

源于社区 服务社区

 中国DevOps社区峰会 2024 · 上海



构建提质增效的研发飞轮

肖然 Thoughtworks





目录

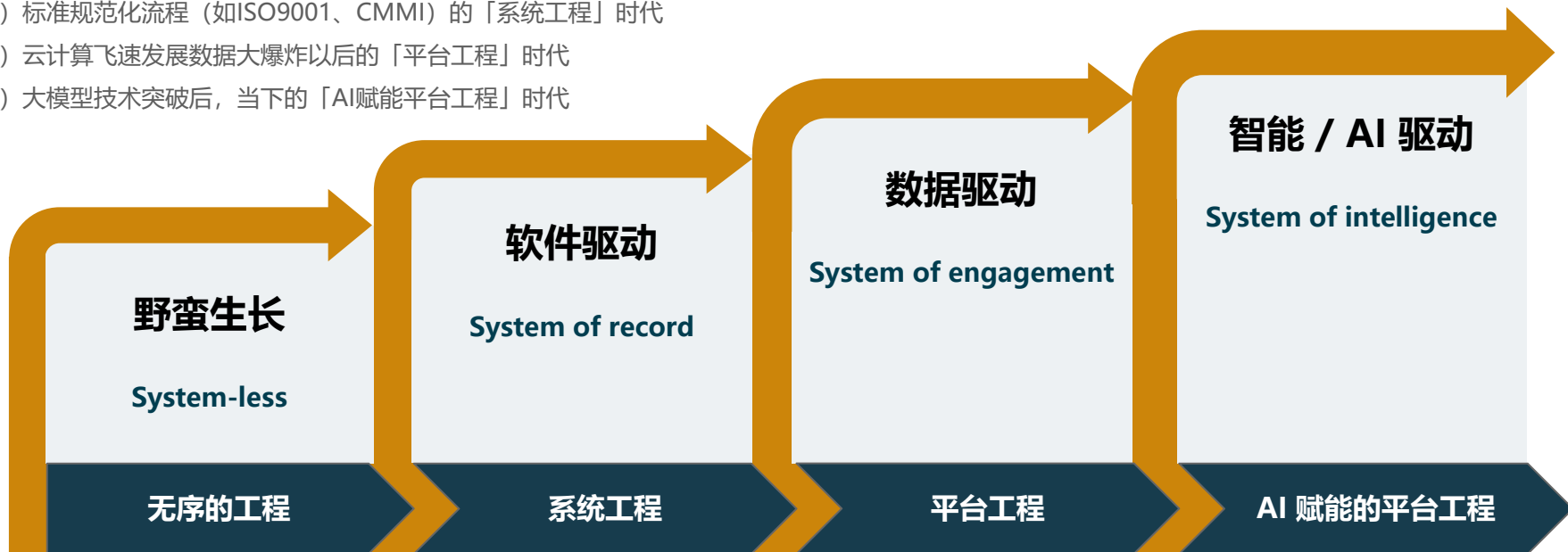
- 1 GenAI 开启研发提质增效新时代
- 2 BizDevSecOps打造提质增效飞轮
- 3 Platform Engineering 新基建
- 4 Thoughtworks研发效能转型框架



软件工程的演进

随着新技术浪潮的演进，软件工程也经历了如下的四个关键阶段：

- 1) 野蛮生长、手工作坊的「无序工程」时代
- 2) 标准规范化流程（如ISO9001、CMMI）的「系统工程」时代
- 3) 云计算飞速发展数据大爆炸以后的「平台工程」时代
- 4) 大模型技术突破后，当下的「AI赋能平台工程」时代



来源： [《Thoughtworks 人工智能+时代的研发范式演进》](#)



软件工程的「变」与「不变」

「变」

1 开发范式转变

- 从工程师主要编写代码 转变为 **通过模型生成代码**
- 从机器语言编程 转变为 **使用自然语言提示编程**
- 从面向对象编程技术栈 转变为 **面向大模型、深度学习的技
术栈**

2 交互范式转变

- 企业应用从以逻辑和过程为主转变为**以具有自主能力的AI Agent为主**
- 业务对企业应用的期待, 转变为 **AI Agent First**
- 从GUI图形界面交互, 转变为 **融合GUI和CUI、多模态融合的交互模式**

3 运营范式转变

- 软件研发流程从双模, 转变为 **更多模态 (如从需求到原型到可工作软件)**
- 从治理软件系统为主, 转变为 **治理大量的AI Agent**
- 研发交付团队趋向**更小型的全栈团队**; 研发组织整体人才结构从金字塔结构趋向转变为**纺锤形结构** (对简单编码任务的人员需求将大幅减少)

「不变」

1 知识转化过程

- 软件工程还是需要将人类在各专业领域的深厚经验和知识体系转换成机器可识别与利用的形态
- 如何在组织内部进行**高效的知识转化**, 依然是软件工程的
核心命题

2 问题定义和 方案设计能力

- 代码是一步步长出来的, **软件的复杂性及熵增**始终存在
- **问题定义和方案设计能力**依然是软件工程的核心能力, 具体如理解问题本质; 对业务需求的理解、分析和拆解; 对复杂系统的快速理解、架构设计和决策、设计取舍等

3 规模化后的 复杂治理

- 软件研发系统和团队的**规模整体的复杂度**依然存在, 甚至随着技术栈的多元化会日趋复杂
- 对于大规模软件系统, **架构、规范、安全的治理**依然是规模化科技组织的工作核心



GenAI使能的提质增效新视角

大模型给我们打开了一条通往“工程标准化”的新通道！

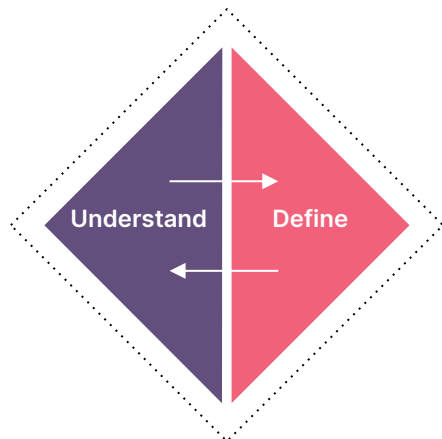


<https://china.architecture2030.org/>

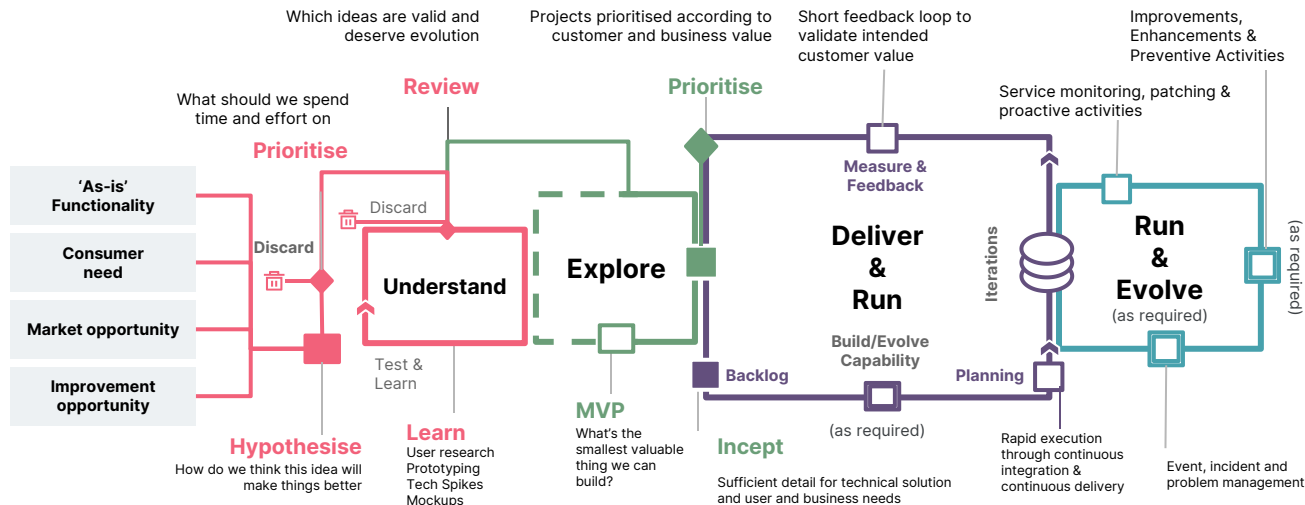


GenAI使能的提质增效新视角

BUILD THE RIGHT THING



BUILD THE THING RIGHT



AI应用带来的标准化可能





面向GenAI开创的新时代， 如何开启研发的提质增效？



BizDevSecOps

BizDevSecOps 成熟度标准是一套端到端的评估体系，它评估业务、研发、安全和运营等方面的整体成熟度。该标准吸收国际先进经验，同时考虑国内企业的特点进行适配，使之能够适应国内实际情况。通过的企业可以获得国内国际双认证。

中国电子学会

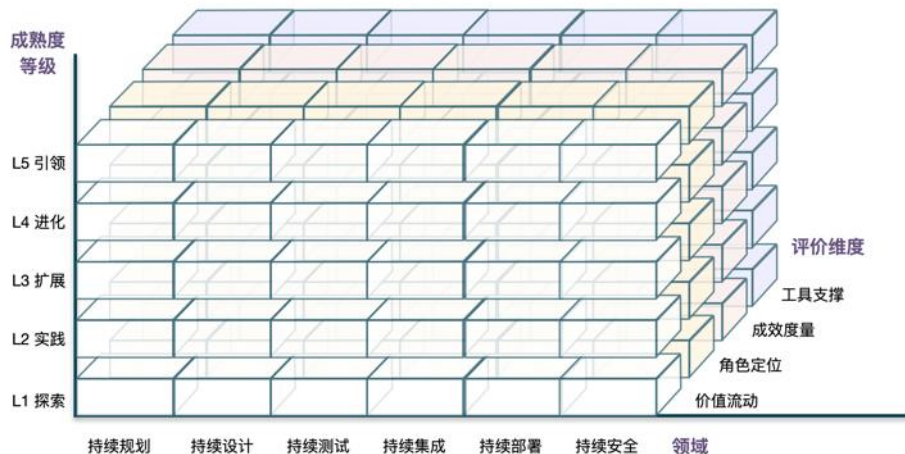
关于征集中国电子学会标准《业务研发安全运维（BizDevSecOps）成熟度模型》参编单位的通知

各有关单位：

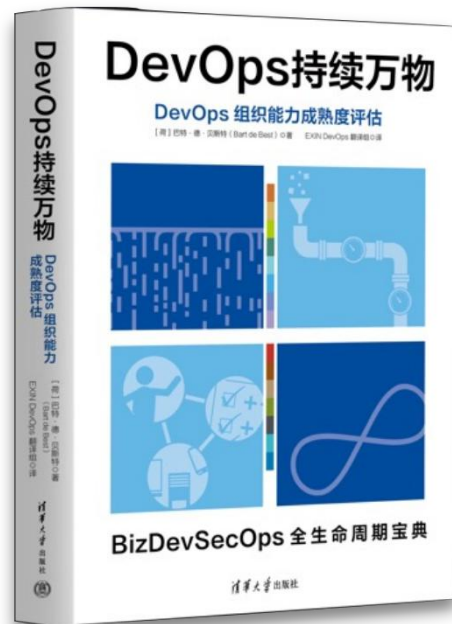
经中国电子学会标准化工作委员会批准，业务研发安全运维（BizDevSecOps）成熟度模型团体标准已完成立项（计划号：JH/CIE 230-2022）。现公开征集参编单位，共同完成标准制定工作。

请有意参加标准制定工作的单位，填写参编单位登记表（见附件），并于 2022 年 7 月 18 日前将加盖公章的电子版反馈至标准编制工作组。

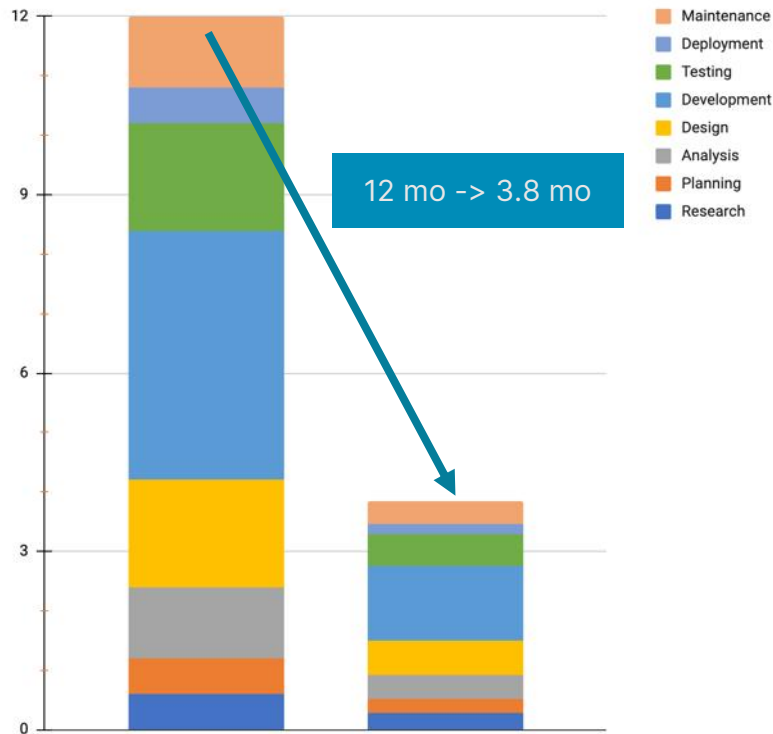
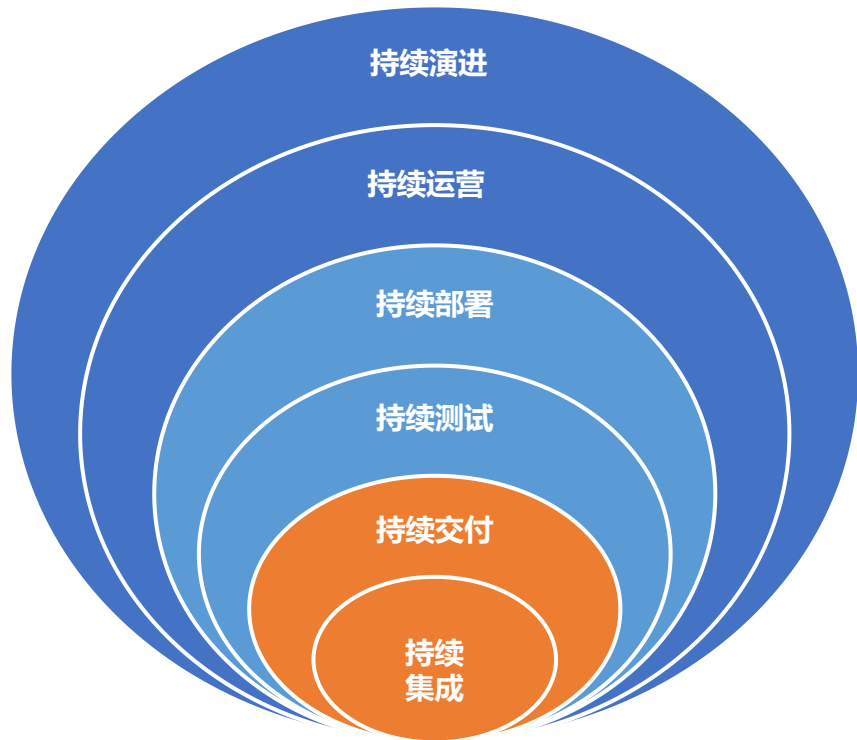
附件：参编单位登记表



《业务研发安全运营一体化能力成熟度模型》 BizDevSecOps Capability Maturity Model
(标准号: T/CIE186-2023)



“持续X” 的提质增效飞轮



来源: Thoughtworks全球分析





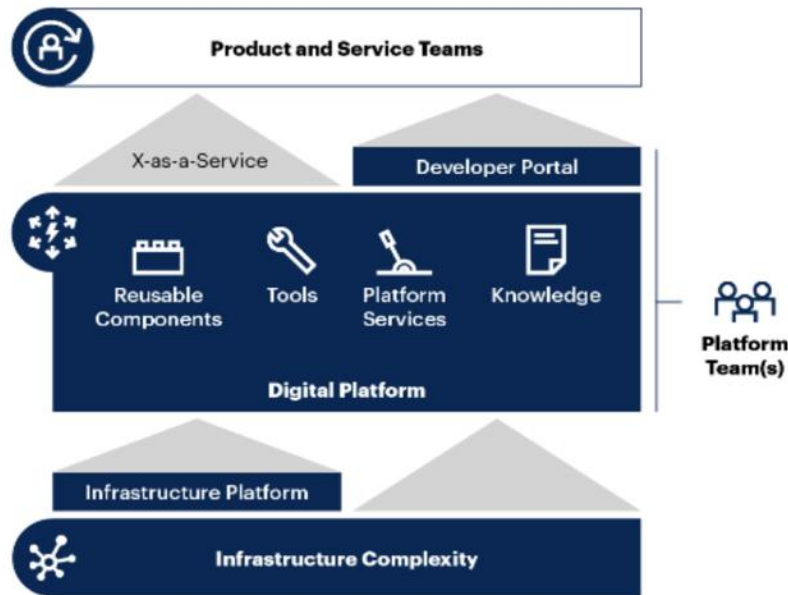
打造提质增效的飞轮， 研发组织需要做什么？



什么是 Platform Engineering?

- ... “Platform Engineering”
The practice of designing, building, and maintaining the underlying infrastructure that enables the development and delivery of software applications and services. It involves creating a self-service set of reusable components, tools, and services used by developers to build and deploy applications more quickly and efficiently.
- The goal of platform engineering is to provide developers with a consistent and reliable set of platform capabilities, so they can focus on meeting their customer needs, enabling organizations to move faster and be more agile in responding to changing market demands.

平台工程团队规模参考：Google 2024 年开发人员约有 27000+，Google 的工程生产力团队约有 2000 人，占 Google 开发人员的 7.4%

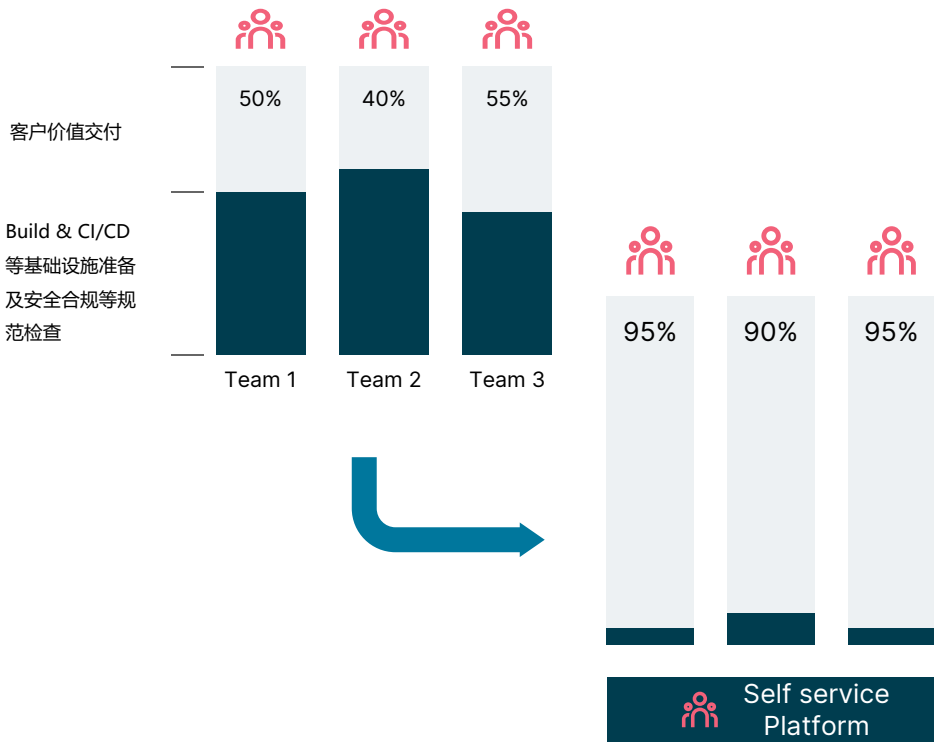


来源：[Gartner 《Gartner Top 10 Strategic Technology Tech Trends for 2024》](#)

为什么要提 Platform Engineering?

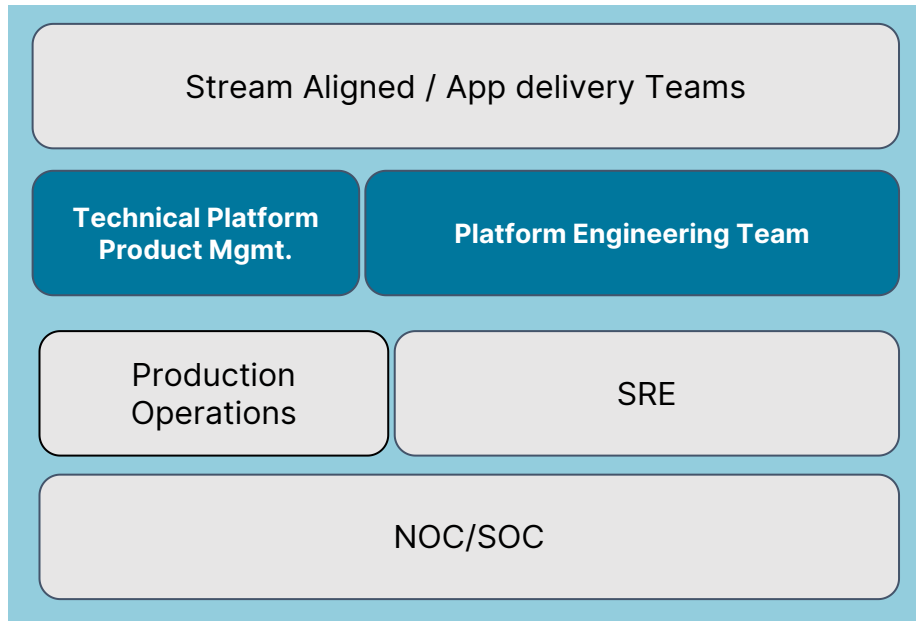


"We realized that the true problem, the true difficulty, and where the greatest potential is – is **building the machine** that makes the machine. In other words, it's building the factory. I'm really thinking of the factory like a product."

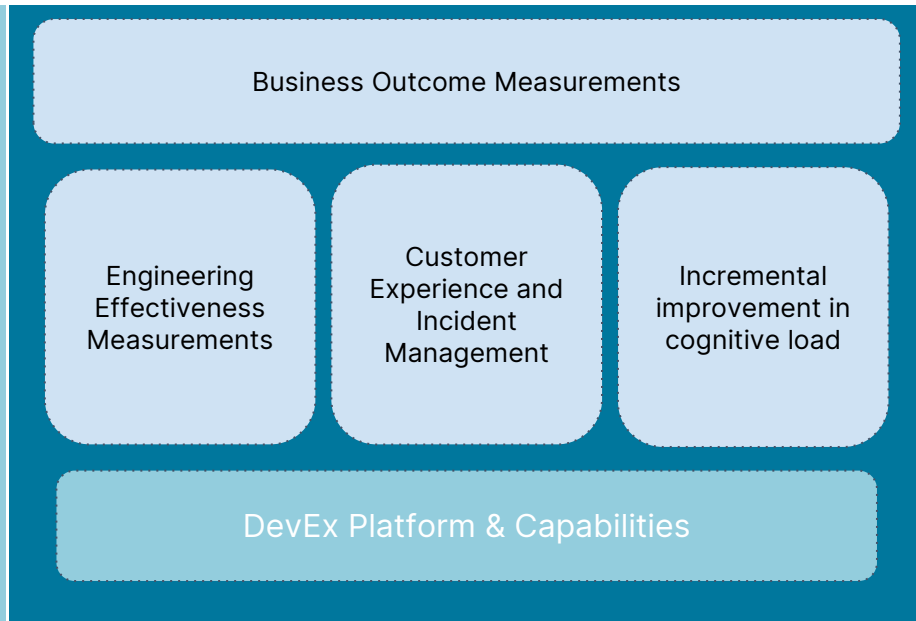


打造 Platform Engineering 新基建

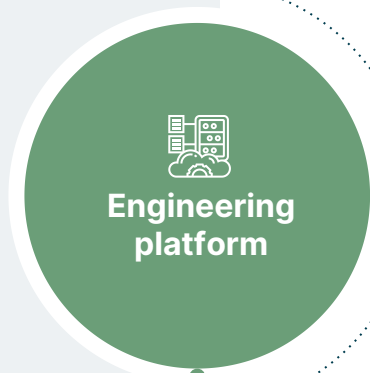
组织生态



产出方向



Thoughtworks 工程平台关键支柱



A technical product or platform which allows a product team member to function more effectively resulting into business agility with faster time to market



Self service

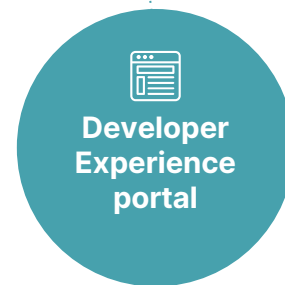
- On demand higher order infrastructure (Elastic Infrastructure)
- Infra as code with env provisioning pipelines
- Automated path to production (pipeline as code)
- Quality & security pipelines
- Monitoring and observability

Knowledge sharing hub with unified search across all resources, space for collaboration, On-boarding & off-boarding, Sensible defaults, Tutorials, Guides, Recipes for architecture, security & compliance

Discoverability

- Publish and discover
 - API gateway
 - Event hub
 - Data models
- API sdks
- On demand integration service

Real time dashboards of metrics such as 4km, SLI/SLO, Cloud cost insights, Team productivity, Sprint status etc.



Single pane of glass & a self service Portal for product teams leading to elevated developer experience

Thoughtworks研发效能转型方案框架

围绕企业端到端价值流的精益改进

平台工程

建立一套自助式的研发平台，支持端到端的软件开发和运维生命周期。该平台为团队提供了可复用的基础设施和能力，从而降低了开发人员的认知负载，使其能够专注于开发具有业务价值的客户功能，并更快地推向市场。

效能工程

一方面通过提供工具、框架、模式、组件库和脚手架等方式，提高开发人员体验；另一方面通过建立标准和规范、引入度量指标等方式，持续改进研发过程；最终达到提高研发效率的目的。

质量工程

建立自动化、体系化的质量保证策略和方法。包括但不限于基于“测试金字塔”的功能性测试策略；可靠性测试、混沌测试、韧性测试等非功能性的测试；从而确保软件产品的质量，降低缺陷率，减少测试时间和成本。



资产建模

旨在从成本控制和资源使用优化的角度管理和优化企业架构中的各种资产的方法和实践，这些资产包括软件代码、数据、服务和基础架构等。包括但不限于采用面向云架构设计优化成本和保持弹性伸缩、采用领域驱动设计方法确保业务逻辑与软件实现相一致等。从精益的角度看，这也被视为研发效能提升的一种表现。

知识建模

旨在帮助企业将知识和经验有效地捕获、组织和共享的方法和实践。包括但不限于通过统一信息检索和目录、统一文档标准规范、产品驱动的知识库、推动内部中心化知识社群（如 CoE 社区）等。它可以帮助团队进行知识和领导力的系统化的建设和管理，为团队创造端到端知识交流和学习的机会和平台，促进团队协作效率。

价值建模

旨在使得研发团队的运作与企业战略、目标和价值实现保持一致的方法和实践。包括但不限于围绕 BizDevOps 模型梳理团队的岗位职责、通过价值驱动建模和团队拓建模策略来优化团队协作和管理等手段来实现研发效能的提升，并推动组织的价值创造。

持续创新的进化型组织

战略对齐与目标管理体系

提高员工工作动力与目标感。确保所有部门的目标都与公司整体战略一致，有助于推动企业向着更加高效和有益的方向前进，实现更高的生产力和业务增长。

能力图谱和赋能体系

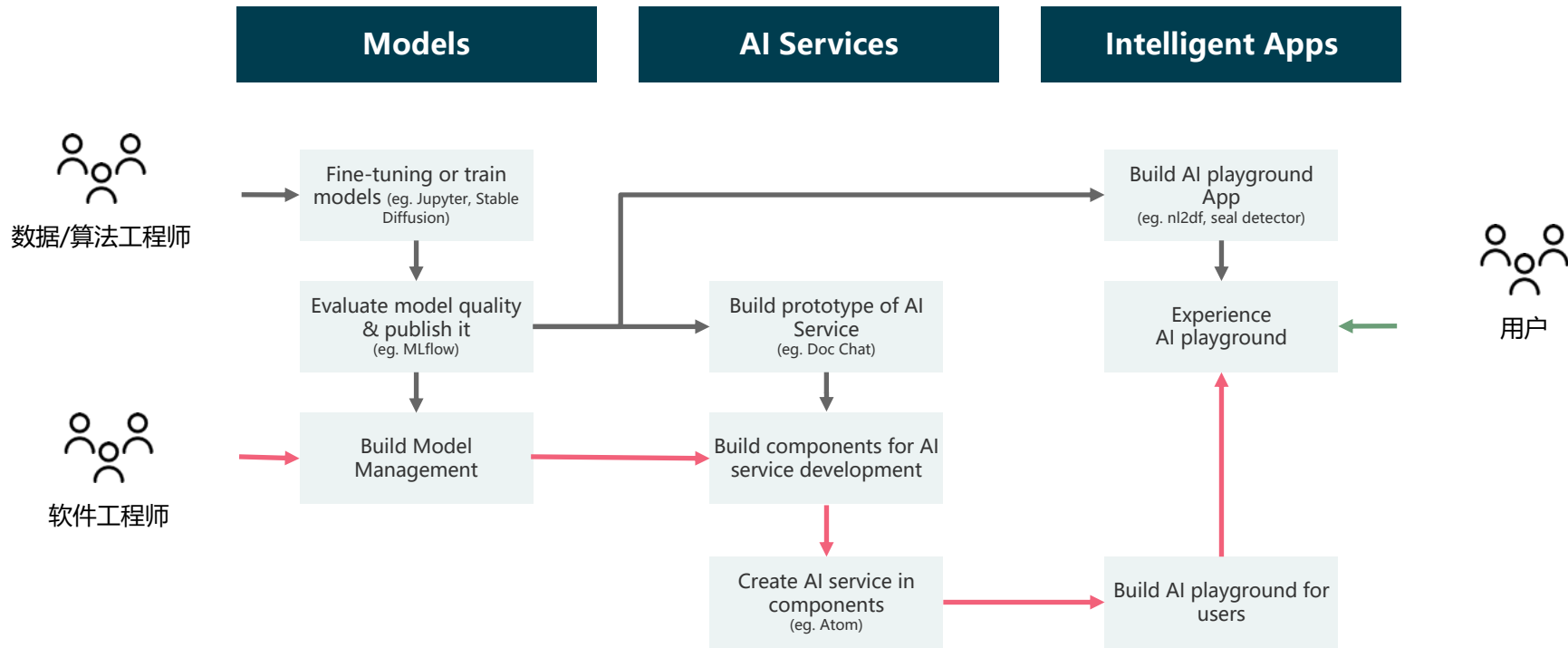
体系化地改善团队工程文化。提高团队成员的参与度，尽可能使得团队每一位成员都拥有端到端协作的技能，最大化团队整体的交付生产力。

数字化人才生态建设与发展体系

打造更精炼的高绩效团队。团队能够更快速地招募到合适的人才，并且储备更好的人才资源来源和配备。



最后一点：以人为本



源于社区 服务社区

THANKS!

