



SOFT130091.01

云原生软件技术

0. 课程介绍



复旦大学软件学院
沈立炜

shenliwei@fudan.edu.cn

授课教师与助教



沈立炜 副教授

办公室：江湾校区二号交叉学科楼D2021

邮箱：shenliwei@fudan.edu.cn

- 软件工程实验室：<http://www.se.fudan.edu.cn>
- 研究方向：软件工程、人机物融合、云原生、边缘计算
- 助教：张皓捷、牛嘉阳

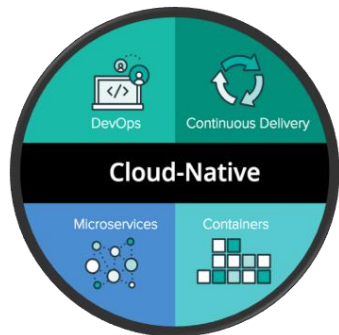
来自于“软件工程”课程

- 当前主流的分布式软件系统有很大一部分都是在云上部署和运行的
- “云原生”（Cloud Native）代表顺应云计算特点的软件应用的典型形态
 - ✓ 在云上部署和运行并不代表一个软件系统就完全符合云计算的特点（高度分布式、可伸缩以及资源按需使用等）
 - ✓ 云计算的原住民，即应用的设计、开发、部署和运行方式都充分考虑了云的特点

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。云原生的代表性技术包括容器、服务网格、微服务、不可变基础设施和声明式API。这些技术能够构建容错性好、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，云原生技术使工程师们能够轻松地对系统作出频繁和可预测的重大变更。

— 云原生计算基金会（CNCF）的定义

云原生软件的特点



IDC预测，到2024年，新增的生产级云原生应用在新应用的占比将从2020年的10%增加到60%；Gartner也预测，到2028年，全球超过95%的企业将在生产中运行容器化应用。

云原生：应用的设计、实现及部署运行方式充分适应云计算平台的特点（如分布式、弹性伸缩）

- ✓ 微服务：物理隔离，轻量级分布式通信，独立部署、独立更新、独立伸缩
- ✓ DevOps：开发运维一体化，持续集成与自动化发布
- ✓ 持续交付：频繁发布、快速交付、快速反馈、快速试错
- ✓ 容器化：服务无差别封装在容器中，运维无需关心服务技术栈

云原生软件的代表性技术-1

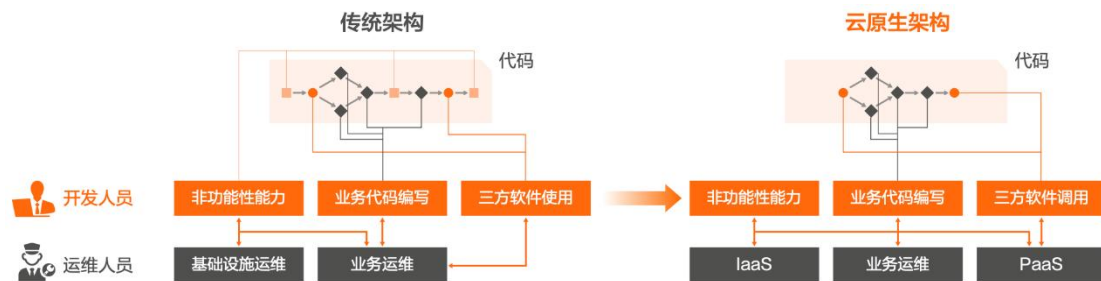
- 微服务（Microservice）：应用软件实现为一组独立开发、独立部署、独立伸缩的服务，服务与服务之间实现了松耦合和物理隔离
- 容器（Container）：应用服务使用轻量级容器进行打包和部署，使得服务的伸缩更加灵活和快速，而基础设施和资源利用率得以提高
- 不可变基础设施（Immutable Infrastructure）：部署服务的服务器在完成部署之后不再发生变化，直至服务实例被销毁后重新创建（以容器化的方式来对服务及运行环境进行整体构建、打包，同时进行镜像版本管理和自动化部署）

云原生软件的代表性技术-2

- 声明式API（Declarative API）：以声明式的方式来定义运行环境（例如“创建10个X服务的实例”），而不需要通过组合多个命令式的API来定义软件运行所需要的运行环境
- 服务网格（Service Mesh）：应用服务之下的一个专门的基础设施层，为每个服务实例提供了轻量级的网络代理，能够在服务开发者无意识的情况下实现服务发现、负载均衡、服务间通信以及相关的可靠性保障（如超时控制、重试熔断等），使得服务开发者可以专注于业务逻辑的实现

云原生软件体系结构

- 出发点：将特定应用业务逻辑实现（如购物流程和积分规则）与通用技术基础设施（如服务间通信及服务的部署、伸缩和可用性保障等）相分离
 - ✓ 非功能性质量特性（如性能、可用性、安全性、韧性等）主要由云计算基础设施提供保障
 - ✓ 应用开发者只需要关注于业务视角的体系结构设计（如服务边界划分和服务间交互逻辑设计）



云原生软件的思想中包含一系列软件体系结构设计原则及设计模式，其中体现了模块化、松耦合、关注点分离等经典软件设计思想，同时充分考虑了云计算的特点

云原生相关的开发过程及管理实践

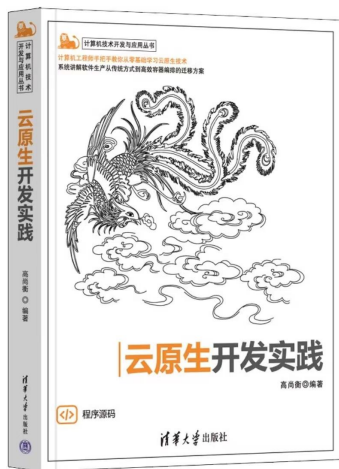
- 开发运维一体化（DevOps）
- 持续交付（Continuous Delivery）
- 康威定律（Conway's law）
 - ✓ 一个系统的设计结构反映了设计这个系统的企业内部的组织结构和沟通关系
 - ✓ 对于采用微服务架构的云原生软件系统而言，每个服务最好都对应一个相对独立的开发团队，服务与服务之间不仅在软件实现上是松耦合的而且对应的开发职责边界清楚，所需要的开发上的沟通关系与组织结构相一致
 - ✓ 因此，打算采用云原生和微服务架构的企业首先要让开发团队的组织结构与所规划的服务划分结构相适应

教学内容

- 云原生软件技术概述、发展与现状
- 云计算概述
- 容器技术与Docker
- 容器编排与Kubernetes
- 微服务架构与应用开发
- 服务网格
- 无服务器技术
- 开发运维一体化与可观测性技术
- 云边协同与KubeEdge
- 云原生开源社区

课程教材

- 自编讲义与ppt
- 参考资料



<https://docs.docker.com> (docker官方文档)

<https://kubernetes.io/docs/> (kubernetes官方文档)

<https://developer.aliyun.com/topic/cn-architecture-paper> (阿里云原生架构白皮书)

课程实践内容

- 针对一个前后端分离系统进行云原生技术实践
 - ✓ 小粒度模块开发与镜像管理
 - ✓ K8S实践
 - ✓ 微服务架构改造
 - ✓ 服务网格实践
- 开展方式
 - ✓ 小组分工与协作
 - ✓ 课堂分享与交流
 - ✓ 助教交流与探讨

实践方案

- 分组完成，5人一组
- 设组长1名，负责协调和沟通
- 项目评分：小组整体评分+个人贡献互评

成绩评定

- 平时表现： 5%
- 课程Lab： 45-50%
- 期末考试： 45-50%

课程微信群

- 课程通知
- 资料分享
- 问题解答
- 技术交流

群聊：25软院云原生课程群



SOFT130091.01

云原生软件技术

End

0. 课程介绍