

《微服务拆分设计规范指南》解读

中国信息通信研究院云计算与大数据研究所





目录

1

微服务架构应用现状

2

微服务架构适用场景

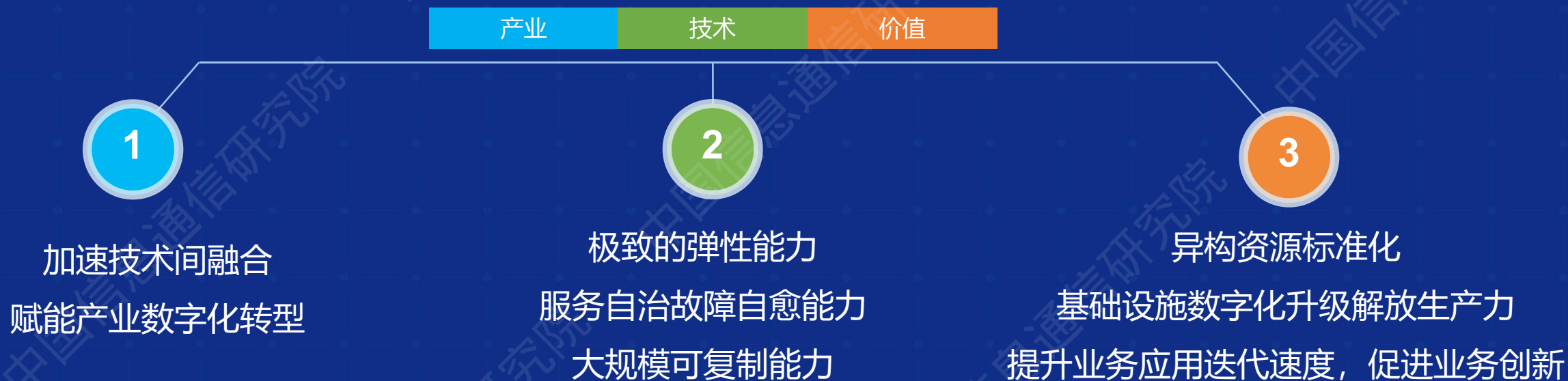
3

微服务拆分设计规范指南



重新认识云原生

云原生是**面向云应用**设计的一种思想理念，是能够充分**发挥云效能的最佳实践路径**，可帮助企业构建弹性可靠、松耦合、易管理可观测的应用系统，提升交付效率，降低运维复杂度。代表技术包括不可变基础设施、服务网格、声明式API及Serverless等。

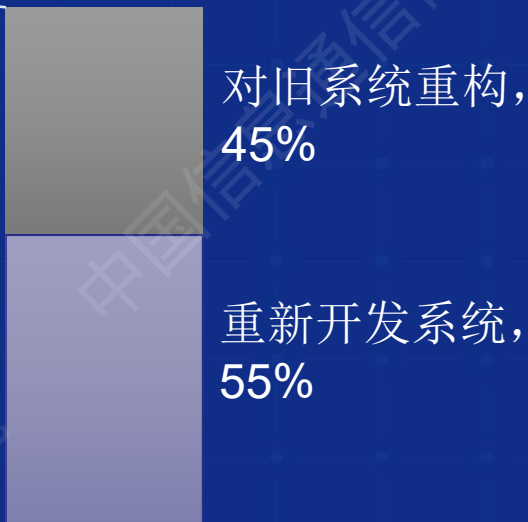
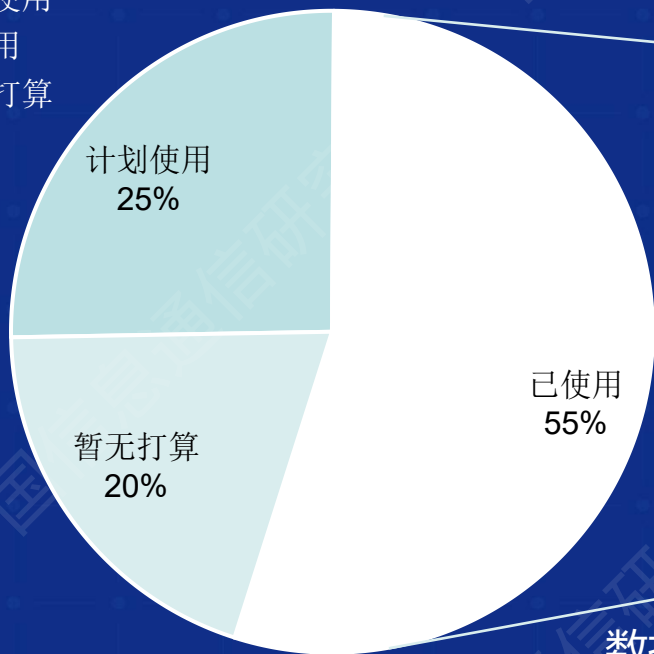


■ 微服务架构应用逐步落地，提升研发效率

云原生时代中，微服务架构应用逐步落地。目前市场上微服务架构的应用**占比已过半**，计划使用与已使用的比例**高达80%**。

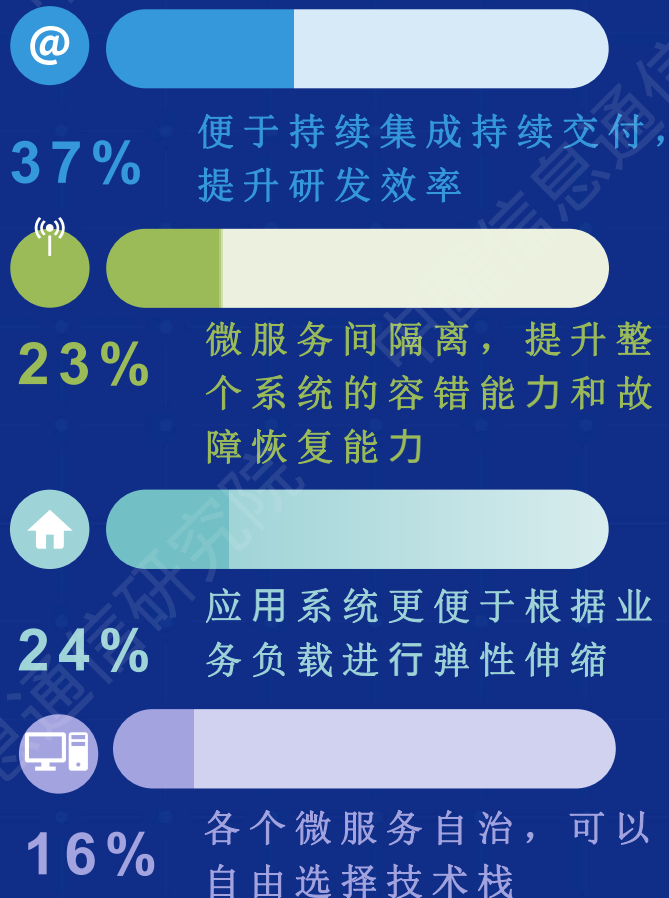
微服务架构使用情况

- 计划使用
- 已使用
- 暂无打算



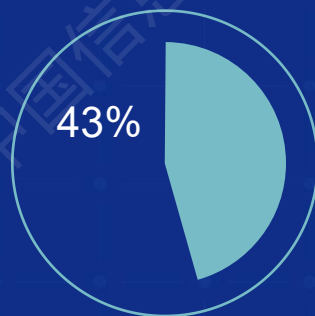
数据来源：《云原生用户调查问卷2021》

对应用系统开发方面进行提升



■ ■ ■ 微服务平台能力及拆分设计亟需规范

企业面临挑战 微服务架构应用开发



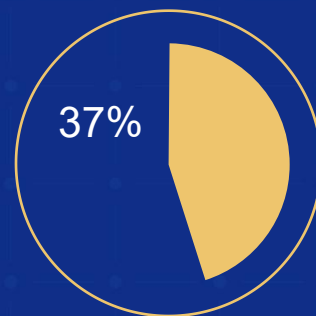
平台能力

缺乏平台提供微
服务管理能力



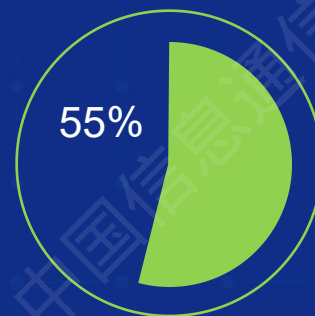
拆分设计

微服务拆分缺乏
标准化规范指引



组织架构

企业现有组织架
构不适合微服务
架构的开发



典型案例

微服务应用缺乏
行业典型案例

数据来源：《云原生用户调查问卷2021》



目录

1

微服务架构应用现状

2

微服务架构适用场景

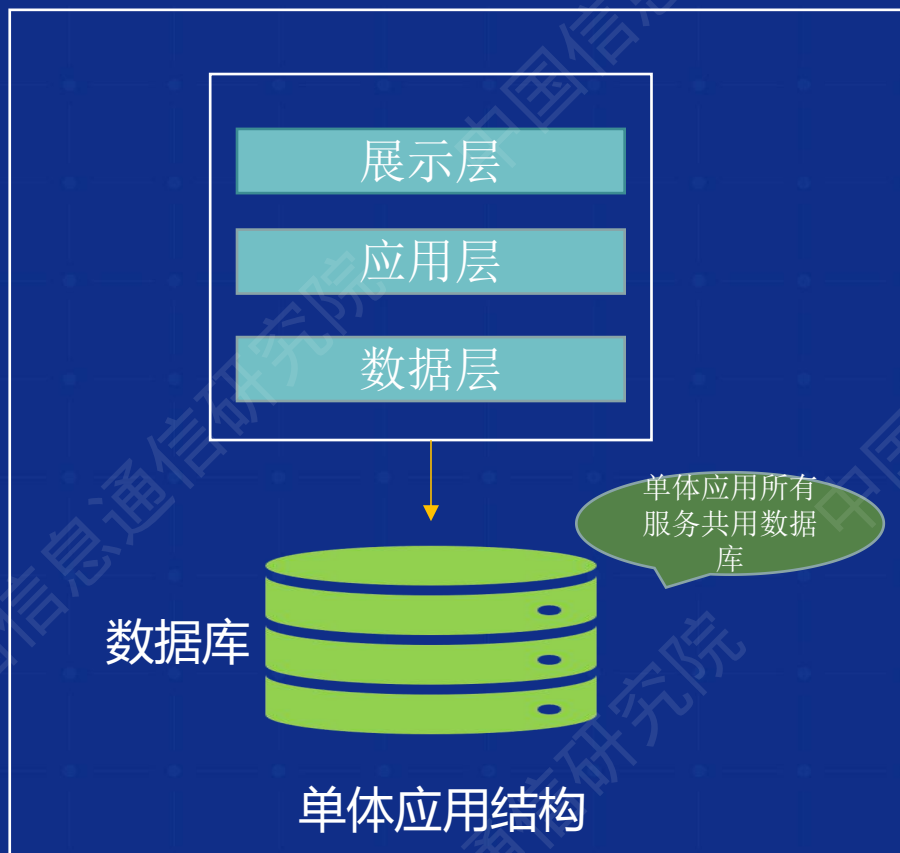
3

微服务拆分设计规范指南

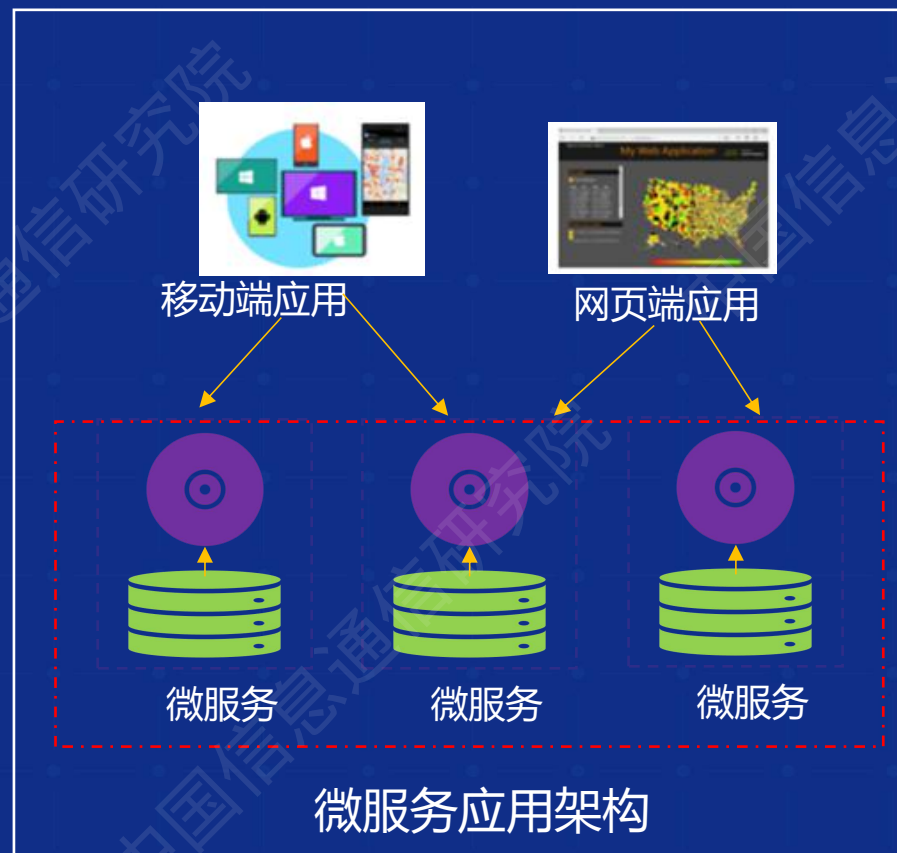


■ 微服务架构 vs 单体架构

单体应用架构向微服务架构的技术演变：微服务架构是一种架构模式，它提倡将单一应用程序划分成一组小的服务，服务之间互相协调、互相配合，为用户提供最终价值。微服务就是把一个单体项目，**拆分为多个微服务**，每个微服务可以**独立技术选型**，独立开发，独立部署，独立运维，并且多个服务互相协调，互相配合，最终完成用户的价值。



VS



■ ■ ■ 微服务架构 vs 单体架构

微服务架构下，每一个微服务应用都应专注于一项小而专的业务单元。微服务应用是实现技术异构性的同时，兼具可独立部署、高弹性、高可扩展性以及开放性等优势。

	单体架构	微服务架构
业务复杂度	适用于业务复杂度相对较低的应用	适用于业务复杂度相对较高的应用
运维复杂度	使用唯一部署包完成系统的部署和发布，系统运维复杂度相对较低	多个微服务应用分布式部署，服务应用实例可弹性扩缩，服务编排和系统运维复杂度大幅度提升
团队组织	以技术为中心的专职团队	以业务为中心的跨职能敏捷型团队
人才储备	技能单一	一专多能
业务变化的频繁度	适用于业务基本上很少发生变化，不需要经常迭代更新的场景。	业务经常变化，需要不断持续迭代的情况。

■ ■ ■ 微服务架构适用场景

01

业务快速开发

具有业务快速开发迭代的需求

02

业务规模庞大

系统模块关系复杂，耦合度高，单次上线更新无法快速完成

03

业务健壮性

单一服务的不可用不会影响系统其他服务的可用性

04

技术创新和异构

微服务应用相互独立，允许每个服务使用不同技术栈实现

05

服务复用

助力技术中台的落地，提供公共的微服务模块

06

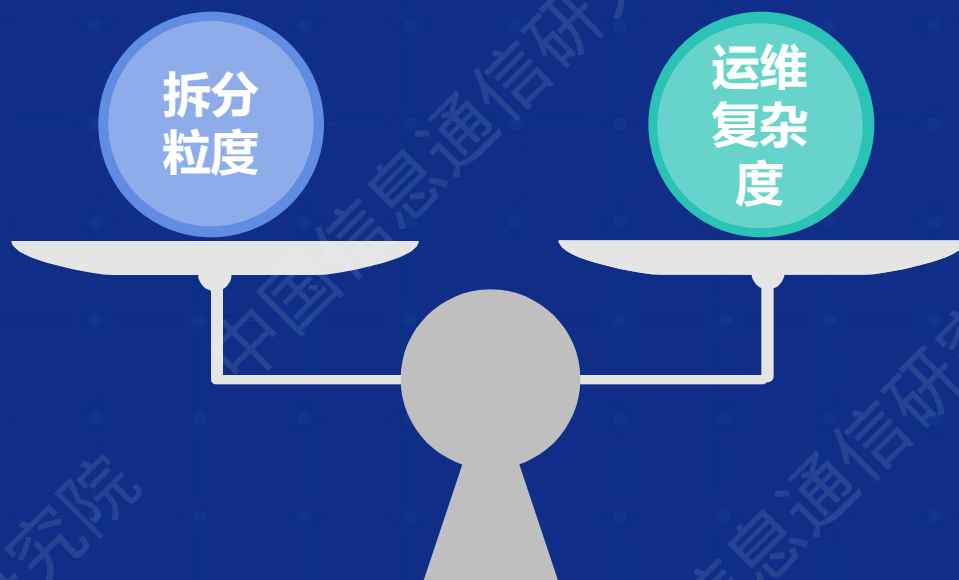
业务弹性需求

可对单独微服务应用进行扩缩容

■ 微服务拆分设计总体思想

将应用系统由单体架构或SOA架构转为微服务架构过程中，对应用系统按照一定原则规范进行服务拆分的过程。

- 高内聚、低耦合，降低系统复杂度
- 独立发布，独立部署



- 微服务数量巨大，将导致应用运维复杂度大幅度提高



目录

1

微服务架构应用现状

2

微服务架构适用场景

3

微服务拆分设计规范指南



■ 微服务拆分设计规范指南 - 目的

《微服务拆分设计规范指南》旨在从**业务、技术、组织、人才**等方面指导企业进行应用系统的微服务拆分设计，最大化发挥微服务架构的优势，为企业业务发展提供服务，解决企业在应用微服务架构中的挑战。

提供微服务拆分设计指导

推广微服务架构应用

解决微服务应用拆分痛点

评估微服务拆分效果

■ 微服务拆分设计规范指南 - 组织架构

在设计微服务团队组织架构时，也需要匹配微服务间的交互特点，如单一职责，高内聚低耦合等，并根据康威定律以及逆康威定律保持组织架构和应用系统的一致性。

01

敏捷管理团队

02

团队规模适中

03

团队负责少量服务

04

全生命周期负责

05

技术委员会

06

团队间有限合作

■ ■ ■ 微服务拆分设计规范指南 - 人才储备

建议企业/组织在确定进行微服务架构改造之前，预先评估企业相关人员的技术和经验水平，有意识地进行相关人才的储备和培养



■ ■ ■ 微服务拆分设计规范指南 - 总体原则

微服务拆分设计总体指导原则可在领域驱动设计（DDD）模型思想基础上进行，最终构建高内聚、低耦合的微服务架构应用系统。



围绕业务
建模



微服务自治



去中心化



可独立部署



单一职责



最终一致性

■ 微服务拆分设计规范指南 - 实施方法

微服务应用的架构设计可分为：**应用架构、数据架构、技术架构、部署架构、应用质量体系**五个视角，以应用架构为基础，数据架构和技术架构为开发依据，最后确定部署架构并考虑应用质量体系对应用架构的影响。



■ 微服务拆分设计规范指南 - 实施路径 - 业务需求分析

从业务角度对应用系统的需求和复杂度进行分析，目标是**划分出业务场景**，识别单一场景下包含的事件并为系统提供应用架构视图。



■ 微服务拆分设计规范指南 - 实施路径 - 微服务边界分析

以业务需求为基础，遵循高内聚低耦合的原则来划分微服务的边界。主要使用**四色建模法**和**事件风暴法**。无论使用哪种方法，在划分边界后还需根据技术需求调整以满足非功能性需求。

四色建模

1. 寻找可追溯事件
2. 确定对应的足迹和时标
3. 寻找时标周围的人事物
4. 确定人事物的角色
5. 补充相关信息
6. 合并时标对象
7. 划分限界上下文

事件风暴

1. 识别领域事件
2. 识别决策命令
3. 识别领域名词
4. 划分限界上下文

■ 微服务拆分设计规范指南 - 实施路径 - 技术需求分析

从非功能性角度出发，对应用的技术栈，业务负载以及安全性等需求进行分析，对微服务边界分析提供**辅助决策**，同时确定**系统功能之间的交互方式**。

应考虑系统功能变更频率的不同要求

变更
频率

应考虑系统功能对伸缩性或健壮性的不同要求

伸缩性

应考虑系统功能对安全性的不同要求

安全性

应考虑系统功能对技术选型的不同诉求

技术
选型

■ 微服务拆分设计规范指南 - 实施路径 - 依赖关系分析

通过分析限界上下文之间的逻辑关系，确定微服务之间的**依赖关系**。
依赖关系的识别有助于确定最终的实现优先级。

确定限界上下文
映射关系

确定微服务依赖
关系

避免循环依赖

确定系统总体视图

分析组织架构依赖
关系

■ ■ ■ 微服务拆分设计规范指南 - 开发规范

微服务定义



微服务所从属应用以及微服务的名字和其它扩展信息

微服务接口



遵循“见名知意”的原则，建议微服务接口符合Swagger规范

源码管理



关注代码所有权，建议根据团队规模分为强所有权、弱所有权以及集体所有权

配置管理



建立机制保证配置的简洁和有效性

部署管理

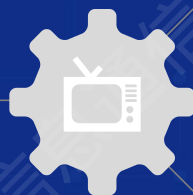


统一部署脚本，通过灰度发布、预发布环境等降低发布风险

文档管理



文档分级管理



■ ■ ■ 微服务拆分设计规范指南 - 可运维性评估

微服务拆分设计应当充分考虑服务拆分后企业现有**平台的支撑能力、服务治理能力、服务可维护性**等，从IT基础设施、服务治理能力、部署、监控、性能以及安全等方面考量微服务拆分的合理性。

基础设施
支撑能力

服务治理
能力

分布式事务

日志

部署发布
能力

变更能力

监控能力

故障处理
能力

安全能力

灾备能力

■ 微服务拆分设计规范指南 - 微服务拆分效果收益评估

微服务拆分效果收益评估应从是否达到预期目的出发，建议分为事前、事中以及事后三个阶段来开展评估。



事前

对预期的效果收益
设置相关目标，构
建较为完善的指标
体系



事中

通过设置若干检查
点对指标进行评估
以确认确实是否正
确，是否需要调整
现有实现方式



事后

系统正式上线后，
对系统状态和组织
管理进行考察，评
估实际服务能力提
升情况

■ ■ ■ 微服务拆分设计规范指南 - 微服务拆分效果收益评估

微服务拆分效果收益评估建议分为**服务治理收益**以及**组织管理收益**两部分





与我联系



尹夏梦雪

13901150020

yinxiamengxue@caict.ac.cn



云原生 产业大会

原生蓄力 云领未来

THANKS!

CAICT 中国信通院

