



下载APP



## 24 | 为什么微服务架构少不了微服务网关？

2022-02-07 姚秋辰

《Spring Cloud 微服务项目实战》

课程介绍 &gt;



讲述：姚秋辰

时长 18:25 大小 16.87M



你好，我是姚秋辰。

今天我们来了解一个新的平台类组件：Spring Cloud Gateway（以下简称 Gateway）。它在微服务架构中扮演的角色是“微服务网关”。

一听到“网关”这个词儿，你第一个想到的一定是 Nginx。没错，Nginx 作为网关领一哥，在大中小厂里的应用面那可不是一般的广泛。那究竟是 Nginx 过气了，还是 Spring Cloud Gateway 太牛了，以至于我们要引入一个新的网关组件呢？

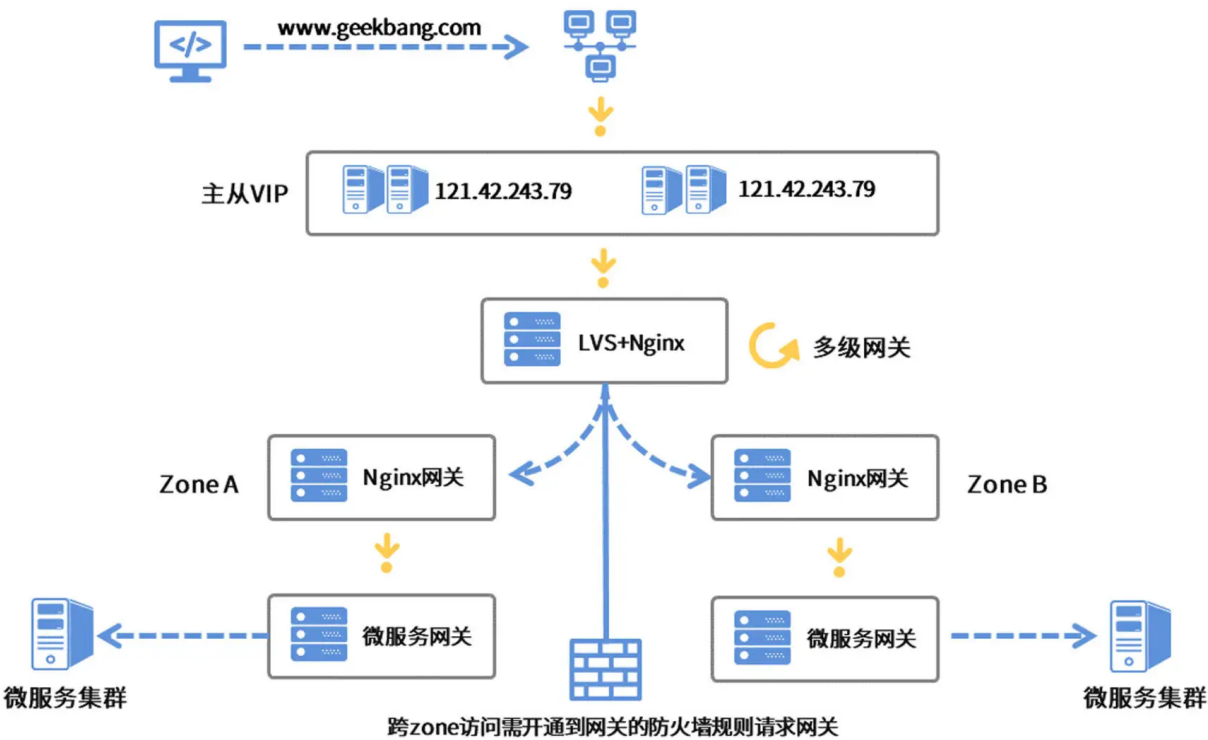


其实都不是，Gateway 并没有抢 Nginx 的地盘，此网关非彼网关，Nginx 和 Gateway 在微服务体系中的分工是不一样的。**Gateway 作为更底层的微服务网关，通常是作为外部 Nginx 网关和内部微服务系统之间的桥梁，起了这么一个承上启下的作用。**

也许你会想，企业级应用不就是在最外层架设一道网关层么？其实不然，在大型微服务应用中，我们往往会搭建多个网关组件，这些网关的应用场景也各有不同。

## 大型微服务应用中的多层网关

接下来，我们就通过一个例子，了解一个服务请求从浏览器发出，直到抵达后台服务的整个过程。你可以在图中看到，这个服务请求都经历了哪些网关层。然后，我会再带你分析，为何需要 Gateway 横在中间做这么一层网络转发。



首先，网址解析的第一步是 DNS 解析。当用户在浏览器里输入一串网址时，这个网址会被 DNS 层解析成一个可被访问的 IP 地址。为了**避免单点故障**，我们可以在这一层加个双保险，比如将域名映射成两个 IP 地址做主备，又或者根据用户 IP 所属区域做 Loadbalancer，将请求导向就近的 IP 地址，这两种方式都是可以的。

在这里你会发现，每一个 IP 地址背后都别有洞天，因为它们只是所谓的虚 IP，后面会映射到一个大型的网关集群，这个集群便是我们业务系统对外的第一道网关。在这个环节中，使用最广泛而且最经济实惠的技术选型就是 Nginx 反向代理。因为它拥有超强的并发能力，而且很节省内存资源。

到这还没完，刚才我提到在大厂里往往会有多个网关组件。也就是说，一个请求抵达最外层的 Nginx 服务之后，还可能会经历多级 LVS+Nginx 集群的转发。大公司之所以这么玩，主要是出于对网络安全的考虑。他们往往会根据业务系统的属性和安全级别来设置不同的网络分区，而这些网络分区之间是相互独立的，分区之间需要开通白名单或者防火墙才能打通连接。比如有的网络分区可以直接对外，而有的高安全级别的分区（比如金融类业务）则部署在更底层的 Secure Zone 当中。这就和我们使用跳板机访问线上机房是一个道理。

然后，请求经过了多级网关服务的转发，抵达了最后的微服务层。在这一层上，Gateway 就需要出马来负责请求转发了。

那 Gateway 早不出现晚不出现，偏偏在请求抵达微服务的最后一刻冒了出来，它的底层逻辑是什么呢？

Gateway 既然叫“微服务网关”，就说明它自己就是一个微服务。换句话说，它也是 Nacos 服务注册中心的一员。既然 Gateway 能连接到 Nacos，那么就意味着它可以轻松获取到 Nacos 中所有服务的注册表。这样一来，Gateway 就可以根据本地的路由规则，将请求精准无误地送达到每个微服务组件中。

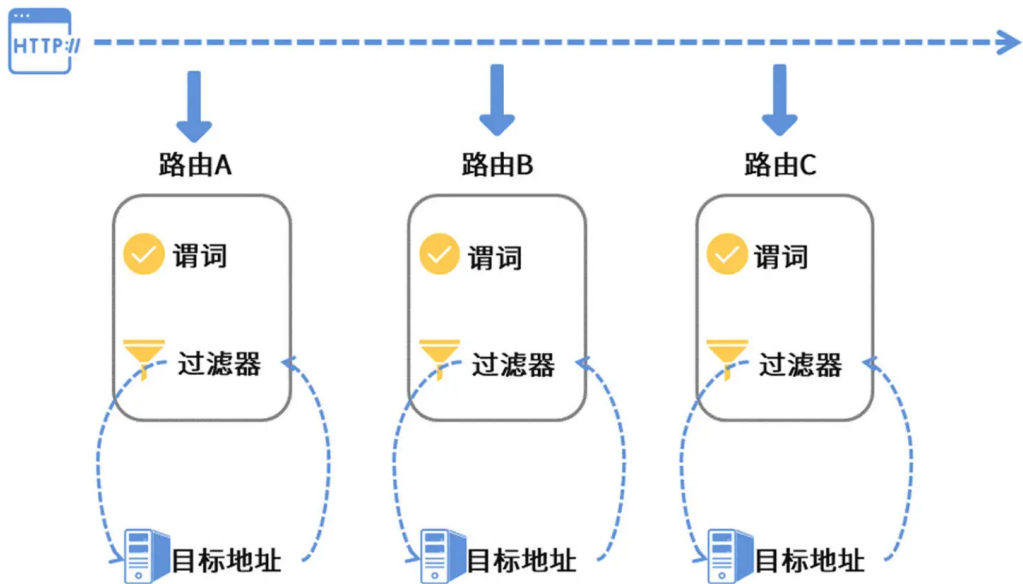
使用 Gateway 有一个显而易见的好处，那就是**高可扩展性**。当你对后台的微服务集群做扩容或缩容的时候，Gateway 可以从 Nacos 注册中心轻松获取所有服务节点的变动，不需要任何额外的配置，一切都在无感知的情况下自然而然地发生。如果使用其他技术方案，你可能还需要花些力气修改 VIP Pool 中的节点列表，将新增的机器手动添加到列表中，还要把移除的机器从列表中删除。

Gateway 的另一个优点就是**高度可定制化**。它提供了一种对开发人员非常友好的方式，可以让你通过 Java 代码去定制各种复杂的路由逻辑，还可以使用 Filter 对请求进行加工。

那么接下来，我就带你了解 Gateway 的几个核心功能模块，看一看它是如何组装路由规则的。

## Gateway 路由规则

Gateway 的路由规则主要有三个部分，分别是路由、谓词和过滤器。我这里画了一张图来表示 Gateway 的路由结构。



路由

**路由是 Gateway 的一个基本单元**，每个路由都有一个目标地址，这个目标地址就是当前路由规则要调用的目标服务。那么一条路由规则在什么情况下会去调用目标服务呢？这就要看路由的谓词设置了。

谓词

**所谓谓词，实际上是路由的判断规则**，一个路由中可以添加多个谓词的组合。如果一个服务请求满足某个路由里设置的所有的谓词规则，那么就说明这个请求是当前路由的心动女神，这时候 Gateway 就会把请求转发到路由中设置的目标地址。

打个比方，你可以为某个路由设置一条谓词规则，约定访问路径的匹配规则为 Path=/bingo/\*，在这种情况下只有以 /bingo 打头的请求才会被当前路由选中。

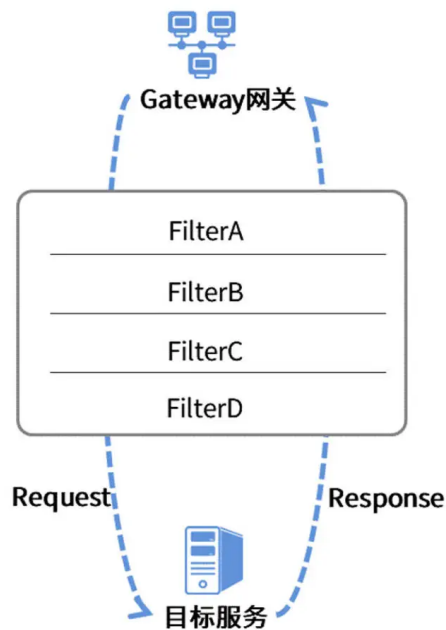
Gateway 为我们提供了非常丰富的内置谓词，你可以通过内置谓词构建复杂的路由条件，甚至连“整点秒杀”这个场景都能在网关层做控制。我将在下一课带你来了解 Gateway 有哪些内置谓词，以及如何通过 Gateway 的谓词工厂创建一个自定义谓词。

现在你已经了解了谓词和路由是怎么配合工作的，其实 Gateway 里通常会配置多个路由单元。因为在真实项目里，每个微服务都有不同的路由规则，但每个请求只能被一个路由规则选中。那如果某个请求同时匹配上了多个路由，该选择哪个路由呢？Gateway 提供了一

种“优先级”设置，你可以通过设置路由的优先级参数来调整生效的先后顺序，我将在下一节课里带你了解优先级设定的具体代码。

## 过滤器

你一定注意到了我在路由的框里还画了一个“过滤器”，还连了两条虚线到路由的“目标地址”，那么过滤器和路由、目标地址之间是什么关系呢？其实 Gateway 在把请求转发给目标地址的过程中，把这个任务全权委托给了 Filter（过滤器）来处理。我用一幅图为你比划一下 Filter 做了什么事儿。



Gateway 组件使用了一种 FilterChain 的模式对请求进行处理，每一个服务请求（Request）在发送到目标服务之前都要被一串 FilterChain 处理。同理，在 Gateway 接收服务响应（Response）的过程中也会被 FilterChain 处理一把。

Gateway 的过滤器主要分为两种，一种是 GlobalFilter，也就是“**全局过滤器**”；另一种是 GatewayFilter，也就是对指定路由生效的“**局部过滤器**”。

全局过滤器继承自 GlobalFilter 接口，它的作用大多是“例行公事”，也就是一些底层能力的支持。比如，RouteToRequestUrlFilter 这个全局过滤器就是用来解析“目标服务地址”的。



除此之外，Gateway 还有一系列用来做路径转发、请求跨域、WebSocket、WebClient 和 Loadbalancer 功能支持的全局过滤器。如果你想深入了解，可以参考 GatewayAutoConfiguration 的源码，这个类是 Gateway 的自动装配器，里面包含了大量 GlobalFilter 的声明。就算你不做任何配置，项目在初始化的时候也会把一大家子全局过滤器添加到上下文中。

GatewayFilter 也就是局部过滤器，它的功能可就多了。Gateway 提供了一系列的内置过滤器，可以实现对 Request/Response 的修改、请求路径修改、调用重试、限流等等功能。当然了，你也可以通过 Gateway 的扩展接口实现一个自定义过滤器并应用到路由规则中。

我将在第 26 课里通过网关限流和全局跨域功能带你了解局部过滤器和全局过滤器的使用，如果你感兴趣，也可以自己预习一下。

到这里，我们就了解了 Gateway 的路由、谓词和过滤器之间是如何协同工作的。接下来我就来带你回顾一下这一节的重点内容吧。

## 总结

今天我们对 Gateway 的路由构成做了分析，知道了谓词和过滤器在路由规则中发挥的作用。在日常开发过程中，路由往往不是由开发人员负责管理的模块，通常会有专门的横向团队负责维护路由表，对路由表的修改也需要经过安全团队的审批，才能对外发布。这是大部分外企使用的管理方式，在老外的观念里，稳定和安全是摆在第一位的。

不过呢，在追求效率的互联网公司里，一线开发团队也可以直接修改对外的路由规则，比如淘系 mtop 接口（不知道现在有没有被其它轮子替换掉）就是授权给开发团队修改的对外网关层，国内倡导的则是糙快猛怎么快怎么来。这种国内外文化的不同，也会体现在公司治理中（965 WLB 对比 996 ICU）。

同理，很多开源项目中也渗入了开发公司的企业文化基因，比如阿里系的很多组件喜欢堆叠眼花缭乱的功能，而 Spring 原生的组件就比较大道至简。当然了，两者没有明显优劣，就像老外玩不转复杂的双 11 优惠计算，就喜欢 amazon 和 ebay 这种简单购物体验，而国内用户不管优惠规则多么复杂都玩得游刃有余并且乐此不疲。

扯远了，回到 Gateway 上面来。没啥重要的事儿嘱咐了，就强调一个事儿吧，那就是尽量别把 Gateway 作为直接对外的网关层，内部服务治理用用就好了，对外还是交给 nginx 这类高性能网关来做。

## 思考题

除了路由功能以外，你还能想到哪些可以挂载在网关上的功能？我举个例子：限流。也欢迎你把想到的功能分享到评论区。

好啦，这节课就结束啦。欢迎你把这节课分享给更多对 Spring Cloud 感兴趣的朋友。我是姚秋辰，我们下节课再见！

分享给需要的人，Ta 购买本课程，你将得 20 元

 生成海报并分享

 赞 2  提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 23 | 调用链追踪：如何通过 ELK 实现日志检索？

下一篇 25 | 微服务网关：Gateway 中的路由和谓词有何应用？

## 精选留言 (3)

 写留言



peter

2022-02-07

请教老师两个问题：

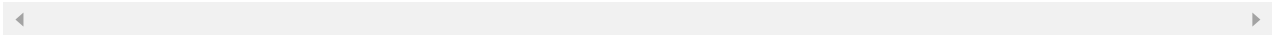
Q1：“可扩展”和“可伸缩”的区别？有的文章把“可扩展”和“可伸缩”当做两个不同的概念来讲解，但我总觉得这两个概念是一回事，请问它们的主要区别是什么？

Q2：老师能联系极客的编辑建立一个学习群吗？老师忙，可以不用进群，但读者可以相互学习。

展开 ∨

作者回复: Q1：看个人习惯了，可伸缩一般用来形容集群弹性，快速扩容缩容，峰值流量的时候弹出算力。可扩展我一般形容微观的层面，比如框架的具体扩展点等等。

Q2：好像平台有个要求是讲师不能私拉学员进群，大家可以留言区继续愉快吹水



共 2 条评论 >



2



**BOB**

2022-02-09

针对路由的访问鉴权可以放在微服务网关吗？



**Geek\_e93c48**

2022-02-07

问个问题，could实战项目姚老师啥时候结课？（迫不急的催更中）

编辑回复: 3月中旬更新完毕~

