云原生技术实践白皮书

（2019 年）

（征求意见稿）

云计算开源产业联盟

OpenSource Cloud Alliance for industry，OSCAR

2019年4月

版权声明

本白皮书版权属于云计算开源产业联盟，并受法律保护。转载、摘编或利用其它方式使用本调查报告文字或者观点的，应注明“来源：云计算开源产业联盟”。违反上述声明者，本联盟将追究其相关法律责任。

编写说明

牵头编写单位：中国信息通信研究院
参与编写单位：华为技术有限公司、阿里云计算有限公司、腾讯云计算（北京）有限公司、毕威拓科技（中国）有限公司、蚂蚁金服、灵雀云、上海道客网络科技有限公司、中国电信股份有限公司智能网络与终端研究院、中兴通讯股份有限公司、国际商业机器（中国）有限公司、时速云、博云、烽火通信科技股份有限公司、无锡华云数据技术服务有限公司、四川长虹电器股份有限公司

编写组成员：

中国信息通信研究院：栗蔚、陈屹力、刘如明、闫丹、郑立
华为技术有限公司：俞岳、张红、高巍
阿里云计算有限公司：李鹏、穆寰、朱松
腾讯云计算（北京）有限公司：黄文俊，卢萌凯，罗茂政，张浩，肖雨浓
毕威拓科技（中国）有限公司:刘鹏
蚂蚁金服：李克鹏、俞仁杰、盛延敏
灵雀云：陈凯、刘嘉伟
上海道客网络科技有限公司：郭峰
中国电信智能终端研究院：陈泳、乔宏明、毛斌宏
中兴通讯股份有限公司：余鹏
国际商业机器（中国）有限公司：郭迎春
时速云：涂家英、魏巍、王金秀
博云：赵安全、李守超、邱世达、刘对
烽火通信科技股份有限公司：田兵
无锡华云数据技术服务有限公司：吴涛
四川长虹电器股份有限公司：李强

前 言

随着云原生技术理念的在行业内进一步的实践发展，云原生架构完成了 IT 架构在云计算时代的进化升级。以 CI/CD、DevOps、微服务架构为代表的云原生技术以其高效稳定、快速响应的特点驱动引领企业的业务发展，帮助企业构建更加适用于云上的应用服务。对企业而言，新旧 IT 架构的转型与企业数字化的迫切需求也为云原生技术提供了很好的契机，云原生技术在行业的应用持续深化。

本白皮书重点介绍云原生技术概念、技术实践以及发展趋势。本书首先梳理了云原生技术理念特点以及与传统架构的对比，然后分析了深度学习、区块链、边缘计算和传统行业的互联网化应用等典型应用场景的特征，旨在从架构、研发流程等角度为企业或组织从传统单体架构过渡到云原生架构提供现参考。本书文末对于云原生技术的发展趋势做了进一步探讨，并给出了在云原生技术体系里的标准化建议。

目 录

版权声明 2

一、 云原生简介 1

(一) “云原生”为何而生? 1

(二) 初识云原生 2

(三) 我国云原生产业现状 3

二、 云原生架构的关键技术 4

(一) 云原生架构的关键技术 4

(二) 云原生架构的典型技术特征 8

(三) 云原生应用的优势 8

三、 云原生技术的典型应用场景 10

(一) 深度学习应用场景 10

(二) 区块链应用场景 12

(三) 边缘计算场景 14

(四) 传统行业互联网化应用场景 16

四、 云原生产业发展趋势 17

(一) 云原生助力人工智能从云端扩展至边缘 17

(二) 运营商搭载云原生强化 5G 网络能力 18

(三) 云原生领域的标准化进程将持续深化 19

(四) 联盟组织在云原生产业中的角色日趋重要 20

附录: 云原生落地案例 21

(一) 招商银行数字化转型实例 21

(二) 云原生技术在网商银行核心系统的应用实例 25

(三) 云原生技术助力广汽丰田数字化转型实践 26

(四) 中石油梦想云平台在能源行业的落地应用 29

一、 云原生简介

(一) “云原生”为何而生？

过去十年，云计算技术风起云涌，云的形态也在不断演进。虚拟化技术助推物理资源上云，然而基于传统技术栈构建的应用包含了太多开发需求（后端服务、开发框架、类库等），但传统的虚拟化平台只能提供基本运行的资源，云端强大的服务能力红利还并没有完全得到释放。从直接具备应用开发环境的 PaaS 平台，再到万物皆服务的 Serverless 平台，云端提供服务逐渐丰富而强大，用户管理的功能则逐渐下沉化繁为简。云平台可提供整套的开发环境，使用户专注在价值密度更高的业务逻辑开发上。

另一方面，云平台日益发展丰富的同时，应用开发架构也应逐渐演进去适应云平台，以便充分发挥云平台的能力。云上的应用应该如何适应现有的云计算架构？云原生的理念应运而生，Pivotal 的 Matt Stine 于 2013 年首次提出云原生的概念，并一直延续使用至今。

经过几年的发展，云原生的理念不断丰富和落地。中国信息通信研究院经过多方面研究，总结云原生的概念为“适合云的应用”和“好用的云架构”。基于云原生的技术和管理方法，更好地把业务生于云或迁移到云平台，从而享受云的高效和持续的服务能力。

(二) 初识云原生

云原生是一系列云计算技术体系和企业管理方法的集合，既包含了实现应用云原生化的方法论，也包含了落地实践的关键技术。云原生应用利用容器、服务网格、微服务、不可变基础设施和声明式 API 等代表性技术，来构建容错性好、易于管理和便于观察的松耦合系统，结合可靠的自动化手段可对系统做出频繁、可预测的重大变更，让应用随时处于待发布状态，云原生技术有利于各组织在公有云、私有云和混合云等新型动态环境中，构建和运行可弹性扩展的应用，借助平台的全面自动化能力，跨多云构建微服务，持续交付部署业务生产系统。[1]

快速响应市场需求已经成为企业竞争的决胜因素，持续交付使开发人员可以在短时间存在的特性分支上工作，定期向主干合并，同时始终让主干保持可发布状态，能做到在正常工作时段里按需进行一键式发布，提升开发的效率。但是复杂传统应用的单体架构模式在代码维护与集成编译方面困难重重，难以做到持续交付。微服务架构的引入使复杂应用的持续交付成为可能，服务拆分可以是多个业务团队并行开发的基础，微服务把同一个小业务的人员汇聚在一起，进一步加速了开发效率。

在部署方面，虚拟机分钟级的弹性不再满足快速扩缩容的需求，更加轻量级的容器技术成为微服务部署的最佳载体。容器技术很好的

[1] 引自 CNCF 对云原生的定义

解决了应用移植过程的环境一致性问题，使微服务实现快速弹性的部署。敏捷开发带来应用的快速迭代，同时也增加了版本发布的风险与业务运维的复杂度。DevOps 理念提倡开发、测试、运维之间的高度协同，从而在完成高频率部署的同时，提高生产环境的可靠性、稳定性、弹性以及安全性，这在很大程度上消除了频繁发布的风险。

图 1：基础架构，交付和应用之间的关系图

(三) 我国云原生产业现状

云原生理念在我国经过几年的推广普及，已经逐步为企业接受，云原生产业已步入快速发展期。过去两年，容器技术的应用持续深化，以容器及其编排技术为核心的生态，逐渐扩展至涵盖微服务、DevOps、服务监测分析、应用管理的完整闭环。

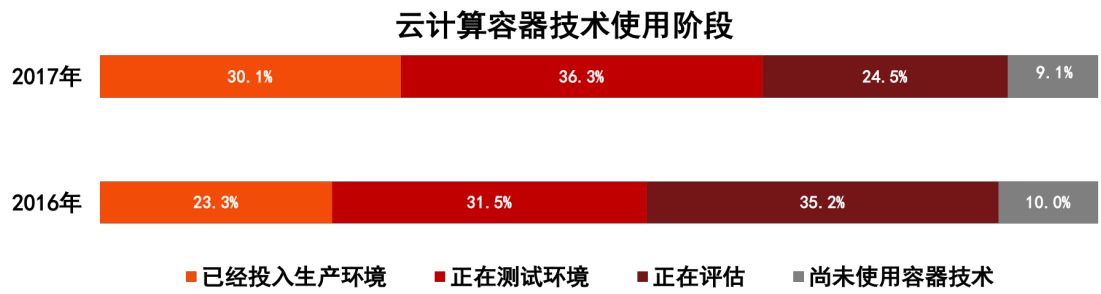

图 2：云计算容器技术采纳率[2]

华为云、阿里云、腾讯云等巨头云服务商以强大的综合云服务能力推动着云原生技术的发展变革，细分生态领域的企业级产品服务也不断涌现，提供更加聚焦的精细化服务。在过去几年中国企业的开源社区贡献率持续增长，不断有新的开源项目反哺社区，已成为国际开源社区的重要力量。

二、 云原生架构的关键技术

(一) 云原生架构的关键技术

1. 容器技术

容器是一种轻量级的虚拟化技术，能够在单一主机上提供多个隔离的操作系统环境[3]，通过一系列的 namespace 进行进程隔离，每个容器都有唯一的可写文件系统和资源配额。容器技术分为运行时和编排两层，运行时负责容器的计算、存储、网络等，编排层负责容器集群的调度、服务发现和资源管理。

[2] 数据来源：《中国私有云发展调查报告（2018 年）》
[3] 定义引自《中国云计算开源产业发展白皮书第二部分:基于容器技术的产业》

容器服务提供高性能可伸缩的容器应用管理服务，容器化应用的生命周期管理可以提供多种应用发布方式。容器服务简化了容器管理集群的搭建工作，整合了调度、配置、存储、网络等，打造云端最佳容器运行环境。**使用容器技术，用户可以将微服务及其所需的所有配置、依赖关系和环境变量打包成容器镜像，轻松移植到全新的服务器节点上**，而无需重新配置环境，这使得容器成为部署单个微服务的最理想工具。

2. 微服务

微服务是指将大型复杂软件应用拆分成多个简单应用，每个简单应用描述着一个小业务，系统中的各个简单应用可被独立部署。[4]**各个微服务之间是松耦合的，可以独立地对每个服务进行升级、部署、扩展和重新启动等流程，从而实现频繁更新而不会对最终用户产生任何影响。**相比传统的单体架构，微服务架构具有降低系统复杂度、独立部署、独立扩展、跨语言编程等特点。

与此同时，架构的灵活、开发的敏捷同时带来了运维的挑战。微服务框架作为微服务开发和运行治理的必要支撑，帮助实现微服务注册、发现、治理等能力，目前，在微服务技术架构实践中主要有侵入式架构和非侵入式架构两种实现形式。侵入式架构是指服务框架嵌入程序代码，实现类的继承，其中以 Spring Cloud 最为常见。非侵入式架构则是以代理的形式，与应用程序部署在一起，接管应用程序的

[4] 定义引自《云计算发展白皮书 2018 年》第二章-云计算发展特点

网络且对其透明，以服务网格为代表。在中国信息通信研究院制定的《分布式应用架构技术要求第一部分: 微服务平台》中也对这两种架构进行了详细的描述，并提出了相应的技术要求。

- Spring Cloud

Spring Cloud 是一系列框架的有序集合。它利用 Spring Boot 的开发便利性巧妙地简化了分布式系统基础设施的开发，如服务注册发现、配置中心、消息总线、负载均衡、断路器、数据监控等，都可以用 Spring Boot 的开发风格做到一键启动和部署。Spring Cloud 并没有重复制造轮子，它只是将目前各家公司开发的比较成熟、经得起实际考验的服务框架组合起来，通过 Spring Boot 风格进行再封装屏蔽掉了复杂的配置和实现原理，最终给开发者留出了一套简单易懂、易部署和易维护的分布式系统开发工具包。

- Service Mesh

Service Mesh 处理服务间请求/响应的可靠传递，并可用于服务治理、遗留系统的零侵入接入以及异构框架开发的微服务。Service Mesh 作为服务间通信的基础设施层，是应用程序间通讯的中间层，实现了轻量级网络代理，对应用程序透明，解耦了应用程序的重试/超时、监控、追踪和服务发现。Service Mesh 的开源软件包括 Istio、Linkderd、Envoy、SOFAMesh、Dubbo Mesh 等。

3. DevOps

DevOps 是一组过程、方法与系统的统称，用于促进开发（应用程序/软件工程）、技术运营和质量保障（QA）部门之间的沟通、协作与整合。它的出现是由于软件行业日益清晰地认识到：为了按时交付软件产品和服务，开发和运营工作必须紧密合作。[5]

DevOps 旨在统一软件开发和软件操作，与业务目标紧密结合，在软件构建、集成、测试、发布到部署和基础设施管理中大力提倡自动化和监控。**DevOps 的目标是缩短开发周期，增加部署频率，更可靠的发布。**用户可通过完整的工具链，深度集成代码仓库、制品仓库、项目管理、自动化测试等类别中的主流工具，实现零成本迁移，快速实践 DevOps。

DevOps 帮助开发者和运维人员打造了一个全新空间，构建了一种通过持续交付实践去优化资源和扩展应用程序的新方式。DevOps 和云原生架构的结合能够实现精益产品开发流程，适应快速变化的市场，更好的服务企业的商业目的。

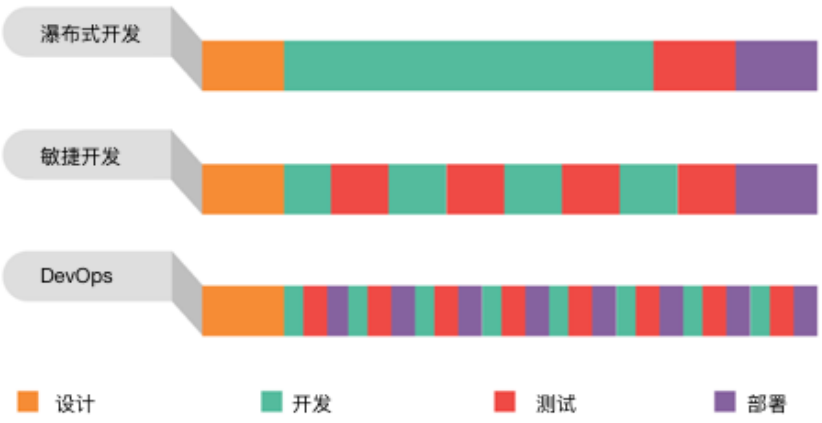

图 3：DevOps 与传统瀑布式开发、敏捷开发的开发周期对比

[5] 引自《研发运营一体化（DevOps）能力成熟度模型》标准中对 DevOps 的定义。

(二) 云原生架构的典型技术特征

云原生架构具有如下典型技术特征:

采用轻量级的容器。云原生应用程序是打包为轻量级容器的独立自治服务的集合。与虚拟机相比容器可以实现更加快速的扩展，优化基础架构资源的利用率。

设计为松散耦合的微服务。属于同一业务的服务通过微服务框架相互发现，它们可作为独立的服务而存在，并利用弹性基础架构和应用架构进行高效扩展。

通过 API 进行交互协作。云原生服务使用轻量级 API 进行交互，比如基于 RESTFul、gRPC 或 NATS 等协议。

使用最佳语言和框架开发。云原生应用的每项服务可以使用最适合该功能的语言和框架开发。云原生应用程序是多语言的;服务使用各种语言、运行时和框架。开发微服务的细粒度方法使各个服务能够为特定功能选择最佳语言和框架。

通过 DevOps 流程进行管理。云原生应用的每项服务都有一个独立的生命周期，通过敏捷的 DevOps 流程进行管理。多个持续集成/持续部署流水线可以协同工作以部署和管理云原生应用程序。

(三) 云原生应用的优势

云原生应用可充分利用云平台服务优势。云原生应用可以快速构建并部署到平台上，平台提供了简单快捷的扩展能力并与硬件解耦，

提供了更大的灵活性、弹性和跨云环境的可移植性。云原生将云计算作为竞争优势，原生云意味着将云目标从 IT 成本节约转向业务增长引擎。

云原生应用使团队专注于弹性设计。在传统的旧基础设施故障时，服务是可能会受到影响的。在一个云原生的世界中，团队特别关注于为弹性和高可用进行架构设计。云原生焦点帮助开发人员和架构师设计不受环境中故障影响的在线系统，快速弹性的重建和保持系统可用。

云原生应用具备多云间扩展的灵活性。公有云的厂商绑定现象一直为人诟病，但是使用支持云原生技术的云平台，企业可以将构建在任何(公有或私有)云上的应用快速迁移，兼具不同云服务商的优势服务能力无需担心锁定。

	部署 可预测性	抽象性	弹性能力	开发运维模式	服务架构	恢复能力
云原生应用	可预测	操作系统 抽象	弹性调度	DevOps	微服务 解耦架构	自动化运维 快速恢复
传统应用	不可预测	依赖 操作系统	资源冗余多 缺乏扩展能力	瀑布式开发 部门孤立	单体耦合 架构	手动运维 恢复缓慢

表 1：云原生应用与传统应用的对比

三、 云原生技术的典型应用场景

云原生技术涵盖的范围非常广泛，针对不同的应用场景，云原生解决的关键问题以及对应的技术栈选择都各不相同。人工智能、区块链等新兴技术领域对使用者的能力要求较高，复杂的服务依赖配置使的这些技术难以赢得更多受众。云原生技术在这些场景的应用很大程度上降低了技术的使用门槛，为新兴技术的快速普适推广铺平了道路。

同时，在互联网+和新商业业态的冲击下，传统行业正处于技术架构转型的十字路口，天然基于云服务的云原生模式无疑能给出重要参考意义。如何落地云原生技术正逐步成为行业用户的焦点。

(一) 深度学习应用场景

深度学习领域需要处理的三个核心问题是性能、效率和成本。利用云原生技术，形成以容器服务为核心，以云原生技术作为基础架构的深度学习解决方案，无缝的整合了云的计算、存储、负载均衡等服务，同时贯穿了深度学习的全生命周期。

深度学习本质上是一个实验科学，需要不断地组合和尝试不同的算法和类库。深度学习软件版本迭代非常快速，新算法层出不穷。TensorFlow、PyTorch 等深度学习框架和各种新算法，公有云平台如何帮助用户快速搭建并投入实验研究，是用户首要的关注点。深度学习需要海量的计算力，但是 GPU 资源昂贵，更低成本的共享资源的高效利用方式是场景普适的瓶颈。对于一个深度学习的试验周期，可以

分为四个部分：准备数据、模型开发、模型训练阶段和模型推理。每个阶段有不同的工作任务，用户便捷的的使用深度学习平台，以便灵活的处理阶段性任务也是重要考量。借助云原生技术，上述问题在很大程度上得以解决。

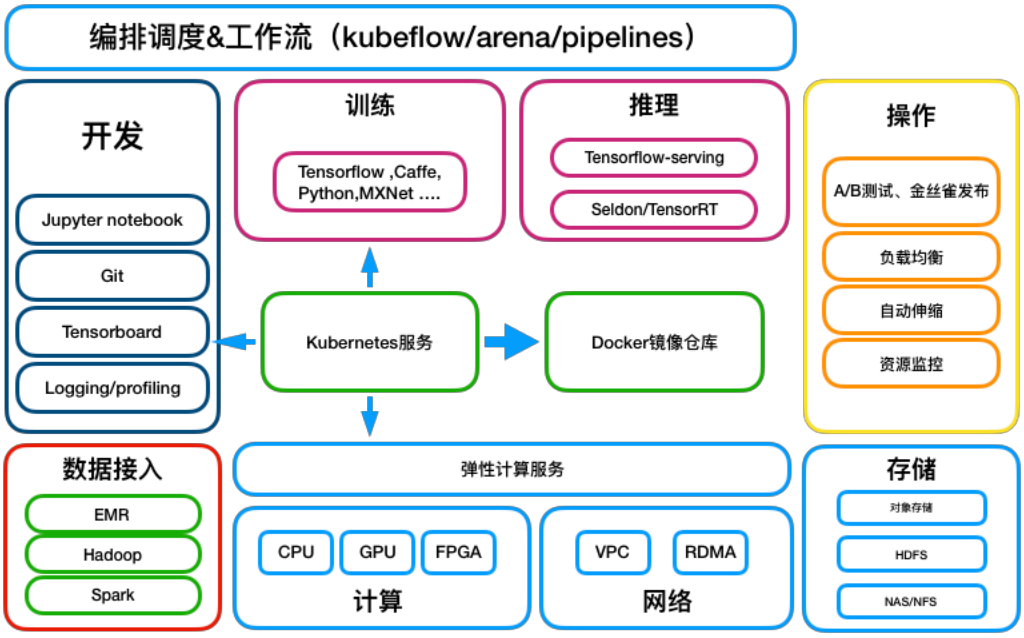

图 4：云原生深度学习方案的基础架构模型

容器化封装深度学习框架，繁琐的配置文件、依赖关系统一打包，实现框架部署的环境一致，极大提升了应用的可移植性，方便用户快速构建实验环境；

服务目录的形式提供多种板卡驱动，能大幅降低 GPU 资源驱动的问题。不同 GPU 卡的驱动程序各不相同，新分配资源的驱动安装复杂易错，非常影响用户使用的体验，将驱动程序以服务方式提供，实现一键配置，降低使用门槛，增强使用体验；

GPU 多机多卡的高效调度，极大降低了深度学习的成本，提升效率,将深度学习平台的普适性大大拓展。以 Kubernetes 为核心，通过

Device Plugin 和 Scheduler Extender 机制整合 GPU 板卡,实现按显存或按卡的调度方式，提升系统利用率。

(二) 区块链应用场景

在企业级的分布式环境以及云环境部署配置部署区块链并非易事,这涉及到区块链相关工具的配置和调用、区块链网络拓扑的设计、证书和密钥的安全分发、组件和服务的高可用性、业务处理能力的弹性扩展、数据的持久化等方面的考虑和设计，需要开发者对区块链相关技术有深入的了解,需要专业和完善的企业基础架构和资源服务的支撑。此外，区块链的配置和部署过程涉及到大量的配置对象，过程繁琐且互相关联，出错概率很高，需要频繁地进行端到端测试才能确保区块链的正确配置和部署，耗费的时间数以小时甚至数以天计。在这种情况下,开发者无法聚焦于区块链上层应用的开发和业务创新的思考上，极大影响了应用和解决方案的快速迭代、快速上线。

区块链业务创新面临巨大的挑战,主要包括需要对区块链底层技术有较深了解，配置部署技术复杂度高、耗时长；二次开发技术难度大，相关平台技术学习曲线陡峭，延缓迭代速度；区块链所需的基础资源和服务选型繁多，整合难度大，投入和质量难以把控；部署平台和环境安全保障薄弱，缺乏企业级安全管控和风险预防能力；服务质量难以达到生产级别要求，运维流程和手段不成熟、不统一。

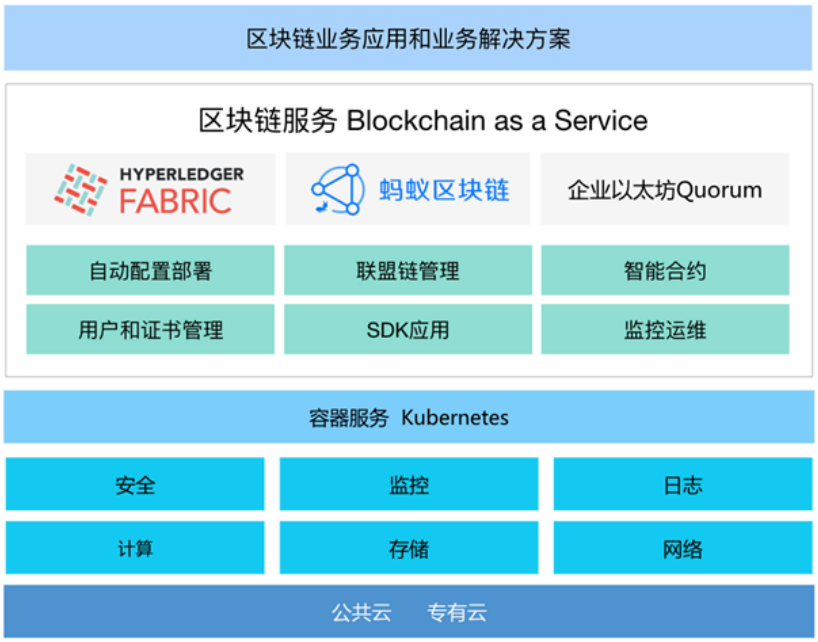

图 5：区块链场景下的云原生架构图

云原生容器区块链解决方案核心思想在于使用容器封装区块链节点，通过容器集群实现区块链网络的编排、创建、运维和资源管理。其具体优势表现在以下几个方面：

容器技术提供标准化的软件打包、分发的能力，保证了运行环境的一致性，与底层环境解耦。这使得企业得以在各种异构、混合云环境实现低成本、高效的区块链系统部署。

依托编排调度工具为区块链实现统一的资源管理和调度。区块链系统底层依赖包括计算、存储、网络、负载均衡、域名解析、加密硬件等各种资源，云原生强大的编排调度功能极大降低了系统设计和运维流程的复杂度。还可以借助编排调度工具来充分利用底层的可信计算能力，如 Intel SGX，保障秘钥签发和验证等关键信息不被篡改、窃取。

Helm Chart 将应用部署化繁为简。区块链网络的拓扑、节点类型、创建流程、服务依赖性均非常复杂，而以 Helm Chart 为代表的应用编排技术可以实现将相关的设计、部署的最佳实践集成到编排模板中，降低企业应用区块链的技术门槛，以标准化的方式实现了区块链网络的自动化编排部署。

云原生技术完美匹配区块链安全机制。区块链网络由于涉及多家参与企业的业务逻辑和数据，以及相关智能合约的安全风险挑战突出，需要严格的安全保障、资源隔离和权限控制等机制。而集群所提供的 namespace 隔离机制、network policy 网络策略、config map 和 secret 等机制均可无缝地与区块链安全治理机制进行整合，提供了坚实的底层安全保障。

此外，云原生技术还有助于实现区块链系统以及区块链应用的持续交付能力，帮助企业更快地实现业务上链。

(三) 边缘计算场景

随着互联网智能终端设备数量的急剧增加，以及 5G 和物联网时代的到来，传统云计算中心集中存储、计算的模式已经无法满足终端设备对于时效、容量、算力的需求。将云计算的能力下沉到边缘侧、设备侧，并通过中心进行统一交付、运维、管控，将是云计算的重要发展趋势。

边缘计算按功能角色主要分为三个部分:

云--传统云计算的中心节点，有丰富的云计算产品形态和资源，是边缘计算的管控端，负责全网算力和数据的统一管理、调度、存储。

边--又称基础设施边缘(Infrastructure Edge)，属于云计算的边缘节点，靠近设备和数据源，拥有充足的算力和存储容量。例如传统云计算的 CDN 节点，物联网场景中的设备控制中心。

端--又称设备边缘(Device Edge)，主要指终端设备，如手机、汽车、智能家电、工厂设备、传感器等，是边缘计算的『最后一公里』。

边缘计算目前面临的主要挑战有:

- 云边端协同: 统一的交付、运维、管控标准。
- 安全: 边缘服务和边缘数据的安全风险控制难度较高。
- 网络: 边缘网络的可靠性和带宽限制。
- 异构资源: 对不同硬件架构、硬件规格、通信协议的支持。

云原生技术的核心价值之一是通过统一的标准实现在任何基础设施上提供和云上一致的功能和体验。借助云原生技术，可以实现云-边-端一体化的应用分发，解决在海量边、端设备上统一完成大规模应用交付、运维、管控的诉求。**安全方面**，云原生技术可以提供容器等更加安全的工作负载运行环境，以及流量控制、网络策略等能力，能够有效提升边缘服务和边缘数据的安全性；**边缘网络环境方面**，基于云原生技术的边缘容器能力，能保证弱网、断网的自治性，提供有效的自恢复能力，同时对复杂的网络接入环境有良好的兼容性；**异构资源兼容性方面**，云原生技术的适用性逐步提升，在物联网领域，云

原生技术已经能够很好的支持多种 CPU 架构(x86-64/arm/arm64)和通信协议，并实现较低的资源占用。

(四) 传统行业互联网化应用场景

传统行业正在经历由数字业务战略推动的转型实践，在提效降本的目标下，尝试创新与发现新的营收来源。数字化转型的程度和成效，在很大程度上影响着整个行业的竞争能力。“互联网+”的大背景下，传统企业互联网特征的业务正在快速崛起，持续交付快速迭代成为这些应用的强烈需求。传统行业转型的瓶颈看似千差万别，但归纳起来有如下共同特点:

项目周期短，需求快速变化。在当前互联网快速发展的驱使下，外部环境的变化日益加快，伴随而来的是 IT 对业务需求的快速响应要求，业务的快速迭代、敏捷交付等需求已经变成企业常态。

互联网高并发，不可预测的承载需求。随着网联化的持续推进，互联网形态的业务日渐丰富。相比较过去传统业务，网联业务具有更强的互联网业务形态，在诸如抢购、秒杀、网促等场景下，要求 IT 架构能更好的支撑高并发、高弹性的业务需求。

兼顾数据安全和用户体验。私有化部署在很大程度上保证了业务数据的安全，但企业自建数据中心的承载规模有限，无法应对特定场景下的访问量激增问题。为了兼顾数据安全要求与用户流畅体验保障，公有云、私有云混合部署规划需要平衡考虑。

面对复杂的、快速变化的的互联网市场竞争，云原生技术可以帮助企业构造一个可扩展的、敏捷的、高弹性的、高稳定性的业务系统。基于容器核心的云原生容器平台，为应用提供标准化敏捷基础架构，充分满足业务的弹性需求，同时利用云原生平台实现跨平台资源横向打通，提供一致的交付体验。建立全流程 DevOps 精益协作，形成以交付为核心，适配多种研发模式的一体化流程协作体系，提升交付效能。

四、 云原生产业发展趋势

(一) 云原生助力人工智能从云端扩展至边缘

微服务架构的盛行正在加速 AI 生态系统中功能解耦。AI 模型中不同粒度的功能如分类、聚类、识别、预测、自然语言处理、卷积、重现等都再向微服务化转变。借助云原生技术，云端智能正在加速向边缘智能扩展。要实现边缘智能化，需要把云上的模型，快速迁移到线下，将云上智能改造为边缘可用的轻量级智能，适配边缘软硬件环境和使用场景。

容器作为一种轻量级操作系统隔离技术，发挥着重要作用，这种轻量级的打包技术将微服务之间依赖关系与预配置文件打包，统一环境和预配置并输出标准化的镜像，很好的解决环境一致性问题，实现快速的扩展和移植，将云端的应用快速在边缘部署。同时，数以亿计

边缘设备汇聚成大量的小型数据中心，利用容器编排引擎，可以轻松完成超多规模集群的分发调度，实现云端的互联互通。

(二) 运营商搭载云原生强化 5G 网络能力

5G 是 ITU、3GPP 等国际标准组织定义的新一代移动通信标准，被广泛认为是将改变社会的通信技术。2017 年启动的 5G R15 标准定义了服务化的系统架构,采用云原生技术中的微服务模式定义了近20个网络能力（NF）接口，强调自包含、独立管理等微服务原则，对传统通信网中紧耦合和功能过于集中的网元进行重构和分解。在 NF 接口的技术实现上，R15 标准可以使用 NFV/SDN 标准化 IT 技术，底层是云计算中通用的 REST 风格接口、x86 硬件平台和虚拟化技术。通过上述标准，5G 网络解决了原有通信网软硬件平台封闭、弹性不足，业务部署和变更困难等固有问题，也为云原生技术应用到 5G 网络及其承载的上层应用打好基础。

云计算技术已应用于现有的通信网络，包括基于 OpenStack 平台实现网元虚拟化和快速部署，基于开放的编排标准实现资源编排。这些能力对于 5G 网络愿景来说还远远未够，包括分钟级的网络能力部署和变更要求、和云计算主流的容器管理平台 Kubernetes 的互操作等。

在已经发布的 ETSI、NGMN 等机构的 5G 白皮书上，通信运营商、设备商都提出了引入云原生技术到 5G 网络中以实现其愿景，相关的

建议包括：对 NF 服务更轻量级的容器化封装，使用 Kubernetes 等云计算动态平台实现弹性部署和开发运营一体化。参与云原生基金会旗下各个开源项目，确保与 5G 业务需求的匹配并增加所需的功能；改造边缘机房为数据中心实现低时延和本地管控的行业应用。

从目前已经试点商用的 5G 案例和业务初期的需求来看，云原生技术非常适合应用于 5G 多接入边缘计算场景，充分利用 5G 用户面/控制面分离、云化改造的边缘机房和 IT 基础设施等能力，通过云原生技术与不同合作伙伴云化应用更为无缝地集成，实现低时延、分钟级弹性部署、支持异构多平台等特性的 5G 应用。

从 5G 核心网乃至通信网的中长期演进来看，云原生技术可以应用到网络能力的研发运营一体化流程中，与持续发展的白盒化通信软硬件平台、人工智能技术相辅相成，促成通信网真正实现开放、智能、敏捷、软硬件解耦、云网融合的转型。

(三) 云原生领域的标准化进程将持续深化

我国云原生技术的标准建设仍处于起步阶段，标准较为分散、不成体系。需要国内的标准化组织统筹规划，做好云原生领域的标准规划和顶层设计，充分发挥标准的引领指导作用。

中国信息通信研究院一直紧跟云计算领域风向，密切关注云原生技术的发展变化，相继制定了《可信云开源解决方案评估方法第二部分：容器类》、《分布式应用架构技术能力要求第一部分：微服务平

台》、《研发运营一体化（DevOps）能力成熟度模型》系列标准、《函数即服务评估方法》等云原生技术领域的一系列标准规范，对相关产品的服务能力进行要求，促进行业了健康有序发展，引导服务贴近产业实际需求。

云原生技术仍在不断成熟发展，越来越多的开源技术被引入生产，但是开源技术的大规模商用需要充分的评估与实践，而基于开源技术的商业发行版差异性较大，服务水平参差不齐。以用户需求为中心的技术规范和能力要求将会持续深化。

(四) 联盟组织在云原生产业中的角色日趋重要

虽然云原生技术已在行业中广泛应用，但行业间的应用程度存在较大差异。产业链的分布较为分散，技术提供方与业务需求方难以实现需求的精准对接，需要中立的第三方机构来承担桥接作用，将云原生领域的分散力量凝聚起来，进一步推动云原生应用的均衡发展。

由中国信息通信研究院牵头，联合云原生技术提供商与众多行业的用户代表共同成立的云原生产业联盟，一直致力于促进我国云原生行业间的交流，加强技术服务商与行业用户间的沟通，推进云原生技术在我国的快速发展与落地实践。联盟作为云原生行业的对话平台，将行业的资源汇聚一起，共同构建技术带动实践、实践反哺技术的良性生态，不断提升着云原生技术在市场的影响力。

附录：云原生落地案例

(一) 招商银行数字化转型实例

云计算、大数据、区块链、移动互联、人工智能等一系列的新一代信息技术的发展和应用，开启了金融科技的黄金时代。金融科技以技术和数据为驱动，用创新的方法改变金融业务信息的不对称，降低交易成本，改善服务体验，金融科技也成为诸多传统金融企业保持继续增长的战略目标。

客户期望不断提高，云计算、移动互联等技术的发展，使得每个金融客户对每一笔服务的交易都能够快速、直观、便捷地掌握。同时希望金融服务商在每个数字化渠道提供的服务体验安全且个性化。金融机构面临巨大压力，希望赶上数字化浪潮，并最终占领更多市场份额。然而技术的发展和运用也可能对金融行业的稳定性造成冲击。金融业历来是强监管、强安全、高复杂度的行业，伴随金融监管体制改革，监管部门对金融 IT 系统的建设和运维有非常严格的要求。金融行业云原生实践的一些显著特征如下:

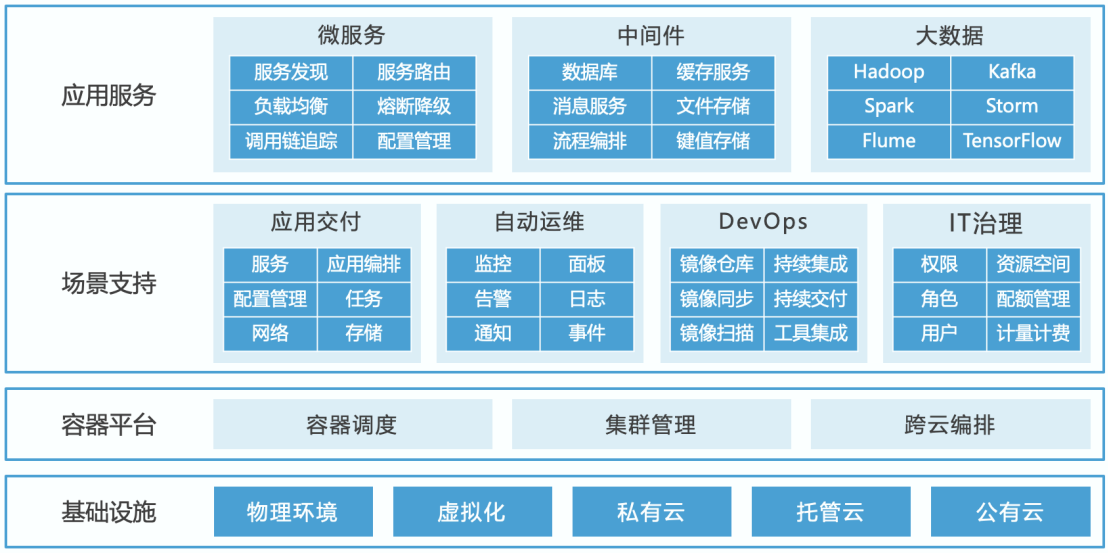

图 6: 金融行业云原生平台典型架构图

银行业“一部两中心”的 IT 建设特色决定了平台的构建顺序。负责架构的信息科技部、负责研发的软件开发中心及负责运维的数据中心各自承担不同角色，项目的分期建设顺序和牵头部门有关。

作为 IT 架构新成员的云原生平台涉及多级系统的对接。云原生平台作为全新的技术实践，需要融入到现有的软件工程及数据运维的管理体系中，可能会涉及全行级 CMP 云管平台及企业 ITSM 系统的对接。

强监管是银行业的重要特点。这就要求整体解决方案满足自主可控的要求，部分项目会要求厂商定向开源代码。生产环境和测试环境有隔离要求，DevOps 的工作范围需有清晰边界，只能延伸至准生产环境。在安全性方面，监管方要求应用间满足点到点的访问控制，这对容器网络方案有较高要求。同时 Https 的服务需要支持国际标准和国家标准的加密算法，后端存储的配置不能包含明文信息等等。

高可用金融行业应用辐射多地理区域/多业务区域，对系统稳定性和容灾能力要求严格，大型建设项目都涉及“两地三中心”、“异地多活”等方案要求，稳定可靠高于业务创新。

招商银行一直对 IT 高度重视，把 IT 作为银行核心竞争力，是新技术的先行者。招商银行在云计算、大数据、机器学习、网络安全、区块链等技术领域都走在行业前列，推动银行业向互联网化、数据化、智能化转型。招商银行还不断将 IT 成功经验和金融 IT 成熟的解决方案开放给金融同业，输出自身金融科技，推动普惠金融。

招商银行 IT 战略要求就是要建立 PaaS 统一管理平台，基于容器技术的新一代企业级 PaaS 平台，整体提升银行应用开发水平和应用交付的速度，为需求快速响应、快速交付以及业务系统的快速迭代提供保障。建成的 PaaS 统一管理平台，有助于招商银行统一对原有 PaaS 平台、容器服务和其他类型的中间件服务进行管理维护，给开发人员提供便利性和可操作性。

平台以容器新一代应用交付件为中心，全方位支持应用创建、编译、集成、部署和运行的每一个环节，并提供一个高效、高可用的运行环境。产品支持一键部署任意容器化应用，并提供自动修复、自动扩展。

图 7: 招商银行云原生平台架构简图

平台优势及创新

- 基于容器化平台技术，提供高资源利用率、弹性伸缩、自动化秒级扩展的平台基础架构。
- 建构企业级 PaaS 云计算平台，提供分布式高可靠、覆盖从开发到测试和部署运行全链条的 DevOps 环境；提供日志、监控、告警分析的智能化运维手段。
- 深度融合 SpringCloud 微服务框架，支持灵活、敏捷的业务快速迭代模式。
- 创新性的支持多集群管理功能，支持异地多中心部署方式，可在异地间实现服务负载的分流、故障切换和统一管理。
- 深度集成 DevOps 主流工具 Jenkins，通过多种手段灵活使用 Jenkins 引擎，包括利用可视化编排工具设计构建、流水线；并且可通过 DSL 插件，在 Jenkins 中调用 PaaS 平台各项功能，完成服务更新、负载均衡设置、自动通知等功能。
- 通过 Service Catalog 技术支持服务市场功能，极大提升企业环境内跨平台间的服务发现、服务管理和服务消费的能力。

(二) 云原生技术在网商银行核心系统的应用实例

网商银行作为首批互联网银行，其金融 IT 系统建设面临较多问题：无线下网点强依赖互联网入口服务的经营模式；金融业务的快速开发与迭代，灵活的金融产品与服务；海量客户与交易，高并发和高性能；数据和服务的可伸缩性、低成本资源调度。

通过微服务技术，包括高性能分布式服务框架、微服务治理中心、Service Mesh 解决方案等，网商银行实现了高可伸缩性和高容错性，满足大规模部署下的性能要求。

通过云原生应用 PaaS 平台和容器技术，将大规模金融级运维能力与渐进式的云原生架构转型方案相结合，网商在 2018 年双十一大促之前完成了底层改造，基于云原生技术实现大规模容器编排的异地多活架构，支持云原生应用的部署和运行，提供镜像管理和集群管理能力，支持多租户，提升研发效率和自动化水平，降低成本和业务技术风险。

通过 DevOps 云原生技术，通过可视化的方式，对云上云下业务、微服务、作业及函数等进行灵活流程编排，有效解决金融业务运维服务的自动化、定制化及灵活的流程调度编排需求，以持续交付实践不断提高研发效率。

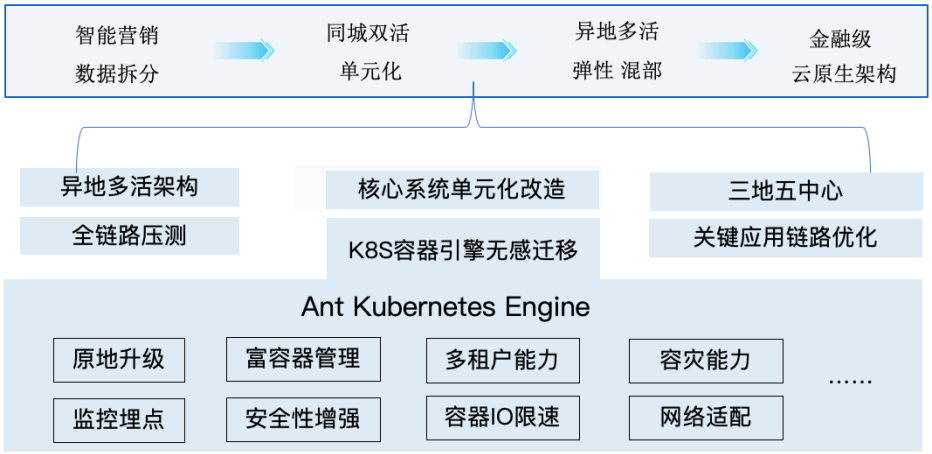

图 8: 云原生技术在网商银行核心系统应用的架构图

基于云原生技术，网商银行成为中国为数不多的将核心系统架构在金融云上的银行。在 2018 年双十一大促期间，网商银行底层架构全面升级，具备上千节点、上万容器组的编排调度，支撑了整体业务容量和峰值 TPS 的大幅增长。

(三) 云原生技术助力广汽丰田数字化转型实践

汽车行业正迅速步入数字化时代。车企服务的对象发生变化，从购车市场转为覆盖后车市场的全周期，通过互联网渠道直面客户，服务客户急速增多。为适配客户快速变化的需求，互联网营销成为常态。业务开展承担新的使命，对业务交付的数量、周期、复杂度都提出新挑战。

目前汽车制造处于“+互联网”和“互联网+”的进程中，面临着互联网业态模式和架构挑战。对业务价值链进行识别，既要满足对产品配置、价格管理、合同管理等稳态业务的支撑，又需要实现商机管理、营销策略、营销管理、电子商务等敏捷业务的快速迭代。两种业

态将长期共存。汽车制造正逐渐从传统汽车生产商和销售商，转变为移动服务、自动驾驶和娱乐、车内体验的供应服务商。新的商业模式，要求充分利用创新的技术，融入到业务的每一个角落。软件定义汽车的时代即将来临。

广汽丰田（GTMC）自 2016 年开始引入移动互联网、电商等数字化营销系统，逐步布局汽车后服务市场，为更好更快迎合客户需求变化，掌握市场转换的主动权，对丰云行为代表的互联网应用进行全面的推广，通过触点连接客户并提供便捷用车和增值服务。同时，积极开拓在线支付、车联网、二手车交易等新型汽车服务业务场景，积累了丰富的实践经验。

广汽丰田积极拥抱变革，充分利用容器、微服务、DevOps 云原生转型方法和手段，驱动技术与汽车场景业务深度融合，建立业务与技术之间良性循环。

战略性构建 GTMC 容器云平台。通过平台实现对丰云行 APP、二手车、在线支付、优惠券等核心互联网应用承载。以多租户的形式提供弹性计算、数据持久化、应用发布等面向敏捷业务服务，并实现高水平资源隔离。标准化交付部署，快速实现业务扩展，满足弹性要求。利用平台健康检查、智能日志分析和监控告警等手段及时洞察风险，保障云平台和业务应用稳定运行。

数字混合云交付。采用私有云+公有云的混合交付模式，按照服务的敏态/稳态特性和管控要求划分部署，灵活调度公有云资源来满

足临时突发或短期高 TPS 业务支撑的需求。利用 PaaS 平台标准化的环境和架构能力，实现私有云和公有云一致交付体验。

图 9：广汽丰田云平台架构示意图

深度融合微服务治理体系，实现架构的革新和能力的沉淀，逐步形成 GTMC 支撑数字化应用的业务中台。通过领域设计、系统设计等关键步骤，对原来庞大的丰云体系应用进行微服务拆分，形成能量、社群、用户、车辆、订单等多共享业务服务，同步制定了设计与开发规范、实施路径和配套设施，形成一整套基于微服务的分布式应用架构规划、设计方法论。

DevOps 理念贯穿始终。通过 DevOps 平台规避软件黑盒，从软件生命周期的源头开始把控，实现对核心代码资产的自主、透明管理，避免对开发商的过度依赖。利用 DevOps 平台的可视化界面，实现全流程自动化、透明化、标准化，实现业务功能的迭代变更的核心掌控。

➢ 实践效果

广汽丰田采用云原生技术在多云环境部署混合云平台，推进丰云行体系应用的架构革新。满足多样化的业务上云需求，满足业务高可

用、高性能、高扩展性、高伸缩性和高安全性要求，为互联网场景业务开展提供有效支撑。提升不同场景下的互联网业务资源使用效率，同时建立以容器为核心的应用交付和运维管理标准，并制定微服务架构应用管理规范。

加强技术管控，提升交付速度。建立适配广汽丰田的 DevOps 实践规范，配合敏捷化开发模式，通过 DevOps 平台实现快速、持续、可靠、规模化地交付业务应用，应用交付周期从 2 个月缩短到 1 个月。

通过敏捷基础架构能力，加快业务创新步伐。持续推进车联网、共享出行等新型场景布局，为广汽丰田数字化转型发展提供有力支撑。

(四) 中石油梦想云平台在能源行业的落地应用

2016 年，中石油提出了“共享中国石油”战略，在这一背景下，对于利用信息化手段实现上中下游产业的一体化协同发展有了更大的需求。“勘探开发梦想云平台”是在中国石油上游业务信息化建设蓝图指导下，以“两统一、一通用”为核心，以集成与共享为目标，建立上游业务统一数据库、统一技术平台，改变以往信息系统建设模式，为上游业务应用开发提供开放、稳定、高效、安全的统一标准规范和技术框架。勘探开发梦想云平台是中石油主营业务第一个共享智能平台，也是国内油气能源行业主营业务第一个共享数字智能平台。

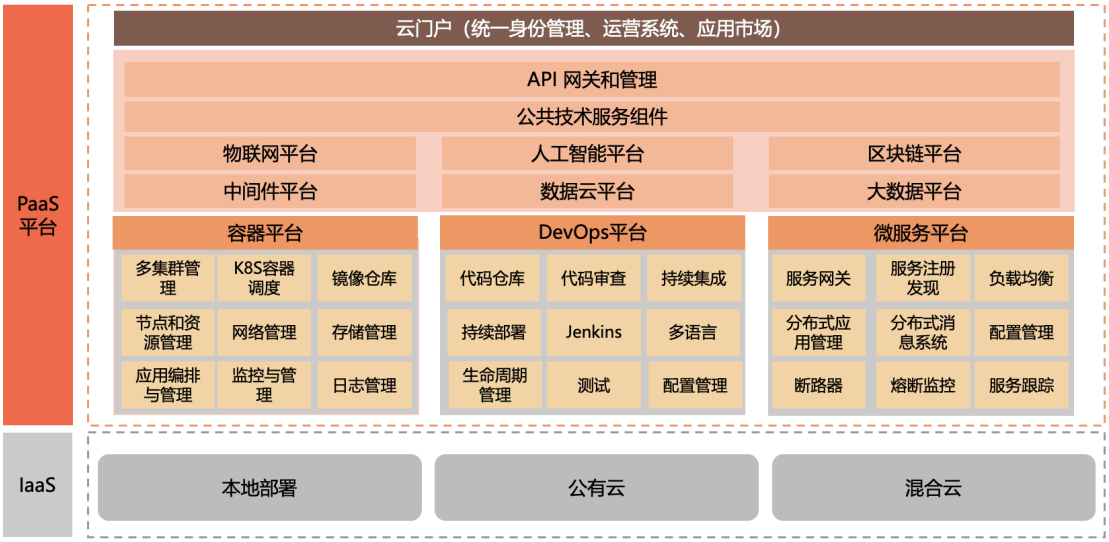

图 10：中石油梦想云平台架构模型

梦想云平台以云原生技术栈为核心，以容器云平台为基础，为应用运行提供丰富的基础技术服务，支撑业务应用环境的快速构建及自适应的弹性调整，实现从资源交付到应用交付的模式转变。平台运维基于 DevOps 理念，提供完整的一体化工具链，支撑应用的全生命周期（需求-设计-开发-测试-发布-运维）统一管理和安全管控，实现开发运维从传统模式到敏捷模式转变；平台构建完全基于服务化的技术框架，提供完善的微服务治理工具及相关开发规范，支撑云原生应用开发和传统应用集成，实现从传统应用架构到云原生架构的过渡；基于开放服务框架，提供勘探开发专业软件集成接口，支撑专业软件提供商自主对接云平台，实现专业软件集成应用。基于应用商店，为合作伙伴提供业务应用及服务组件在线运营环境，支撑上游业务应用共享生态建设。

实践效果：

- 中石油上游业务实现了服务和应用共享及运营。勘探开发相关

的业务规则及软件成果标准化、组件化、引擎化，实现 IT 资产化管理;

- 提升交付效率与质量，应用服务目录提供可依赖的服务组件。为上游业务用户提供统一 APP 式自助服务入口;
- DevOps 的落地，实现了应用交付过程的规范化，基于敏捷理念，通过流水线的高度自动化，实现应用的高效迭代交付，快速响应业务需求;
- 在应用运行管理方面，实现了高效、低成本和可视化运维;
- 通过微服务改造和实施，开发团队可以将工作重心聚焦于业务需求，实现了模块间的松耦合，减少沟通成本，缩短交付周期，完成企业低成本创新。