



链滴

# spring cloud - 简介

作者: [fjun](#)

原文链接: <https://ld246.com/article/1533263700785>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p>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，如服务发现注册、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发工具包。 </p>

## 

<p>Spring Cloud 的子项目，大致可分成两类，一类是对现有成熟框架“ Spring Boot 化” 的封装抽象，也是数量最多的项目；第二类是开发了一部分分布式系统的基础设施的实现，如 Spring Cloud Stream 扮演的就是 kafka, ActiveMQ 这样的角色。对于我们想快速实践<a href="https://ld246.co/forward?goto=https%3A%2F%2Fbaidu.com%2Fitem%2F%25E5%25BE%25AE%25E6%29C%258D%25E5%258A%25A1" target="\_blank" rel="nofollow ugc">微服务</a>的开发者来，第一类子项目就已经足够使用，如： </p>

- <ul>

- <li>

### <p>Spring Cloud Netflix<br>

是对 Netflix 开发的一套分布式服务框架的封装，包括服务的发现和注册，负载均衡、断路器、RST 客户端、请求路由等。 </p>

- </li>
- <li>

### <p>Spring Cloud Config<br>

将配置信息中央化保存, 配置 Spring Cloud Bus 可以实现动态修改配置文件 </p>

- </li>
- <li>

### <p>Spring Cloud Bus<br>

分布式消息队列，是对 Kafka, MQ 的封装 </p>

- </li>
- <li>

### <p>Spring Cloud Security<br>

对 Spring Security 的封装，并能配合 Netflix 使用 </p>

- </li>
- <li>

### <p>Spring Cloud Zookeeper<br>

对 Zookeeper 的封装，使之能配置其它 Spring Cloud 的子项目使用 </p>

- </li>
- <li>

### <p>Spring Cloud Eureka</p>

<p>Spring Cloud Eureka 是 Spring Cloud Netflix 微服务套件中的一部分，它基于 Netflix Eureka 做了二次封装，主要负责完成微服务架构中的服务治理功能。 </p>

- </li>

- </ul>

## 

<p>Spring Cloud 对于中小型互联网公司来说是一种福音，因为这类公司往往没有实力或者没有足够的资金投入去开发自己的分布式系统基础设施，使用 Spring Cloud 一站式解决方案能在从容应对业务发展的同时大大减少开发成本。同时，随着近几年微服务架构和 Docker 容器概念的火爆，也会让 Spring Cloud 在越来越来越“云”化的软件开发风格中立有一席之地，尤其是在目前五花八门的分布式解决方案中提供了标准化的、全站式的技术方案，意义可能会堪比当年 Servlet 规范的诞生，有效推进务端软件系统技术水平的进步。 </p>