



# GOTC 2023

## 全球开源技术峰会

THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE

---

# OPEN SOURCE, INTO THE FUTURE #

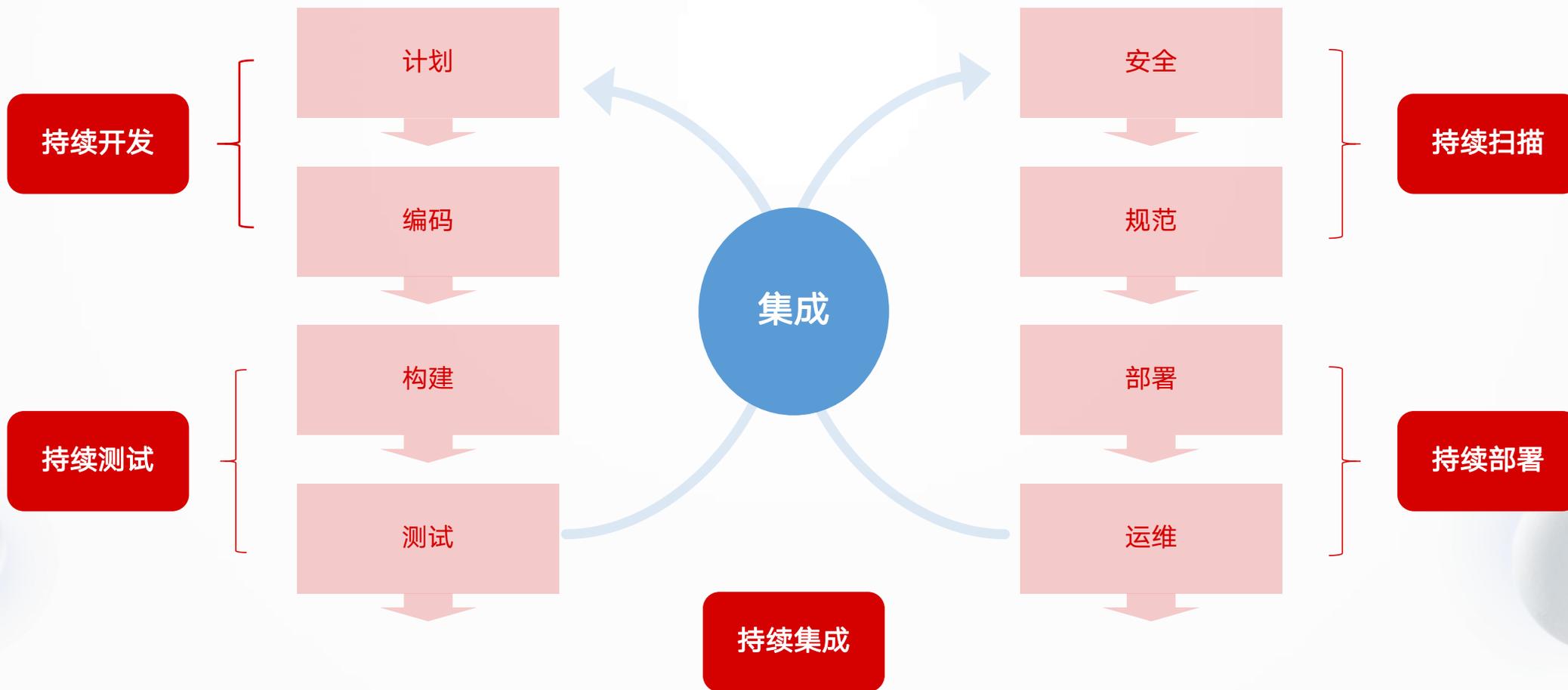
---

大型金融企业敏捷研发平台的建设实践与分享

建信金科 沈阳 2023年05月28日

# 平台建设历程

## DevOps的范围和作用



# 平台建设历程



## 里程碑



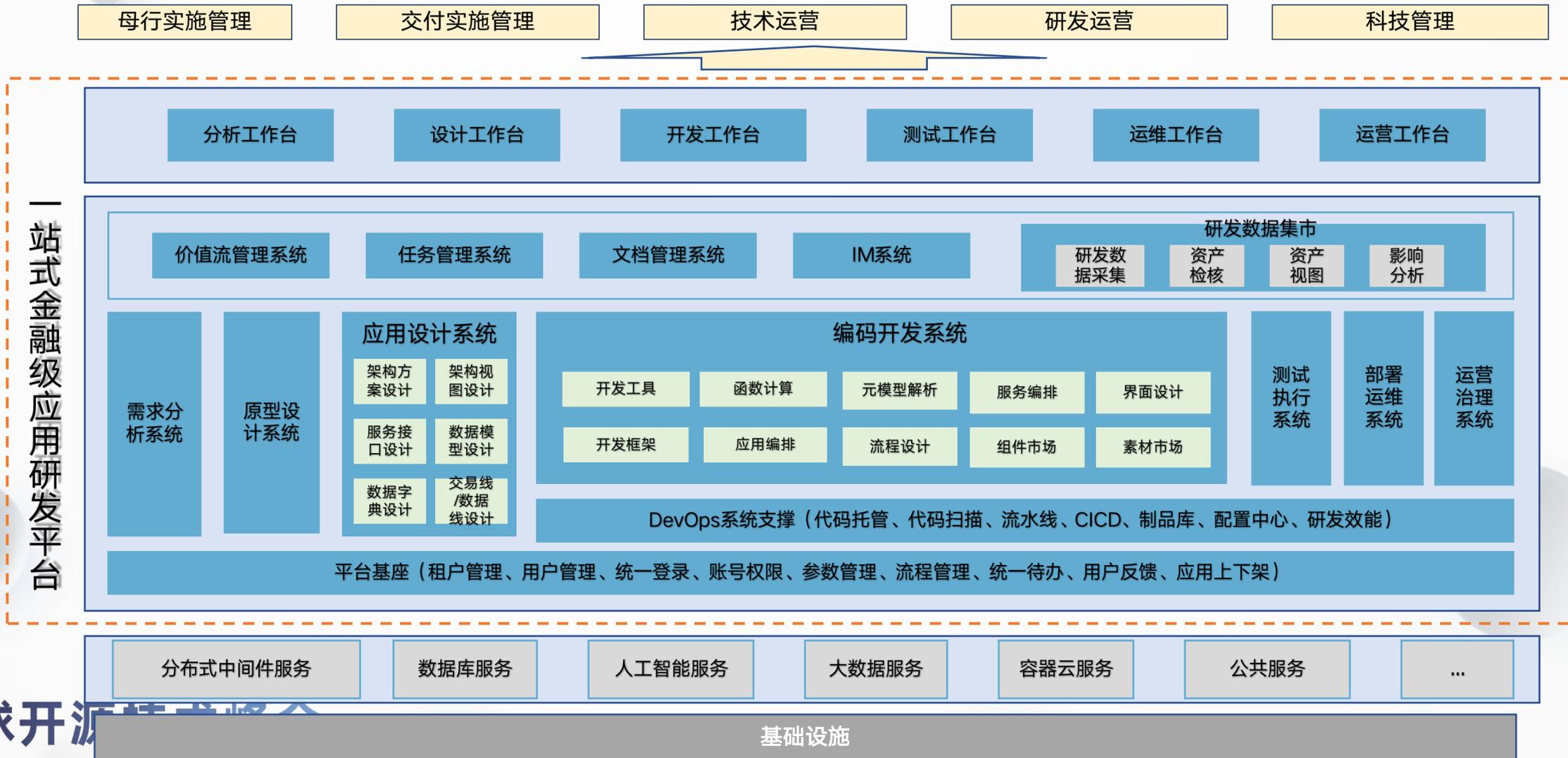
## 全球开源技术峰会

THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE

# 平台建设历程

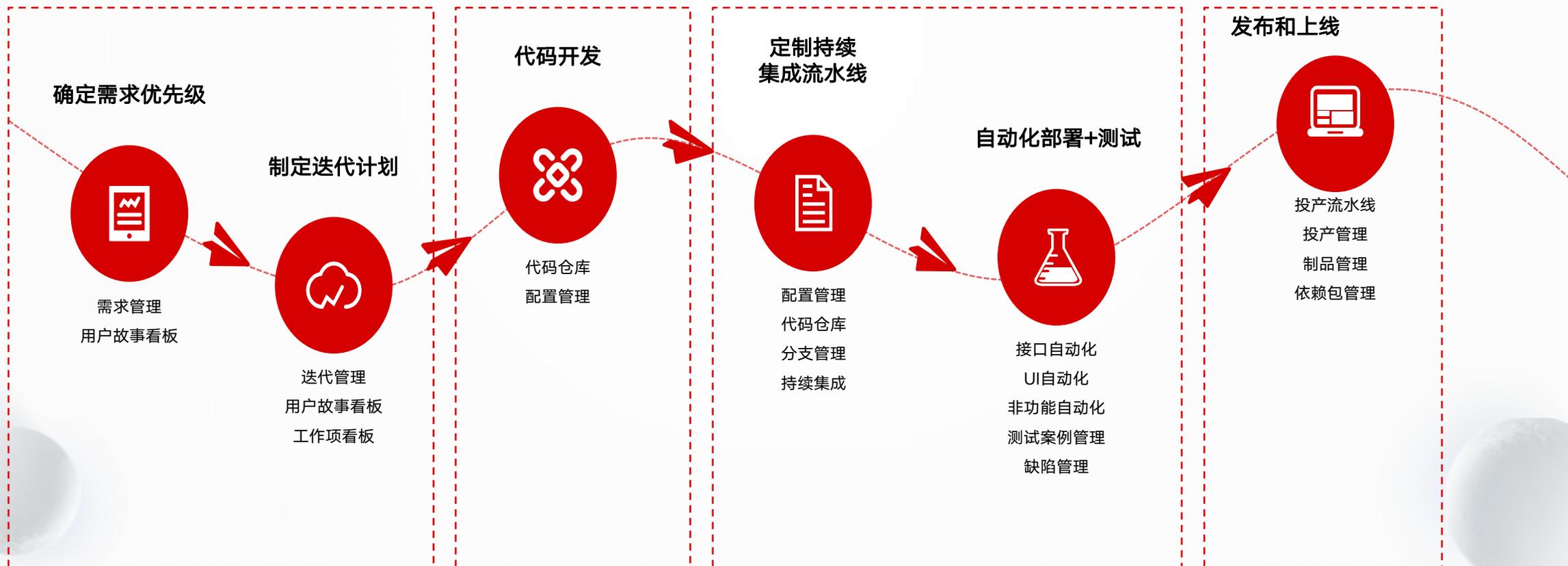


## 平台架构



一站式金融级应用研发平台

## 典型实践



敏捷研发管理

统一代码仓库

持续集成流水线

版本发布上线

## 技术特点



容器化和K8S管理，方便扩缩容和扩展



建设平台基座和研发框架，支持第三方参与开发



基于开源软件封装和定制



定义自动化工具接入的标准和规范



与管理系统松耦合集成



支持包含服务端、移动端研发在内的所有技术栈

## 统一 vs 多元

1

### 现状系统种类多，非标准

Java、NodeJS、C/C++、Go、Python、Android、IOS、区块链、AI…  
同样是Java，也有千差万别。版本、目录结构、分支模式、开发模式…  
代码仓库也不一样，CC、Git、SVN等，都要支持

2

### 工具链，使用的工具多样，如何支持？

Gradle、Maven、Pip、Yum、Ansible、统一部署平台、Jmeter、测试工具、扫描工具…  
种类多，版本多，工具多，怎么集成。

3

### 现存习惯

各事业群有自己的私服、编译器、依赖源…

## 研发体验 vs 环境限制

### 一、开发环境隔离

不能直连外网，不能直连生产

开发过程需要的各种依赖包、资源人肉建设

### 二、开发、功能测试、非功能测试网络隔离

### 三、事业群跨地域有自己的机房，跨域使用效率

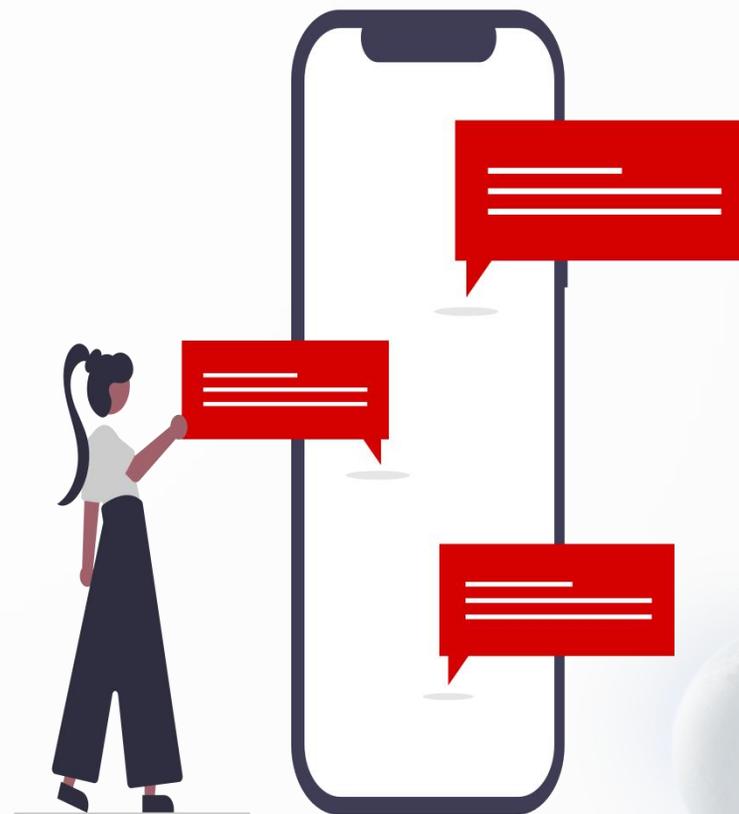
### 四、移动端、小程序的开发测试环境



# 困难与对策

## 全面推广 vs 支持压力

- 一、使用平台的用户和系统越多，需求和反馈就越多，支持工作也越来越大
- 二、平台团队是教练还是保姆？
- 三、有限的资源和无限的需求如何平衡



- 一、如何定位：目的是管理还是赋能。
- 二、如何推广：从上往下的考核，还是从下往上的推动。如何不反感？
- 三、不仅管过程，还要管资产



# 困难与对策

投入 vs 价值

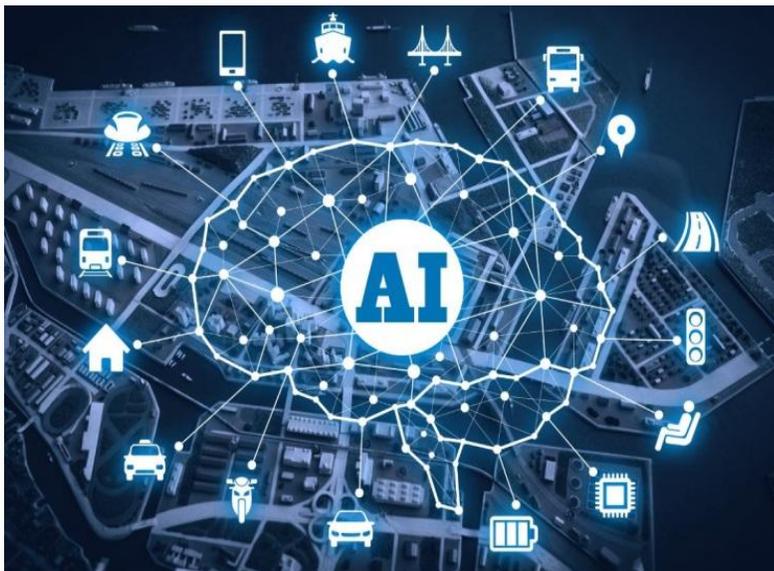
一、用户多、系统多，平台在企业内部如何运营平台？

二、如何衡量平台的效果？价值？



# 困难与对策

标准化 vs 新技术



越来越多的新技术，如何支持？

AI、区块链、物联网、小程序…



基于云原生的DevOps

全球开源技术峰会

THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE

产品打磨 vs 需求众多



## 1. 需求方多且强

业务部门、管理部门、研发团队、关键角色…

## 2. 计划赶不上变化

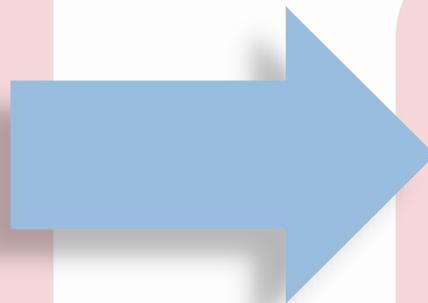
发展方向可能会受到影响和调整

## 3. 时间紧要求多

数据源还在分析，几百个度量指标的需求就来了

### 五大原则

- 快速交付
- 将几乎所有事情自动化
- 把所有的东西都纳入版本控制
- 内建质量
- 赋权于团队



### 实践核心

- 敏捷项目管理
- 持续集成
- 持续部署
- 持续交付
- 自动化测试
- 监控/度量反馈

# 体会与思考

一些感悟

GOTC



勇于尝试



携手外部



互相学习



持续改进

## 全球开源技术峰会

THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE

# 体会与思考

## 后续建设方向



## ALL IN IDE

### 触达和衔接

解决的是开发人员要同时面对多个系统、多套环境来回切换使用的问题，通过IDE这个门户来统一对接使用，体现的是IDE的整合能力

### 技术资产的组织使用

解决的是不同类型的技术资产，有了后端的研发平台来完成管理以后，还需要有一个对应的开发者直接可用的工具，来直观地获取、引用、组合、编排这些技术资产

### 资产使用的最佳实践组合

对于技术资产的最佳使用方式，需要将其通过特定的开发模式和工程化手段固化下来，而IDE则为这些开发模式和工程化提供高效的工具化支撑

# 全球开源技术峰会



# THANKS