

前言

1.1. 课程内容

1.2. 技术要求

1.3. 听课对象, 开局爬楼

1.4. 网站观看建议

1.5. 持续学习

2. 理论介绍

3. 回顾SpringCloud第一季

前言

1.1. 课程内容

SpringCloud + SpringCloud Alibaba

1.2. 技术要求

Java8 + Maven + Git

Github + Nginx + RabbitMQ + SpringBoot2.0

2018年发布的SpringCloud第一季(必要条件)

1.3. 听课对象, 开局爬楼

- 学生 + 零基础 + 非科班 + ...
- 阶段 零->初->中->高
 - 1-4 零基础
 - 5-9 初级
 - 10-16 中级
 - 17-21 高级
 - 有基础从17章看

1.4. 网站观看建议

- 关闭弹幕
- 调整播放速度
- 不跟风学习

1.5. 持续学习

- 少说话, 多动手
- 不在能知, 乃在能行
- 语言少了, 思想就出来了

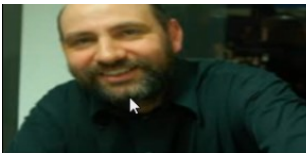
2. 理论介绍

理论介绍见<<微服务架构2020.pptx>>



01.微服务架构概述

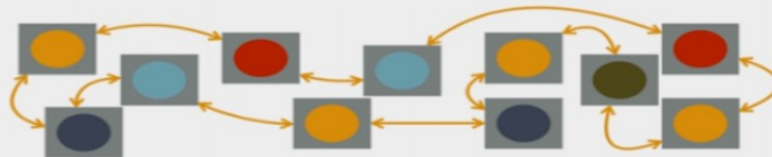
- 什么是微服务
- 主题词01：95后数字化生活-落地维度
- 主题词02：分布式微服务架构-落地维度



Microservices - the new architectural style

Martin Fowler, Mar 2014

The **microservice architectural style** is an approach to developing a **single application** as a suite of **small services**, each running in its **own process** and **communicating** with **lightweight mechanisms** often an HTTP resource API. These services are **built around business capabilities** and **independently deployable** by fully automated deployment machinery. There is a bare minimum of centralized management of these services, which may be written in different programming languages and use different data storage technologies.



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。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协作（通常是基于HTTP协议的RESTful API）。每个服务都围绕着具体业务进行构建，并且能够被独立的部署到生产环境、类生产环境等。另外，应当尽量避免统一的、集中式的服务管理机制，对具体的一个服务而言，应根据业务上下文，选择合适的语言、工具对其进行构建。



主题词:95后互联网时代下的数字化生活

哪些维度？



主题词:基于分布式的微服务架构

满足哪些维度？

支撑起这些维度的具体技术？？？？

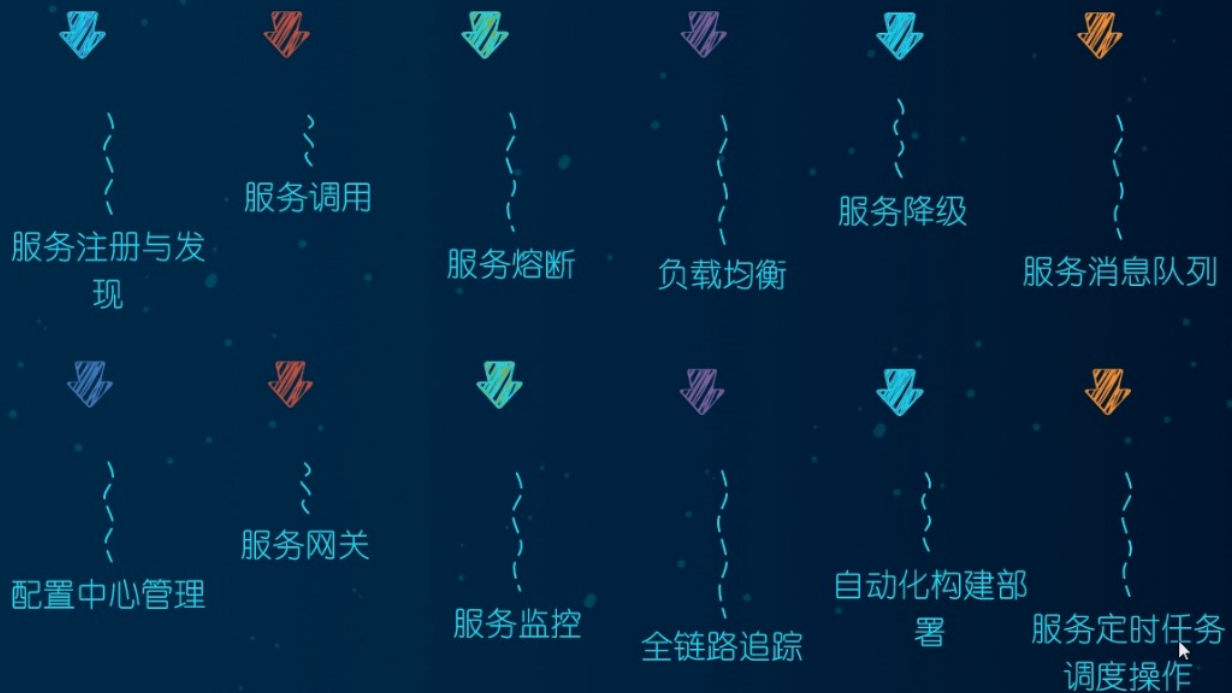
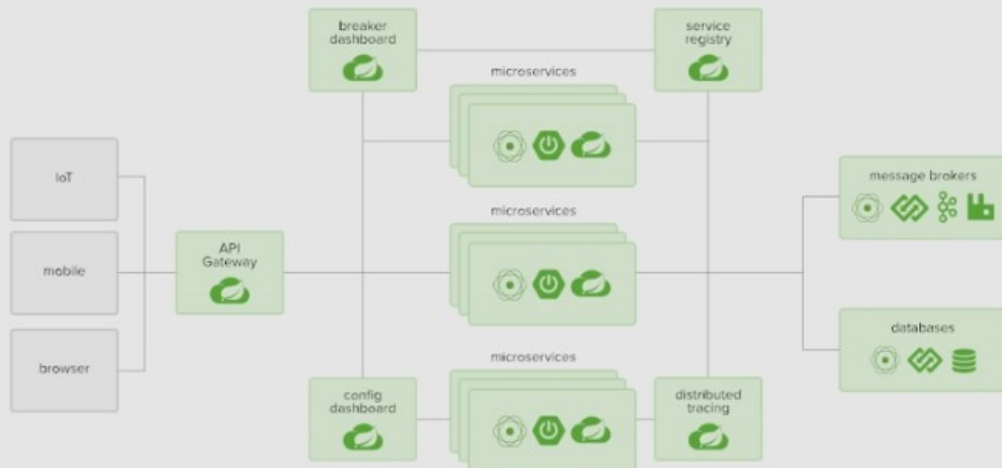


Spring Cloud



COORDINATE ANYTHING: DISTRIBUTED SYSTEMS SIMPLIFIED

Building distributed systems doesn't need to be complex and error-prone. Spring Cloud offers a simple and accessible programming model to the most common distributed system patterns, helping developers build resilient, reliable, and coordinated applications. Spring Cloud is built on top of Spring Boot, making it easy for developers to get started and become productive quickly.



02.SpringCloud简介

■ 是什么













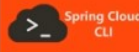



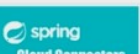




■ 符合微服务技术维度

SpringCloud=分布式微服务架构的一站式解决方案，是多种微服务架构落地技术的集合体，俗称微服务全家桶

■ 猜猜SpringCloud这个大集合里有多少种技术？

Spring Cloud集成相关优质项目推荐

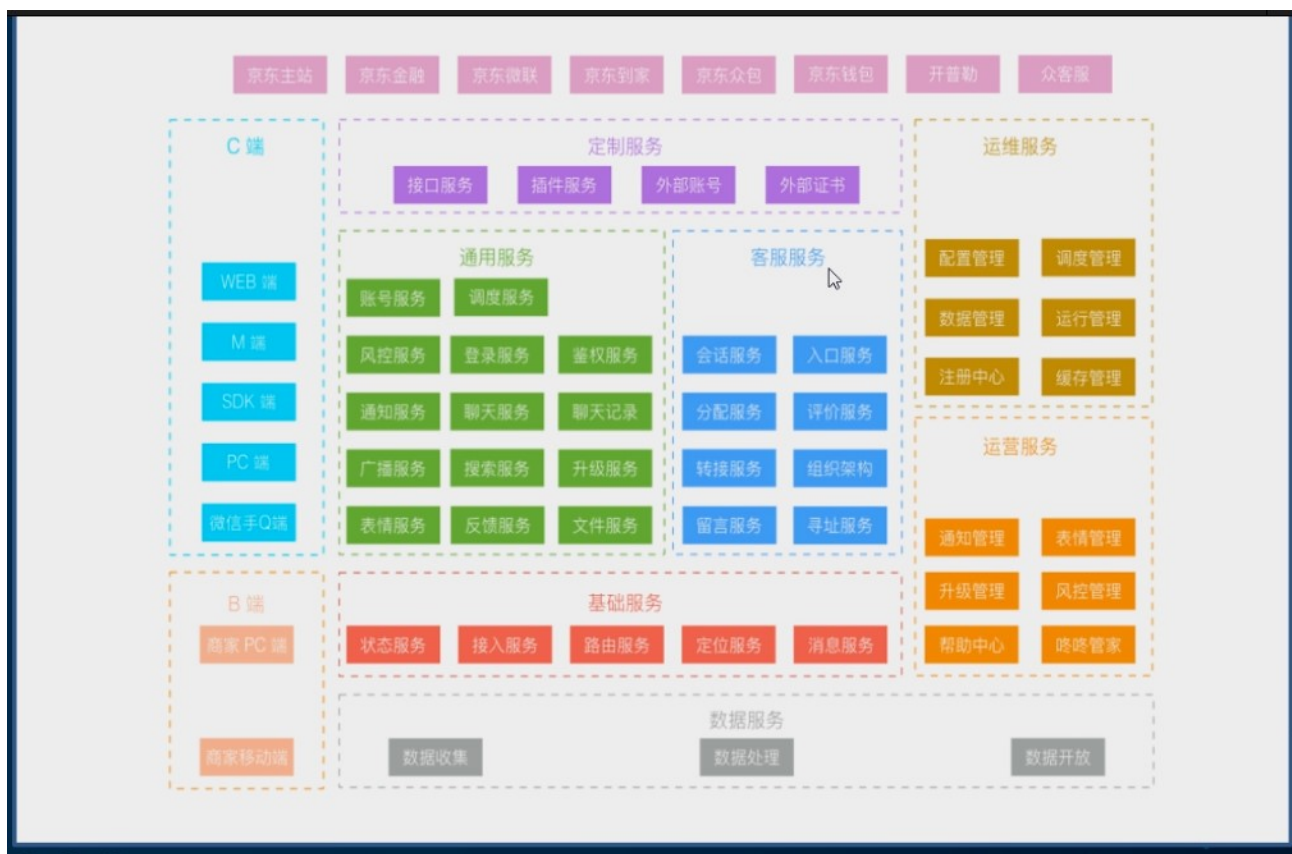
这些项目是Spring Cloud官方项目或是对Spring Cloud进行了有益的补充以及基于Spring Cloud二次开发。

 Spring Cloud Config Spring 配置管理工具包，让您可以把配置放在远程服务器，集中化管理配置，目前支持本地存储、Git及Subversion。	 Spring Cloud Bus Spring 事件、消息总线，用于在集群（例如，配置发布事件）中传播状态变化，可与Spring Cloud Config联合实现热部署。	 Eureka Netflix 云端服务发现，一个基于Netflix的服务，用于定位服务，以实现云端的服务器发现和故障转移。	 Hystrix Netflix 断路器、管理管理工具，旨在通过断路器控制服务和第三方服务的节点，从而对延迟和故障提供强大的容错能力。	 Zuul Netflix Zuul 是在云平台上提供动态路由、负载均衡、安全等功能的网关。Zuul 相当于设备端和Netflix流应用的Web网关后端所有请求的部门。	 Archaius Netflix 配置管理API，包含一系列配置管理API，提供动态配置功能，包括安全配置操作、轮询框架、回调机制等功能。	 Consul HashCorp 封装了Consul操作，consul是一个服务发现与配置工具，与Docker容器可以无缝集成。
 Spring Cloud Sleuth Spring 日志收集工具包，封装了Clapper和log-based跟踪以及Zipkin和Trace操作，为SpringCloud应用实现了一种分布式追踪解决方案。	 Spring Cloud Data Flow Pivotal 大数据事件工具，作为Spring XD的替代产品，它是一个混合计算模型，结合了流数据与批量数据的处理方式。	 Spring Cloud Security Spring 基于spring security的安全工具包，为您的应用程序添加安全控制。	 Spring Cloud Zookeeper Spring 提供Zookeeper的工具包，用于使用zookeeper方式的服务器发现和配置管理。	 Spring Cloud Stream Spring 数据流操作库，封装了与Redis、Rabbit、Kafka等发送接收消息。	 Spring Cloud CLI Spring 基于Spring Boot CLI，可以让你以命令方式快速建立云组件。	 Ribbon Netflix 提供云端负载均衡，有多种负载均衡策略可供选择，可配合服务发现和断路器使用。
 Feign OpenFeign Feign是一种声明式、模板化的HTTP客户端。	 Spring Cloud Task Spring 提供云端计划任务管理、任务调度。	 Spring Cloud Connectors Spring 便于云端应用程序在多种PaaS平台连接后端，如：数据库和消息代理服务。	 Spring Cloud Cluster Spring 提供Leader选举，如：Zookeeper、Redis、Hazelcast、Consul等常见状态模式的抽象和实现。	 Spring Cloud Starters Pivotal Spring Boot式的启动项目，为Spring Cloud提供开箱即用的依赖管理。	 Spring Cloud for Cloud Foundry Pivotal 通过CloudFoundry快速部署到CloudFoundry，CloudFoundry是VMware推出的开源PaaS云平台。	 Turbine Netflix Turbine是聚合服务端发送事件流数据的一个工具，用来收集并汇总Hystrix的metrics情况。

SpringCloud俨然已成为微服务开发的主流技术栈，在国内开发者社区非常火爆。

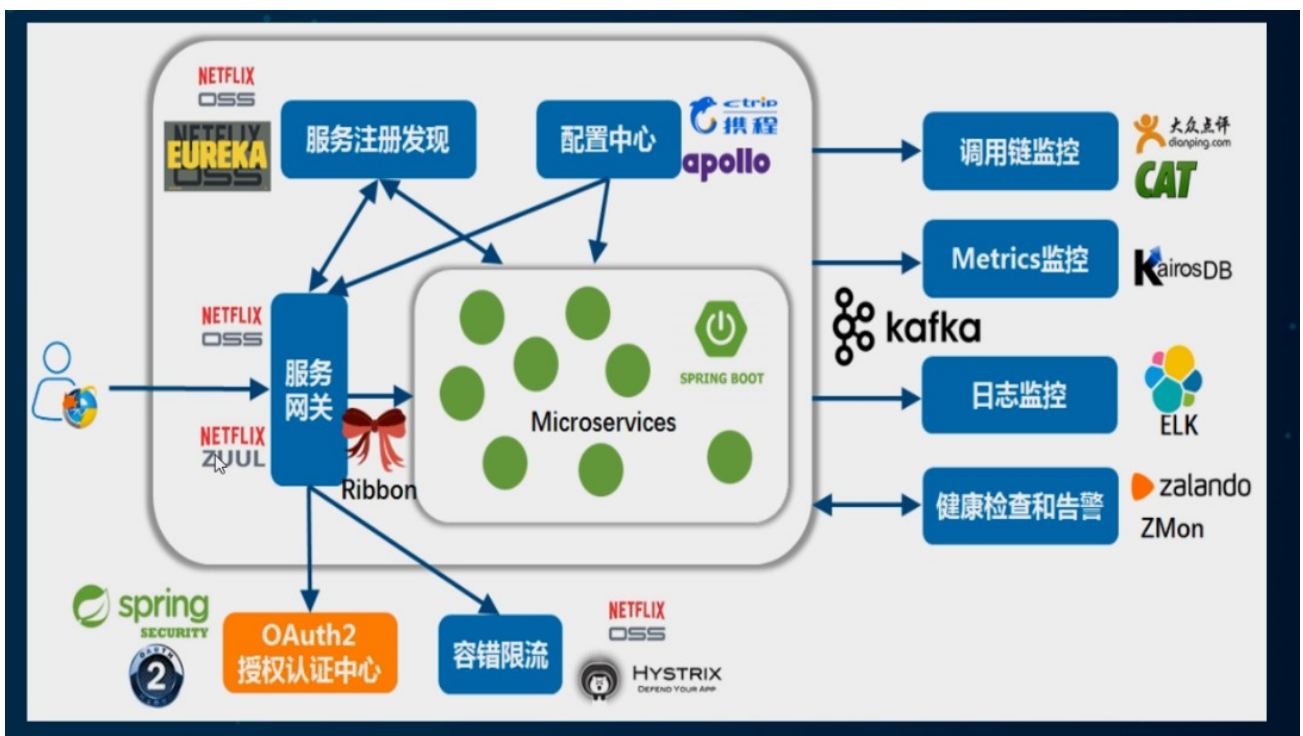
“微”力十足，互联网大厂微服务架构案例

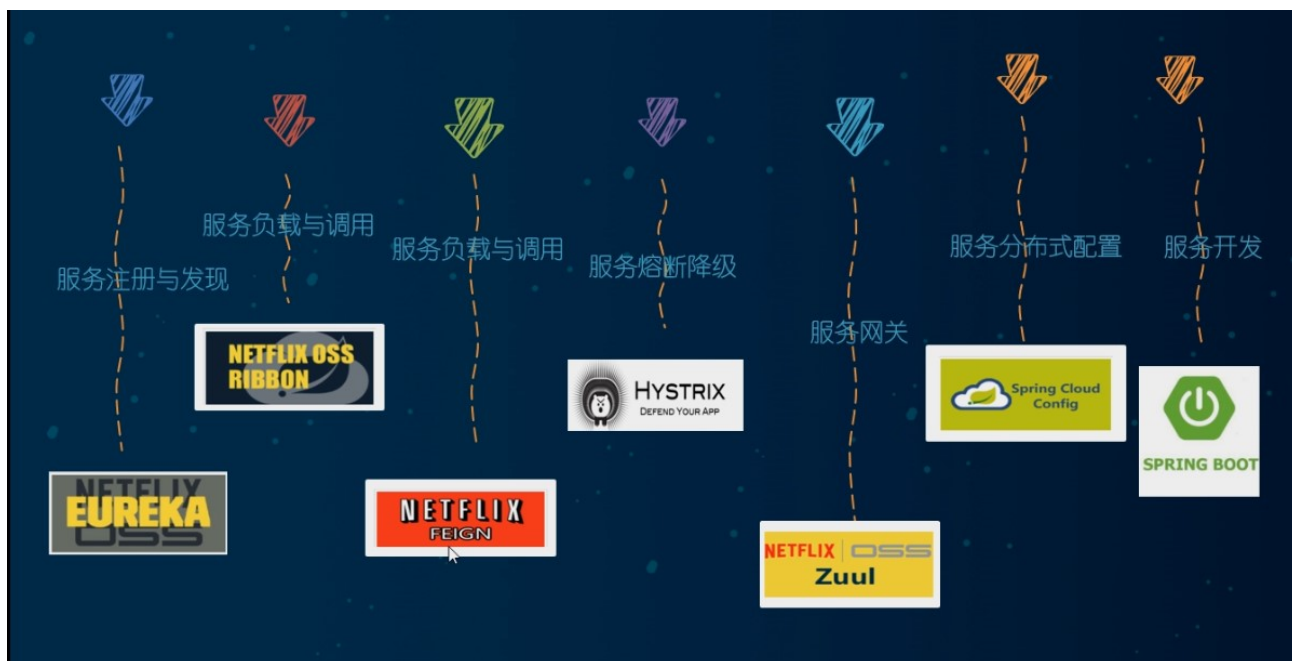




03.SpringCloud技术栈

□ 各个技术栈功能介绍





3. 回顾SpringCloud第一季

- 回顾2018年第一季SpringCloud版本

```

<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-dependencies</artifactId>
  <version>Dalston.SR1</version>
  <type>pom</type>
  <scope>import</scope>
</dependency>

<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-dependencies</artifactId>
  <version>1.5.9.RELEASE</version>
  <type>pom</type>
  <scope>import</scope>
</dependency>

```