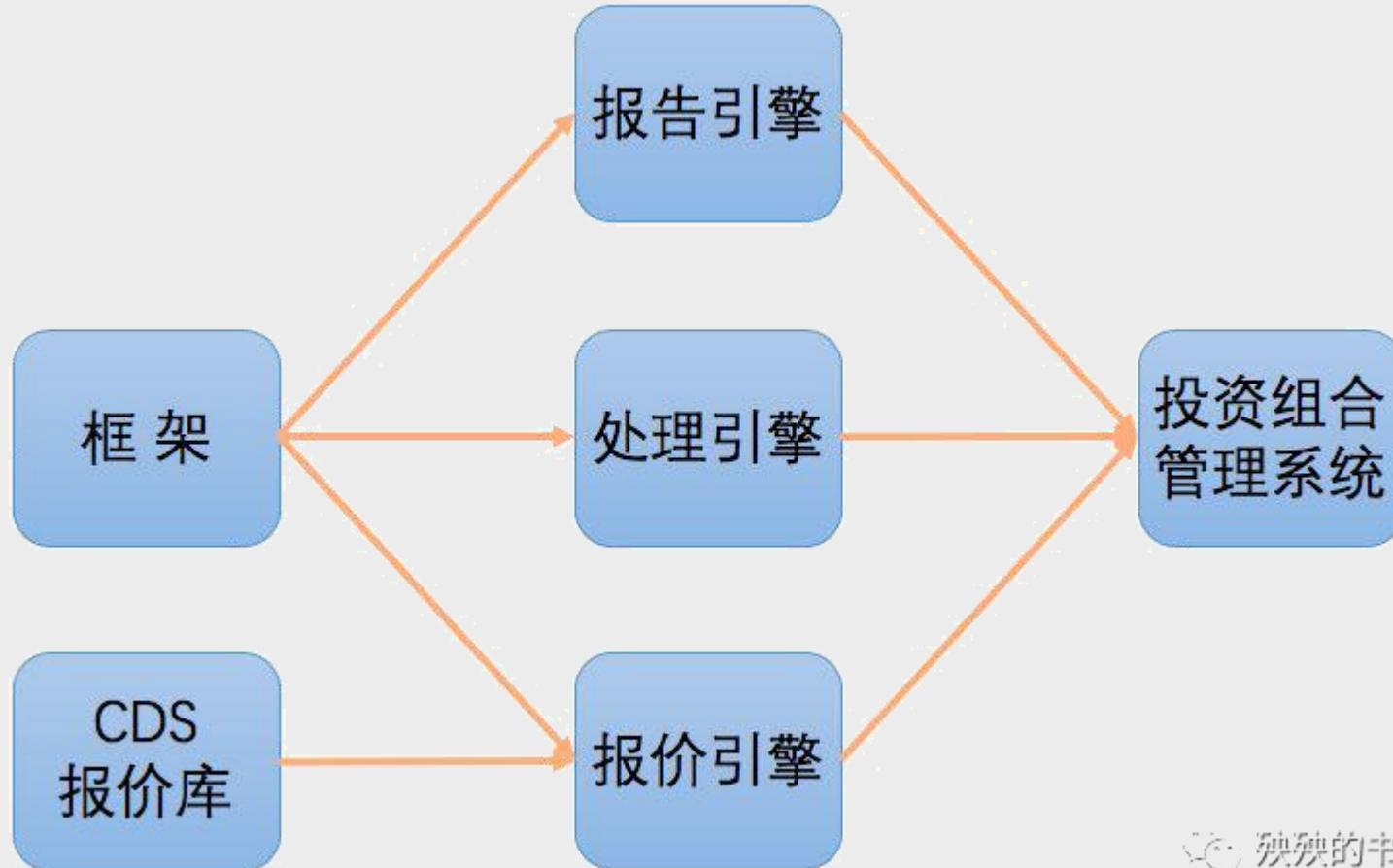
Stage 4: Resume Production



9、管理依赖

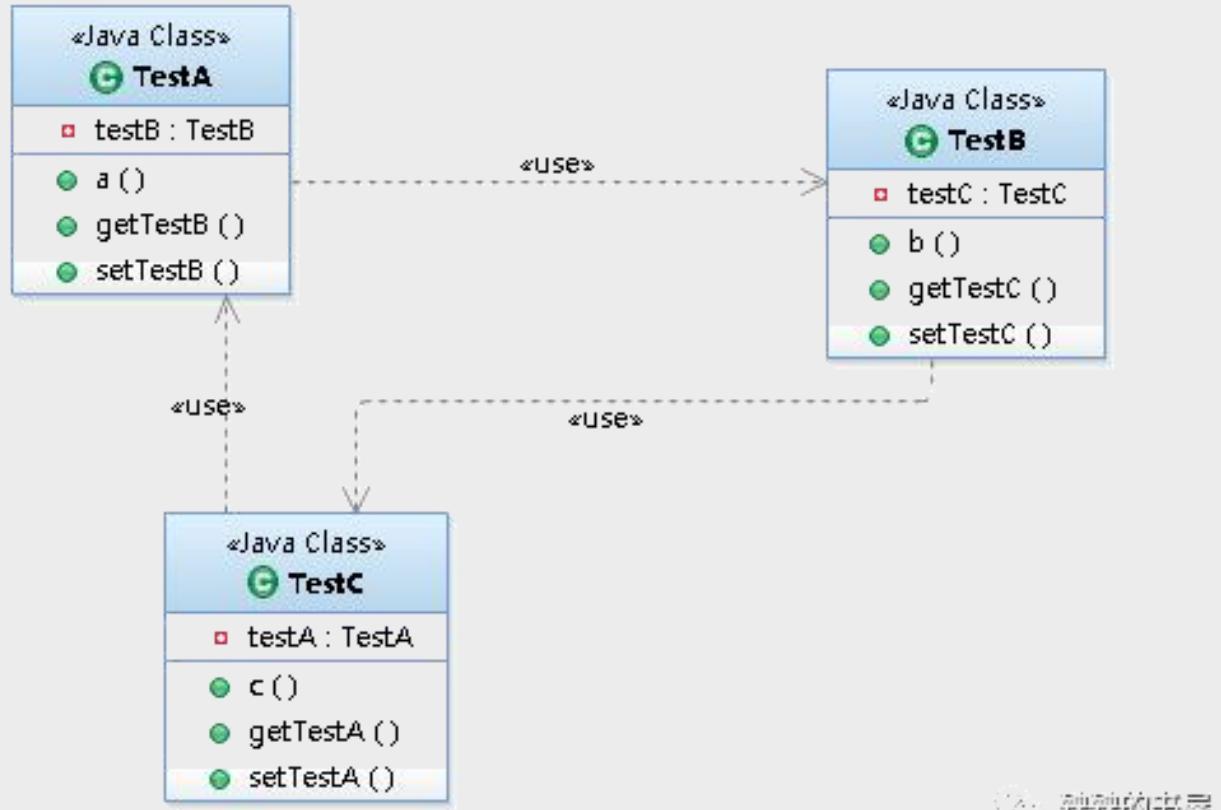


凤凰项目 (The Phoenix Project) —— 管理依赖



凤凰项目 (The Phoenix Project) —— 管理依赖

循环依赖 & 依赖地狱 & 构建阶梯



© 沫沫的书屋

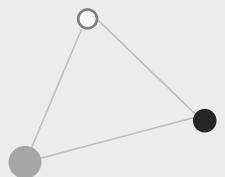
组件A 组件B



© 沫沫的书屋

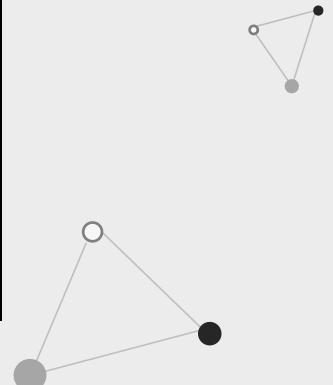
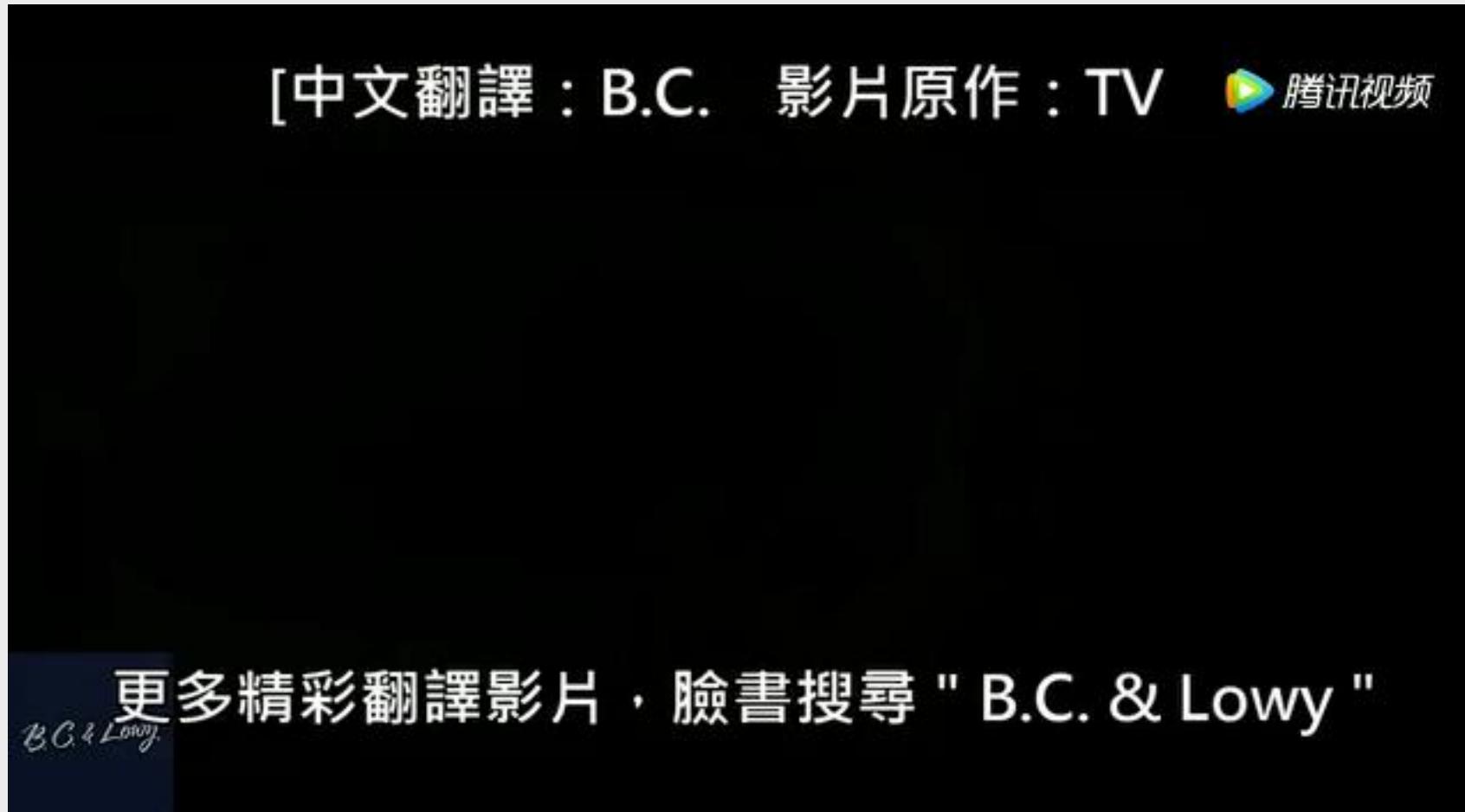


10、规模化

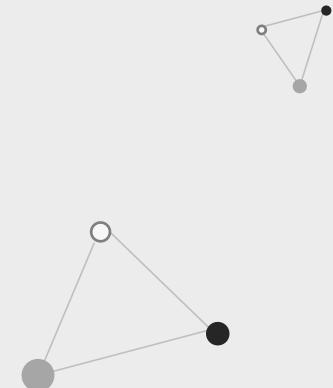
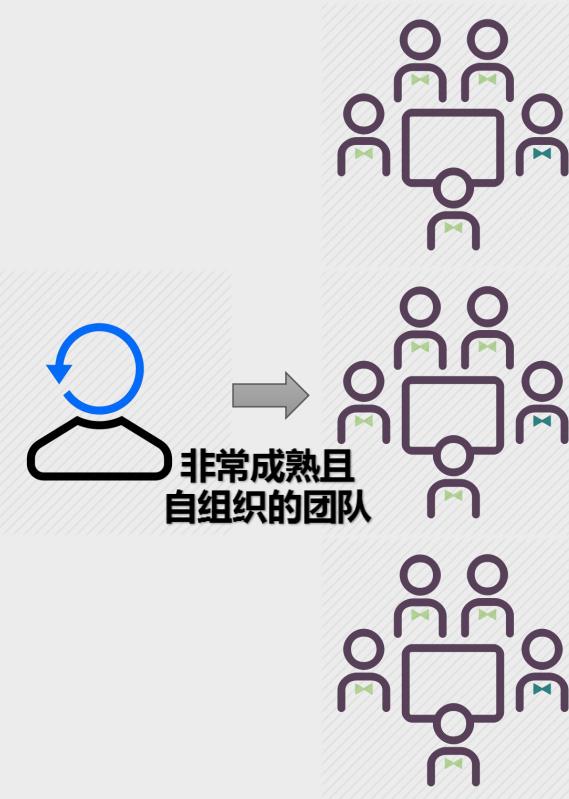
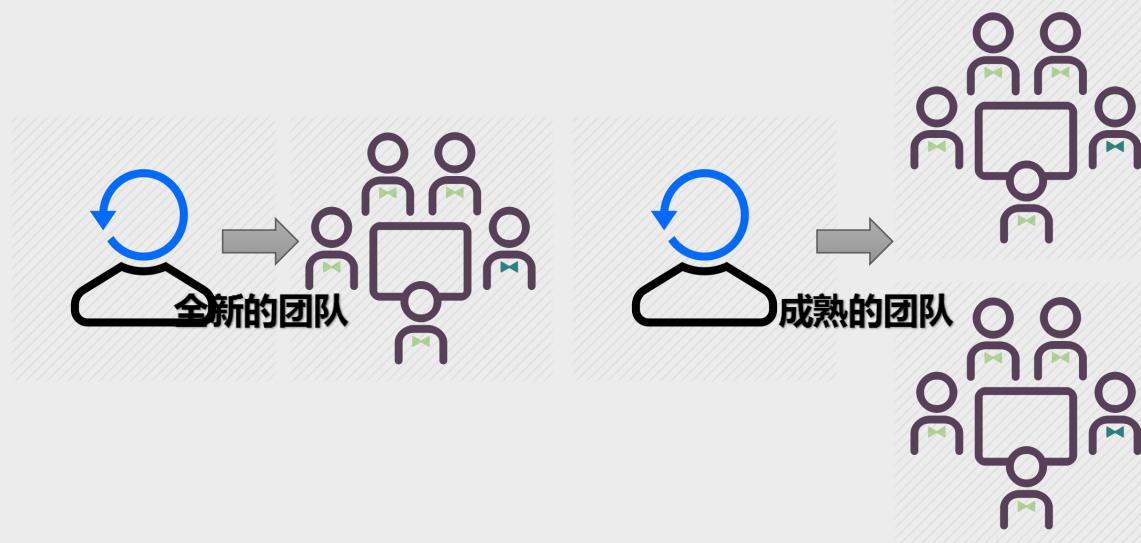


凤凰项目 (The Phoenix Project) —— 规模化

多元化的团队



凤凰项目 (The Phoenix Project) —— 规模化

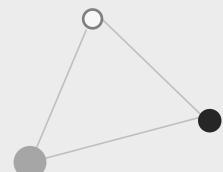


守破離

Shu

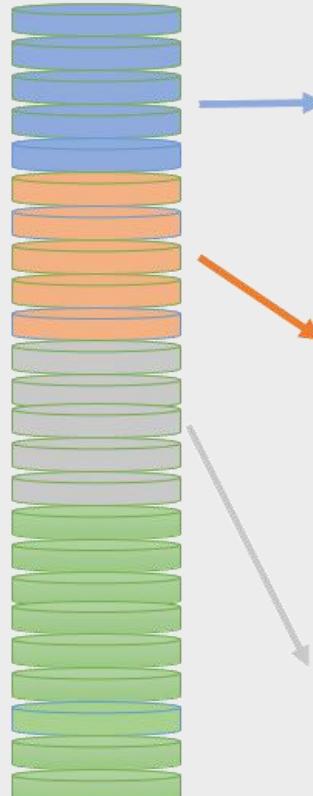
Ha

Ri

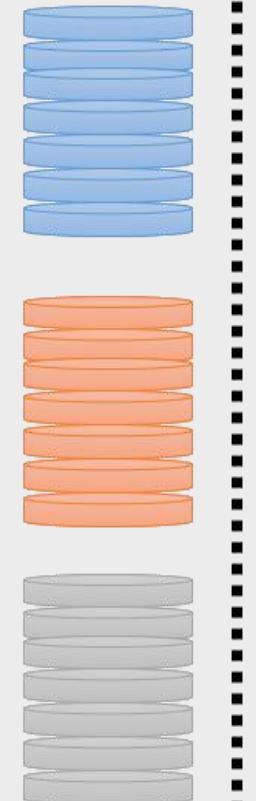


凤凰项目 (The Phoenix Project) —— 规模化

(总) 产品列表



各团队自己的产品列表



版本规划

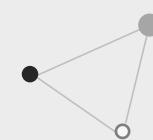
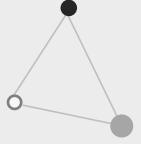


检视和调整

时间

02

什么是DevOps ?



什么是DevOps

“**DevOps** is development and operations collaboration”

“**DevOps** is using automation”

“**DevOps** is small deployments”

It's DevOps!

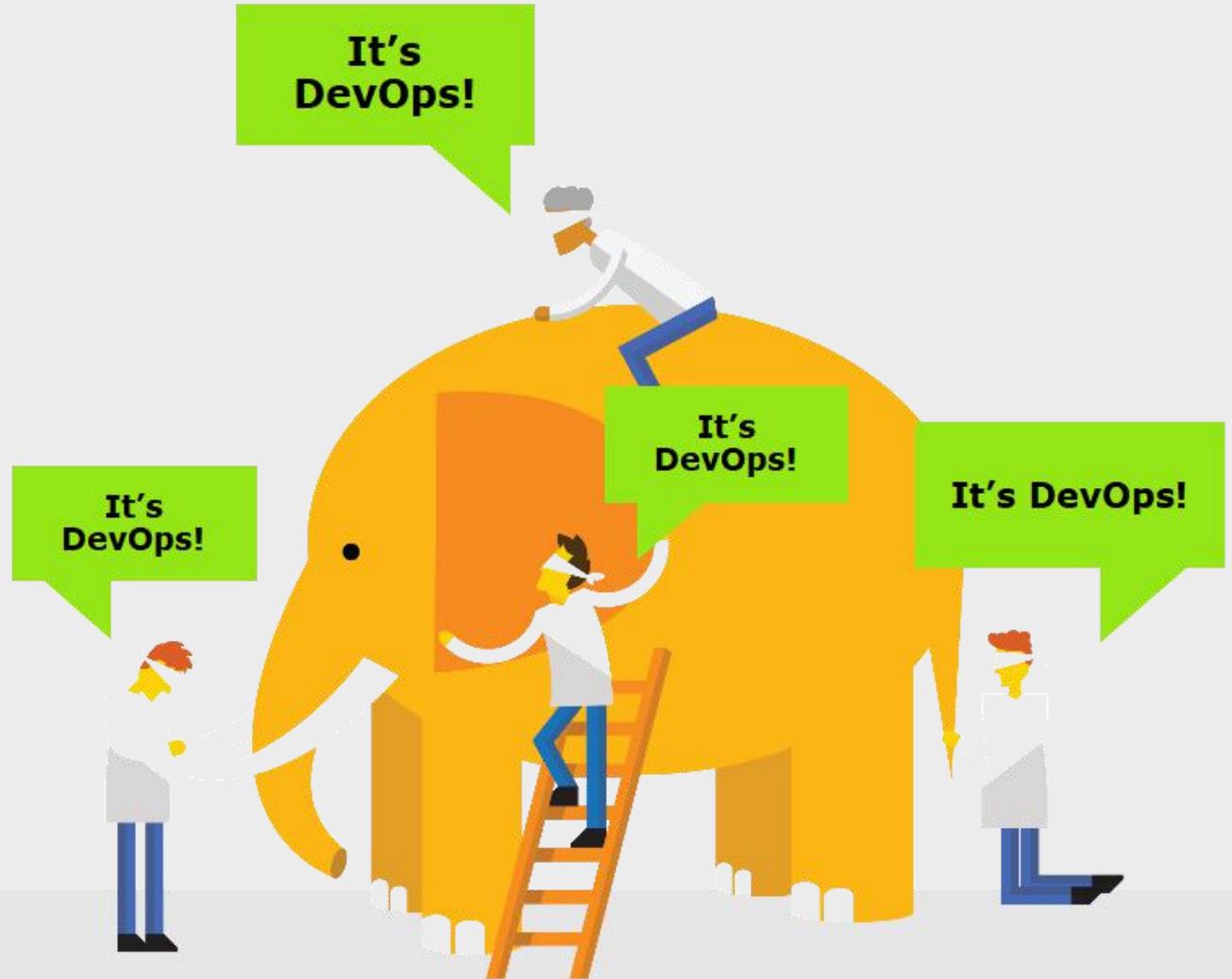
It's DevOps!

It's DevOps!

“**DevOps** is treating your infrastructure as code”

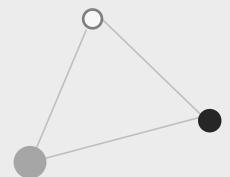
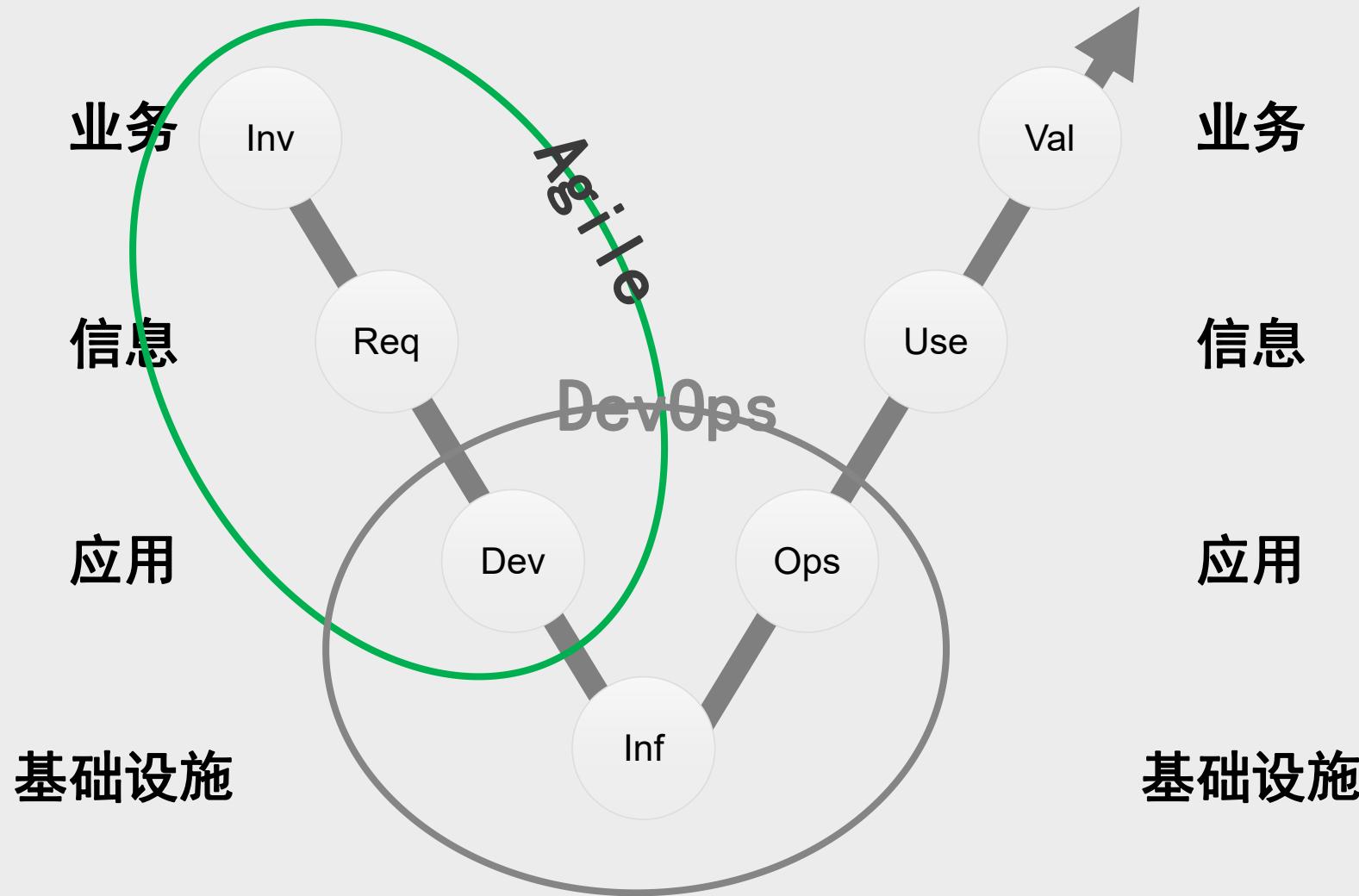
“**DevOps** is feature switches”

“**DevOps** is continuous monitoring”

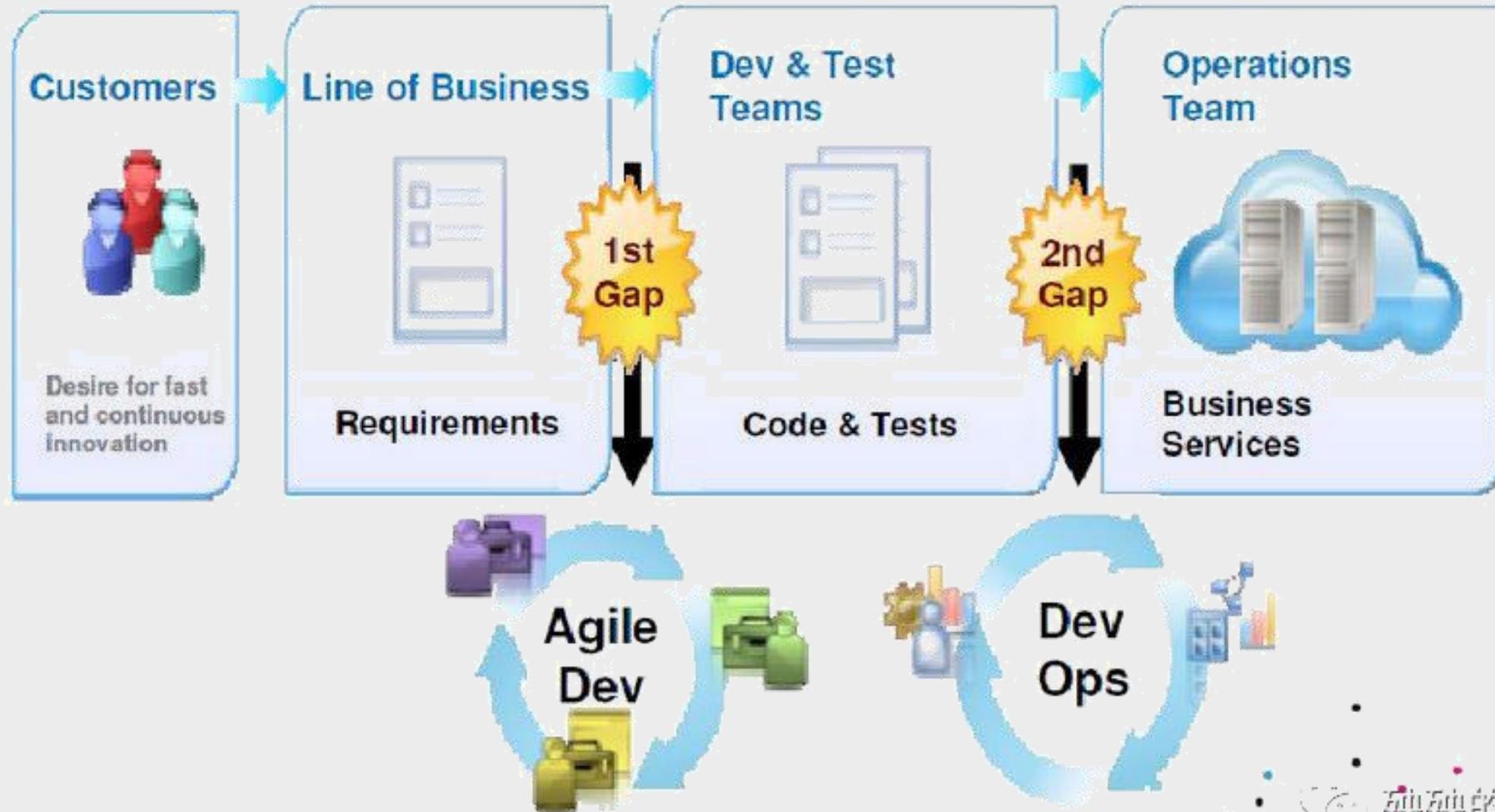


什么是DevOps

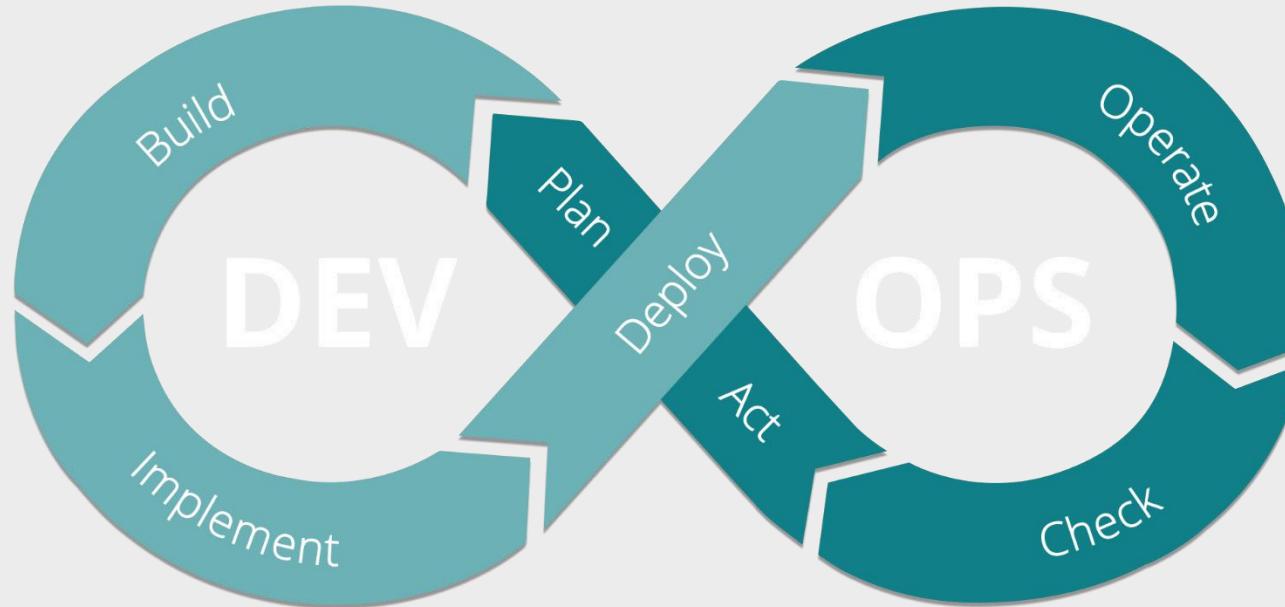
狭义的DevOps与广义的DevOps



什么是DevOps



什么是DevOps



- DevOps 是 “开发” 和 “运维”的缩写。
- DevOps是一组最佳实践，强调IT专业人员（开发人员、操作人员、支持人员）在应用和服务声明周期中的协作和沟通。
- 强调整个组织的合作以及交付和基础设施变更的自动化，从而实现持续集成、持续部署和持续交付。
- DevOps意味着在业务中建立了一条IT服务供应链，与其他的供应链嵌入业务相同。



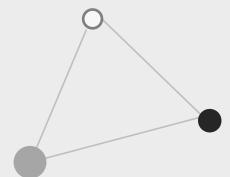
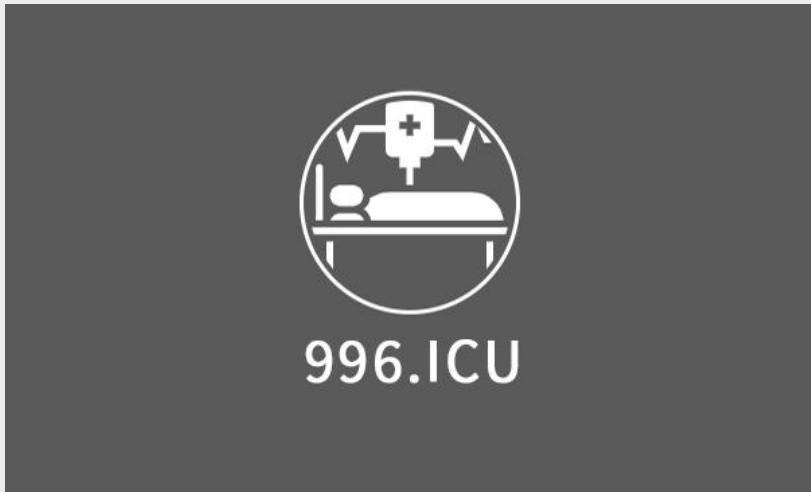
什么是DevOps



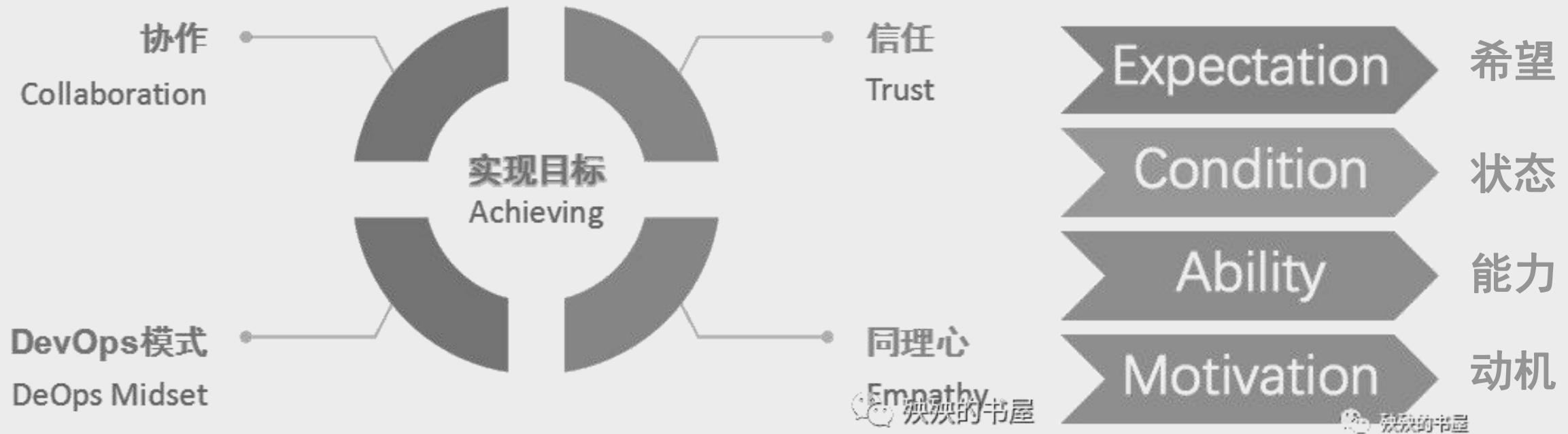
——资源来自《Effective DevOps》



什么是DevOps

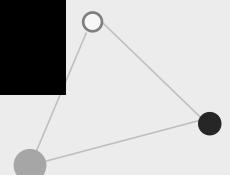


什么是DevOps

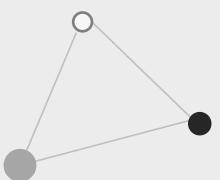


什么是DevOps

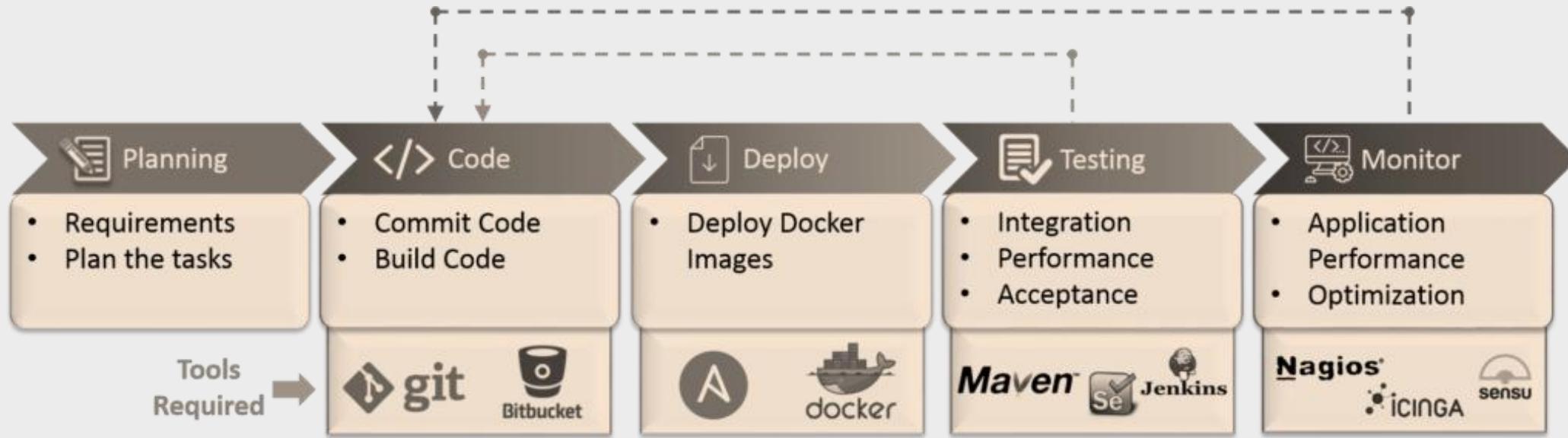
优酷



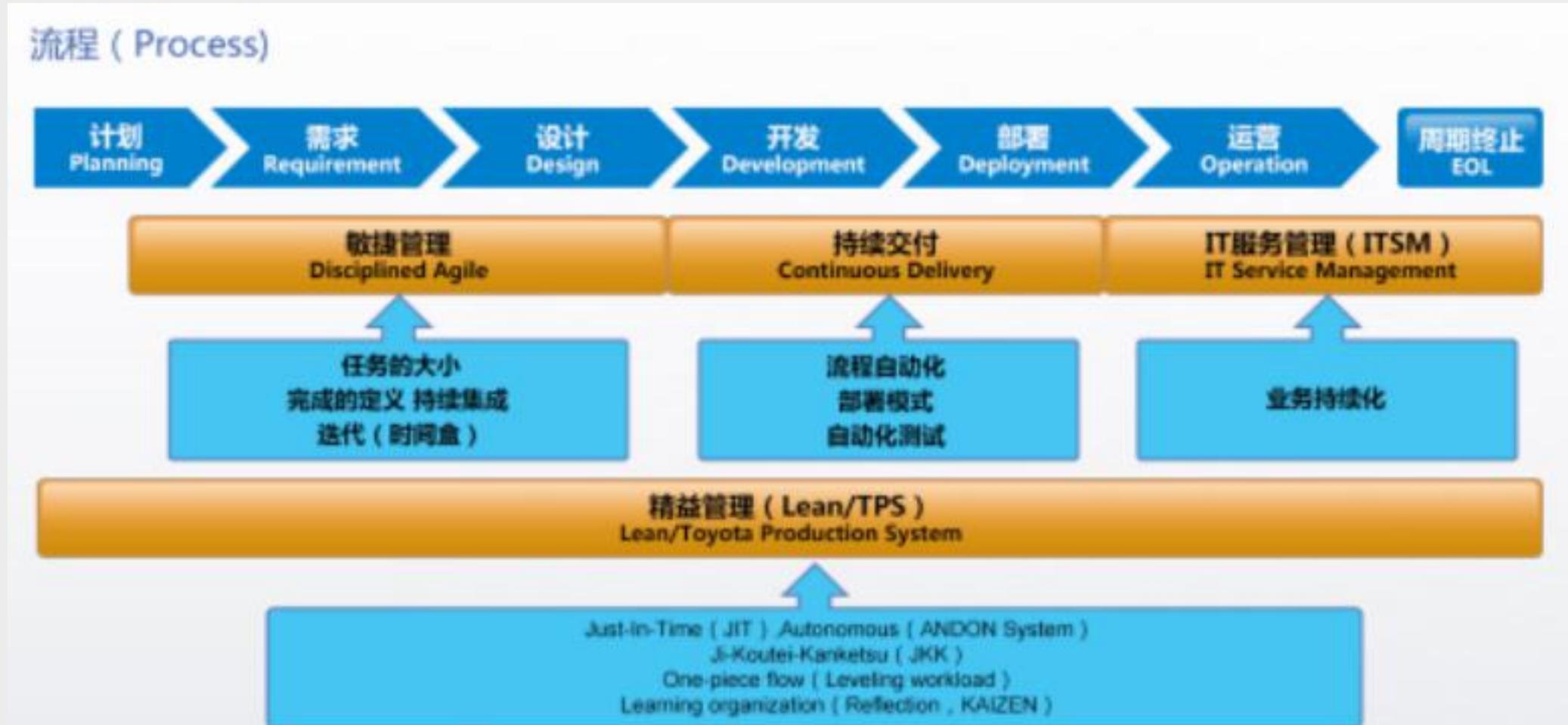
DevOps的流程



什么是DevOps



什么是DevOps



注：以精益管理为基石，以敏捷开发、持续交付和轻量级ITSM为支撑，其考试知识面广、注重实操、实践作业落地性强，再配以基于DevOps的实践沙盘，实乃广大互联网从业者提升及证明DevOps能力的有效途径！

03

项目生命周期与DevOps

Application

■ Lifecycle Management 生命周期管理

■ 聚焦于应用的开发阶段

■ 应用的操作和维护是一个输出服务的循环过程，该循环基于应用的声明周期。

Service

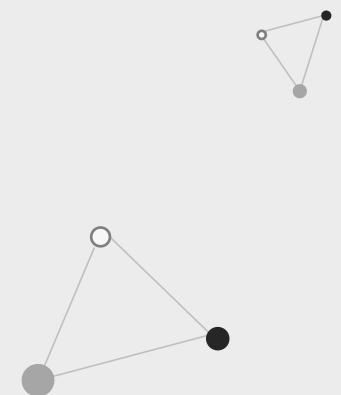
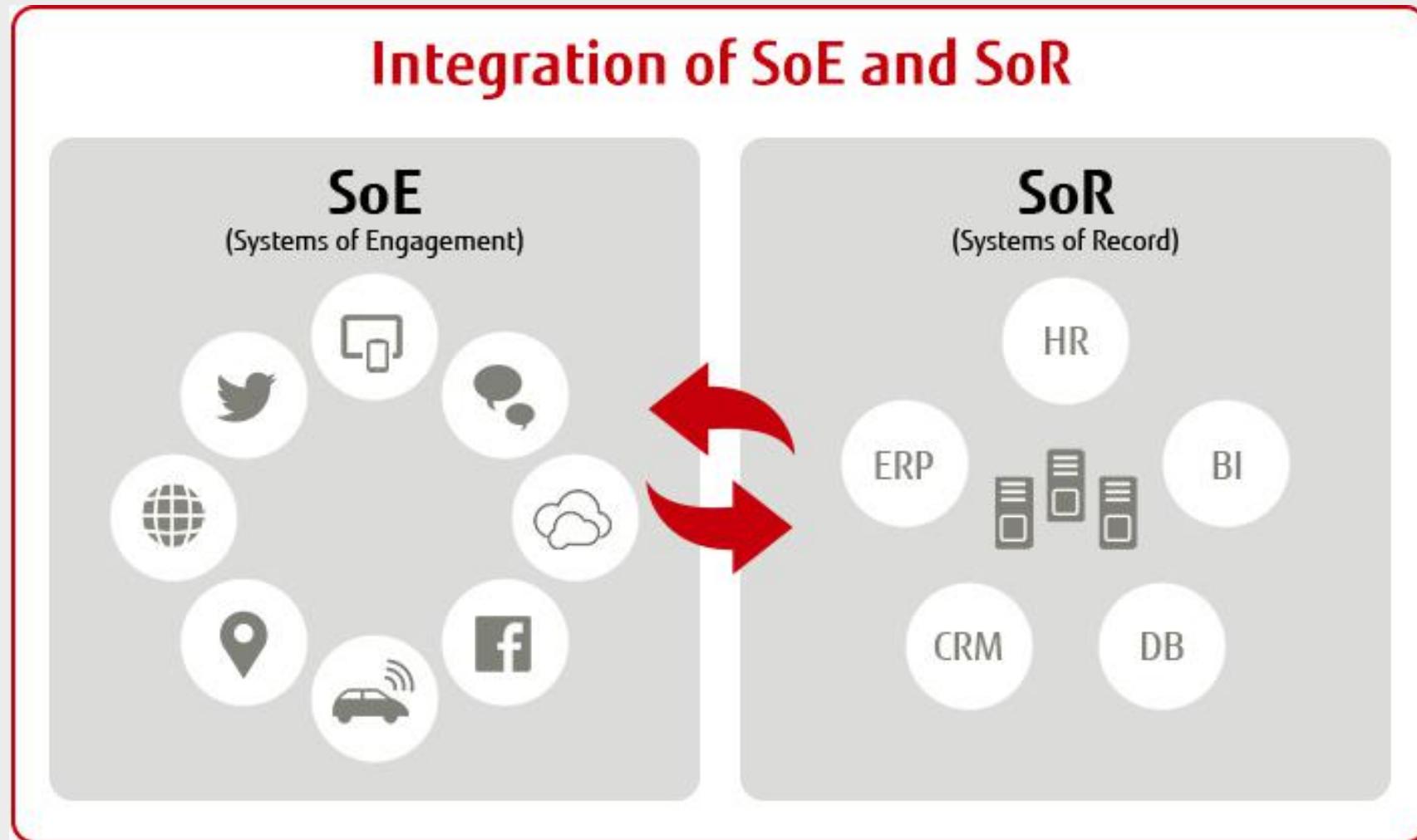
■ Lifecycle Management 生命周期管理

■ 在应用的开发阶段，聚焦于服务交付的准备

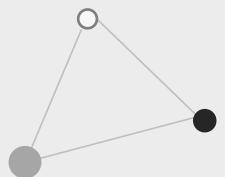
■ 服务交付有其自身的声明周期——战略、设计、转换、运营和持续改进



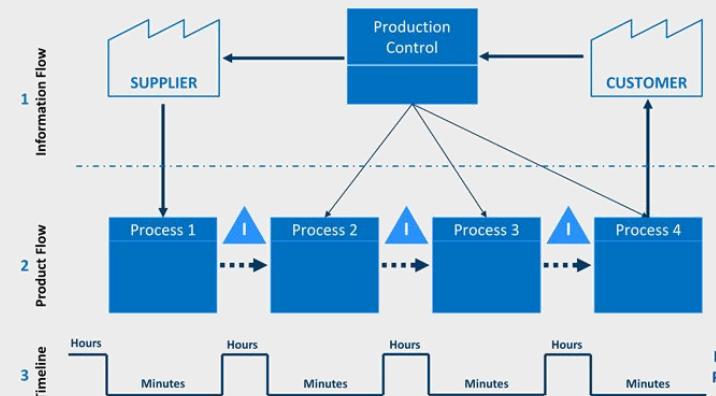
“项目”生命周期与DevOps



Part 1 : 规划、需求与设计



“项目”生命周期与DevOps —— 规划、需求与设计

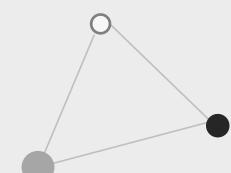


项目的愿景，目标和价值在开始时就达成一致

设计了整个过程的价值流图

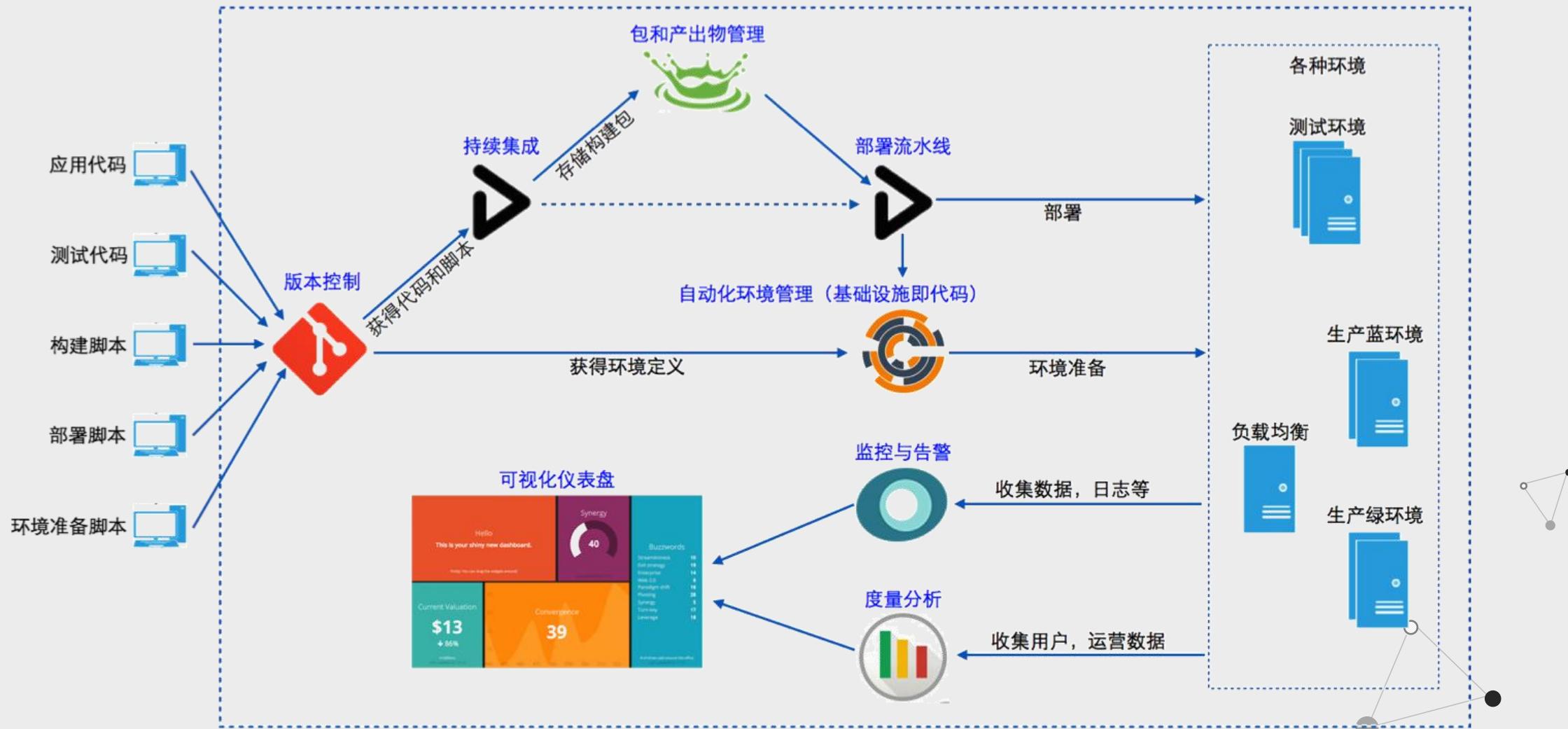


使用自动化过程进行验收测试，
性能测试和部署。



“项目”生命周期与DevOps——规划、需求与设计

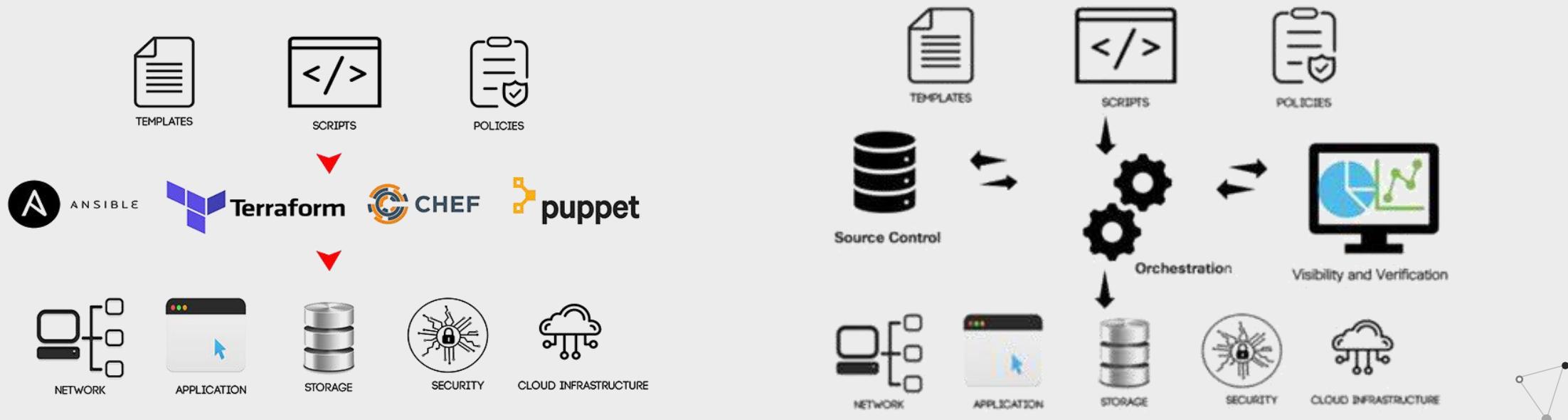
基础设施即代码



“项目”生命周期与DevOps——规划、需求与设计

基础设施即代码

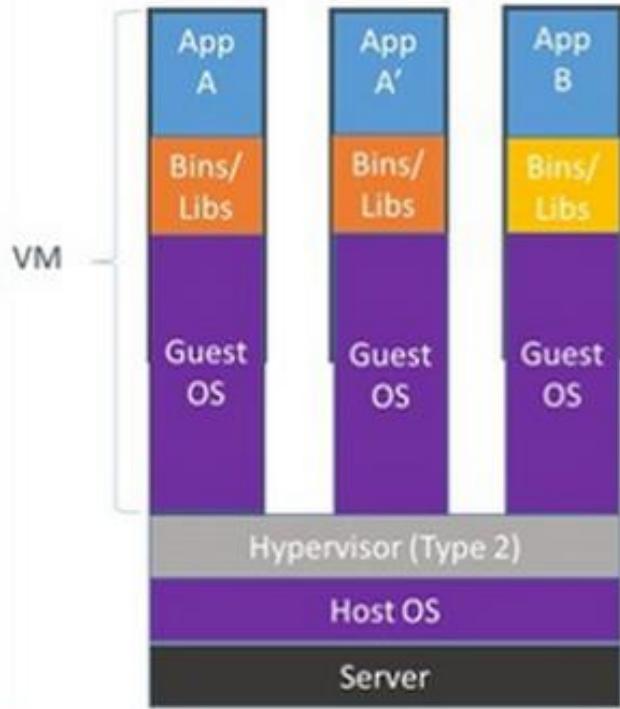
Infrastructure as Code



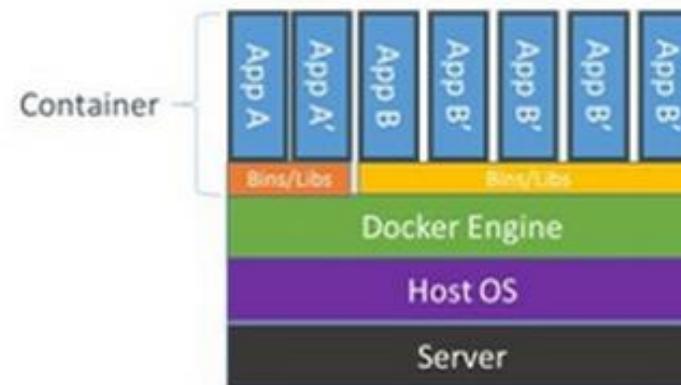
“基础设施即代码是一种使用心得技术来构建和管理动态基础设施的方式。它把基础设施、工具和服务以及对基础设施的管理本身作为一个软件系统，采纳软件工程实践以结构化的安全方式来管理对系统的变更。” ——《Infarftruce As Code》

“项目”生命周期与DevOps —— 规划、需求与设计

Containers vs. VMs



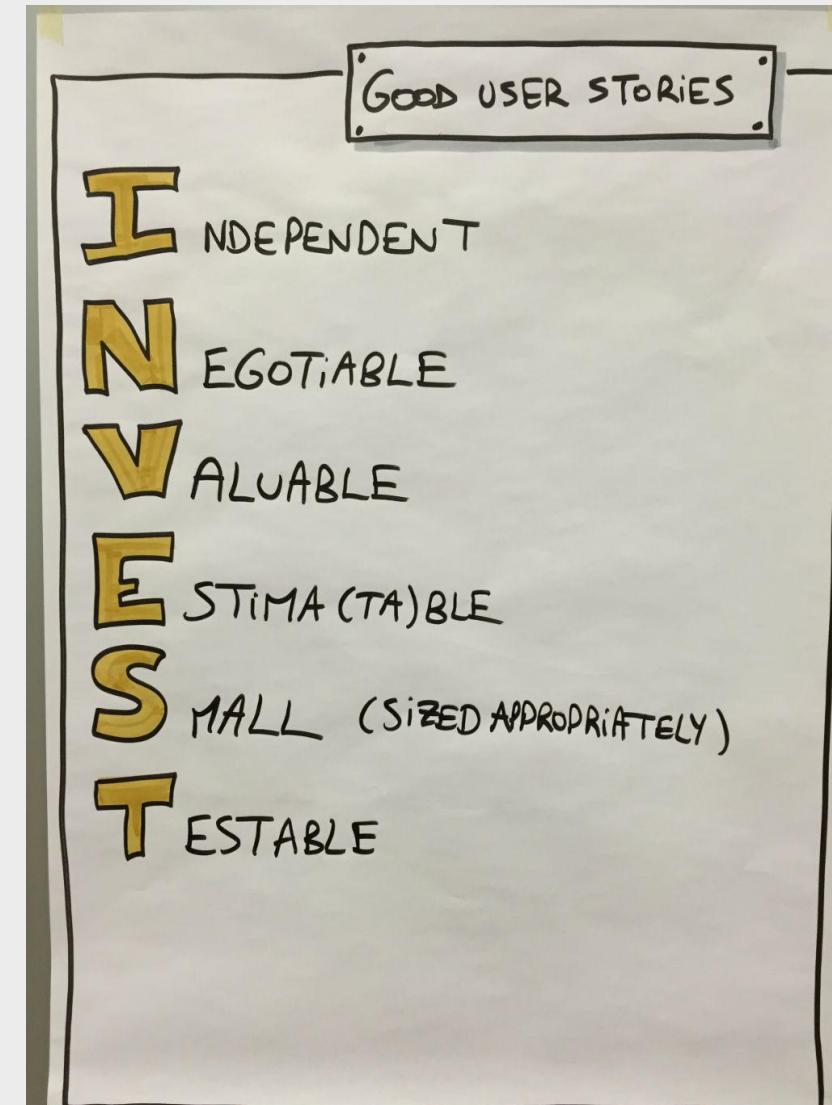
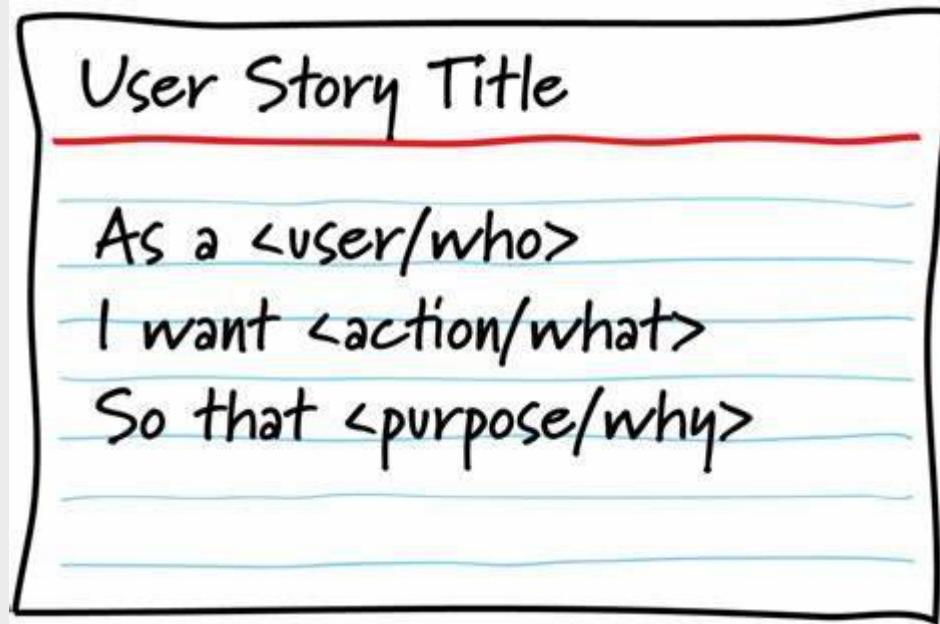
Containers are isolated,
but share OS and, where
appropriate, bins/libraries



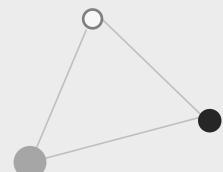
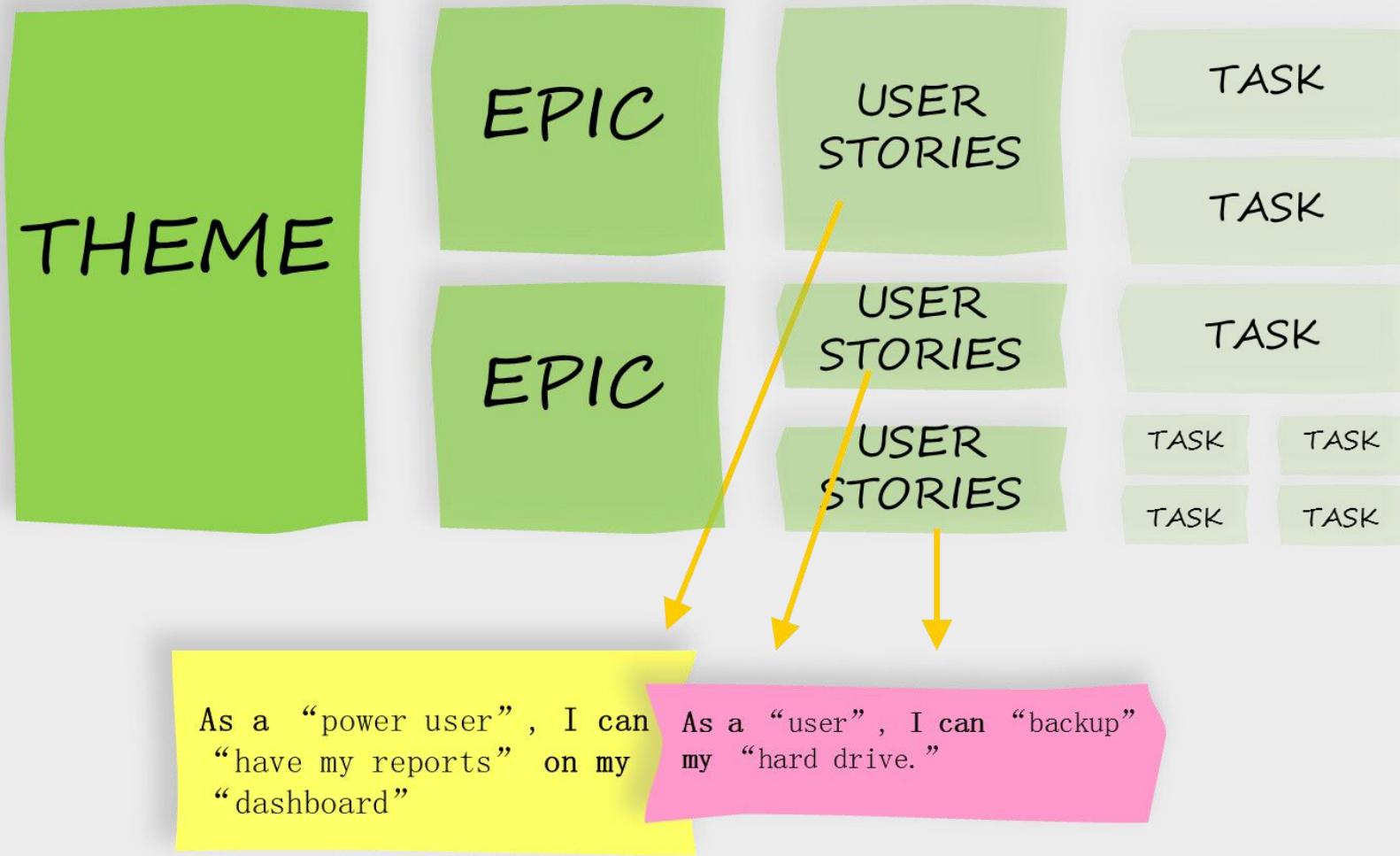
虚拟机（VM）是操作系统级别的模拟

容器（Container）是应用级别的模拟

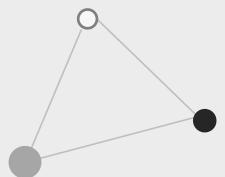
“项目”生命周期与DevOps —— 规划、需求与设计



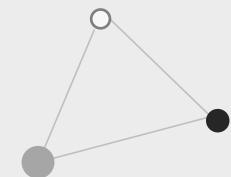
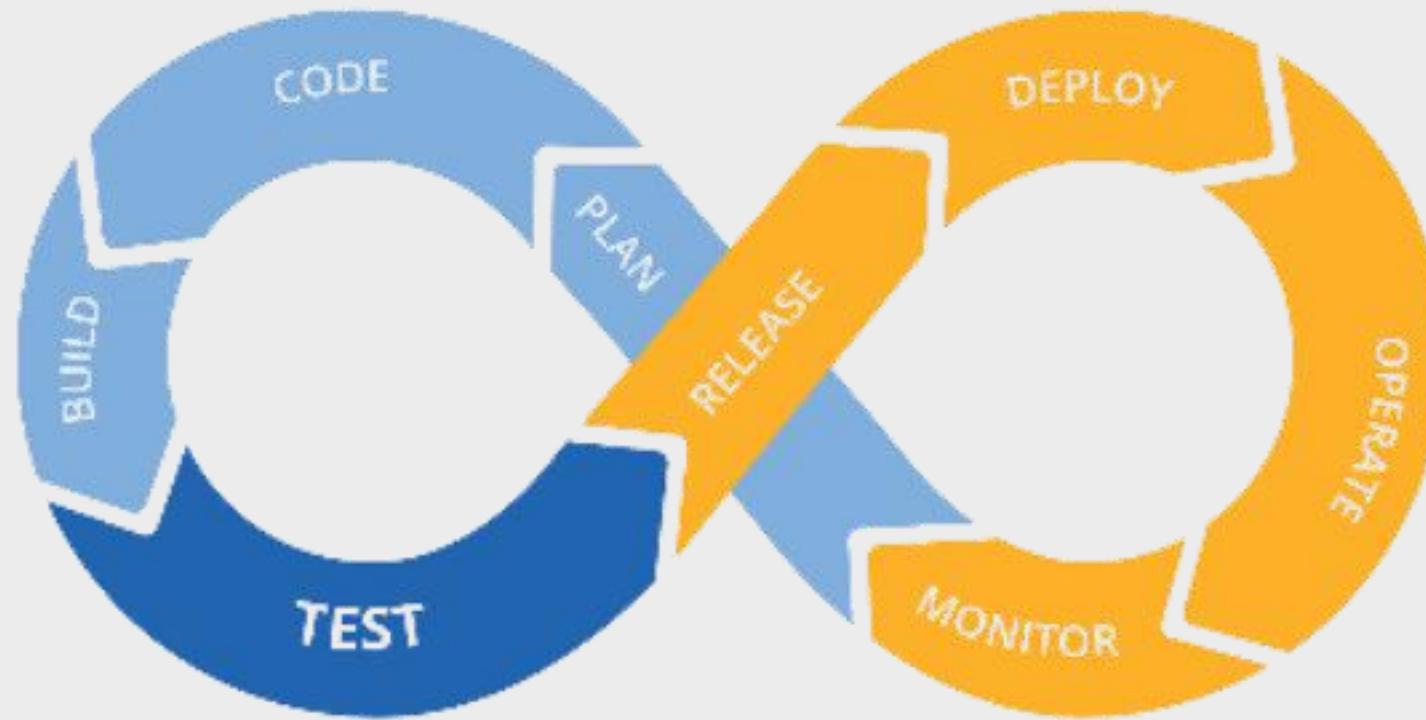
“项目”生命周期与DevOps —— 规划、需求与设计



Part 2 : 开发 和 部署

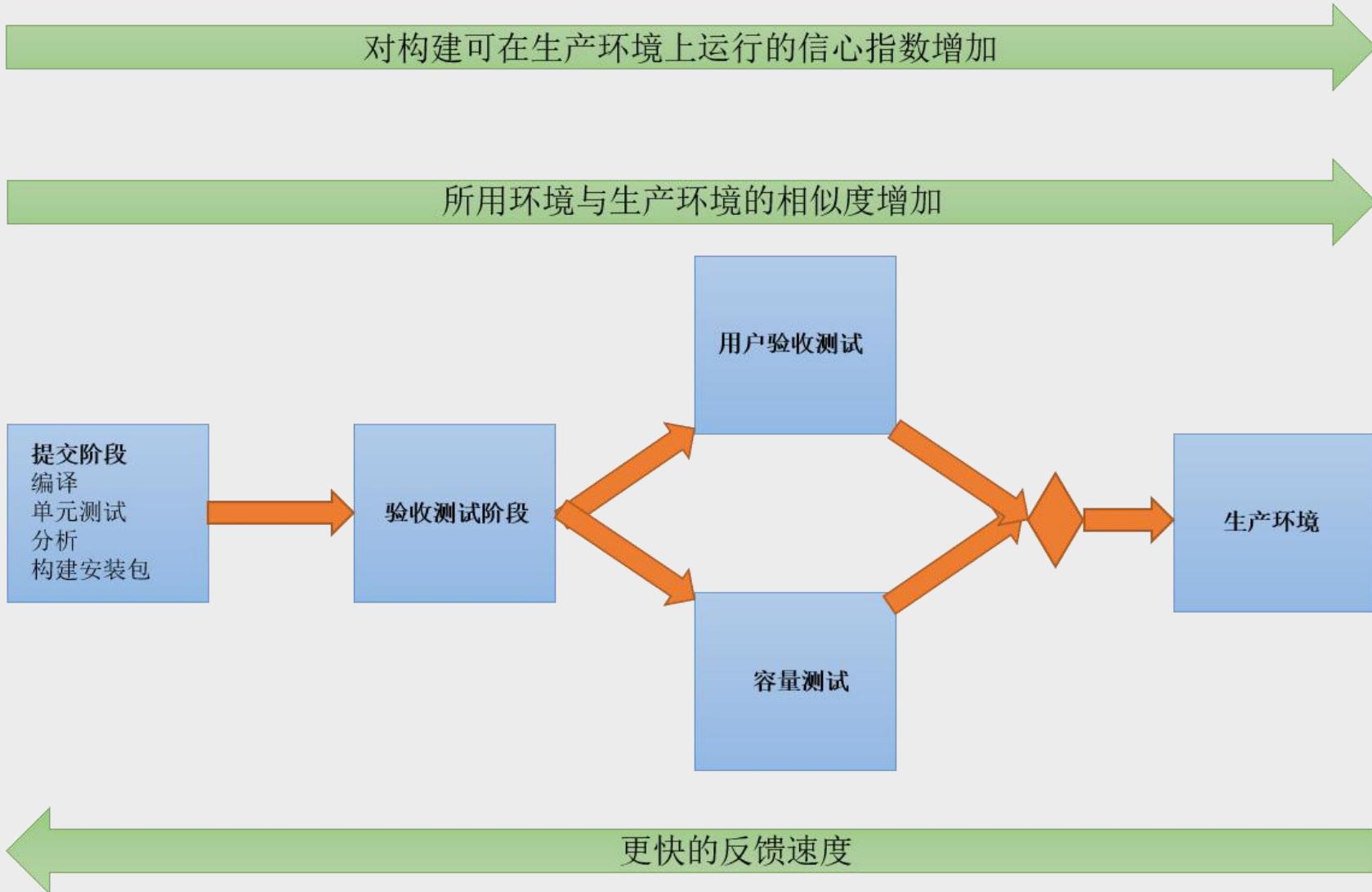


“项目”生命周期与DevOps —— 开发和部署



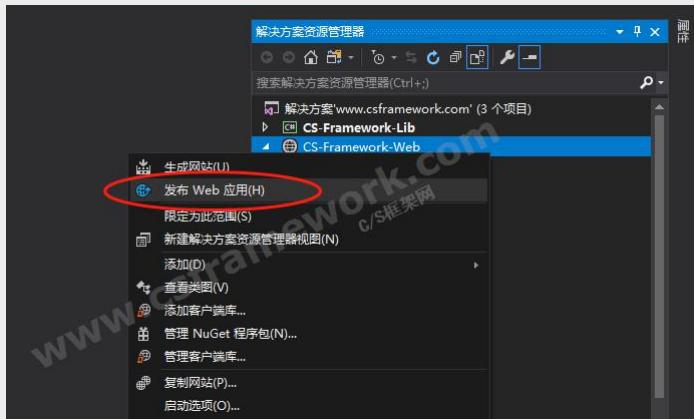
“项目”生命周期与DevOps —— 开发和部署

回顾：部署流水线

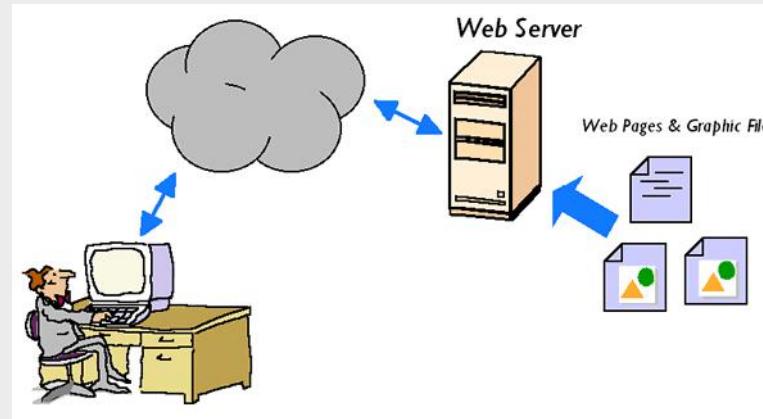


“项目”生命周期与DevOps —— 开发和部署

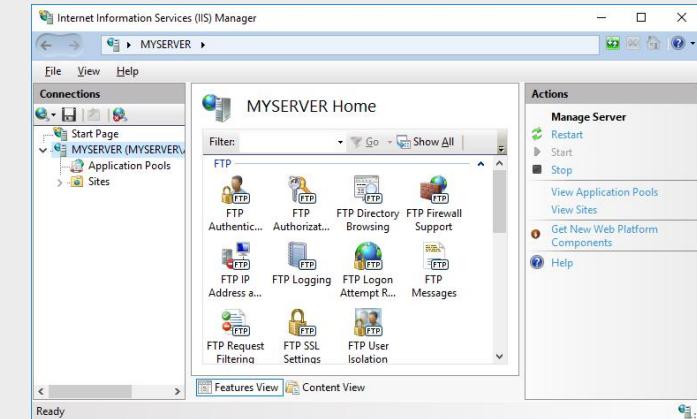
软件交付中的常见的反模式



手工部署软件



开发完成之后才向类生产环境部署

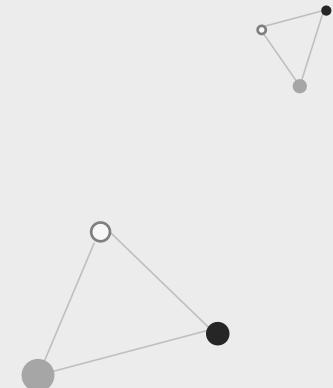
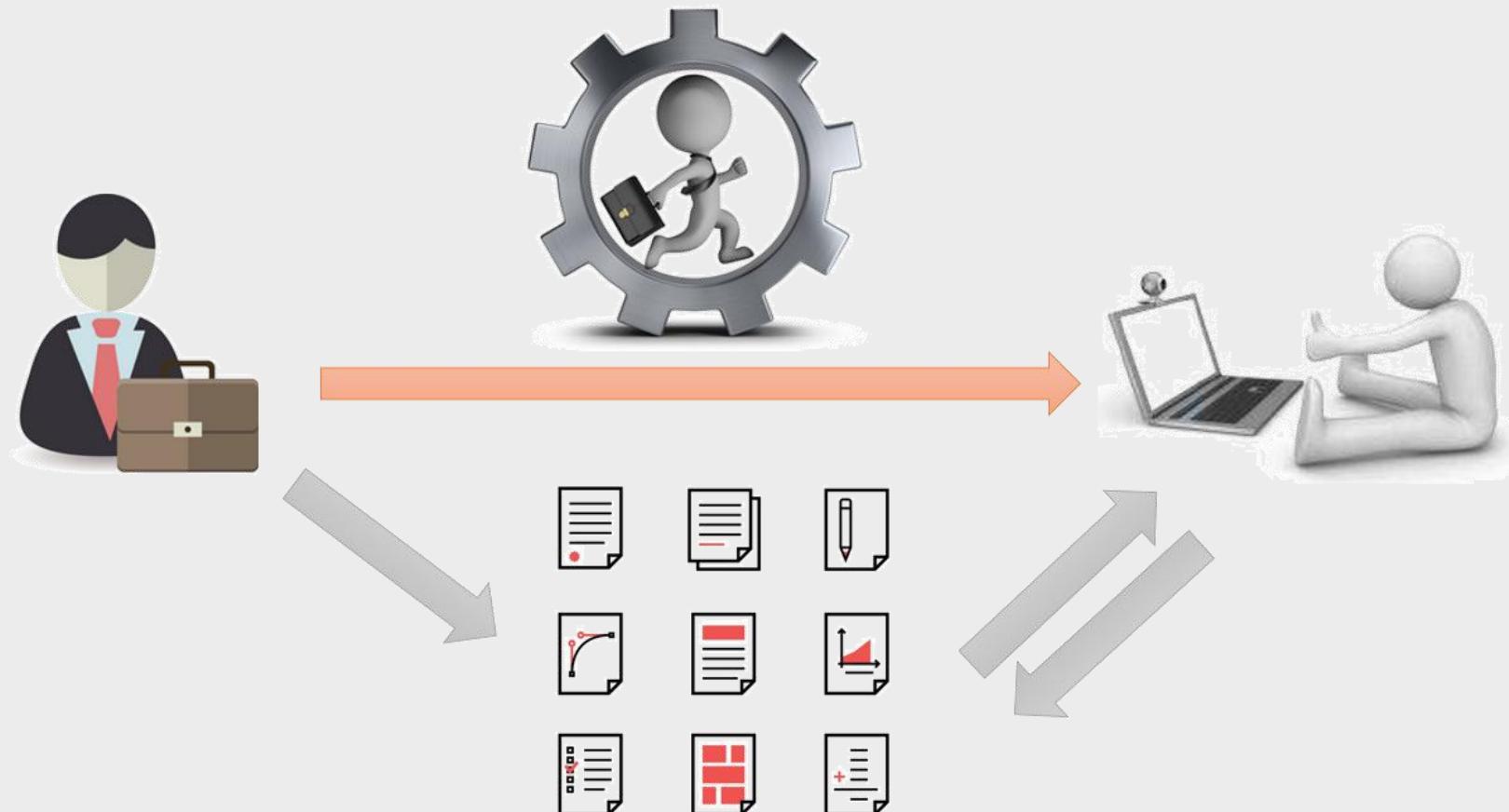


生产环境的手工配置管理



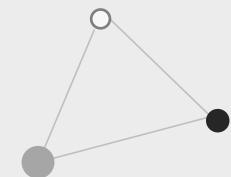
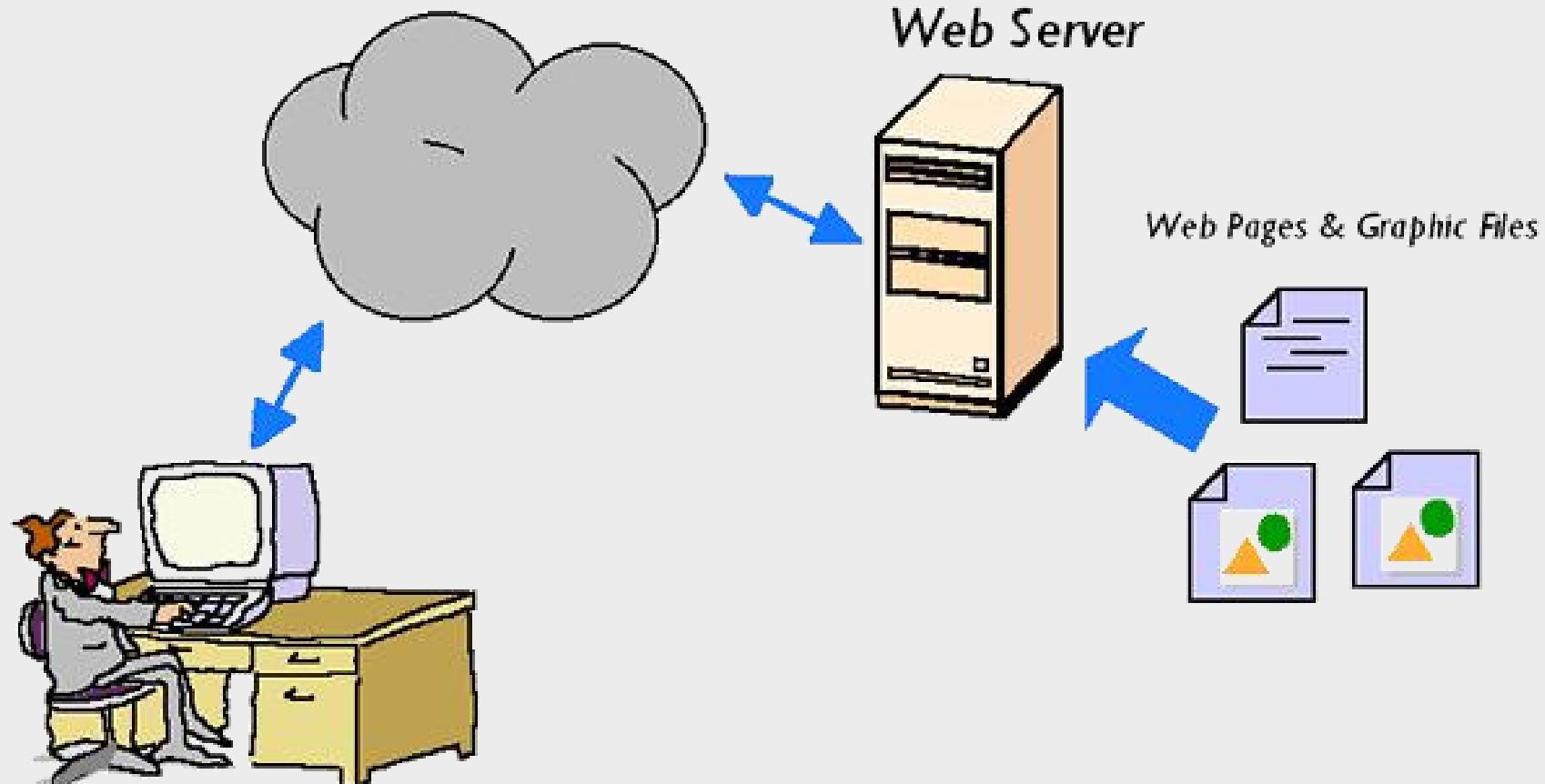
“项目”生命周期与DevOps —— 开发和部署

反模式：手工部署软件



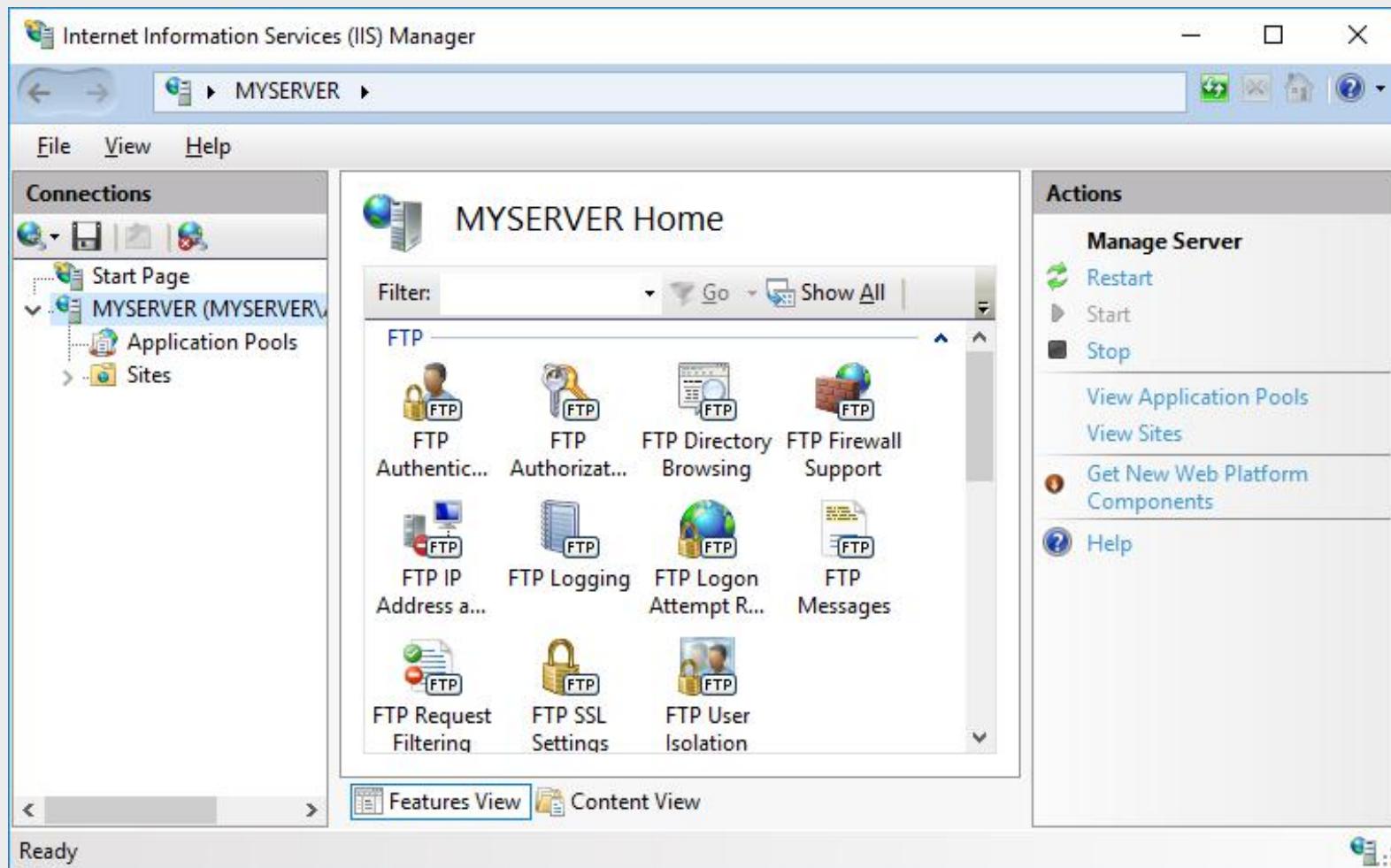
“项目”生命周期与DevOps —— 开发和部署

反模式：开发完成之后才向类生产环境部署

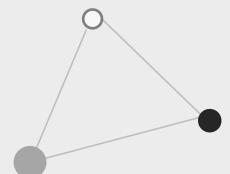


“项目”生命周期与DevOps —— 开发和部署

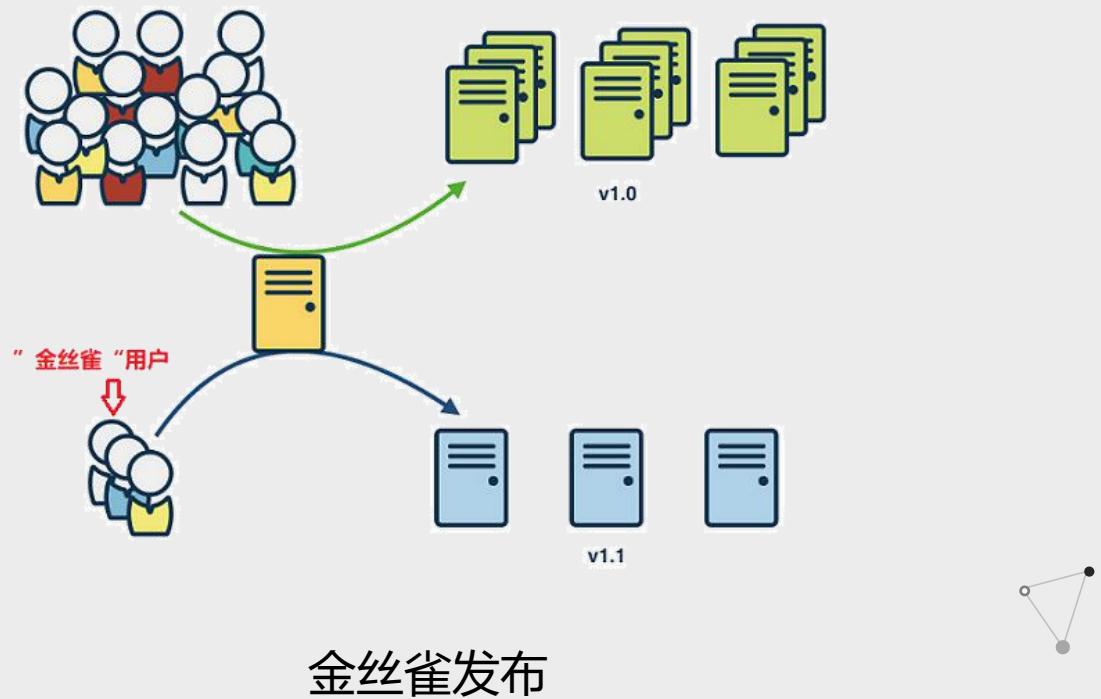
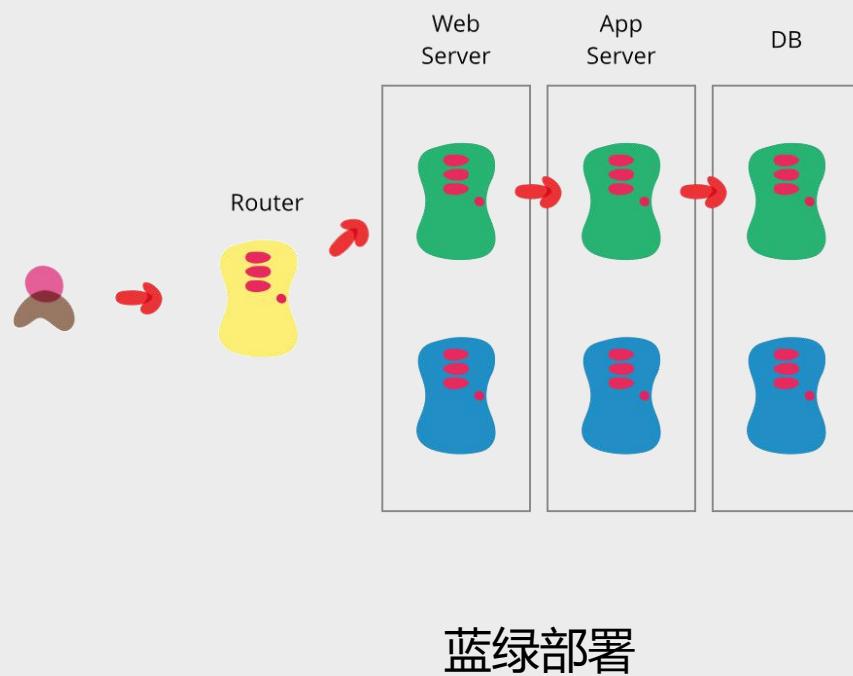
反模式：开发完成之后才向类生产环境部署



**Homework : 了解了以上三个反模式的特点，那么
我们如何做的更好？（如何改进？）**

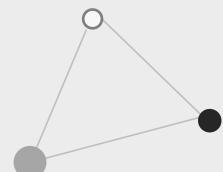
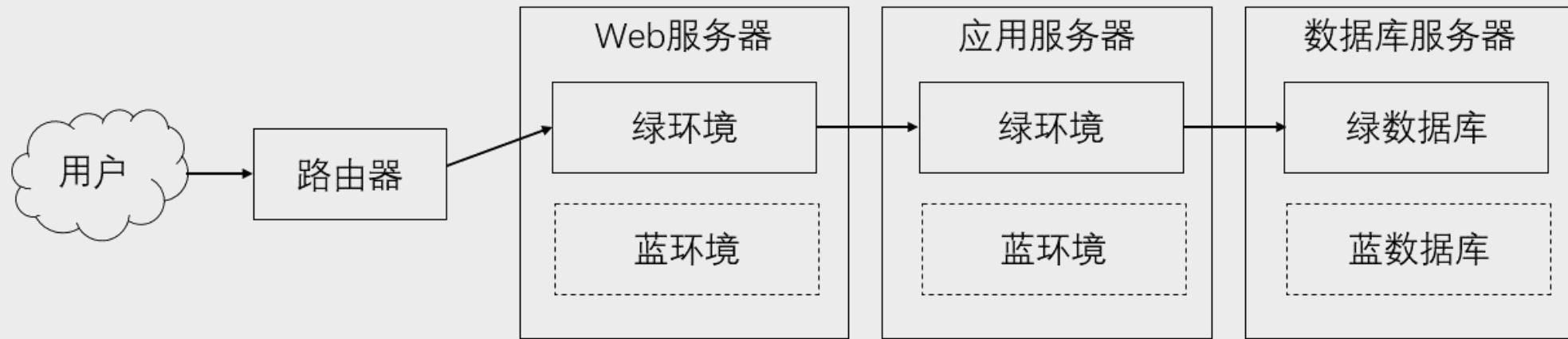


“项目”生命周期与DevOps —— 开发和部署



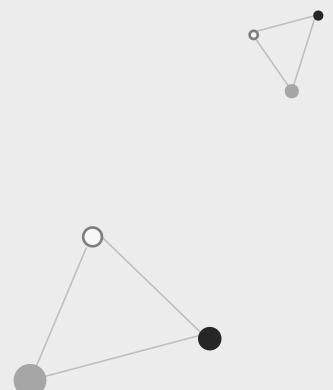
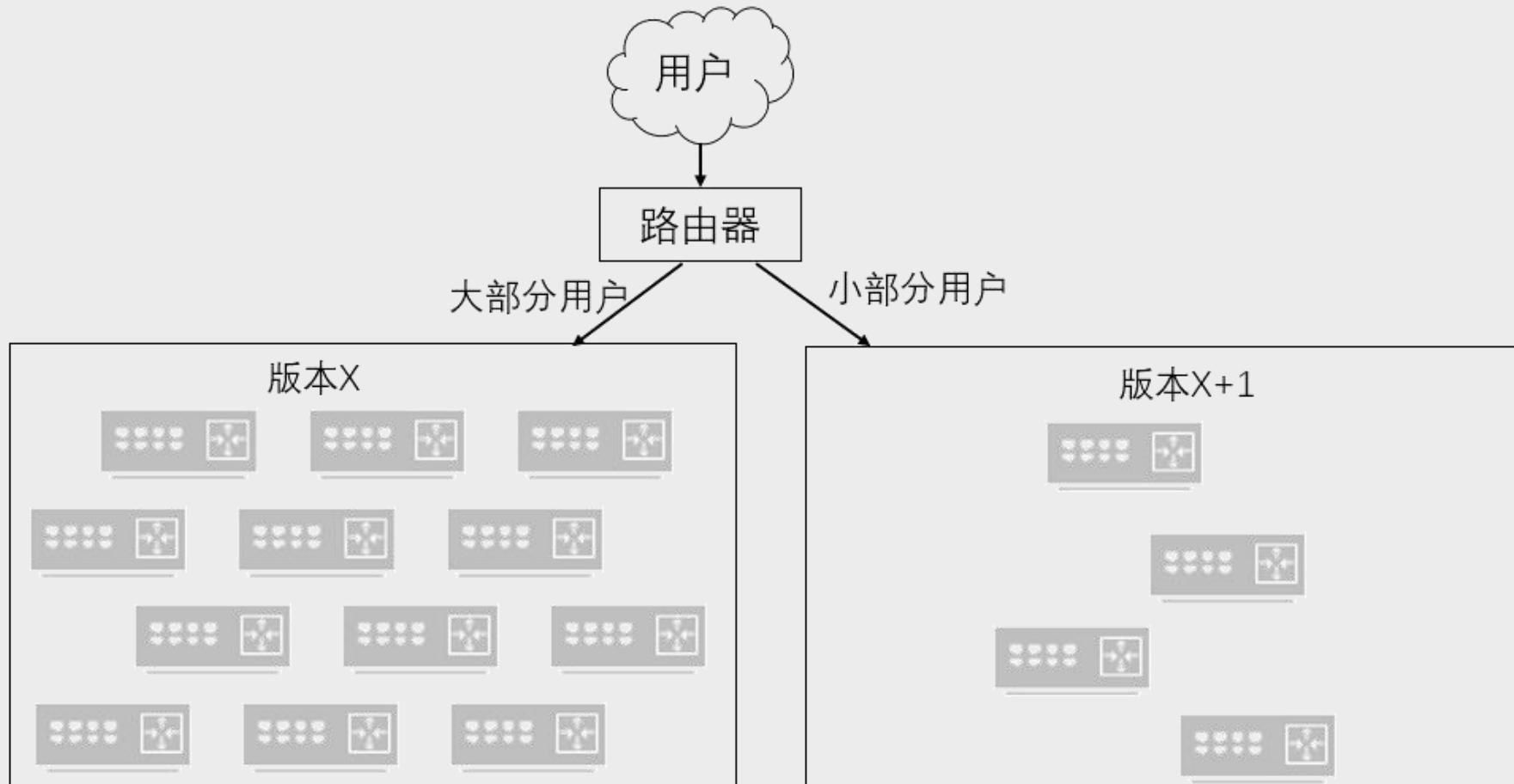
“项目”生命周期与DevOps —— 开发和部署

蓝绿部署

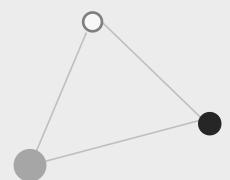


“项目”生命周期与DevOps —— 开发和部署

金丝雀发布

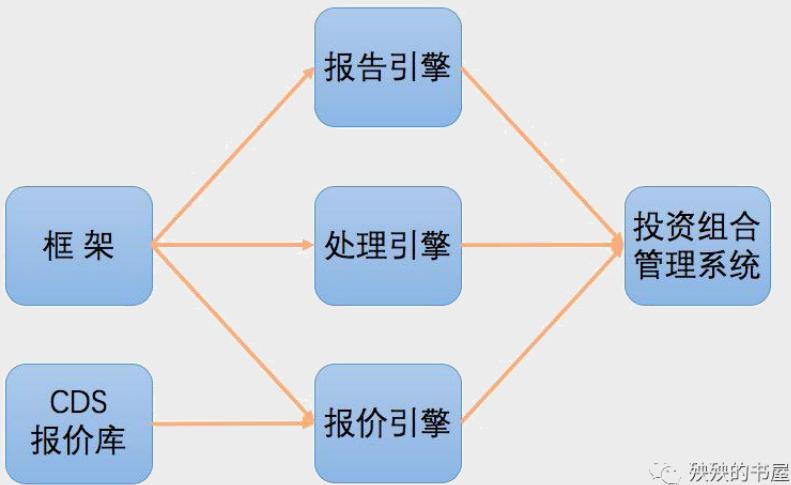


Part 3 : 运维 和 规模化

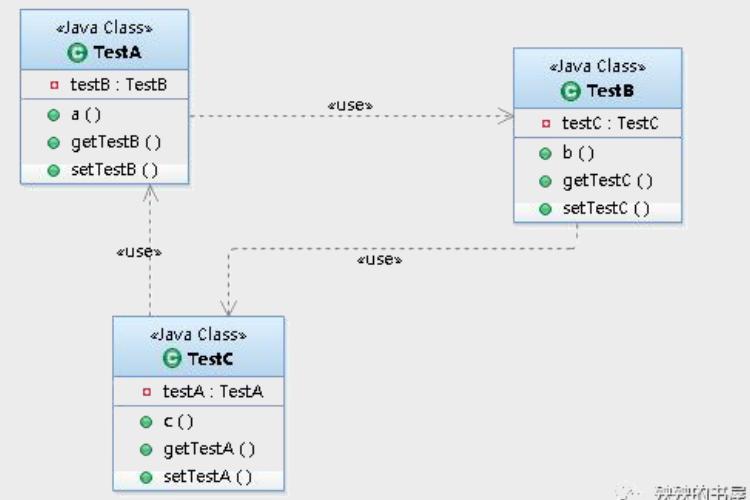


“项目”生命周期与DevOps —— 运维和规模化

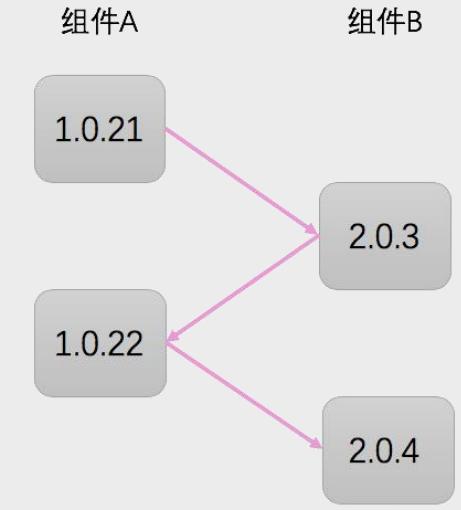
回顾：依赖管理



© 独特的书屋



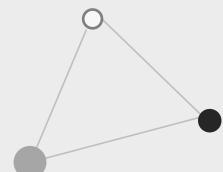
© 独特的书屋



“项目”生命周期与DevOps —— 运维和规模化

业务联系性

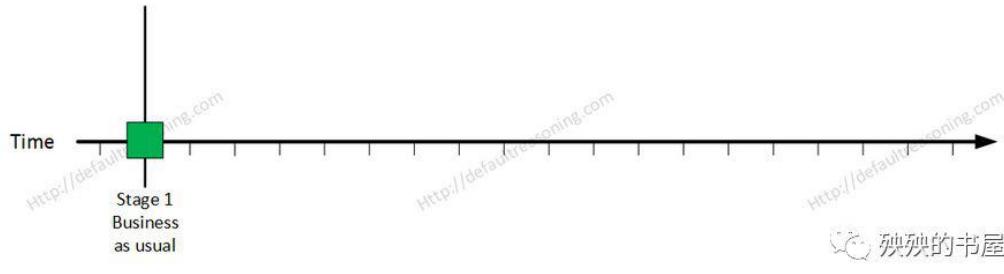
- High availability of IT Services is critical to the survival of the business as whole.
IT服务的高可用性对整个企业的生存至关重要。
- Use risk reduction measures and recovery options.
使风险降低措施和恢复选项
- Ongoing maintenance of the recovery capability is essential.
持续维护恢复能力至关重要。



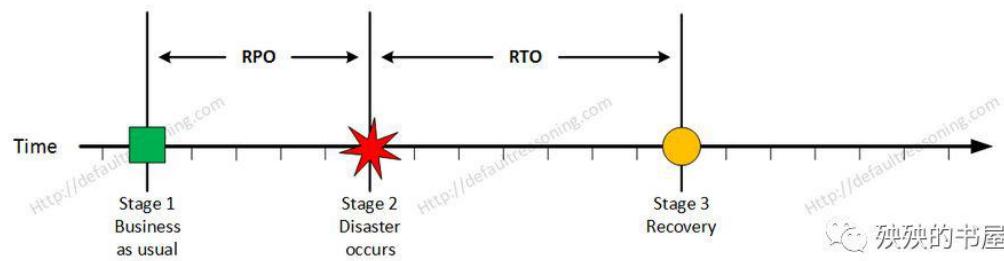
“项目”生命周期与DevOps —— 运维和规模化

回顾：RPO & RTO（业务连续性）

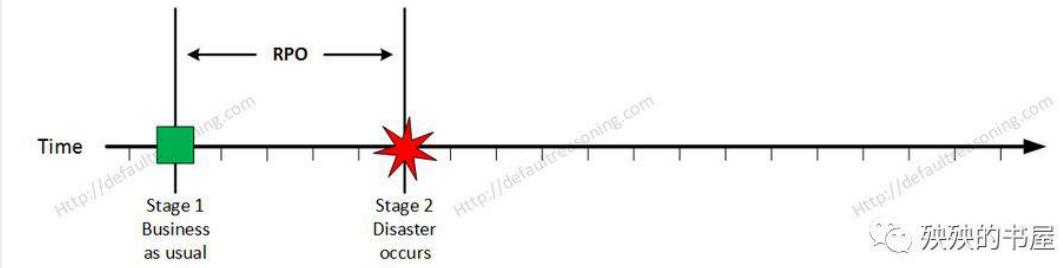
Stage 1: Business as usual



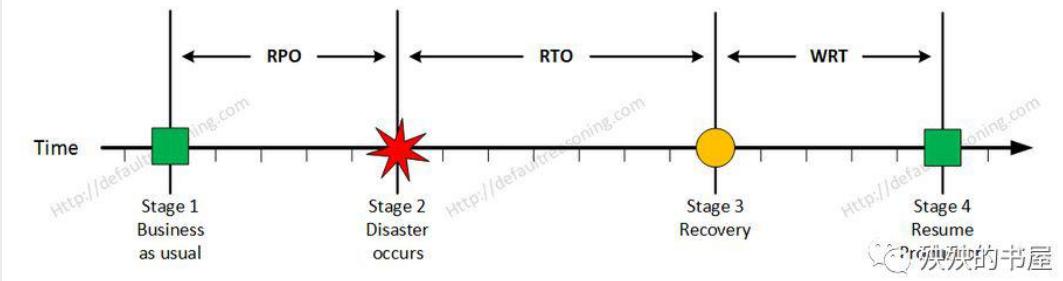
Stage 3: Recovery



Stage 2: Disaster occurs

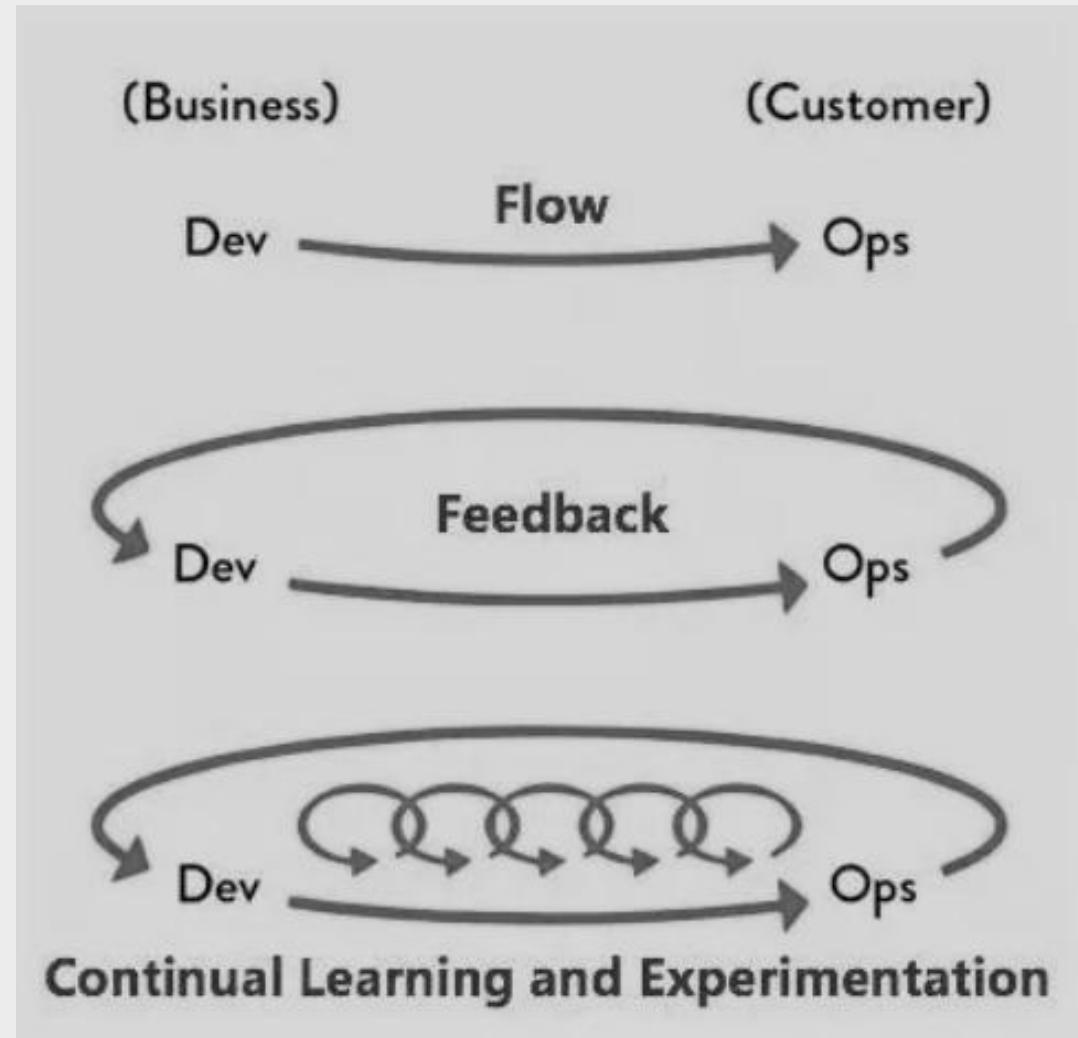


Stage 4: Resume Production

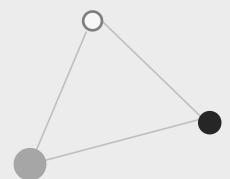


“项目”生命周期与DevOps —— 运维和规模化

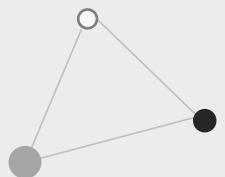
规模化：DevOps 实施的三步工作法



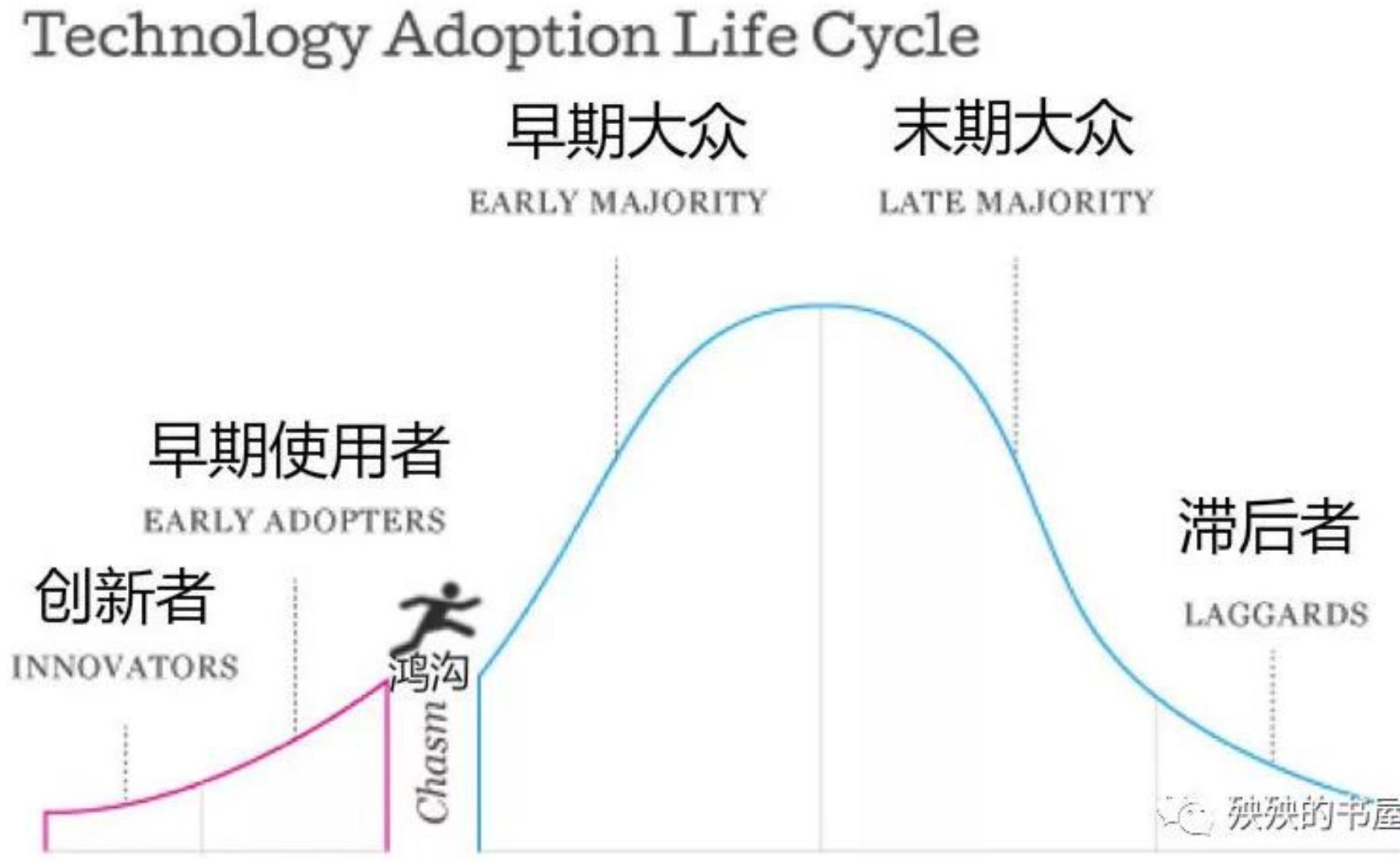
End : 生命周期的终结



选择试点项目和团队的考量

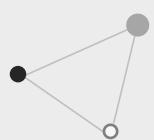
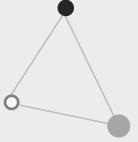


“项目”生命周期与DevOps —— 开发和部署



04

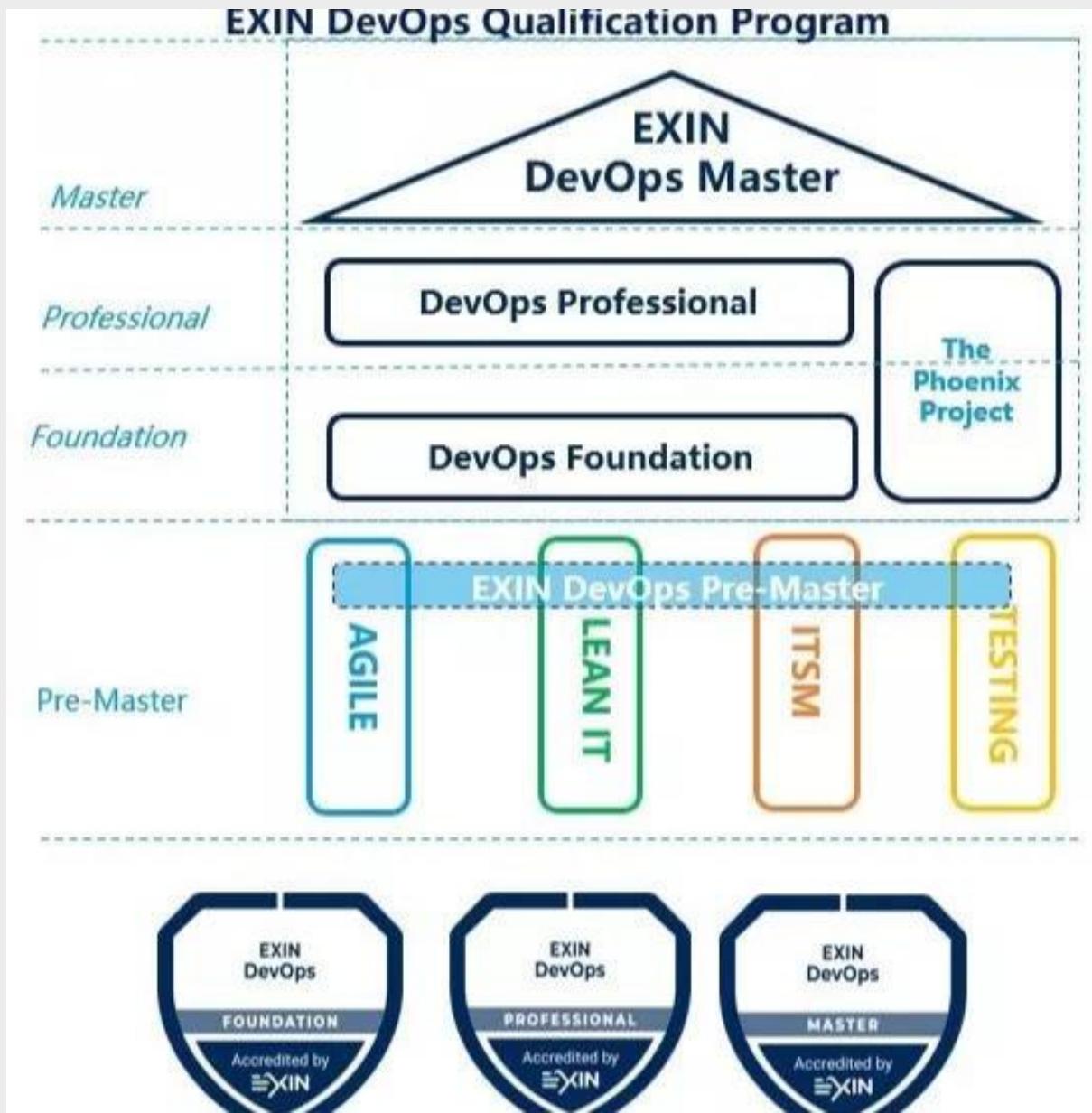
DevOps认证





国际信息科学考试学会（Exam Institute for Information Science,简称EXIN）由荷兰经济事务部于1984年创办，现今已经从荷兰政府部门独立成立了EXIN基金会。EXIN是一家面向全球ICT从业人员的中立认证考试机构。EXIN是全球IT服务管理最佳实践知识体系 - ITIL®的创始机构之一，开发并拥有ITIL®版本一和版本二全系列试题的知识产权。自1993年起EXIN面向全球提供ITIL®全系列认证。迄今为止，来自全球120多个国家和地区累计两百万以上的IT职业人士已经获得了EXIN颁发的IT管理资格认证。^[1] 荷兰国际信息科学考试学会作为全球知名的企业数字化转型权威定标认证机构，在全球认证了三百万名企业高级管理人员。

DevOps 认证



Certificate
EXIN DevOps Master

Presented to:
Zhenhuan Wang

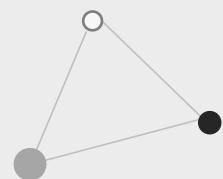
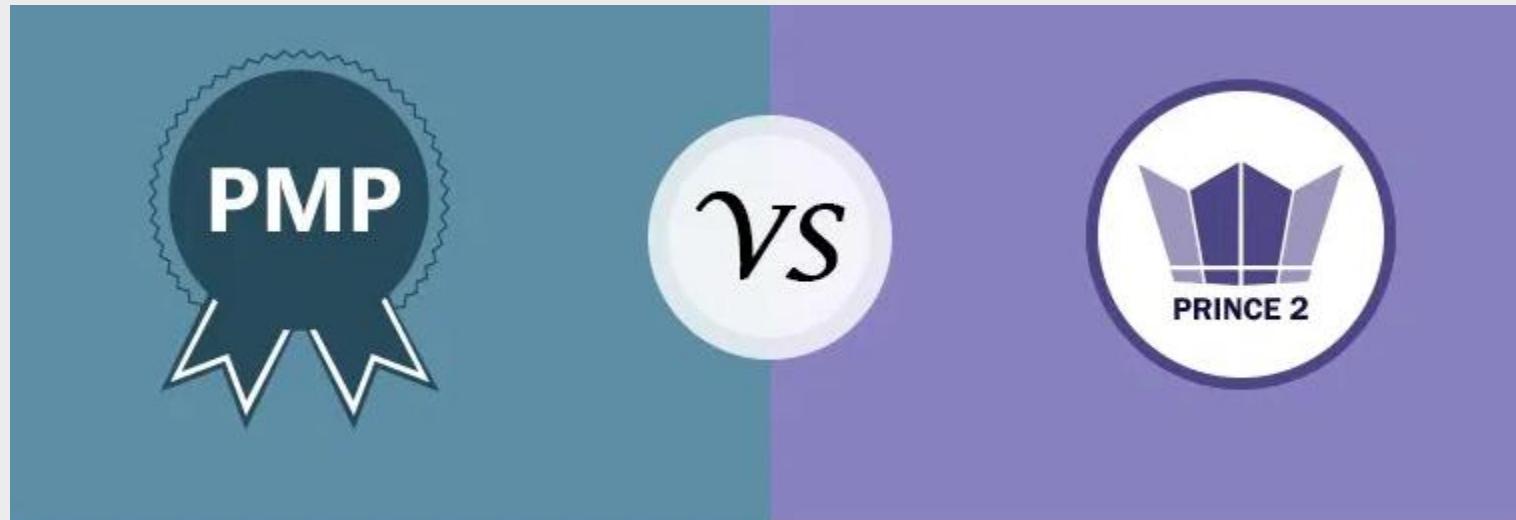
16 December 2018



drs. Bernd W.E. Taselaar
Chief Executive Officer
6241332-3075-5080

EXIN
The global independent certification institute for ICT Professionals

其他认证 —— 项目管理

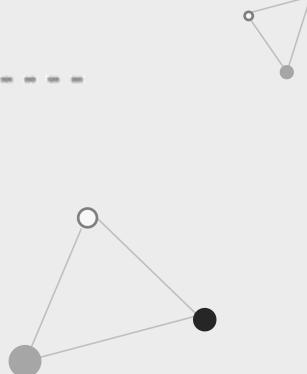


其他认证 —— 敏捷项目管理 (Scrum Alliance)

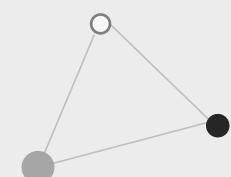
Scrum Alliance Certification Paths 2020



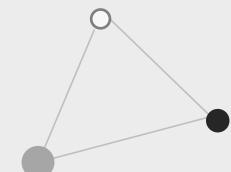
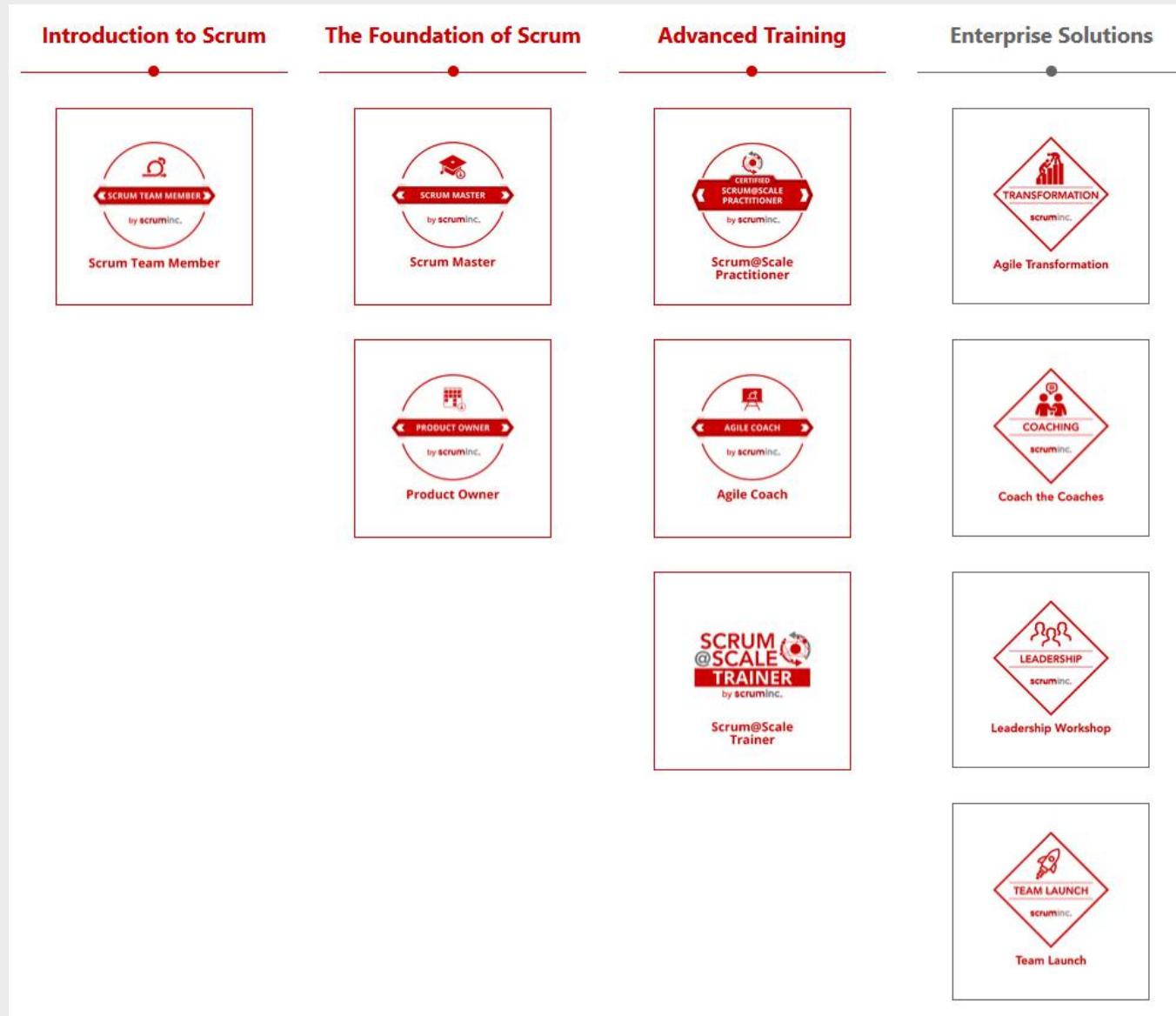
Associated Certifications



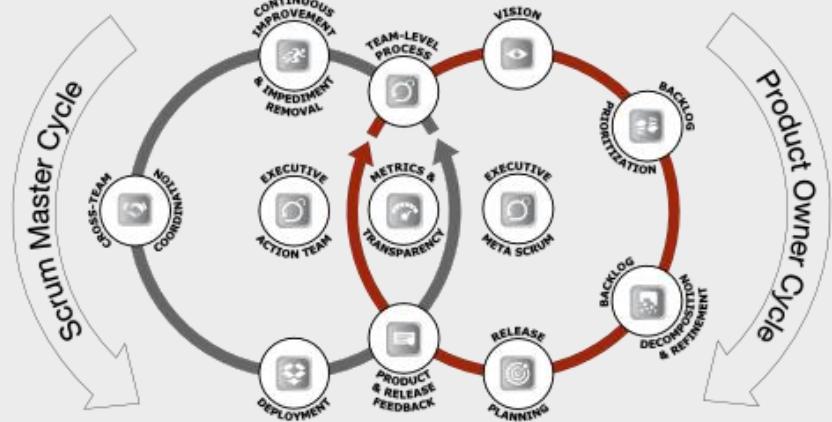
其他认证 —— 敏捷项目管理 (Scrum.Org)



其他认证 —— 敏捷项目管理 (ScrumInc)

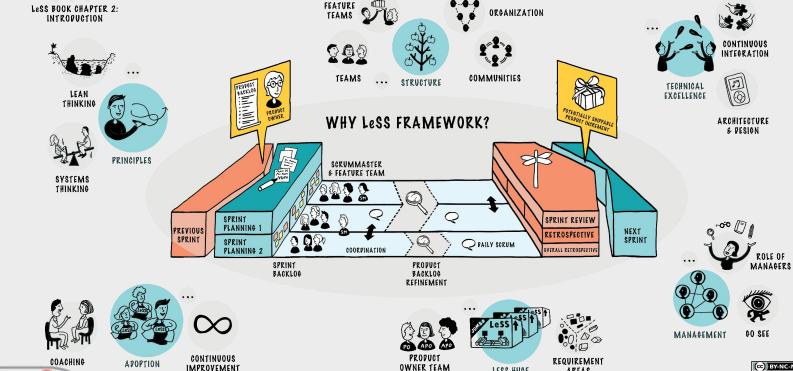


其他认证 —— 规模化敏捷 (Scale Agile)

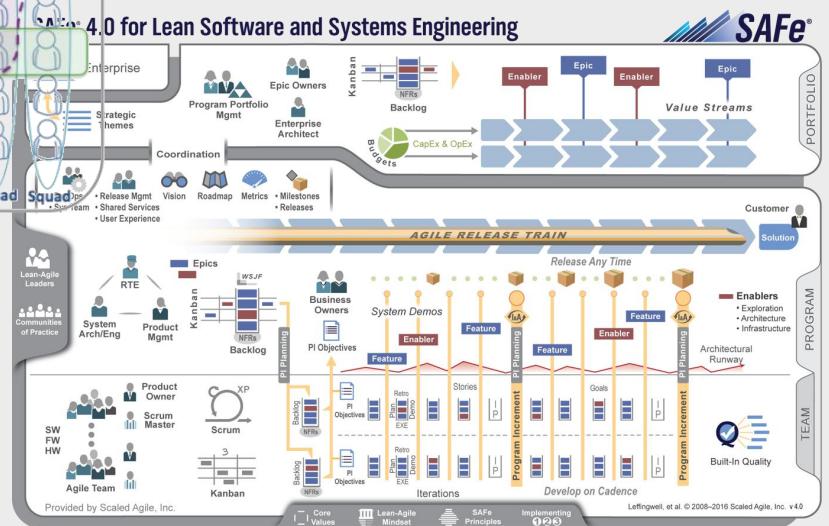
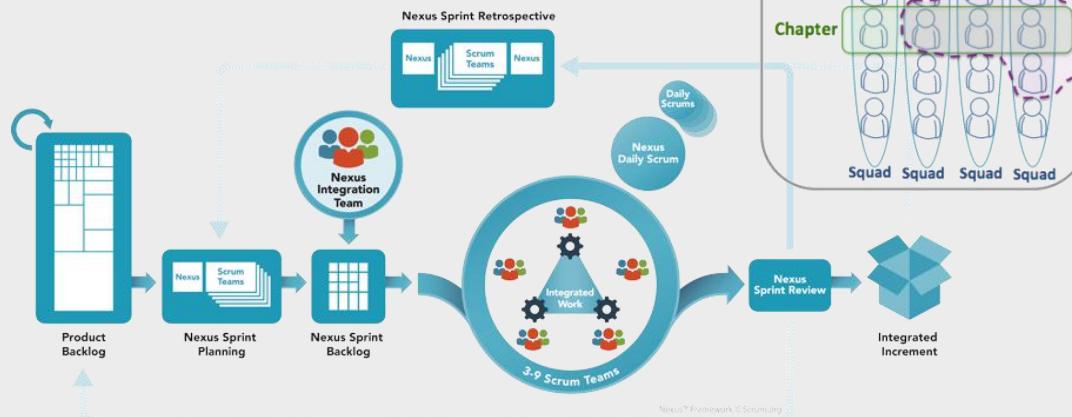


Scaling Agile @ Spotify with Tribes, Squads, Chapters & Guilds

Henrik Kniberg & Anders Ivarsson
Oct 2012



NEXUS™ FRAMEWORK



朝夕教育
謝 謝

