

Spring Cloud

- 1 微服务是什么
- 2 Spring Cloud是什么
- 3 Spring Cloud Eureka
- 4 Spring Cloud Ribbon
- 5 Spring Cloud OpenFeign
- 6 Spring Cloud Hystrix
- 7 Spring Cloud Gateway
- 8 Spring Cloud Config
- 9 Spring Cloud Alibaba是什么
- 10 Spring Cloud Alibaba Nacos
- 11 Spring Cloud Alibaba Sentinel
- 12 Spring Cloud Alibaba Seata

🏠 首页 > Spring Cloud

Spring Cloud是什么

< 上一节

下一节 >

Spring Cloud 是一款基于 Spring Boot 实现的微服务框架。Spring Cloud 源自 Spring 社区，主要由 Pivotal 和 Netflix 两大公司提供技术迭代和维护。

随着微服务的火爆流行，国内外各大互联网公司都相继分享了他们在微服务架构中，针对不同场景出现的各种问题的解决方案和开源框架。

- **服务治理**：阿里巴巴开源的 Dubbo 和当当网在其基础上扩展出来的 DubboX、Netflix 的 Eureka 以及 Apache 的 Consul 等。
- **分布式配置管理**：百度的 Disconf、Netflix 的 Archaius、360 的 QConf、携程的 Apollo 以及 Spring Cloud 的 Config 等。
- **批量任务**：当当网的 Elastic-Job、LinkedIn 的 Azkaban 以及 Spring Cloud 的 Task 等。
- **服务跟踪**：京东的 Hydra、Spring Cloud 的 Sleuth 以及 Twitter 的 Zipkin 等。
-

以上这些微服务框架或解决方案都具有以下 2 个特点：

- 对于同一个微服务问题，各互联网公司给出的解决方案各不相同。
- 一个微服务框架或解决方案都只能解决微服务中的某一个或某几个问题，对于其他问题则无能为力。

这种情况下，搭建一套微分布式微服务系统，就需要针对这些问题从诸多的解决方案中做出选择，这使得我们不得不将大量的精力花费在前期的调研、分析以及实验上。

Spring Cloud 被称为构建分布式微服务系统的“全家桶”，它并不是某一门技术，而是一系列微服务解决方案或框架的有序集合。它将市面上成熟的、经过验证的微服务框架整合起来，并通过 Spring Boot 的思想进行再封装，屏蔽调其中复杂的配置和实现原理，最终为开发人员提供了一套简单易懂、易部署和易维护的分布式系统开发工具包。

Spring Cloud 中包含了 spring-cloud-config、spring-cloud-bus 等近 20 个子项目，提供了服务治理、服务网关、智能路由、负载均衡、断路器、监控跟踪、分布式消息队列、配置管理等领域的解决方案。

Spring Cloud 并不是一个拿来即可用的框架，它是一种微服务规范，共有以下 2 代实现：

- 第一代实现：Spring Cloud Netflix
- 第二代实现：Spring Cloud Alibaba

这里我们介绍的 Spring Cloud 特指 Spring Cloud 的第一代实现。

Spring Cloud 常用组件

Spring Cloud 包括 Spring Cloud Gateway、Spring Cloud Config、Spring Cloud Bus 等近 20 个服务组件，这些组件提供了服务治理、服务网关、智能路由、负载均衡、熔断器、监控跟踪、分布式消息队列、配置管理等领域的解决方案。

Spring Cloud 的常用组件如下表所示。

Spring Cloud 组件	描述
-----------------	----



Spring Cloud Netflix Eureka	Spring Cloud Netflix 中的服务治理组件，包含服务注册中心、服务注册与发现机制的实现。
Spring Cloud Netflix Ribbon	Spring Cloud Netflix 中的服务调用和客户端负载均衡组件。
Spring Cloud Netflix Hystrix	人称“豪猪哥”，Spring Cloud Netflix 的容错管理组件，为服务中出现的延迟和故障提供强大的容错能力。
Spring Cloud Netflix Feign	基于 Ribbon 和 Hystrix 的声明式服务调用组件。
Spring Cloud Netflix Zuul	Spring Cloud Netflix 中的网关组件，提供了智能路由、访问过滤等功能。
Spring Cloud Gateway	一个基于 Spring 5.0，Spring Boot 2.0 和 Project Reactor 等技术开发的网关框架，它使用 Filter 链的方式提供了网关的基本功能，例如安全、监控/指标和限流等。
Spring Cloud Config	Spring Cloud 的配置管理工具，支持使用 Git 存储配置内容，实现应用配置的外部化存储，并支持在客户端对配置进行刷新、加密、解密等操作。
Spring Cloud Bus	Spring Cloud 的事件和消息总线，主要用于在集群中传播事件或状态变化，以触发后续的处理，例如动态刷新配置。
Spring Cloud Stream	Spring Cloud 的消息中间件组件，它集成了 Apache Kafka 和 RabbitMQ 等消息中间件，并通过定义绑定器作为中间层，完美地实现了应用程序与消息中间件之间的隔离。通过向应用程序暴露统一的 Channel 通道，使得应用程序不需要再考虑各种不同的消息中间件实现，就能轻松地发送和接收消息。
Spring Cloud Sleuth	Spring Cloud 分布式链路跟踪组件，能够完美的整合 Twitter 的 Zipkin。

注：Netflix 是美国的一个在线视频网站，它是公认的大规模生产级微服务的杰出实践者，微服务界的翘楚。Netflix 的开源组件已经在其大规模分布式微服务环境中经过了多年的生产实战验证，成熟且可靠。

Spring Boot 和 Spring Cloud 的区别与联系

Spring Boot 和 Spring Cloud 都是 Spring 大家族的一员，它们在微服务开发中都扮演着十分重要的角色，两者之间既存在区别也存在联系。

1. Spring Boot 和 Spring Cloud 分工不同

Spring Boot 是一个基于 Spring 的快速开发框架，它能够帮助开发者迅速搭 Web 工程。在微服务开发中，Spring Boot 专注于快速、方便地开发单个微服务。

Spring Cloud 是微服务架构下的一站式解决方案。Spring Cloud 专注于全局微服务的协调和治理工作。换句话说，Spring Cloud 相当于微服务的大管家，负责将 Spring Boot 开发的一个个微服务管理起来，并为它们提供配置管理、服务发现、断路器、路由、微代理、事件总线、决策竞选以及分布式会话等服务。

2. Spring Cloud 是基于 Spring Boot 实现的

Spring Cloud 是基于 Spring Boot 实现的。与 Spring Boot 类似，Spring Cloud 也为提供了一系列 Starter，这些 Starter 是 Spring Cloud 使用 Spring Boot 思想对各个微服务框架进行再封装的产物。它们屏蔽了这些微服务框架中复杂的配置和实现原理，使开发人员能够快速、方便地使用 Spring Cloud 搭建一套分布式微服务系统。



3. Spring Boot 和 Spring Cloud 依赖项数量不同

Spring Boot 属于一种轻量级的框架，构建 Spring Boot 工程所需的依赖较少。

Spring Cloud 是一系列微服务框架技术的集合体， 它的每个组件都需要一个独立的依赖项（Starter POM）， 因此想要构建一套完整的 Spring Cloud 工程往往需要大量的依赖项。

4. Spring Cloud 不能脱离 Spring Boot 单独运行

Spring Boot 不需要 Spring Cloud， 就能直接创建可独立运行的工程或模块。

Spring Cloud 是基于 Spring Boot 实现的， 它不能独立创建工程或模块， 更不能脱离 Spring Boot 独立运行。

注意：虽然 Spring Boot 能够用于开发单个微服务， 但它并不具备管理和协调微服务的能力， 因此它只能算是一个微服务快速开发框架， 而非微服务框架。

Spring Cloud 版本

Spring Cloud 包含了许多子项目（组件）， 这些子项目都是独立进行内容更新和迭代的， 各自都维护着自己的发布版本号。

为了避免 Spring Cloud 的版本号与其子项目的版本号混淆， Spring Cloud 没有采用常见的数字版本号， 而是通过以下方式定义版本信息。

```
{version.name} . {version.number}
```

Spring Cloud 版本信息说明如下：

- **version.name**：版本名，采用英国伦敦地铁站的站名来命名，并按照字母表的顺序（即从 A 到 Z）来对应 Spring Cloud 的版本发布顺序，例如第一个版本为 Angel，第二个版本为 Brixton（英国地名）， 然后依次是 Camden、Dalston、Edgware、Finchley、Greenwich、Hoxton 等。
- **version.number**：版本号， 每一个版本的 Spring Cloud 在更新内容积累到一定的量级或有重大 BUG 修复时， 就会发布一个 “service releases” 版本， 简称 SRX 版本， 其中 X 为一个递增的数字， 例如 Hoxton.SR8 就表示 Hoxton 的第 8 个 Release 版本。

Spring Cloud 版本选择

在使用 Spring Boot + Spring Cloud 进行微服务开发时， 我们需要根据项目中 Spring Boot 的版本来决定 Spring Cloud 版本， 否则会出现许多意想不到的错误。

Spring Boot 与 Spring Cloud 的版本对应关系如下表（参考自 [Spring Cloud 官网](#)）。

Spring Cloud	Spring Boot
2020.0.x （Ilford）	2.4.x, 2.5.x （从 Spring Cloud 2020.0.3 开始）
Hoxton	2.2.x, 2.3.x （从 Spring Cloud SR5 开始）
Greenwich	2.1.x
Finchley	2.0.x
Edgware	1.5.x



Dalston	1.5.x
---------	-------

注意：Spring Cloud 官方已经停止对 Dalston、Edgware、Finchley 和 Greenwich 的版本更新。

除了上表中展示的版本对应关系之外，我们还可以使用浏览器访问 <https://start.spring.io/actuator/info>，获取 Spring Cloud 与 Spring Boot 的版本对应关系（JSON 版）。

```
01.  {
02.     .....
03.     "bom-ranges":{
04.         .....
05.         "spring-cloud":{
06.             "Hoxton.SR12":"Spring Boot >=2.2.0.RELEASE and <2.4.0.M1",
07.             "2020.0.4":"Spring Boot >=2.4.0.M1 and <2.5.6-SNAPSHOT",
08.             "2020.0.5-SNAPSHOT":"Spring Boot >=2.5.6-SNAPSHOT and <2.6.0-M1",
09.             "2021.0.0-M1":"Spring Boot >=2.6.0.M1 and <2.6.0-SNAPSHOT",
10.             "2021.0.0-SNAPSHOT":"Spring Boot >=2.6.0-SNAPSHOT"
11.         },
12.         .....
13.     },
14.     .....
15. }
```

< 上一节

下一节 >

推荐阅读

Go语言排序（借助sort.Interface接口）	memset函数及其用法，C语言memset函数详解
Java保存图书信息	使用exec命令操作文件描述符
三元组顺序表，稀疏矩阵的三元组表示及（C语言）实现	<fmt:timeZone>标签
《Web前端开发技术——HTML5、CSS3、JavaScript（第3版）》PDF	《明解C语言(中级篇)》PDF下载（高清完整版）
Go语言集合的用法	什么是Java，Java语言简介

精美而实用的网站，分享优质编程教程，帮助有志青年。千锤百炼，只为大作；精益求精，处处斟酌；这种教程，看一眼就倾心。

[关于网站](#) | [联系我们](#) | [网站地图](#)

Copyright ©2012-2023 biancheng.net

biancheng.net

