



链滴

springcloud(一): 大话 Spring Cloud

作者: [911708498](#)

原文链接: <https://ld246.com/article/1543310472931>

来源网站: [链滴](#)

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

研究了一段时间Spring Boot了准备向Spring Cloud进发，公司架构和项目也全面拥抱了Spring Cloud。在使用了一段时间后发现Spring Cloud从技术架构上降低了对大型系统构建的要求，使我们以非常的成本（技术或者硬件）搭建一套高效、分布式、容错的平台，但Spring Cloud也不是没有缺点，小独立的项目不适合使用。

Spring Cloud是什么？

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并没有重复制造轮子，它只是将目前各家公司开的比较成熟、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开发工具包。

微服务是可以独立部署、水平扩展、独立访问（或者有独立的数据库）的服务单元，springcloud就这些微服务的大管家，采用了微服务这种架构之后，项目的数量会非常多，springcloud作为大管家要管理好这些微服务，自然需要很多小弟来帮忙。

主要的小弟有：Spring Cloud Config、Spring Cloud Netflix (Eureka、Hystrix、Zuul、Archaius)、Spring Cloud Bus、Spring Cloud for Cloud Foundry、Spring Cloud Cluster、Spring Cloud Consul、Spring Cloud Security、Spring Cloud Sleuth、Spring Cloud Data Flow、Spring Cloud Stream、Spring Cloud Task、Spring Cloud Zookeeper、Spring Cloud Connectors、Spring Cloud Starters、Spring Cloud CLI，每个小弟身怀独门绝技武功高强下面来做一一介绍。

核心成员

Spring Cloud Netflix

这可是个大boss，地位仅次于老大，老大各项服务依赖与它，与各种Netflix OSS组件集成，组成微务的核心，它的小弟主要有Eureka, Hystrix, Zuul, Archaius... 太多了

Netflix Eureka

服务中心，云端服务发现，一个基于 REST 的服务，用于定位服务，以实现云端中间层服务发现和故障转移。这个可是springcloud最牛鼻的小弟，服务中心，任何小弟需要其它小弟支持什么都需要从这里来拿，同样的你有什么独门武功的都赶紧过报道，方便以后其它小弟来调用；它的好处是你不需要直找各种什么小弟支持，只需要到服务中心来领取，也不需要知道提供支持的其它小弟在哪里，还是几小弟来支持的，反正拿来用就行，服务中心来保证稳定性和质量。

Netflix Hystrix

熔断器，容错管理工具，旨在通过熔断机制控制服务和第三方库的节点,从而对延迟和故障提供更强大容错能力。比如突然某个小弟生病了，但是你还需要它的支持，然后调用之后它半天没有响应，你却知道，一直在等等这个响应；有可能别的小弟也正在调用你的武功绝技，那么当请求多之后，就会发严重的阻塞影响老大的整体计划。这个时候Hystrix就派上用场了，当Hystrix发现某个小弟不在状态稳定立马马上让它下线，让其它小弟来顶上来，或者给你说不用等了这个小弟今天肯定不行，该干嘛赶紧干嘛去别在这排队了。

Netflix Zuul

Zuul 是在云平台上提供动态路由,监控,弹性,安全等边缘服务的框架。Zuul 相当于是设备和 Netflix 流行的 Web 网站后端所有请求的前门。当其它门派来找大哥办事的时候一定要先经过zuul,看下有没有刀子什么的给拦截回去，或者是需要找那个小弟的直接给带过去。

Netflix Archaius

配置管理API，包含一系列配置管理API，提供动态类型化属性、线程安全配置操作、轮询框架、回调

制等功能。可以实现动态获取配置，原理是每隔60s（默认，可配置）从配置源读取一次内容，这样改了配置文件后不需要重启服务就可以使修改后的内容生效，前提使用archaius的API来读取。

Spring Cloud Config

俗称的配置中心，配置管理工具包，让你可以把配置放到远程服务器，集中化管理集群配置，目前支本地存储、Git以及Subversion。就是以后大家武器、枪火什么的都集中放到一起，别随便自己，方便以后统一管理、升级装备。

Spring Cloud Bus

事件、消息总线，用于在集群（例如，配置变化事件）中传播状态变化，可与Spring Cloud Config合实现热部署。相当于水浒传中日行八百里的神行太保戴宗，确保各个小弟之间消息保持畅通。

Spring Cloud for Cloud Foundry

Cloud Foundry是VMware推出的业界第一个开源PaaS云平台，它支持多种框架、语言、运行时环境云平台及应用服务，使开发人员能够在几秒钟内进行应用程序的部署和扩展，无需担心任何基础架构问题

其实就是与CloudFoundry进行集成的一套解决方案，抱了Cloud Foundry的大腿。

Spring Cloud Cluster

Spring Cloud Cluster将取代Spring Integration。提供在分布式系统中的集群所需要的基础功能支，如：选举、集群的状态一致性、全局锁、tokens等常见状态模式的抽象和实现。

如果把不同的帮派组织成统一的整体，Spring Cloud Cluster已经帮你提供了很多方便组织成统一的具。

Spring Cloud Consul

Consul 是一个支持多数据中心分布式高可用的服务发现和配置共享的服务软件,由 HashiCorp 公司用 Go 语言开发, 基于 Mozilla Public License 2.0 的协议进行开源. Consul 支持健康检查,并允许 HTTP 和 DNS 协议调用 API 存储键值对.

Spring Cloud Consul 封装了Consul操作，consul是一个服务发现与配置工具，与Docker容器可以缝集成。

其它成员

Spring Cloud Security

基于spring security的安全工具包，为你的应用程序添加安全控制。这个小弟很牛鼻专门负责整个帮的安全问题，设置不同的门派访问特定的资源，不能把秘籍葵花宝典泄漏了。

Spring Cloud Sleuth

日志收集工具包，封装了Dapper和log-based追踪以及Zipkin和HTrace操作，为SpringCloud应用现了一种分布式追踪解决方案。

Spring Cloud Data Flow

Data flow 是一个用于开发和执行大范围数据处理其模式包括ETL，批量运算和持续运算的统一编程型和托管服务。

对于在现代运行环境中可组合的微服务程序来说，Spring Cloud data flow是一个原生云可编配的务。使用Spring Cloud data flow，开发者可以为像数据抽取，实时分析，和数据导入/导出这种常

用例创建和编配数据通道 (data pipelines) 。

Spring Cloud data flow 是基于原生云对 spring XD的重新设计，该项目目标是简化大数据应用的发。Spring XD 的流处理和批处理模块的重构分别是基于 Spring Boot的stream 和 task/batch 的微务程序。这些程序现在都是自动部署单元而且他们原生的支持像 Cloud Foundry、Apache YARN、A ache Mesos和Kubernetes 等现代运行环境。

Spring Cloud data flow 为基于微服务的分布式流处理和批处理数据通道提供了一系列模型和最佳践。

Spring Cloud Stream

Spring Cloud Stream是创建消息驱动微服务应用的框架。Spring Cloud Stream是基于Spring Boo创建，用来建立单独的 / 工业级spring应用，使用spring integration提供与消息代理之间的连接。据流操作开发包，封装了与Redis,Rabbit、Kafka等发送接收消息。

一个业务会牵扯到多个任务，任务之间是通过事件触发的，这就是Spring Cloud stream要干的事了

Spring Cloud Task

Spring Cloud Task 主要解决短命微服务的任务管理，任务调度的工作，比如说某些定时任务晚上就一次，或者某项数据分析临时就跑几次。

Spring Cloud Zookeeper

ZooKeeper是一个分布式的，开放源码的分布式应用程序协调服务，是Google的Chubby一个开源实现，是Hadoop和Hbase的重要组件。它是一个为分布式应用提供一致性服务的软件，提供的功能括：配置维护、域名服务、分布式同步、组服务等。ZooKeeper的目标就是封装好复杂易出错的关键务，将简单易用的接口和性能高效、功能稳定的系统提供给用户。

操作Zookeeper的工具包，用于使用zookeeper方式的服务发现和配置管理，抱了Zookeeper的大腿。

Spring Cloud Connectors

Spring Cloud Connectors 简化了连接到服务的过程和从云平台获取操作的过程，有很强的扩展性，以利用Spring Cloud Connectors来构建你自己的云平台。

便于云端应用程序在各种PaaS平台连接到后端，如：数据库和消息代理服务。

Spring Cloud Starters

Spring Boot式的启动项目，为Spring Cloud提供开箱即用的依赖管理。

Spring Cloud CLI

基于 Spring Boot CLI，可以让你以命令行方式快速建立云组件。

技术架构图如下：

