



厦门大学国家示范性软件学院  
Software School of Xiamen University



# 微服务架构 简介

赖永炫 博士/教授  
厦门大学 软件工程系

<http://mocom.xmu.edu.cn/lai>

[laiyx@xmu.edu.cn](mailto:laiyx@xmu.edu.cn)

# 提纲



- 1. 单块架构系统及其面临的挑战
- 2. 什么是微服务架构
- 3. 微服务架构的核心特征
- 4. 微服务架构的优缺点



# 当前进度

---

- 1. 单块架构系统及其面临的挑战
- 2. 什么是微服务架构
- 3. 微服务架构的核心特征
- 4. 微服务架构的优缺点



# 单块架构系统及其面临的挑战

## ■ 应用的三层架构

层的概念：

- 层能够被单独构造；
- 层具有区别于其他层的显著特点；
- 层与层之间能够互相连接、互相支撑、互相作用，相互协作构成一个整体。
- 层的内部，可以被替换成其他可工作的部分，但对整体的影响不大。

WEB 应用中不同职责的部分逐渐被定义在了不同的层次，每一层负责的部分更趋向于具体化，细致化，三层架构通常包括表示层、业务逻辑层以及数据访问层。

# 单块架构系统及其面临的挑战



## ■ 非三层架构

有些人认为，对于一个WEB应用程序，其被自动地分成了三层架构，因为它有三个分离的部件，如图：





# 单块架构系统及其面临的挑战

## ■ 非三层架构



- 这三个部分看起来虽然满足“层”的概念，但它并不是我们所说的软件架构的层：
  1. 浏览器可以**独立存在于WEB应用程序之外**，WEB应用程序也可以被不同的浏览器访问，因此浏览器不是WEB应用程序的部分。
  2. 数据库服务器也可以**独立存在于应用程序之外**，因此它也不是应用程序的一部分。

# 单块架构系统及其面临的挑战



## ■ 单块架构应用

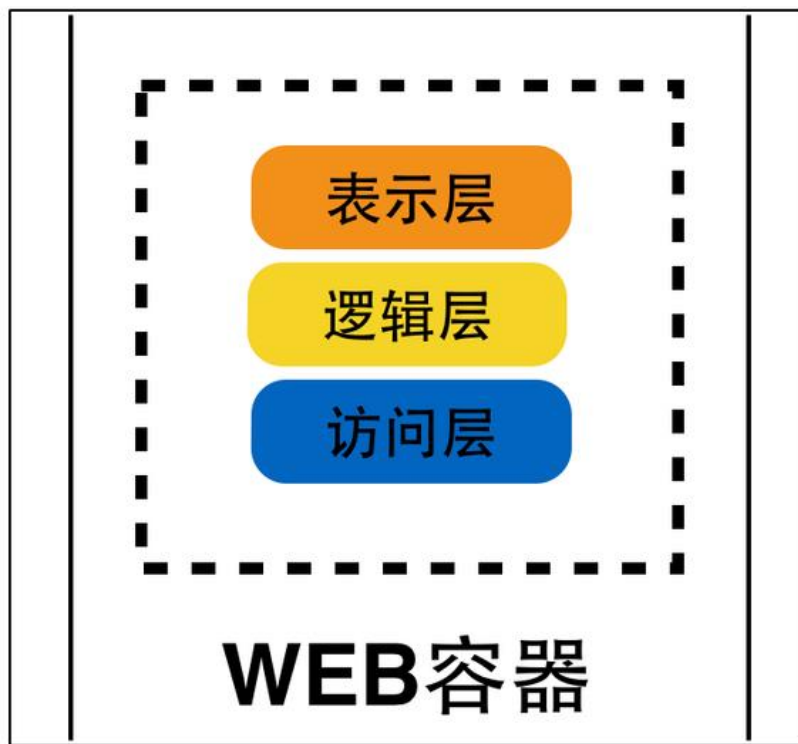
- 虽然软件的三层架构帮助我们将应用在**逻辑上**分成了三层，但它并不是物理上的分层。
- 即便我们将应用架构分成了所谓的三层，经过开发团队对不同层的代码实现，经历过编译、打包、部署后，不考虑负载均衡以及水平扩展的情况，最终还是运行在同一个机器的**同一个进程中**。



# 单块架构系统及其面临的挑战

## ■ 单块架构应用

- 对于这种功能集中、代码和数据中心化、一个发布包、部署后运行在同一进程的应用程序，我们通常称之为单块架构应用。





# 单块架构系统及其面临的挑战



## ■ 单块架构应用的优势

- 易于开发
- 易于测试
- 易于部署
- 易于水平伸缩



# 单块架构系统及其面临的挑战

- 随着互联网行业的迅猛发展，需求不断的增加以及用户量的不断增加，单块架构面临着越来越多的挑战：
  - 维护成本增加
  - 持续交付周期长
  - 新人培养周期长
  - 技术选型成本高
  - 可伸缩性差
  - 构建全功能团队难



# 当前进度

---

- 1. 单块架构系统及其面临的挑战
- 2. 什么是微服务架构
- 3. 微服务架构的核心特征
- 4. 微服务架构的优缺点



# 什么是微服务架构

- 与传统IT行业的“服务架构”概念不一样的，微服务更强调的是一种**独立测试、独立部署、独立运行**的软件架构模式。
- 每个服务运行在其独立的进程中，服务与服务间采用**轻量级的通信机制**互相沟通（通常是基于HTTP协议的RESTful API）。
- 每个微服务可以使用**不同的开发语言**，**不同框架**，以及**不同的数据存储技术**，因此相比与“服务架构”，它是一种高度自治的、非集中式管理的、细粒度的业务单元。



# 什么是微服务架构

■ 微服务本身没有严格定义，可以从以下四个方面考虑：

## 1. 小，且专注于做一件事情

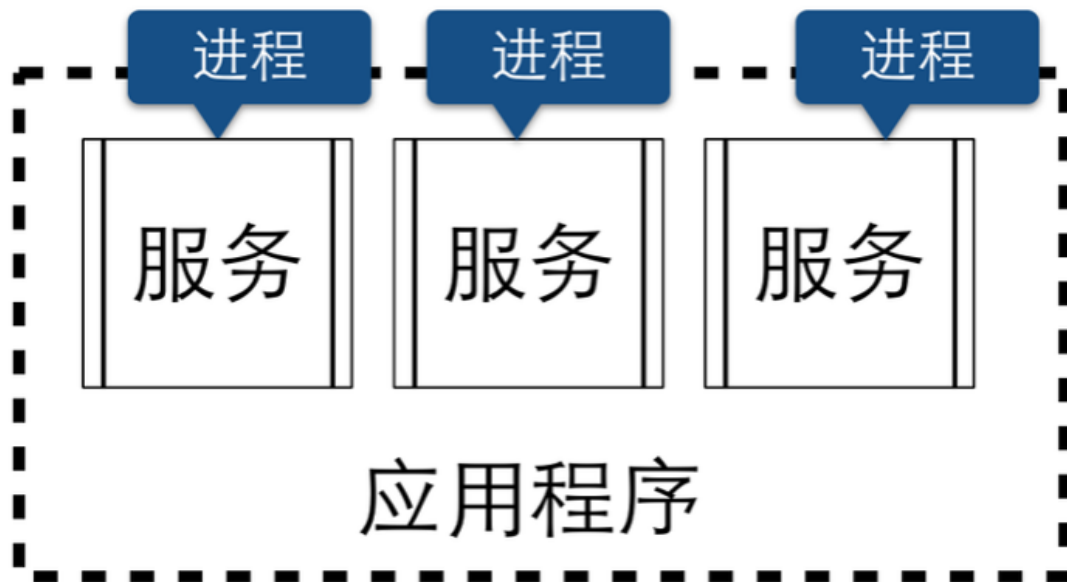
- 微服务的每个服务都是很小的应用。微服务中提到的“微”或“小”并不是由代码行数衡量。
- 微服务的“微”是不可衡量、看不见、摸不着的“微”。考虑服务的“微”时，至少遵循两个原则：**业务独立性和团队自主性**。
- 微服务的服务专注于做一件事情。遵循“**高内聚，低耦合**”的原则。



# 什么是微服务架构

## 2. 运行在独立的进程中

- 在传统的单块架构应用中，通常将应用程序的代码分成逻辑上的三层，但它并不是物理上的分层。最后，应用程序会运行在同一个机器的同一个进程中。
- 但在微服务的架构里，应用程序由多个服务组成，每个服务都是一个具有**高度自治性的独立业务实体**。
- 通常情况下，每个服务都能够运行在一个独立的操作系统进程中，这就意味着不同的服务能非常容易被部署到不同的主机上。





# 什么是微服务架构

## 3. 轻量级的通信机制

- 所谓轻量级通信机制，通常指基于**语言无关**、**平台无关**、**代码无关**的这类协议，例如我们熟悉的XML或者JSON，他们的解析和使用与语言无关、平台无关。
- 另外，基于HTTP协议，能让服务间的通信变得无状态化，目前大家所熟悉的**REST(Representational State Transfer)**就是服务之间互相协作常用的轻量级通信机制之一。
- 对于微服务而言，通过使用**语言无关**、**平台无关**的轻量级通信机制，使服务与服务之间的协作变得更加简单、标准化。



# 什么是微服务架构

## 4. 松耦合，独立部署

- 在传统的单块架构应用里，由于所有的功能都存在同一个代码库里。因此，当修改了该代码库的某个功能，在后续的测试过程中都需要做回归测试。功能和功能之间存在着**强耦合**关系。







# 什么是微服务架构

## 4. 松耦合，独立部署

- 微服务架构中，每个服务都是一个**独立的业务单元**，服务和  
服务之间是**松耦合**的。当对某个服务进行改变时，对其他的  
服务并不会产生影响。





# 当前进度

---

- 1. 单块架构系统及其面临的挑战
- 2. 什么是微服务架构
- 3. 微服务架构的核心特征
- 4. 微服务架构的优缺点



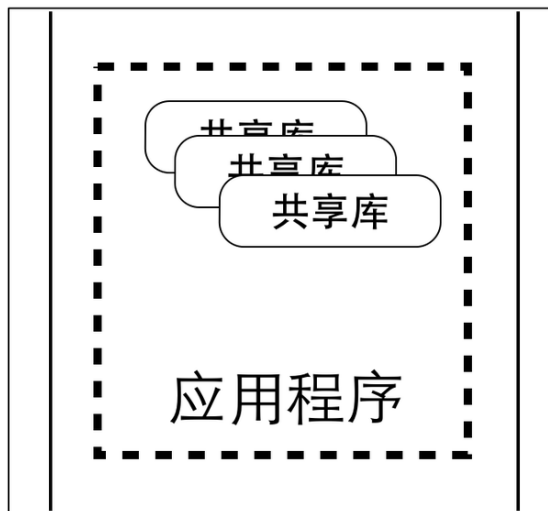
# 微服务架构的核心特征

## ■ 服务与组件

- 在软件领域，我们也将**组件**定义为应用软件构建中**独立的单元**，它的最大特点是，对整个应用软件而言，组件能够被**容易的替代或者更新**。

## □ 共享库作为组件

- 传统实现组件的方式采用和应用程序一样的编程语言，构建独立的**共享库 (Library)**，从而达到解耦和复用的效果。
- 共享库是**语言相关、平台相关**，并且是和应用程序运行在**同一个进程中的**。任何库的变更都将导致整体应用的**重新发布**。





# 微服务架构的核心特征

## ■ 服务与组件

### □ 微服务作为组件

- 微服务也可以作为组件。把微服务当成应用程序中的组件，其中一个最主要的好处是微服务可以**独立部署**。
- 通过服务来实现组件，将应用**拆散为一系列的服务**，运行在**不同的进程中**时，任何一个服务的局部变化只需重新部署对应的服务本身。
- 把服务当成组件的另外一个优点是在组件和组件之间、或者组件和应用程序之间定义了**清晰的、跨语言、跨平台的接口**。



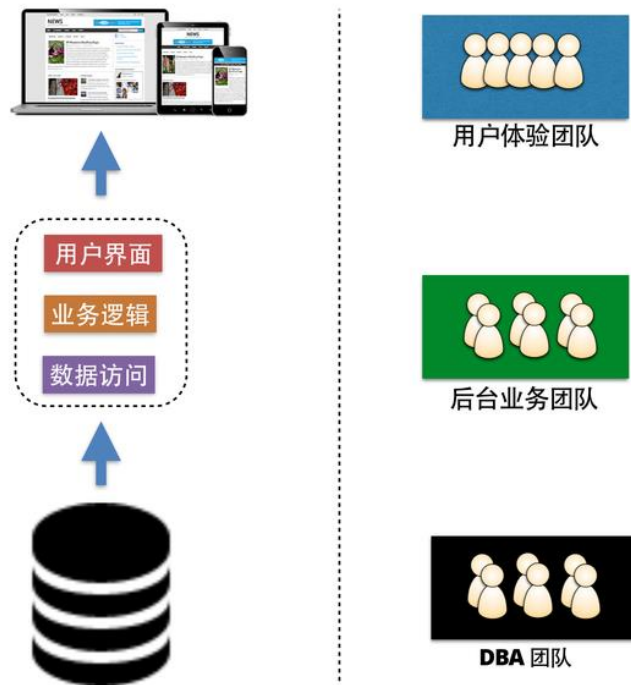


# 微服务架构的核心特征

## ■ 围绕业务组织团队

### □ 传统开发模式

- 在单块应用架构的时代，为了节省成本、快速实现目标，企业或者组织一般都会根据技能类型的差异化来划分团队。
- 实际上，当团队被按照这个策略划分后，即便是某些简单的需求变更，都有可能造成不同团队之间跨组织、跨团队的协作，耗费很高的跨团队的沟通和协作成本。





# 微服务架构的核心特征

## ■ 围绕业务组织团队

### □ 微服务架构开发模式

- 微服务架构的开发模式不同于传统方式，它倡导**围绕应用程序为核心**，按**业务能力**来划分为不同的团队。
- 每个团队都要求能够对每个服务，将其对应的业务领域的**全部功能实现**，譬如对于某业务需求的更改，从用户体验界面到业务逻辑实现，再到数据的存储和迁移等。

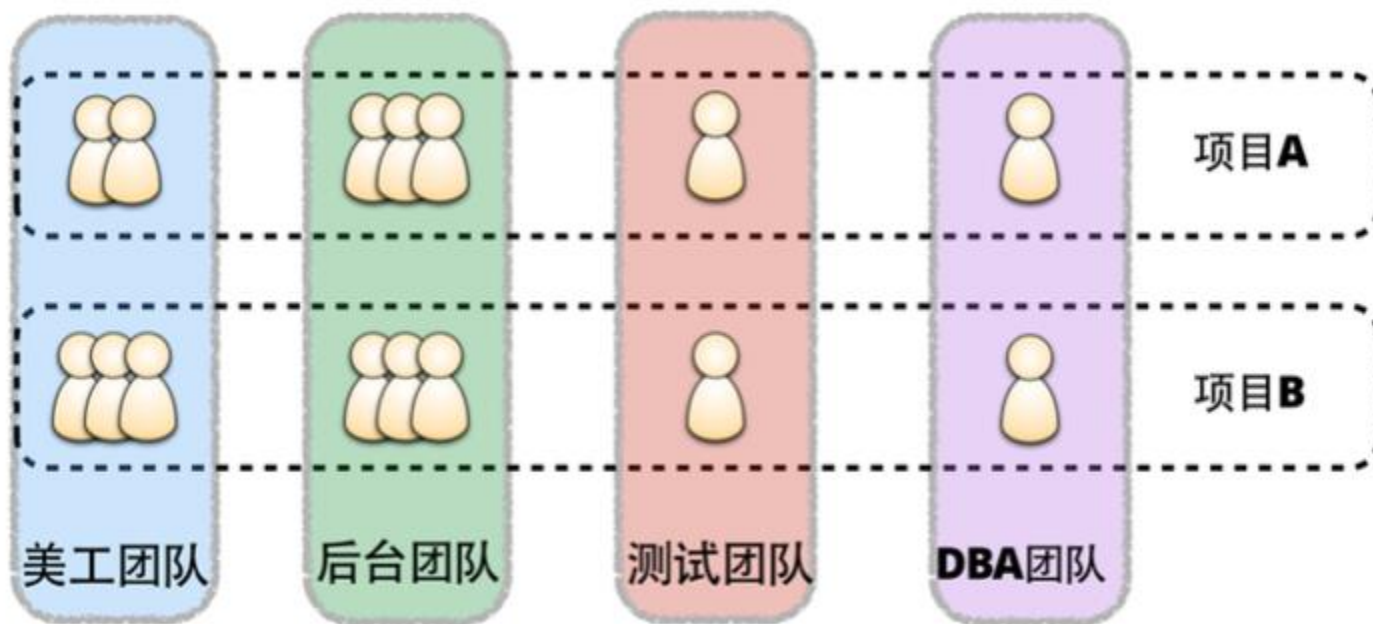


# 微服务架构的核心特征

## ■ 关注产品而不是项目

### □ 传统应用开发

- 传统的应用开发都是基于**项目模式**的。
- 当项目启动后，企业或者组织会从开发团队中**抽出一部分资源**、从测试团队中**抽出一部分资源**，同时也从其他不同的技能团队中**抽出不同的资源**，组成一个项目团队，然后设置一个**时间期限**，让大家完成项目。





# 微服务架构的核心特征

## ■ 关注产品而不是项目

### □ 微服务开发

- 服务架构倡导的是避免采用这种项目模式，更倾向于让开发团队负责整个产品的生命周期。
- 即对于开发产品的团队而言，产品就是团队的，也是每个成员的。团队中的每个人都有责任、有义务确保产品的快速发展以及演进。
- 亚马逊CTO Werner Vogels，曾经说过一句经典的名言，“You build it, you run it”。





# 微服务架构的核心特征

## ■ 去中心化

### □ 传统的单块应用架构

- 传统的单块应用架构，倾向于采用统一的技术平台或方案来解决所有问题。
- 单块架构系统，初始的技术选型严重限制将来采用不同语言或框架的能力。
- 如果想尝试新的编程语言或者框架，没有完备的功能测试集，很难平滑的完成替换，而且系统规模越大，风险越高。



# 微服务架构的核心特征

## ■ 去中心化

### □ 微服务架构

- 微服务的架构可以针对不同的业务特征选择不同的技术方案，有针对性的解决具体的业务问题。
- 基于微服务架构，使我们更容易在系统上尝试新的技术或解决方案。
- 譬如说，可以先挑选风险最小的服务作为尝试，快速得到反馈后再决定是否试用于其他服务。这也意味着，即便对一项新技术的尝试失败，也可以抛弃这个方案，并不会对整个产品带来风险。



# 微服务架构的核心特征

## ■ 独立业务数据

### □ 传统的单块应用架构

- 传统的单块应用架构，倾向于采用统一的数据存储平台来存储所有的数据。
- 传统的数据库大多是关系型数据库，存储的数据都是以结构化信息为主，但随着互联网的快速发展，数据的结构并不具有确定性，或者说结构发生变化的频率非常快，因此，对于如何有效维护业务数据，也成了一个难题，相应的维护成本越来越高。

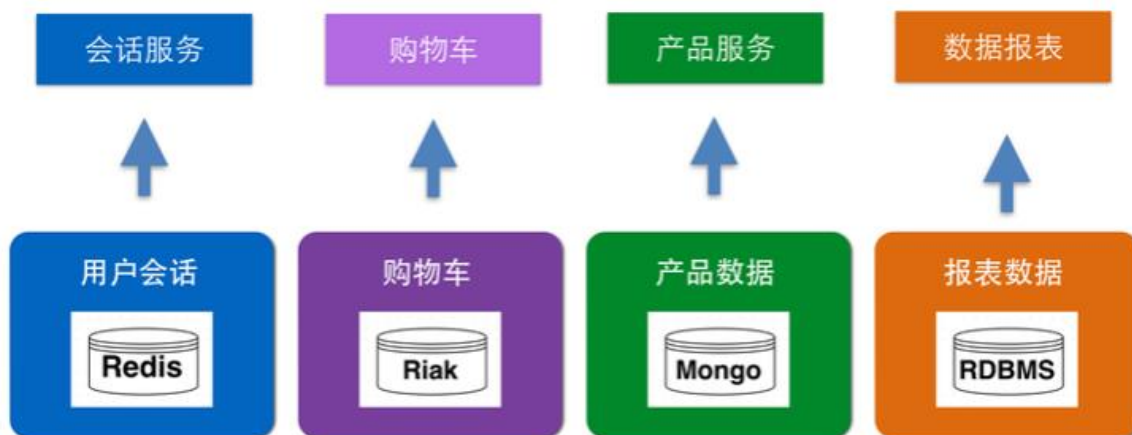


# 微服务架构的核心特征

## ■ 独立业务数据

### □ 微服务架构

- 微服务架构，提倡具有业务属性的**独立单元**或者**服务自身**维护其相关的业务数据。
- **具有业务属性的服务单元**能够有权利**管理其相关的数据**，同时能够随着业务的发展，不断更新业务数据。
- 每个业务单元**只关心自己的业务数据**，因此可以选择最合适的工具或者产品来存储以及管理数据。





# 微服务架构的核心特征

## ■ 基础设施自动化

- 微服务架构将应用程序本身分成多个小的服务，每个服务都是一个独立的部署单元。
- 微服务的实践，对持续交付和部署流水线要求非常高。
- 微服务的粒度越细，就意味着需要部署的业务单元就越多，业务单元越多，就需要更稳定的基础设置自动化机制，能够创建运行环境，安装依赖，部署应用等。





# 微服务架构的核心特征

## ■ 演进式架构设计

- **架构设计**应该是随着业务的发展而不断发展，随着需求的变化而**不断变化**的。
- 在传统的单块设计中，企业或者组织通常是希望构建一个**大而全、无所不能**的平台。
- 在技术发展如此之快的今天，单一的技术平台已经无法适应市场的快速变化，组织应该随着业务的发展，随着企业的发展，**不断尝试并改进架构设计**，真正做到**业务驱动架构**，**架构服务于业务**。



# 当前进度

---

- 1. 单块架构系统及其面临的挑战
- 2. 什么是微服务架构
- 3. 微服务架构的核心特征
- 4. 微服务架构的优缺点



# 微服务架构的优缺点

## ■ 微服务架构的优点

- ✓ 每个服务都很简单，只关注于一个业务功能。
- ✓ 每个微服务可以由不同的团队独立开发。
- ✓ 微服务是松散耦合的。
- ✓ 微服务可以通过不同的编程语言与工具进行开发。





# 微服务架构的优缺点

## ■ 微服务架构的缺点

- ✓ 运维成本过高
- ✓ DevOps是必须的
- ✓ 接口不匹配
- ✓ 代码重复
- ✓ 分布式系统的复杂性



AGREE

谢谢聆听  
**Thank you**